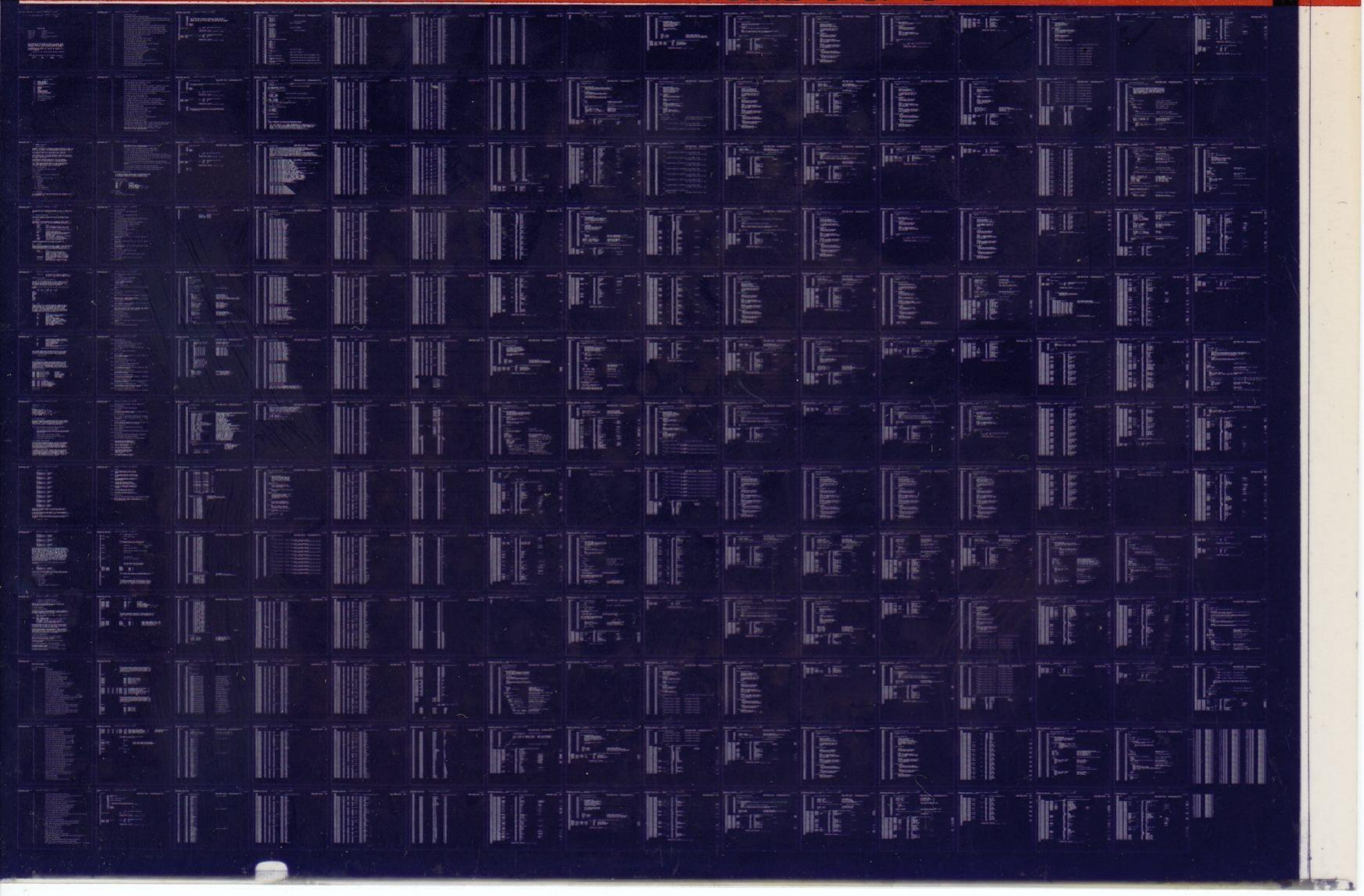
ML11

LOGIC TEST
CZMLADO

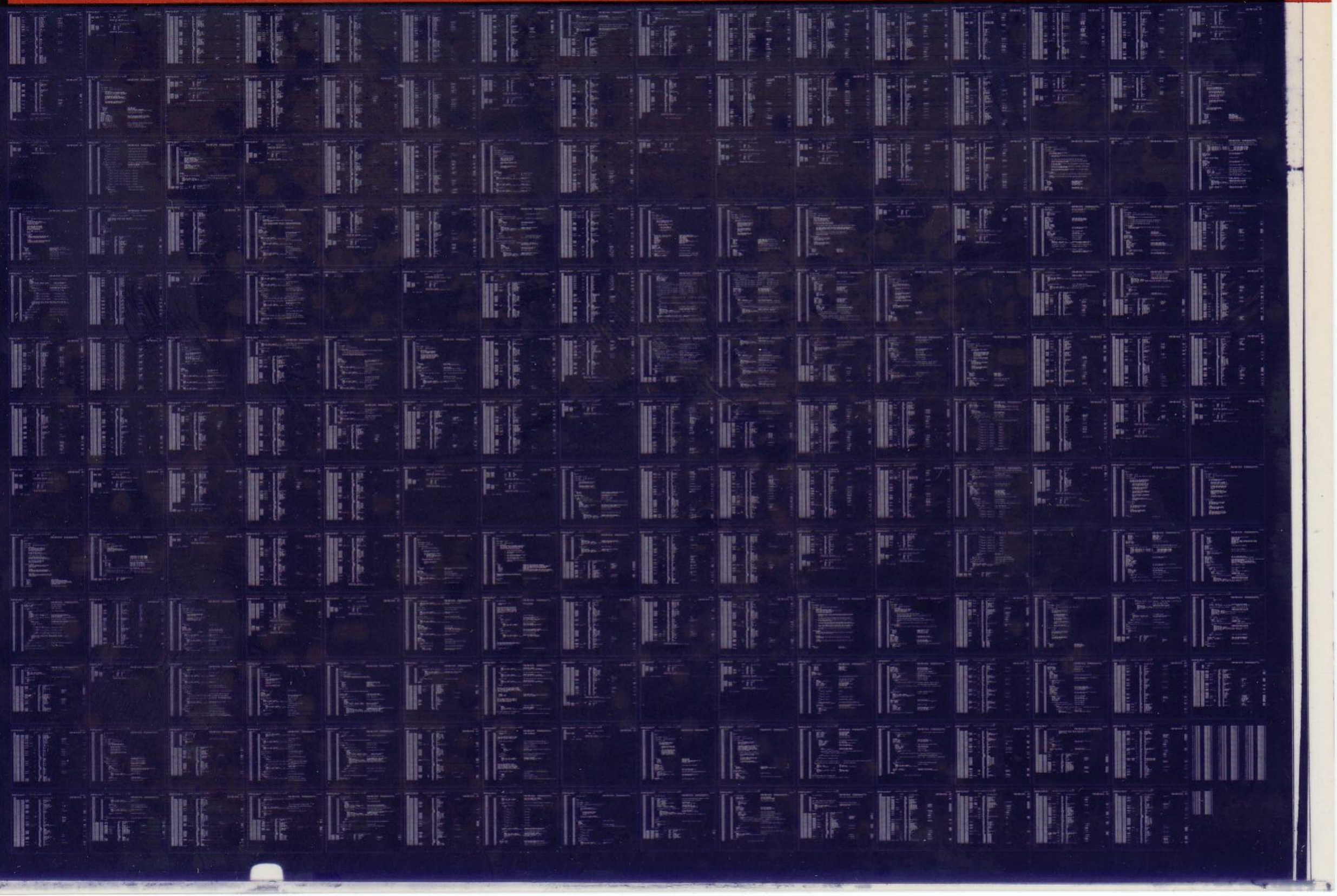
AH-S390D-MC
FICHE 1 OF 3



ML11

LOGIC TEST
CZMLADO

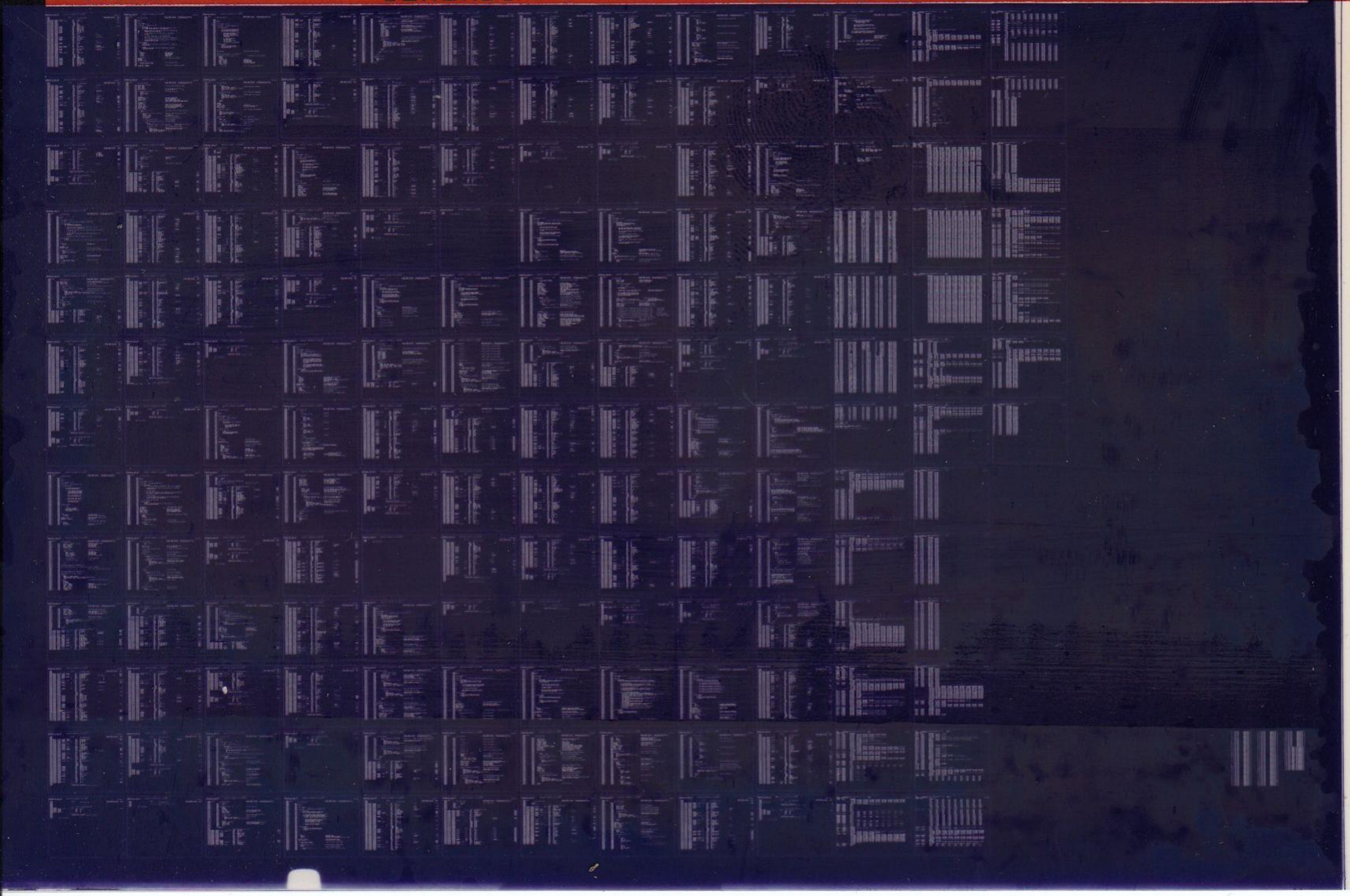
AH-S390D-MC
FICHE 2 OF 3



ML11

LOGIC TEST
CZMLADO

AH-S390D-MC
FICHE 3 OF 3



```
1                       .TITLE CZMLADO ML-11 LOGIC TEST
        .SBTTL USER DOCUMENTATION
                                                .REM &
```

IDENTIFICATION

PRODUCT CODE:	AC-S388D-MC
PRODUCT NAME:	CZMLADO ML11 LOGIC TEST
PRODUCT DATE:	19-MAR-82
MAINTAINER:	MEMORY DIAGNOSTIC ENGINEERING
AUTHOR:	D.W. NEALE

TABLE OF CONTENTS

1.0 GENERAL INFORMATION
1.1 PROGRAM ABSTRACT
1.2 SYSTEM REQUIREMENTS
1.3 RELATED DOCUMENTS AND STANDARDS
1.4 ASSUMPTIONS

2.0 OPERATING INSTRUCTIONS
2.1 COMMANDS
2.2 SWITCHES
2.3 FLAGS
2.4 HARDWARE QUESTIONS
2.5 SOFTWARE QUESTIONS
2.6 EXTENDED P-TABLE DIALOGUE
2.7 QUICK STARTUP PROCEDURE

3.0 ERROR INFORMATION

4.0 PERFORMANCE AND PROGRESS REPORTS

5.0 DEVICE INFORMATION TABLES

6.0 TEST SUMMARIES

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

DIAGNOSTIC ENGINEERING WAS CONTRACTED BY MEMORY ENGINEERING TO MAKE THIS DIAGNOSTIC IN ORDER TO AID MEMORY ENGINEERING TO DESIGN AND DEBUG THE ML-11, AID FIELD SERVICE FOR FIELD REPAIRS AND INSTALLATIONS THE ML-11.

THIS DIAGNOSTIC PRODUCT WILL BE DESIGNED TO TEST FROM ONE TO EIGHT ML-11A OR ML-11B UNITS OFF A SINGLE RH11 OR RH70 CONTROLLER.

THE FUNCTIONAL LEVEL (FRU) OF THIS DIAGNOSTIC PRODUCT WILL BE TO THE LOGIC FUNCTION LEVEL. UPON DETECTION OF AN ERROR BY THE DIAGNOSTIC, THE LOGIC FUNCTION AND RESPECTIVE MODULE WHICH IT IS LOCATED ON WILL BE PRINTED TO THE OPERATOR.

THIS DIAGNOSTIC HAS BEEN WRITTEN FOR USE WITH THE DIAGNOSTIC RUNTIME SERVICES SOFTWARE (SUPERVISOR). THESE SERVICES PROVIDE THE INTERFACE TO THE OPERATOR AND TO THE SOFTWARE ENVIRONMENT.

THIS PROGRAM CAN BE USED WITH XXDP+, ACT, APT, SLIDE AND PAPER TAPE. FOR A COMPLETE DESCRIPTION OF THE RUNTIME SERVICES, REFER TO THE XXDP+ USER'S MANUAL. THERE IS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES IN SECTION 2 OF THIS DOCUMENT.

1.2 SYSTEM REQUIREMENTS

PDP-11 WITH MINIMUM OF 28K WORDS OF MEMORY

CONSOLE TERMINAL

RH11 OR RH70

1 TO 8 ML-11A OR ML-11B DRIVES ON INTERMIXED BUS

XXDP+ LOAD MEDIA

1.3 RELATED DOCUMENTS AND STANDARDS

1. SUPPRGC.DOC
2. SUPINT.MEN
3. SUPFUN.C
4. XXDPPLUS.DOC
5. BLISS LANGUAGE GUIDE
6. BLISS-16 USER'S GUIDE

1.4 ASSUMPTIONS

IT WILL BE ASSUMED THAT PRIOR TO THE RUNNING OF THIS DIAGNOSTIC THAT ALL APPROPRIATE CPU, MAIN MEMORY AND RH CONTROLLER DIAGNOSTICS HAVE BEEN SUCCESSFULLY RUN.

THIS DIAGNOSTIC WILL HOWEVER PERFORM MINIMAL RH TESTS TO ENSURE ITS EXISTANCE AND BASIC FUNCTIONALITY BEFORE LOGIC TESTS ARE ALLOWED TO RUN.

2.0 OPERATING INSTRUCTIONS

THIS SECTION CONTAINS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES. FOR DETAILED INFORMATION, REFER TO THE XXDP+ USER'S MANUAL (CHQUS).

2.1 COMMANDS

THERE ARE ELEVEN LEGAL COMMANDS FOR THE DIAGNOSTIC RUNTIME SERVICES (SUPERVISOR). THIS SECTION LISTS THE COMMANDS AND GIVES A VERY BRIEF DESCRIPTION OF THEM. THE XXDP+ USER'S MANUAL HAS MORE DETAILS.

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT TEST THAT WAS INTERRUPTED (AFTER ^C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP+ MONITOR (XXDP+ OPERATION ONLY!)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE CONSIDERED TO BE ACTIVE AT START TIME
DROP	DEACTIVATE A UNIT
PRINT	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED BY THE DIAGNOSTIC - SECTION 4.0)
DISPLAY	TYPE A LIST OF ALL DEVICE INFORMATION
FLAGS	TYPE THE STATE OF ALL FLAGS (SEE SECTION 2.3)
ZFLAGS	CLEAR ALL FLAGS (SEE SECTION 2.3)

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO YOU MAY, FOR EXAMPLE, TYPE "STA" INSTEAD OF "START".

2.2 SWITCHES

THERE ARE SEVERAL SWITCHES WHICH ARE USED TO MODIFY SUPERVISOR OPERATION. THESE SWITCHES ARE APPENDED TO THE LEGAL COMMANDS. ALL OF THE LEGAL SWITCHES ARE TABULATED BELOW WITH A BRIEF DESCRIPTION OF EACH. IN THE DESCRIPTIONS BELOW, A DECIMAL NUMBER IS DESIGNATED BY "DDDDD".

SWITCH	EFFECT
/TESTS:LIST	EXECUTE ONLY THOSE TESTS SPECIFIED IN THE LIST. LIST IS A STRING OF TEST NUMBERS, FOR EXAMPLE - /TESTS:1:5:7-10. THIS LIST WILL CAUSE TESTS 1,5,7,8,9,10 TO BE RUN. ALL OTHER TESTS WILL NOT BE RUN.
/PASS:DDDDD	EXECUTE DDDDD PASSES (DDDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 2.3.
/EOP:DDDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDDD = 1 TO 64000)

/UNITS:LIST TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED IN THE LIST. LIST EXAMPLE - /UNITS:0:5:10-12 USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

START/TESTS:1-5/PASS:1000/EOP:100

THE EFFECT OF THIS COMMAND WILL BE: 1) TESTS 1 THROUGH 5 WILL BE EXECUTED, 2) ALL UNITS WILL TESTED 1000 TIMES AND 3) THE END OF PASS MESSAGES WILL BE PRINTED AFTER EACH 100 PASSES ONLY. A SWITCH CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. YOU MAY, FOR EXAMPLE, TYPE "/TES:1-5" INSTEAD OF "/TESTS:1-5".

BELOW IS A TABLE THAT SPECIFIES WHICH SWITCHES CAN BE USED BY EACH COMMAND.

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
PRINT					
DISPLAY					X
FLAGS					
ZFLAGS					
EXIT					

2.3 FLAGS

FLAGS ARE USED TO SET UP CERTAIN OPERATIONAL PARAMETERS SUCH AS LOOPING ON ERROR. ALL FLAGS ARE CLEARED AT STARTUP AND REMAIN CLEARED UNTIL EXPLICITLY SET USING THE FLAGS SWITCH. FLAGS ARE ALSO CLEARED AFTER A START COMMAND UNLESS SET USING THE FLAG SWITCH. THE ZFLAGS COMMAND MAY ALSO BE USED TO CLEAR ALL FLAGS. WITH THE EXCEPTION OF THE START AND ZFLAGS COMMANDS, NO COMMANDS AFFECT THE STATE OF THE FLAGS; THEY REMAIN SET OR CLEARED AS SPECIFIED BY THE LAST FLAG SWITCH.

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIMc SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBE*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXE*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	"BELL" ON ERROR

UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)
ISR	INHIBIT STATISTICAL REPORTS (DOES NOT APPLY TO DIAGNOSTICS WHICH DO NOT SUPPORT STATISTICAL REPORTING)
IDU	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST
EVL	EXECUTE EVALUATION (ON DIAGNOSTICS WHICH HAVE EVALUATION SUPPORT)

*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

SEE THE XXDP+ USER'S MANUAL FOR MORE DETAILS ON FLAGS. YOU MAY SPECIFY MORE THAN ONE FLAG WITH THE FLAG SWITCH. FOR EXAMPLE, TO CAUSE THE PROGRAM TO LOOP ON ERROR, INHIBIT ERROR REPORTS AND TYPE A "BELL" ON ERROR, YOU MAY USE THE FOLLOWING STRING:

```
/FLAGS:LOE:IER:BOE
```

2.4 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?" YOU MUST ANSWER "Y" AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN "PRELOADED" USING THE SETUP UTILITY (SEE CHAPTER 6 OF THE XXDP+ USER'S MANUAL). WHEN YOU ANSWER THIS QUESTION WITH A "Y", THE RUNTIME SERVICES WILL ASK FOR THE NUMBER OF UNITS (IN DECIMAL). YOU WILL THEN BE ASKED THE FOLLOWING QUESTIONS FOR EACH UNIT.

PARAMETER CODING CALLS

```
GPRMA   MSGH1,0,0,0,177777,YES          ;RH ADDRESS
GPRMD   MSGH2,2,0,77,11,70,YES          ;RH TYPE
GPRMD   MSGH3,4,0,777,0,777,YES         ;RH VECTOR ADDRESS
GPRMD   MSGH4,6,D,77,1,16.,YES          ;NUMBER OF ARRAYS
GPRML   MSGH5,10,1,YES                  ;DRIVE OPTION TYPE
GPRMD   MSGH6,12,0,7,0,7,YES            ;DRIVE NUMBER
GPRML   MSGH7,14,1,YES                  ;PARITY DISABLED
```

PARAMETER CODING MESSAGES

```
MSGH1:  .ASCIZ  /RH ADDRESS?/
MSGH2:  .ASCIZ  /IS RH AN '70' OR '11?/
MSGH3:  .ASCIZ  /RH VECTOR ADDRESS?/
MSGH4:  .ASCIZ  /NUMBER OF ARRAY MODULES?/
MSGH5:  .ASCIZ  /IS DRIVE OPTION AN ML11A?/
MSGH6:  .ASCIZ  /ML-11 DRIVE NUMBER?/
MSGH7:  .ASCIZ  /IS PARITY DISABLED?/
```

SAMPLE DIALOGE

```
DR> S.A  <CR>
```

```
CHANGE HW <L>  ?  Y  <CR>

# UNITS <D>  ? 1  <CR>

UNIT 0
RH ADDRESS <O> 176400  ?  <CR>
IS RH AN '70' OR '11' <O>  ?  <CR>
RH VECTOR ADDRESS <O>  204  ?  <CR>
NUMBER OF ARRAYS MODULES  ? <D>  16  ?  14  <CR>
IS DRIVE OPTION AN ML11A ? <L>  Y  ?  <CR>
ML-11 DRIVE NUMBER ? <O>  0  ?  <CR>
IS PARITY DISABLED ? <L>  N  ?  <CR>
```

2.5 SOFTWARE QUESTIONS

AFTER YOU HAVE ANSWERED THE HARDWARE QUESTIONS OR AFTER A RESTART OR CONTINUE COMMAND, THE RUNTIME SERVICES WILL ASK FOR SOFTWARE PARAMETERS. THESE PARAMETERS WILL GOVERN SOME DIAGNOSTIC SPECIFIC OPERATION MODES. YOU WILL BE PROMPTED BY "CHANGE SW (L) ?" IF YOU WISH TO CHANGE ANY PARAMETERS, ANSWER BY TYPING "Y".

THE FOLLOWING OPTIONS ARE AVAILABLE FOR OPERATOR SELECTION:

1. ENABLE PRINTING OF DRIVE SERIAL NUMBERS.
2. LIMIT THE NUMBER OF ERRORS PRINTED PER EACH DETECTED ERROR. A MAXIMUM OF 10 ERRORS WILL BE PRINTED PER ERROR BEFORE THE ERROR IS EXITED.
3. ENABLE RH AND ML11 REGISTER DUMP ON ERRORS.
4. ABORTING PROGRAM EXECUTION AFTER ONE COMPLETE PASS.
5. ENABLE MANUAL INTERVENTION TESTS.

2.6 EXTENDED P-TABLE DIALOGUE

WHEN YOU ANSWER THE HARDWARE QUESTIONS, YOU ARE BUILDING ENTRIES IN A TABLE THAT DESCRIBES THE DEVICES UNDER TEST. THE SIMPLEST WAY TO BUILD THIS TABLE IS TO ANSWER ALL QUESTIONS FOR EACH UNIT TO BE TESTED. IF YOU HAVE A MULTIPLEXED DEVICE SUCH AS A MASS STORAGE CONTROLLER WITH SEVERAL DRIVES OR A COMMUNICATION DEVICE WITH SEVERAL LINES, THIS BECOMES TEDIOUS SINCE MOST OF THE ANSWERS ARE REPETITIOUS.

TO ILLUSTRATE A MORE EFFICIENT METHOD, SUPPOSE YOU ARE TESTING A FICTIONAL DEVICE, THE XY11. SUPPOSE THIS DEVICE CONSISTS OF A CONTROL MODULE WITH EIGHT UNITS (SUB-DEVICES) ATTACHED TO IT. THESE UNITS ARE DESCRIBED BY THE OCTAL NUMBERS 0 THROUGH 7. THERE IS ONE HARDWARE PARAMETER THAT CAN VARY AMONG UNITS CALLED THE Q-FACTOR. THIS Q-FACTOR MAY BE 0 OR 1. BELOW IS A SIMPLE WAY TO BUILD A TABLE FOR ONE XY11 WITH EIGHT UNITS.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  0<CR>
Q-FACTOR (O) 0 ?  1<CR>

UNIT 2
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  1<CR>
Q-FACTOR (O) 1 ?  0<CR>

UNIT 3
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  2<CR>
Q-FACTOR (O) 0 ?  <CR>

UNIT 4
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  3<CR>
Q-FACTOR (O) 0 ?  <CR>

UNIT 5
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  4<CR>
Q-FACTOR (O) 0 ?  <CR>

UNIT 6
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  5<CR>
Q-FACTOR (O) 0 ?  <CR>

UNIT 7
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  6<CR>
Q-FACTOR (O) 0 ?  1<CR>

UNIT 8
CSR ADDRESS (O)  160000<CR>
SUB-DEVICE # (O) ?  7<CR>
Q-FACTOR (O) 1 ?  <CR>
```

NOTICE THAT THE DEFAULT VALUE FOR THE Q-FACTOR CHANGES WHEN A NON-DEFAULT RESPONSE IS GIVEN. BE CAREFUL WHEN SPECIFYING MULTIPLE UNITS!

AS YOU CAN SEE FROM THE ABOVE EXAMPLE, THE HARDWARE PARAMETERS DO NOT VARY SIGNIFICANTLY FROM UNIT TO UNIT. THE PROCEDURE SHOWN IS NOT VERY EFFICIENT.

THE RUNTIME SERVICES CAN TAKE MULTIPLE UNIT SPECIFICATIONS HOWEVER. LET'S BUILD THE SAME TABLE USING THE MULTIPLE SPECIFICATION FEATURE.

```
        # UNITS (D) ?  8<CR>
```

```
UNIT 1
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  0,1<CR>
Q-FACTOR (O) 0 ?  1,0<CR>

UNIT 3
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  2-5<CR>
Q-FACTOR (O) 0 ?  0<CR>

UNIT 7
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  6,7<CR>
Q-FACTOR (O) 0 ?  1<CR>
```

AS YOU CAN SEE IN THE ABOVE DIALOGUE, THE RUNTIME SERVICES WILL BUILD AS MANY ENTRIES AS IT CAN WITH THE INFORMATION GIVEN IN ANY ONE PASS THROUGH THE QUESTIONS. IN THE FIRST PASS, TWO ENTRIES ARE BUILT SINCE TWO SUB-DEVICES AND Q-FACTORS WERE SPECIFIED. THE SERVICES ASSUME THAT THE CSR ADDRESS IS 160000 FOR BOTH SINCE IT WAS SPECIFIED ONLY ONCE. IN THE SECOND PASS, FOUR ENTRIES WERE BUILT. THIS IS BECAUSE FOUR SUB-DEVICES WERE SPECIFIED. THE "-" CONSTRUCT TELLS THE RUNTIME SERVICES TO INCREMENT THE DATA FROM THE FIRST NUMBER TO THE SECOND. IN THIS CASE, SUB-DEVICES 2, 3, 4 AND 5 WERE SPECIFIED. (IF THE SUB-DEVICE WERE SPECIFIED BY ADDRESSES, THE INCREMENT WOULD BE BY 2 SINCE ADDRESSES MUST BE ON AN EVEN BOUNDARY.) THE CSR ADDRESSES AND Q-FACTORS FOR THE FOUR ENTRIES ARE ASSUMED TO BE 160000 AND 0 RESPECTIVELY SINCE THEY WERE ONLY SPECIFIED ONCE. THE LAST TWO UNITS ARE SPECIFIED IN THE THIRD PASS.

THE WHOLE PROCESS COULD HAVE BEEN ACCOMPLISHED IN ONE PASS AS SHOWN BELOW.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  0-7<CR>
Q-FACTOR (O) 0 ?  0,1,0,...,1,1<CR>
```

AS YOU CAN SEE FROM THIS EXAMPLE, NULL REPLIES (COMMAS ENCLOSING A NULL FIELD) TELL THE RUNTIME SERVICES TO REPEAT THE LAST REPLY.

2.7 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

1. BOOT XXDP+
2. GIVE THE DATE
3. TYPE "R CZMLA"
4. TYPE "START"
5. ANSWER THE "CHANGE HW" QUESTION WITH "Y"

6. ANSWER ALL THE HARDWARE QUESTIONS

7. ANSWER THE ''CHANGE SW'' QUESTION WITH 'N''

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE DEFAULTS FOR FLAGS AND SOFTWARE PARAMETERS. THESE DEFAULTS ARE DESCRIBED IN SECTIONS 2.3 AND 2.5.

3.0 ERROR INFORMATION

3.1 TYPES OF ERROR MESSAGES

THERE ARE THREE LEVELS OF ERROR MESSAGES THAT MAY BE ISSUED BY A DIAGNOSTIC: GENERAL, BASIC AND EXTENDED. GENERAL ERROR MESSAGES ARE ALWAYS PRINTED UNLESS THE ''IER'' FLAG IS SET (SECTION 2.3). THE GENERAL ERROR MESSAGE IS OF THE FORM:

```
NAME  TYPE  NUMBER  ON UNIT NUMBER  TST NUMBER PC:XXXXXX
ERROR MESSAGE
```

,WHERE: NAME = DIAGNOSTIC NAME
TYPE = ERROR TYPE (SYS FATAL, DEV FATAL, HARD OR SOFT)
NUMBER = ERROR NUMBER
UNIT NUMBER = 0 - N (N IS LAST UNIT IN PTABLE)
TST NUMBER = TEST AND SUBTEST WHERE ERROR OCCURRED
PC:XXXXXX = ADDRESS OF ERROR MESSAGE CALL

BASIC ERROR MESSAGES ARE MESSAGES THAT CONTAIN SOME ADDITIONAL INFORMATION ABOUT THE ERROR. THESE ARE ALWAYS PRINTED UNLESS THE ''IER'' OR ''IBR'' FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL MESSAGE.

EXTENDED ERROR MESSAGES CONTAIN SUPPLEMENTARY ERROR INFORMATION SUCH AS REGISTER CONTENTS OR GOOD/BAD DATA. THESE ARE ALWAYS PRINTED UNLESS THE ''IER'', ''IBR'' OR ''IXR'' FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

3.2 SPECIFIC ERROR MESSAGES

ERROR MESSAGES ARE HANDLED VIA A DICTIONARY STRUCTURE.

WORDS AND PHRASES ARE MULTIPLY REFERENCED USING ONLY ONE COPY OF THE WORD OR PHRASE IN CORE.

THIS PERMITS EXTENSIVE ERROR MESSAGE PRINTING AT MINIMAL STORAGE REQUIREMENTS.

THE FOLLOWING DEMONSTRATES TYPICAL ERROR MESSAGES:

```
ML11 DVC FTL ERR 00077 ON UNIT 07 TST 027 SUB 002 PC: 050432
ASYNCHRONOUS MODULE FAILURE
EXCESSIVE DATA ERRORS DURING INITIAL ARR RD_WRT

ML11 DVC FTL ERR  00112 ON UNIT 03 TST 037 SUB 000 PC: 056466
ASYNCHRONOUS MODULE FAILURE
```

ARRAY ADRS MULTIPLEXER FAILURE
FAILED AT DSA: 000000

3.2.1 ERROR NUMBER DEFINITION

ERROR NO.	FAILING LOGIC
1	DRIVE DID NOT RESPOND WITHIN 1.5 US
2	DSA REG READ/WRITE ERROR DURING DRIVE SEL TEST
3	UNIQUE DRIVE SELECTION ERROR
4	ML REGISTER READ WRITE ONES/ZEROES ERROR
5	ML REGISTER SHIFTING ONES/ZEROES ERROR
6	ML REGISTER INITIALIZATION ERROR
7	CONTROL BUS BAD PARITY NOT DETECTED. (CPAR NOT SET)
8	CONTROL BUS GOOD PARITY NOT DETECTED. (CPAR SET)
9	CONTROL BUS BAD PARITY GENERATED. (MCPE SET)
10	ARRAY SIZING LOGIC ERROR
11	GO BIT NOT CLR AFTER NOOP FUNCTION
12	ILF BIT SET DURING NOOP FUNCTION
13	OPI BIT SET DURING NOOP FUNCTION
14	GO BIT NOT SET DURING WRITE CHECK FUNCTION
15	DRY BIT CLEARED WHILE GO CLEARED DURING WRITE CHECK FUNCTION TEST
16	DRY BIT SET WHEN GO SET DURING WRITE CHECK FUNCTION
17	ILF SET DURING WRITE CHECK FUNCTION
18	OPI BIT SET DURING WRITE CHECK FUNCTION
19	GO BIT NOT CLEAR AFTER WRITE CHECK FUNCTION 'ASYNC FAILURE'
20	GO BIT NOT CLEAR AFTER WRITE CHECK FUNCTION 'SYNC FUNCTION'
21	DRY BIT NOT SET AFTER WRITE CHECK FUNCTION
22	GO BIT NOT CLEAR AFTER WRITE CHECK FUNCTION 'ASYNC FAILURE'
23	GO BIT NOT CLEAR AFTER WRITE CHECK FUNCTION 'SYNC FAILURE'
24	GO BIT NOT SET DURING WRITE FUNCTION

```
                25              DRY BIT CLEAR WITH GO CLEAR DURING WRITE FUNCTION.
                26              DRY BIT SET WITH GO BIT DURING WRITE FUNCTION.
                27              ILF BIT SET DURING WRITE FUNCTION
                28              OPI BIT SET DURING WRITE FUNCTION
                29              GO BIT NOT CLEAR AFTER WRITE FUNCTION 'ASYNC FAILURE'
                30              GO BIT NOT CLEAR AFTER WRITE FUNCTION 'SYNC FAILURE'
                31              DRY BIT NOT SET AFTER WRITE FUNCTION.
                32              GO BIT NOT CLEAR AFTER WRITE FUNCTION 'ASYNC FAILURE'
                33              GO BIT NOT CLEAR AFTER WRITE FUNCTION 'ASYNC FAILURE'
                34              GO BIT NOT SET DURING READ FUNCTION.
                35              DRY BIT SET WHILE GO CLEAR DURING READ FUNCTION.
                36              DRY BIT SET WHILE GO SET DURING READ FUNCTION.
                37              ILF BIT SET DURING READ FUNCTION.
                38              OPI BIT SET DURING READ FUNCTION
                39              GO BIT NOT CLEAR AFTER READ FUNCTION 'ASYNC FAILURE'
                40              GO BIT NOT CLEAR AFTER READ FUNCTION 'SYNC FAILURE'
                41              DRY BIT NOT SET AFTER READ FUNCTION
                42              GO BIT NOT CLEAR AFTER READ FUNCTION 'ASYNC FAILURE'
                43              GO BIT NOT CLEAR AFTER READ FUNCTION 'SYNC FAILURE'
                44              GO BIT NOT CLEAR AFTER CLEAR FUNCTION
                45              DRY BIT SET WHILE GO SET DURING CLEAR FUNCTION
                46              DRY BIT NOT SET AFTER CLEAR FUNCTION .
                47              ILF BIT SET DURING CLEAR FUNCTION
                48              OPI BIT SET DURING CLEAR FUNCTION
                49              ERROR REGISTER DID NOT CLEAR AFTER DRIVE CLEAR.
                50              COMPOSITE ERROR BIT NOT SET AFTER MLER BITS SET
                51              ATA BIT SETTING ERROR
                52              ATTN BIT SETTING ERROR
                53              ATTN BIT NOT CLEARED BY MLCS1 NOOP FUNCTION
```

```
54          ATA BIT NOT CLEAR AFTER WRITING A ONE TO IT
55          WRITING A ONE TO OTHER DRIVES ATA BIT CLEARED THIS DRIVES ATA BIT
56          GO BIT NOT CLEARED AFTER SEARCH FUNCTION
57          ILF BIT SET DURING SEARCH FUNCTION
58          NO-OP FUNCTION DID NOT CLEAR THE ATA BIT AFTER BEING SET
59          ATTN BIT NOT SET AFTER SEARCH FUNCTION
60          OPI BIT SET DURING SEARCH FUNCTION AT PRESENT ARRAYS
61          OPI BIT NOT SET AFTER SEARCH AT NON-PRESENT ARRAYS
62          GO BIT NOT CLEAR AFTER READ-IN-PRESET FUNCTION
63          ILF BIT SET DURING READ-IN-PRESET FUNCTION
64          OPI BIT SET DURING READ-IN-PRESET FUNCTION
65          ECC INITIALIZE DID NOT CLEAR MEMORY AFTER POWER UP WITH BAD BATTERY BACK-UP
66          GO BIT NOT CLEARED AFTER ILLEGAL FUNCTION
67          ILLEGAL FUNCTION NOT DETECTED
68          OPI BIT SET WITH ILLEGAL FUNCTION
69          RMR BIT NOT SET AFTER MODIFYING REG WITH FUNCTION IN PROGRESS
70          MEMORY ARRAY PROM CHECK SUM ERRORS DURING INITIAL PROM READS
71          NIBBLE OFF SET COUNTS GREATER THAN 14 DETECTED
72          UNS BIT SET WITH GOOD UV DATA
73          UNS BIT SET WITH GOOD UV DATA
74          UNS BIT NOT SET WITH BAD UV DATA
75          UNS BIT NOT SET WITH BAD UV DATA
76          MEMORY ARRAY PROM ROW/COL DATA ORING ERROR
77          BAD NIBBLE THRESHOLD OF 36 EXCEEDED DURING INITIAL ARRAY READ/WRITE TEST
78          UNIQUE PROM SELECTION ERROR
79          FAILURE TO FIND GOOD ROW DURING READ WRITE ARRAY WITH PROM DATA
80          MEMORY ARRAY TIMING AND CONTROL FAILURE TO REFRESH MEMORY
81          DATA ERRORS DETECTED AT LAST BLOCK DURING ADDRESSES COUNTER TEST.  (TEST ABORTED)
```

82	ADDRESS COUNTER ERROR
83	UNIQUE MEMORY ARRAY MODULE SELECTION FAILURE
84	ALL BITS IN ALL NIBBLES TESTED DURING SEQUENCER EXISTENCE TEST WERE IN ERROR (FAIL
85	INTERMEDIATE FAILURE. SOME BITS IN NIBBLES TESTED WERE IN ERROR (CONTINUE TESTING)
86	SYNC BUS DATA BIT WRITE PATH CONTINUITY FAILURE
87	SYNC BUS DATA BIT READ PATH CONTINUITY FAILURE
88	RAM BUS ADRS COUNTER FAILURE TO LOAD/UNLOAD SKIP RAM DURING WRITE FUNCTION
89	RAM BUS ADRS COUNTER FAILURE TO LOAD/UNLOAD SKIP RAM DURING READ FUNCTION
90	SYNC DATA BUS WRITE PATH UNIQUE DATA BIT FAILURE (ALL ONES NIBBLE PATTERN)
91	SYNC DATA BUS WRITE PATH UNIQUE DATA BIT FAILURE (SHIFTED BIT NIBBLE PATTERN)
92	SYNC DATA BUS WRITE PATH UNIQUE DATA BIT FAILURE (ALL ONES NIBBLE PATTERN)
93	SYNC DATA BUS READ PATH UNIQUE DATA BIT FAILURE
94	NIBBLE OFF SET COUNTERS FAILURE
95	CS1 FUNCTION ABORT FAILURE DURING CLASS 'A' ERROR
96	CS1 FUNCTION ABORT FAILURE DURING CLASS 'B' ERROR
97	LBT BIT SET BEFORE A LAST BLOCK TRANSFER
98	DSA REGISTER INCREMENT FAILURE DURING NON LAST BLOCK TRANSFERS.
99	LBT BIT NOT CLEAR AFTER LOADING DSA REG
100	LBT BIT NOT SET AFTER A LAST BLOCK TRANSFER
101	DSA REGISTER INCREMENT FAILURE AFTER A LAST BLOCK TRANSFER
102	IAE BIT NOT SET AT INVALID SECTOR ADDRESSES
103	AOE BIT NOT SET AFTER ADDRESS OVERFLOW
104	SC BIT NOT AFTER CS1 FUNCTION ADORT
105	GOOD DATA BUS PARITY NOT DETECTED
106	GOOD DATA BUS PARITY NOT GENERATED
107	UNS BIT SET AFTER WRITING TO A SECTOR DURING PROM DATA TEST
108	UNS BIT NOT SET WITH BAD UV DATA
109	WCE BIT SET DURING MBUS WRITE/READ FUNCTION TROUBLE SHOOTING LOOP TEST
110	UNIQUE REGISTER SELECTION TEST FAILURE

111	FAILURE TO FIND GOOD MOS RAM ROW DURING ARRAY ADRS MUX TEST (INTERMEDIATE DIAG MSG)
112	UNIQUE ARRAY MODULE ROW/COL ADDRESSING FAILURE
113	DRIVE TYPE REGISTER VALUE WAS NOT CORRECT
114	TRE BIT SET UNEXPECTEDLY DURING A WRITE CHECK TRANSFER (INTERMEDIATE DIAG ERROR)
115	TRE BIT SET UNEXPECTEDLY DURING A WRITE TRANSFER (INTERMEDIATE DIAG ERROR)
116	TRE BIT SET UNEXPECTEDLY DURING A READ TRANSFER (INTERMEDIATE DIAG ERROR)
117	TRE BIT DID NOT SET AFTER A REGISTER MODIFICATION ERROR (EXCEPTION WAS NOT ASSERTED)
118	DATA DIAGNOSTIC REGISTER (D1 D2 E2) INITIALIZATION ERRORS
119	RH CONTROLLER FAILED EXISTANCE PROBE TEST
120	NED BIT SET DURING MASS BUS HANDSHAKE PROBE
121	DATA DIAGNOSTIC REGISTER (D1 D2 E2) ONE'S ZERO'S READ/WRITE ERRORS
122	DATA DIAGNOSTIC REGISTER (D1 D2 E2) SHIFTING 1'S AND 0'S READ/WRITE ERRORS
123	ECC HARD ERROR BIT NOT SET WHEN UNCORRECTABLE ECC ERRORS WERE READ
124	ECC ERROR REGISTER FAILURE TO LATCH FAILING ERROR CORRECTION INFORMATION
125	ECC ERROR REGISTER FAILURE TO CLEAR
126	ECC ERROR LOCATION REGISTER FAILED TO CLOCK IN DSA ADDRESS
127	ECC ERROR LOCATION REGISTER DATA BIT ERRORS
128	ECC ERROR LOCATION REGISTER INITIALIZATION ERRORS
129	ECC HARD ERROR BIT FAILED TO SET NOT SET
130	DATA CHECK ERROR BIT FAILED TO SET / NOT SET
131	CRC DATA BUS FAILURE. BIT CONTINUITY / UNIQUENESS ERRORS
132	UNIQUE NIBBLE CRC GENERATION CODE FAILURES
133	UNIQUE WORD CRC GENERATION CODE FAILURES
134	CORRECTABLE ERROR SYNDRONE DECODE FAILURE. FAILURE TO COMPLIMENT EXPECTED FAILING B
135	CORRECTABLE ERROR SYNDRONE DECODE FAILURE. UNEXPECTED BIT(S) FOUND COMPLIMENTED
136	ECH OR UNC BIT NOT SET DURING READS WITH ECC ERRORS IN CHANNEL > 35
137	ERROR CORRECTION WAS NOT INHIBITED DURING UNCORRECTABLE ECC ERRORS. UNEXPECTED BIT(S) WERE FOUND COMPLIMENTED
138	ECH OR UNC BIT NOT SET WHEN MULTIPLE CHANNEL ERRORS WERE READ

139 ERROR CORRECTION WAS NOT INHIBITED DURING UNCORRECTABLE ECC ERROR.
UNEXPECTED BIT(S) WERE FOUND COMPLIMENTED

140 ECC FAILURE TO DETECT AND CORRECT SINGLE BIT CHANNEL ERRORS

141 ECC FAILURE TO DETECT AND CORRECT MULTIPLE BIT CHANNEL ERRORS

142 FAILURE TO FIND GOOD BLOCK DURING ARRAY SELECT TEST

148 SC BIT SET DURING MASS BUS WRITE OR WRITE CHECK TRANSFERS DURING TEST 21 AND 61

149 VV BIT WAS SET WHEN TESTED AND GOT CLEARED AFTER A READ-IN-PRESET FUNCTION WAS PERFO

150 VV BIT WAS CLEARED WHEN TESTED AND DID NOT GET SET WHEN A READ-IN-PRESET FUNCTION WA

151 THE UNS BIT DID NOT SET WHEN BOX POWER WAS LOST

152 THE VV BIT DID NOT CLEAR WHEN DRIVE AC POWER RESUMED WITH BAD BATTERY BACK-UP

153 DC LO DID NOT INITIATE 'MB DIS LO' RESULTING IN THE ML-11 REGISTERS NOT GETTING CLEA
POWER UP.

154 VV BIT GOT CLEARED AFTER POWER FAIL WITH GOOD BATTERY BACK-UP

155 ECC INIT CLEARED OUT MEMORY AFTER POWER FAILED WITH GOOD BATTERY BACK-UP

156 BATTERY BACK-UP FAILED TO MAINTAIN DATA INTEGRITY DURING POWER FAIL

4.0 PERFORMANCE AND PROGRESS REPORTS

AT THE END OF EACH PASS, THE PASS COUNT IS GIVEN ALONG WITH THE
TOTAL NUMBER OF ERRORS REPORTED SINCE THE DIAGNOSTIC WAS STARTED.
THE "EOP" SWITCH CAN BE USED TO CONTROL HOW OFTEN THE END
OF PASS MESSAGE IS PRINTED. SECTION 2.2 DESCRIBES SWITCHES.

5.0 DEVICE INFORMATION TABLES

HARDWARE DEFAULT PTABLE

```
.WORD   176400          ;RH ADDRESS
.WORD   70              ;RH TYPE
.WORD   204             ;RH VECTOR ADDRESS
.WORD   16.             ;NUMBER OF ARRAY MODULES
.WORD   1               ;IS DRVE OPTION ML11A,  1=16K,  0=64K
.WORD   0               ;ML-11 DRIVE NUMBER
.WORD   0               ;IS PARITY DISABLED,  1=YES,  0=NO
```

6.0 TEST SUMMARIES

TST1. MASS BUS READY

TEST THE RH CONTROLLER FOR EXISTANCE

TST2. MASS BUS HANDSHAKE

TEST MASS BUS ---> UNIBUS COMMUNICATIONS

TST3. DRIVE PRESENT

TEST TO SEE IF THE DRIVE UNDER TEST EXIST.

TST4. DRIVE SELECTION

SEE IF SELECTING OTHER DRIVES ON RH EFFECTS DRIVE UNDER TEST.

TST5. READ WRITE REGISTER ONE'S ZERO'S TEST

TEST REGISTERS READ WRITE CAPABILITY AND UNIQUENESS

TST6. READ WRITE REGISTER SHIFTING ONE'S AND ZEROES

TEST REGISTERS FOR UNIQUE DATA BITS.

TST7. REGISTER INITIALIZATION

TEST REGISTERS FOR CORRECT INIT DATA.

TST8 REGISTER SELECTION TEST

TEST FOR UNIQUE REGISTER SELECTION

TST9. PRINT DRIVE SERIAL NUMBER

PRINT THE CONTENTS OF MLSN IF THE SOFTWARE QUESTION WAS ANSWERED 'YES'.

TST10. C-BUS PARITY

TEST IF DRIVE CAN DETECT BAD PARITY ON C-BUS AND GENERATE GOOD PARITY.

TST11. MEMORY SIZING

SEE IF MEMORY SIZING LOGIC DETECTS AND RECORD CORRECT NUMBER OF ARRAYS PRESENT.

TST12. NO-OP FUNCTION

SEE IF A NO-OP FUNCTION CAN BE EXECUTED.

TST13. WRITE CHECK FUNCTION

SEE IF A WRITE CHECK FUNCTION CAUSES THE DRIVE TO HANG.

TST14. WRITE FUNCTION

SEE IF A WRITE FUNCTION CAUSES THE DRIVE TO HANG.

TST15. READ FUNCTION

SEE IF A READ FUNCTION CAUSES THE DRIVE TO HANG.

TST16. CLEAR FUNCTION

SEE IF A CLEAR FUNCTION CAN BE EXECUTED.

TST17. DIAGNOSTIC REGISTER READ WRITE

TEST THE DIAGNOSTIC REGISTERS FOR 1'S/0'S , SHIFTING 1'S/0'S AND INITIALIZATION

TST18. COMPOSITE ERROR BIT TEST

SEE IF EACH INDIVIDUAL ERROR BIT IN MLER CAUSES A COMPOSITE ERROR.

TST19. ATA BIT

TEST IF THE ATA BIT CAN BE SET AND CLEARED.

TST20. SEARCH FUNCTION

SEE IF A SEARCH FUNCTION CAN BE EXECUTED ON ALL PRESENT ARRAYS.

TST21. POWER FAIL TEST *MANUAL INTERVENTION TEST*
TEST THE VV BIT SETTING/CLEARING, ECC INIT, REGISTER CLEARING UNS BIT SETTING, AC AND DC LOW.

TST22. ILLEGAL FUNCTION

SEE IF WRITING AN ILLEGAL FUNCTION TO CS1 CAN BE DETECTED AND THAT A TRANSFER IS NOT INITIATED.

TST23. REGISTER MODIFICATION REFUSED

TEST TO SEE IF WRITING TO SPECIFIC REGISTERS ARE ABORTED WHILE THE DRIVE IS ACTIVE. SEE IF WRITING TO NON-SPECIFIC REGISTERS ARE ALLOWED WHILE DRIVE IS ACTIVE.

TST24. INITIAL PROM TEST

TEST PROMS FOR EXISTENCE.

TST25. PROM 'OR' FUNCTION TEST

TEST THE PROM DATA ORING FUNC

TST26. UV ERROR TEST

TEST ABILITY OF UV ERR PROMS TO DETECT ALL POSSIBLE CHECK SUM ERRORS.

TST27. INITIAL ARRAY TEST

TEST ARRAY TIMING AND CONTROLFOR EXISTENCE.

TST28. PROM SELECTION TEST

TEST FOR UNIQUE PROM SELECTION.

TST29. READ WRITE MEMORY ARRAY WITH PROM DATA (DIAG MODE)

SEE IF MEMORY CAN BE WRITTEN AND READ.

ALSO FIND ERROR FREE BLOCK OF MEMORY FOR FUTURE TESTS.

TST30. REFRESH TIMING

TEST TO SEE IF MEMORY CAN BE REFRESHED.

TST31. ADDRESS COUNTER

TEST THE ADDRESS COUNTER FOR ABILITY TO COUNT THROUGH ALL POSSIBLE MEMORY ADDRESSES.

TST32. ARRAY MODULE SELECTION

TEST FOR UNIQUE ARRAY MODULE SELECTION

TST33. SEQUENCER EXISTENCE TEST

TEST TO SEE IF BASIC SEQUENCER TIMING EXISTS.

TST34. SYNC DATA BUS CONTINUITY/WRITE PATH

TEST SYNCHRONOUS DATA BUS WRITE PATH FOR CONTINUITY BY READ-ING WRITING ONE'S AND ZERO'S.

TST35. SYNC DATA BUS CONTINUITY/READ PATH

TEST SYNCHRONOUS DATA BUS READ PATH FOR CONTINUITY BY READING WRITING ONE'S AND ZEROES.

TST36. RAM-BUS ADDRESS COUNTER/WRITE PATH

TEST ABILITY OF THE RAM-BUS ADDRESS COUNTERS TO LOAD/UNLOAD THE SKIP DURING WRITE FUNCTIONS.

TST37. RAM BUS ADRS COUNTER/READ PATH

TEST ABILITY OF RAM/BUS ADRS COUNTERS TO LOAD/UNLOAD THE SKIP RAM DURING READ FUNCTIONS.

TST38. SYNC DATA BUS BIT UNIQUENESS/WRITE PATH

TEST SYNCHRONOUS DATA BUS FOR DATA BIT UNIQUENESS BY WRITING SHIFTING PATTERNS OF ONE'S AND ZERO'S TO THE ML.

TST39. SYNC DAT BUS BIT UNIQUENESS/READ PATH

TEST SYNCHRONOUS DATA BUS READ PATH FOR DATA BIT UNIQUENESS

BY WRITING SHIFTING PATTERNS OF ONES AND ZEROES TO THE

TST40. ARRAY ADDRESS MUX

TEST FOR UNIQUE ROW AND COLUMN ADDRESSING

TST41. NIBBLE OFFSET

TEST NIBBLE OFFSET COUNTERS TO COUNT TO 14 NIBBLE DATA TO BE SHIFTED ON DETECTION OF BAD NIBBLES.

TST42. CS1 FUNCTION ABORT

SEE IF A CLASS 'B' ERROR ABORTS A FUNCTION WHILE IN PROGRESS.

SEE IF A CLASS'A' ERROR IS DETECTED BUT FUNCTION IS ALLOWED TO COMPLETE.

TST43. LAST BLOCK INDICATOR

TEST THE LAST BLOCK INDICATOR BIT FOR NOT SETTING BELOW THE LAST AND SETTING AND CLRING AT THE LAST BLOCK

TST44. INVALID ADDRESS TEST

FOR ALL ILLEGAL DSA ADDRESSES READ THE IAE BIT SET.

TST45. ADDRESS OVERFLOW

TEST FOR AOE ON TRANSFERS WHICH EXTEND BEYOND THE LAST BLOCK.

TST46. SYNC BUS PARITY

TEST FOR BAD PARITY DETECTION AND GOOD PARITY GENERATION.

TST47. WRITE READ MEMORY ARRAY (M-BUS BLOCK MODE)

WRITE READ MEMORY VIA M-BUS BLOCK WITH MINIMUM OVERHEAD

TST48. TEST THE CRC DATA BUS BETWEEN THE CRC GENERATORS AND THE CRC/MBUS DATA MUX FOR CONTINUITY AND BIT UNIQUENESS

TST49. TEST CRC CODES GENERATED FOR ONE CRC GROUP (52 UNIQUE NIBBLES)

TST50. TEST CRC CODES GENERATED FOR ONE CRC GROUP (13 UNIBUS WORDS)

TST51. TEST SYNDRONE DECODE AND ERROR CORRECTION TO DECODE AND CORRECT SINGLE BIT AND MULTIPLE BIT CHANNEL ERRORS

TST52. TEST SYNDRONE DECODE TO DETECT BUT NOT CORRECT UNCORRECTABLE CHANNEL

ERRORS

TST53. TEST SYNDRONE DECODE TO DETECT BUT NOT CORRECT UNCORRECTABLE MULTIPLE CHANNEL ERRORS

TST54. TEST SYNDRONE GENERATION, SYNDRONE DECODE AND ERROR CORRECTION FOR SIGLE BIT CHANNEL ERRORS

TST55. TEST SYNDRONE GENERATION, SYNDRONE DECODE AND ERROR CORRECTION FOR MULTIPLE BIT CHANNEL ERRORS

TST56. TEST THE ECC ERROR REGISTER FOR CLEARING AND LATCHING OF ECC ERROR INFORMATION ON DETECTION OF ECC ERRORS

TST57. TEST THE ECC ERROR LOCATION REGISTER FOR CLOCKING, BIT UNIQUENESS, CLEARING AND LATCHING

TST58. VIA ECC_DM AND ECC_DIS TEST THE ECH BIT FOR SETTING AND NOT SETTING

TST59. VIA ECC_EN, ECC_DM AND ECC_DIS TEST THE DCK BIT FOR SETTING AND NOT SETTING

TST60. PROM DATA TEST

VERIFY THAT CHECK SUM VALUES FOR ALL PROM LOCATIONS ARE CORRECT.

TST61. BATTERY BACK-UP TEST *MANUAL INTERVENTION TEST*
TEST THE DRIVE BATTERY BACK-UP DURING POWER FAILS AND ALSO TEST VV BIT, ECC INIT.

```
                                .SBTTL  PROGRAM HEADER AND TABLES

000000                                  .ENABL  ABS,AMA
002000                          .       =       2000

002000                                  BGNMOD

                                ;++
                                ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
                                ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
                                ;--

002000                                  POINTER ALL

002000                                  HEADER  CZMLA,D,0,1800.,0

                                ;
                                ;       NAMES OF DEVICES SUPPORTED BY THIS PROGRAM
                                ;
002122                                  DEVTYP <ML-11>


                                ;
                                ;       TEST DESCRIPTION
                                ;
002130                                  DESCRIPT <ML-11 LOGIC TEST>


                                ;
                                ;       THE GLOBAL ERROR TABLE (INFORMATION
                                ;       USED IN A CALL TO THE MACRO 'ERROR')
                                ;
002152                          ERRTBL
002152  000000                  ERRTYP::        .WORD   0
002154  000000                  ERRNBR::        .WORD   0
002156  000000                  ERRMSG::        .WORD   0
002160  000000                  ERRBLK::        .WORD   0

                                ;++
                                ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRES OF EACH TEST.
                                ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
                                ;--

002162                                  DISPATCH 61


                                ;++
                                ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
                                ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
                                ; IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES,
                                ; AND IS USED AS A "TEMPLATE" FOR BUILDING THE P-TABLES.
                                ;--
```

```
                                        BGNHW   DFPTBL
                                        .WORD   176400          ;RH ADDRESS
                                        .WORD   70              ;RH TYPE
                                        .WORD   204             ;RH VECTOR ADDRESS
                                        .WORD   1               ;NUMBER OF ARRAY MODULES
                                        .WORD   1               ;IS DRIVE OPTION ML11A,  1=16K,  0=64K
                                        .WORD   0               ;ML-11 DRIVE NUMBER
                                        .WORD   0               ;IS PARITY DISABLED,  1=YES,  0=NO
                                        ENDHW


                                ;++
                                ; THE DEFAULT SOFTWARE P-TABLE CONTAINS VARIOUS DATA USED BY THE
                                ; PROGRAM AS OPERATIONAL PARAMETERS.  THESE PARAMETERS ARE SET
                                ; UP AT ASSEMBLY TIME AND MAY BE VARIED BY THE OPERATOR AT RUN
                                ; TIME.
                                ;--

                                        BGNSW   SFPTBL

                                PRSN::          .WORD   0               ;PRINT SERIAL NUMBER, 1=YES,  0=NO
                                ERRTHR::        .WORD   1               ;ENABLE ERROR THRESHOLD 1=YES, 0=NO
                                REGDMP::        .WORD   0               ;ENABLE REGISTER DUMPS 1=YES, 0=NO
                                ONEPAS::        .WORD   1               ;DROP UNIT AFTER ONE (EOP)
                                INTERVEN::      .WORD   0               ;MANUAL INTERVENTION TESTS 1=YES, 0=NO
                                        ENDSW
```

```
                                ;++
                                ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;--

002412                                  BGNHRD

002414                                  GPRMA   MSGH1,0,0,0,177777,YES
002424                                  GPRMD   MSGH2,2,0,77,11,70,YES
002436                                  GPRMD   MSGH3,4,0,777,0,777,YES
002450                                  GPRMD   MSGH4,6,D,77,1,16.,YES
002462                                  GPRML   MSGH5,10,1,YES
002470                                  GPRMD   MSGH6,12,0,7,0,7,YES
002502                                  GPRML   MSGH7,14,1,YES

002510                                  ENDHRD

002510  122  110  040  MSGH1:  .ASCIZ  /RH CONTROLER BASE REGISTER ADDRESS  ?/
002556  122  110  040  MSGH2:  .ASCIZ  /RH CONTROLER TYPE '70  OR  11'      ?/
002624  122  110  040  MSGH3:  .ASCIZ  /RH CONTROLER VECTOR ADDRESS         ?/
002670  116  125  115  MSGH4:  .ASCIZ  /NUMBER OF ARRAY MODULES IN THIS UUT ?/
002736  111  123  040  MSGH5:  .ASCIZ  /IS UUT DRIVE OPTION TYPE AN 'ML11A' ?/
003004  104  122  111  MSGH6:  .ASCIZ  /DRIVE SELECT NUMBER OF THIS UUT     ?/
003052  111  123  040  MSGH7:  .ASCIZ  /IS PARITY DISABLED IN THIS UUT      ?/
                                .EVEN

                                ;++
                                ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;--

003120                                  BGNSFT

003122                                  GPRML   MSGS1,0,1,YES
003130                                  GPRML   MSGS2,2,1,YES
003136                                  GPRML   MSGS3,4,1,YES
003144                                  GPRML   MSGS4,6,1,YES
003152                                  GPRML   MSGS5,10,1,YES
                                        .EVEN

003160                                  ENDSFT
```

```
003160     120     122     111  MSGS1:  .ASCIZ  /PRINT DRIVE SERIAL NUMBER          ?/          ;PRINT DRIVE SERIAL NUMBER?
003224     105     116     101  MSGS2:  .ASCIZ  /ENABLE ERROR MESSAGE THRESHOLD     ?/          ;ERROR THRESHOLD FLAG
003270     105     116     101  MSGS3:  .ASCIZ  /ENABLE REGISTER DUMP ON ERRORS     ?/          ;REG DUMP FLAG
003334     105     130     111  MSGS4:  .ASCIZ  /EXIT PROGRAM AFTER ONE PASS        ?/          ;EXIT PROG FLAG
003400     105     116     101  MSGS5:  .ASCIZ  /ENABLE MANUAL INTERVENTION TESTS  ?/
                                        .EVEN

                                ;++
                                ; THIS TABLE IS USED BY THE RUNTIME SERVICES
                                ; TO PROTECT THE LOAD MEDIA.
                                ;--

003444                                  BGNPROT

003444  177777                          -1              ;OFFSET INTO P-TABLE FOR CSR ADDRESS
003446  177777                          -1              ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
003450  177777                          -1              ;OFFSET INTO P-TABLE FOR DRIVE NUMBER

003452                                  ENDPROT

003452                          $PATCH::
003452                                  .BLKW   20

003512                                  ENDMOD
```

```
;ML3AD                                                    22-Mar-1982 11:35:31    TOPS-20 Bliss-16 V2(212)
;                                                         22-Mar-1982 11:32:50    PA:<NEALE>ML3AD.BLI.1 (1)

;          0001  MODULE ML3AD =
;          0002  BEGIN
;          0003
;          0004  %SBTTL 'REPORT CODE SECTION'
;          0005
;          0006  REQUIRE 'BLSMAC.REQ';
;          1496
;          1497  !+
;          1498  !  THE REPORT CODING SECTION CONTAINS THE
;          1499  ! 'PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
;          1500  !-
;          1501
;          1502  BGNRPT;
;          1503  RETURN;
;          1504  ENDRPT;


                                        .SBTTL  LRPT REPORT CODE SECTION

004646  000207                  LRPT:   RTS     PC                              ;                       1495

                                ; Routine Size:  1 word
                                ; Maximum stack depth per invocation:  0 words


                                        .SBTTL  L$RPT REPORT CODE SECTION

004650  004767  177772          L$RPT:: JSR     PC,LRPT                         ;                       1503
004654  104425                          TRAP    25
004656  000207                          RTS     PC

                                ; Routine Size:  4 words
                                ; Maximum stack depth per invocation:  0 words
```

```
                1505
                1506  !+
                1507  ! THE AUTODROP CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE
                1508  ! CODE IF THE "ADR" FLAG WAS SET.  THE UNIT(S) UNDER TEST ARE
                1509  ! CHECKED TO SEE IF THEY WILL RESPOND.  THOSE THAT DON'T ARE IMMEDIATELY
                1510  ! DROPPED FROM TESTING.  ISSUE A "DODU" FOR THOSE THAT DON'T RESPOND.
                1511  !-
                1512
                1513  BGNAUTO;
                1514  RETURN;
                1515  ENDAUTO;

                                        .SBTTL  LAUTO REPORT CODE SECTION
004660  000207                  LAUTO:  RTS     PC                              ;  1504

                                ; Routine Size:  1 word
                                ; Maximum stack depth per invocation:  0 words


                                        .SBTTL  L$AUTO REPORT CODE SECTION
004662  004767  177772          L$AUTO::JSR     PC,LAUTO                        ;  1514
004666  104461                  TRAP    61
004670  000207                  RTS     PC

                                ; Routine Size:  4 words
                                ; Maximum stack depth per invocation:  0 words
```

```
116 ;ML3AD
117 ;                   REPORT CODE SECTION                    22-Mar-1982 11:35:31    TOPS-20 Bliss-16 V2(212)
118                                                            22-Mar-1982 11:32:50    PA:<NEALE>ML3AD.BLI.1 (1)
119 ;           1516
120 ;           1517   !+
121 ;           1518   ! THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
122 ;           1519   ! TO NO LONGER BE TESTED.
123 ;           1520   !-
124 ;           1521
125 ;           1522   BGNDU;
126 ;           1523   RETURN;
127 ;           1524   ENDDU;
131
132                                    .SBTTL  LDU REPORT CODE SECTION
136 004672  000207                LDU:  RTS     PC                                ;                          1515
137
138                                    ; Routine Size:  1 word
139                                    ; Maximum stack depth per invocation:  0 words
144
145
149
150                                    .SBTTL  L$DU REPORT CODE SECTION
154 004674  004767  177772         L$DU:: JSR    PC,LDU                            ;                          1523
155 004700  104453                       TRAP   53
156 004702  000207                       RTS    PC
157
158                                    ; Routine Size:  4 words
159                                    ; Maximum stack depth per invocation:  0 words
164
165
166 ;           1525
167 ;           1526   !+
168 ;           1527   ! THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES
169 ;           1528   ! TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK
170 ;           1529   ! TO THE TEST CYCLE.
```

```
     172 ;ML3AD
     173 ;                   REPORT CODE SECTION                    22-Mar-1982 11:35:31    TOPS-20 Bliss-16 V2(212)
     174                                                            22-Mar-1982 11:32:50    PA:<NEALE>ML3AD.BLI.1 (1)
     175 ;           1530    !-
     176 ;           1531
     177 ;           1532    BGNAU;
     178 ;           1533    RETURN;
     179 ;           1534    ENDAU;
     183
     184                                     .SBTTL  LAU REPORT CODE SECTION
     188 004704  000207                 LAU:    RTS     PC                              ;                       1524
     189
     190                                 ; Routine Size:  1 word
     191                                 ; Maximum stack depth per invocation:  0 words
     196
     197
     201
     202                                     .SBTTL  LSAU REPORT CODE SECTION
     206 004706  004767  177772         LSAU::  JSR     PC,LAU                          ;                       1533
     207 004712  104452                         TRAP    52
     208 004714  000207                         RTS     PC
     209
     210                                 ; Routine Size:  4 words
     211                                 ; Maximum stack depth per invocation:  0 words
     216
     217
     218 ;           1535    END
     219 ;           1536
     220 ;           1537    ELUDOM
     224
     225
     226
```

```
                        ;ML3AD
                        ;                REPORT CODE SECTION                    22-Mar-1982 11:35:31    TOPS
                        ;                                                       22-Mar-1982 11:32:50    PA:<
                        ; Size:          20 code + 0 data words
                        ; Run Time:      00:02.9
                        ; Elapsed Time:  00:05.5
                        ; Memory Used:   24 pages
                        ; Compilation Complete
```

```
;ML4AD                                                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (1)

;          0001  MODULE ML4AD =
;          0002  BEGIN
;          0003  !
;          0004  ! PRETTY BLF COMMANDS
;          0005  !
;          0006  !<BLF/LOWERCASE_KEY>
;          0007  !
;          0008  ! REQUIRE
;          0009  !
;          0010
;          0011  %SBTTL 'DECLARATION SECTION'
;          0012
;          0013  require 'BLSMAC.REQ';                          !BLISS INTERFACE MODULE
;          1503
;          1504  !
;          1505  ! CONSTANT LITERALS
;          1506  !
;          1507
;          1508  literal
;          1509      ONE = 1,                                   !DATA BIT OF ONE
;          1510      ONES = %o'177777',                         !DATA PATTERN OF ONES
;          1511      ZERO = 0,                                  !DATA BIT OF ZERO
;          1512      ZEROES = 0,                                !DATA PATTERN OF ZEROES
;          1513      NUM_OF_REG = 22,                           !NUMBER OF BLOCKS IN GLOBAL STORAGE 'ML-REG'
;          1514      FIELD_SIZ = 4,                             !FIELD SIZ FOR FIELD DECLARATIONS 'WORD_MAP'
;          1515      !
;          1516      !MLCS1 FUNCTION CODES
;          1517      !
;          1518      NOOP = 1,                                  !NOOP FUNCTION
;          1519      DRV_CLR = %o'11',                          !DRIVE CLEAR FUNCTION
;          1520      RD_IN_PRE = %o'21',                        !READ IN PRESET FUNCTION
;          1521      SEARCH = %o'31',                           !SEARCH FUNCTION
;          1522      WRT_CHK = %o'51',                          !WRITE CHECK FUNCTION
;          1523      write = %o'61',                            !WRITE FUNCTION
;          1524      read = %o'71',                             !READ FUNCTION
;          1525      !
;          1526      ! DELAY ARGUMENTS
;          1527      !
;          1528      ONE_US = 1,                                !ONE MICRO SECOND DELAY
;          1529      FRTY_US = 40,                              !FORTY MICRO SECOND DELAY
;          1530      TWO_TH_US = 2000,                          !TWO THOUSAND MICRO SECOND DELAY
;          1531      ONE_SEC = 100,                             !ONE SECOND DELAY
;          1532      ONE_MS = 10;                               !ONE MILL SECOND DELAY
;          1533
;          1534  !
;          1535  ! FIELD DECLARATIONS
;          1536  !
;          1537
;          1538  field
;          1539      WORD_MAP =                                 !MAPS GLOBAL STORAGE 'ML_REG' INTO REGISTER PERSONALITIES
;          1540          set
;          1541              REGISTER_ADD = [0, 0, 16, 0],      !REGISTERS ADDRESS
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (1)

```
;ML4AD
;                    DECLARATION SECTION

;          1542          FORCE_HI = [1, 0, 16, 0],          !REGISTERS  FORCED HI BITS
;          1543          FORCE_LO = [2, 0, 16, 0],          !REGISTERS FORCED LO BITS
;          1544          DONT_CARE = [3, 0, 16, 0]          !REGISTERS IGNORE BITS
;          1545          tes,
;          1546      NIB_MAP =                              !MAPS OWN STORAGE NIB_SAVE INTO TEN FOUR BIT NIBBLES
;          1547          set
;          1548          NIB_0 = [0, 0, 4, 0],              !NIBBLE 0 BITS <0:3>
;          1549          NIB_1 = [0, 4, 4, 0],              !NIBBLE 1 BITS <4:7>
;          1550          NIB_2 = [0, 8, 4, 0],              !NIBBLE 2 BITS <8:11>
;          1551          NIB_3 = [0, 12, 4, 0],             !NIBBLE 3 BITS <12:15>
;          1552          NIB_4 = [1, 0, 4, 0],              !NIBBLE 4 BITS <16:19>
;          1553          NIB_5 = [1, 4, 4, 0],              !NIBBLE 5 BITS <20:23>
;          1554          NIB_6 = [1, 8, 4, 0],              !NIBBLE 6 BITS <24:27>
;          1555          NIB_7 = [1, 12, 4, 0],             !NIBBLE 7 BITS <28:31>
;          1556          NIB_8 = [2, 8, 4, 0],              !NIBBLE 8 BITS <32:35>
;          1557          NIB_9 = [2, 12, 3, 0]              !NIBBLE 9 BITS <36:39>
;          1558          tes,
;          1559      DT1_FLD =
;          1560          set
;          1561          E0_5 = [0, 0, 6, 0],
;          1562          C0_5 = [0, 6, 6, 0],
;          1563          CRC_DATA = [0, 13, 1, 0],
;          1564          SGL_DATA = [0, 14, 1, 0],
;          1565          UNC_DATA = [0, 15, 1, 0],
;          1566          EE_DATA = [0, 0, 16, 0],
;          1567          A0_5 = [1, 0, 6, 0],
;          1568          P0_5 = [1, 6, 6, 0],
;          1569          ECA_DATA = [1, 12, 1, 0],
;          1570          B0_5 = [2, 0, 6, 0]
;          1571          tes;
;          1572
;          1573  external
;          1574          REGDMP : volatile,                 !REGISTER DUMP ON ERROR FLAG
;          1575          ONEPAS : volatile,                 !DROP UNIT AFTER ONE PASS
;          1576          INTERVEN : volatile,                       !MANUAL INTERVENTION FLAG
;          1577          PRSN : volatile,                   !PRINT SERIAL NUMBER FLAG
;          1578          ERRTHR : volatile;                 !ENABLE ERROR THRESHOLD FLAG
;          1579
;          1580  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (2)

```
;ML4AD
;                DECLARATION SECTION

!
! OWN STORAGE
!

own
    NIB_SAVE : block [3] field (NIB_MAP) volatile,                  !STORAGE LOCATION TO SAVE NIBBLE DATA READ DURING DIAG MODE
    HW_OR_TBL : vector [127] volatile,                              !STORES HARDWARE ORED PROM DATA DURING PROM OR FUNC TEST
    PTBL_PTR : volatile,                                            !HARDWARE P-TABLE POINTER
    OP_NOM_ARR : volatile,                                          !OPERATORS NUMBER  OF ARRAY INPUTTED
    ARR_INC : volatile,                                             !ARRAY SELECTION INCREMENT VALUE
    GOOD_BLK : volatile,                                            !GOOD BLOCK ADRS
    PAR_DIS : volatile,                                             !PARITY DISABLE FLAG
    CHIP_SIZ : volatile,                                            !MOS RAM CHIP SIZE
    LST_BLK : volatile,                                             !LAST ADDRESSABLE BLOCK
    ARR_16 : volatile,                                              !MAX NUMBER OF ARRAY ALLOWED
    LST_ARR : volatile,                                             !LAST ADDRESSABLE ARRAY
    IO_BUF : vector [256] volatile,                                 !INPUT OUTPUT BUFFER
    STK_OFF : vector [10, byte] volatile,                           !STACK OFFSET STORAGE LOCATION
    stack : vector [198, byte] volatile,                            !STACK OF 198 BYTE LOCATIONS
    PD_TEMP : bitvector [16] volatile,                              !PROM DATA STORAGE LOCATION DURING DIAG MODES
    W_C_SIZE : volatile,                                            !STORES WORD COUNT FOR 16K OR 64K XFERS
    RAS_INC : volatile,                                             !ROW ADRS STROBE INCREMENT
    WT_DATA : volatile,                                             !SAVE WRITE DATA DURING REG READ WRITE TEST
    RD_DATA : volatile,                                             !SAVE READ DATA DURING REG READ WRITE TEST
    DRIVE_TYPE : volatile,                                          !DRIVE TYPE STORAGE LOCATION
    LST_DUT_REG : volatile,                                         !LAST ML-11 REG INDEX FOR THIS TYPE RH CONTROLLER
    REG_INIT_FLG : initial (0) volatile,                            !FLAG TO DETECT DOING REG INIT TEST
    A_CAL : volatile,                                               !CALCULATED CRC STORAGE LOCATION
    B_CAL : volatile,                                               !CALCULATED CRC STORAGE LOCATION
    P_CAL : volatile,                                               !CALCULATED CRC STORAGE LOCATION
    A_GEN : volatile,                                               !GENERATED CRC STORAGE LOCATION
    B_GEN : volatile,                                               !GENERATED CRC STORAGE LOCATION
    P_GEN : volatile,                                               !GENERATED CRC STORAGE LOCATION
    P_CNT : volatile,                                               !ERROR MESSAGE PRINT COUNTER
    LIMIT : volatile,                                               !LIMIT OF ERROR MESSAGES PRINTED
    DROP_CNT : volatile,                                            !COUNTS NUMBER OF PASSES DONE
!
!<BLF/NOFORMAT>
!
   ML_REG: blockvector [NUM_OF_REG,FIELD_SIZ] field(WORD_MAP) !ML11 REGISTERS
      preset (                                                      !STRUCTURE TO STORE INDIVIDULA
                                                                    !REGISTER PERSONALITIES
                                                                    !ie. REGISTER ADDRESSES
                                                                    !BITS FORCED HIGH
                                                                    !BITS FORCED LOW
                                                                    !READ ONLY BITS
          [0,FORCE_HI]    = %o'004000',    !MLCS1

          [0,FORCE_LO]    = %o'173701',
          [0,DONT_CARE]   = %o'160200',
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (2)

```
;ML4AD
;                    DECLARATION SECTION
;
;          1633          [5,FORCE_LO]    = %o'25077',    !MLDS
;          1634          [5,FORCE_HI]    = %o'010600',
;          1635          [5,DONT_CARE]   = %o'000100',
;          1636
;          1637          [6,FORCE_LO]    = %o'014620',   !MLER
;          1638
;          1639          [7,DONT_CARE]   = %o'177400',   !MLAS
;          1640
;          1641          [8,FORCE_LO]    = %o'100000',   !MLPA
;          1642
;          1643          [10,FORCE_LO]   = %o'000020',   !MLMR
;          1644          [10,DONT_CARE]  = %o'177400',
;          1645
;          1646          [11,FORCE_HI]   = %o'000110',   !MLDT
;          1647          [11,FORCE_LO]   = %o'177666',
;          1648          [11,DONT_CARE]  = %o'000001',
;          1649
;          1650          [13,FORCE_LO]   = %o'140300',   !MLE1
;          1651
;          1652          [14,FORCE_LO]   = %o'100300',   !MLE2
;          1653
;          1654          [17,FORCE_LO]   = %o'010000',   !MLEE
;          1655          [21,DONT_CARE]  = %o'000000'    !MLCS2
;          1656                  ) volatile,
;          1657
;          1658
;          1659    REM_TBL:vector [63,byte]          !REMAINDER TABLE
;          1660      preset (                        !STRUCTURE TO STORE PRECALCULATED
;          1661              [0] = %b'000001',       !CRC REMAINDER VALUES FOR CRC CODE
;          1662              [1] = %b'000010',       !GENERATION
;          1663              [2] = %b'000100',
;          1664              [3] = %b'001000',
;          1665              [4] = %b'010000',
;          1666              [5] = %b'100000',
;          1667              [6] = %b'000011',
;          1668              [7] = %b'000110',
;          1669              [8] = %b'001100',
;          1670              [9] = %b'011000',
;          1671              [10] = %b'110000',
;          1672              [11] = %b'100011',
;          1673              [12] = %b'000101',
;          1674              [13] = %b'001010',
;          1675              [14] = %b'010100',
;          1676              [15] = %b'101000',
;          1677              [16] = %b'010011',
;          1678              [17] = %b'100110',
;          1679              [18] = %b'001111',
;          1680              [19] = %b'011110',
;          1681              [20] = %b'111100',
;          1682              [21] = %b'111011',
;          1683              [22] = %b'110101',
;          1684              [23] = %b'101001',
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (2)

```
;ML4AD
;                    DECLARATION SECTION
;
;          1685              [24] = %b'010001',
;          1686              [25] = %b'100010',
;          1687              [26] = %b'000111',
;          1688              [27] = %b'001110',
;          1689              [28] = %b'011100',
;          1690              [29] = %b'111000',
;          1691              [30] = %b'110011',
;          1692              [31] = %b'100101',
;          1693              [32] = %b'001001',
;          1694              [33] = %b'010010',
;          1695              [34] = %b'100100',
;          1696              [35] = %b'001011',
;          1697              [36] = %b'010110',
;          1698              [37] = %b'101100',
;          1699              [38] = %b'011011',
;          1700              [39] = %b'110110',
;          1701              [40] = %b'101111',
;          1702              [41] = %b'011101',
;          1703              [42] = %b'111010',
;          1704              [43] = %b'110111',
;          1705              [44] = %b'101101',
;          1706              [45] = %b'011001',
;          1707              [46] = %b'110010',
;          1708              [47] = %b'100111',
;          1709              [48] = %b'001101',
;          1710              [49] = %b'011010',
;          1711              [50] = %b'110100',
;          1712              [51] = %b'101011',
;          1713              [52] = %b'010101',
;          1714              [53] = %b'101010',
;          1715              [54] = %b'010111',
;          1716              [55] = %b'101110',
;          1717              [56] = %b'011111',
;          1718              [57] = %b'111110',
;          1719              [58] = %b'111111',
;          1720              [59] = %b'111101',
;          1721              [60] = %b'111001',
;          1722              [61] = %b'110001',
;          1723              [62] = %b'100001'}volatile,
;          1724
;          1725  DT_1:blockvector[5,3] field (DT1_FLD)          !DATA TABLE 1
;          1726  preset (                                       !STRUCTURE TO STORE TEST DATA
;          1727              [0,E0_5] = %b'111111',             !FOR TEST 56 'ECC ERROR LOCATION TEST'
;          1728              [0,C0_5] = %b'100100',
;          1729              [0,CRC_DATA] = %b'1',
;          1730              [0,SGL_DATA] = %b'1',
;          1731              [0,UNC_DATA] = %b'0',
;          1732              [0,A0_5] = %b'000000',
;          1733              [0,P0_5] = %b'111111',
;          1734              [0,ECR_DATA] = %b'0',
;          1735              [0,B0_5] = %b'000000',
;          1736              [1,E0_5] = %b'111111',
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (2)

```
;ML4AD
;                    DECLARATION SECTION
;
;          1737          [1,CO_5] = %b'100100',
;          1738          [1,CRC_DATA] = %b'1',
;          1739          [1,SGL_DATA] = %b'1',
;          1740          [1,UNC_DATA] = %b'0',
;          1741          [1,AO_5] = %b'111111',
;          1742          [1,PO_5] = %b'000000',
;          1743          [1,ECR_DATA] = %b'0',
;          1744          [1,BO_5] = %b'000000',
;          1745          [2,EO_5] = %b'111111',
;          1746          [2,CO_5] = %b'000101',
;          1747          [2,CRC_DATA] = %b'0',
;          1748          [2,SGL_DATA] = %b'0',
;          1749          [2,UNC_DATA] = %b'1',
;          1750          [2,AO_5] = %b'000000',
;          1751          [2,PO_5] = %b'111111',
;          1752          [2,ECR_DATA] = %b'1',
;          1753          [2,BO_5] = %b'111111',
;          1754          [3,EO_5] = %b'111111',
;          1755          [3,CO_5] = %b'000101',
;          1756          [3,CRC_DATA] = %b'0',
;          1757          [3,SGL_DATA] = %b'0',
;          1758          [3,UNC_DATA] = %b'1',
;          1759          [3,AO_5] = %b'000000',
;          1760          [3,PO_5] = %b'000000',
;          1761          [3,ECR_DATA] = %b'0',
;          1762          [3,BO_5] = %b'111111',
;          1763          [4,EO_5] = %b'111111',
;          1764          [4,CO_5] = %b'000101',
;          1765          [4,CRC_DATA] = %b'0',
;          1766          [4,SGL_DATA] = %b'0',
;          1767          [4,UNC_DATA] = %b'1',
;          1768          [4,AO_5] = %b'111111',
;          1769          [4,PO_5] = %b'000000',
;          1770          [4,ECR_DATA] = %b'1',
;          1771          [4,BO_5] = %b'111111')volatile,
;          1772
;          1773  !
;          1774  !<BLF/FORMAT>
;          1775  !
;          1776      RH_ADD : volatile,                    !RH CONTROLLER BASE ADDRESS
;          1777      RH_TYP : volatile,                    !RH CONTROLLER TYPE
;          1778      RH_VEC : volatile,                    !RH CONTROLLER VECTOR ADDRESS
;          1779      ML_LUN : volatile,                    !ML LOGICAL UNIT NO.
;          1780      ML_DUT : volatile;                    !ML DRIVE NUMBER
;          1781
;          1782  !
;          1783  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (3)

```
;ML4AD
;                    DECLARATION SECTION

;          1784  !
;          1785  EQUALS;
;          1786  !
;          1787  ! MACRO DEFINITIONS
;          1788  !
;          1789
;          1790  macro
;          1791  !
;          1792  ! REGISTER NAMES:
;          1793  !
;        M 1794      MLCS1 =
;          1795   .ML_REG [0,REGISTER_ADD]%,          !CONTROL AND STATUS REGISTER 1
;        M 1796      MLWC =
;          1797   .ML_REG [1,REGISTER_ADD]%,          !WORD COUNT REGISTER
;        M 1798      MLBA =
;          1799   .ML_REG [2,REGISTER_ADD]%,          !UNIBUS ADDRESS REGISTER
;        M 1800      MLDA =
;          1801   .ML_REG [3,REGISTER_ADD]%,          !DESIRED ADDRESS REGISTER
;        M 1802      MLCS2 =
;          1803   .ML_REG [4,REGISTER_ADD]%,          !CONTROL AND STATUS REGISTER 2
;        M 1804      MLDS =
;          1805   .ML_REG [5,REGISTER_ADD]%,          !DRIVE STATUS REGISTER
;        M 1806      MLER =
;          1807   .ML_REG [6,REGISTER_ADD]%,          !ERROR REGISTER
;        M 1808      MLAS =
;          1809   .ML_REG [7,REGISTER_ADD]%,          !ATTENTION SUMMARY REGISTER
;        M 1810      MLLA =
;          1811   .ML_REG [8,REGISTER_ADD]%,          !LOOK AHEAD REGISTER
;        M 1812      MLPA =
;          1813   .ML_REG [8,REGISTER_ADD]%,          !PROM ADDRESS REGISTER
;        M 1814      MLDB =
;          1815   .ML_REG [9,REGISTER_ADD]%,          !DATA BUFFER REGISTER
;        M 1816      MLMR =
;          1817   .ML_REG [10,REGISTER_ADD]%,         !MAINTENANCE REGISTER
;        M 1818      MLDT =
;          1819   .ML_REG [11,REGISTER_ADD]%,         !DRIVE TYPE REGISTER
;        M 1820      MLSN =
;          1821   .ML_REG [12,REGISTER_ADD]%,         !SERIAL NUMBER REGISTER
;        M 1822      MLE1 =
;          1823   .ML_REG [13,REGISTER_ADD]%,         !ECC CRC WORD REGISTER 1
;        M 1824      MLE2 =
;          1825   .ML_REG [14,REGISTER_ADD]%,         !ECC CRC WORD REGISTER 2
;        M 1826      MLD1 =
;          1827   .ML_REG [15,REGISTER_ADD]%,         !DATA DIAGNOSTIC REGISTER 1
;        M 1828      MLD2 =
;          1829   .ML_REG [16,REGISTER_ADD]%,         !DATA DIAGNOSTIC REGISTER 2
;        M 1830      MLEE =
;          1831   .ML_REG [17,REGISTER_ADD]%,         !ECC ERROR REGISTER
;        M 1832      MLEL =
;          1833   .ML_REG [18,REGISTER_ADD]%,         !ECC ERROR LCOATION REGISTER
;        M 1834      MLPD =
;          1835   .ML_REG [19,REGISTER_ADD]%,         !PROM DATA REGISTER
```

```
;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                    DECLARATION SECTION                  29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (3)
;
;          M 1836      MLBAE =
;            1837    .ML_REG [20,REGISTER_ADD]%,          !BUS ADDRESS EXTENSION REGISTER
;          M 1838      MLCS3 =
;            1839    .ML_REG [21,REGISTER_ADD]%,          !CONTROL AND STATUS REGISTER 3
;            1840    !
;            1841    ! BIT ASSIGNMENTS:
;            1842    !
;          M 1843      SC =
;            1844    (MLCS1)<15,1>%,                      !MLCS1 BIT ASSIGNMENTS
;          M 1845      TRE =
;            1846    (MLCS1)<14,1>%,
;          M 1847      MCPE =
;            1848    (MLCS1)<13,1>%,
;          M 1849      DVA =
;            1850    (MLCS1)<11,1>%,
;          M 1851      RDY =
;            1852    (MLCS1)<7,1>%,
;          M 1853      IE =
;            1854    (MLCS1)<6,1>%,
;          M 1855      GO =
;            1856    (MLCS1)<0,1>%,
;          M 1857      ML_FUNC =
;            1858    (MLCS1)<0,6>%,
;          M 1859      DLT =
;            1860    (MLCS2)<15,1>%,                      !MLCS2 BIT ASSIGNMENTS
;          M 1861      WCE =
;            1862    (MLCS2)<14,1>%,
;          M 1863      PE =
;            1864    (MLCS2)<13,1>%,
;          M 1865      NED =
;            1866    (MLCS2)<12,1>%,
;          M 1867      NEM =
;            1868    (MLCS2)<11,1>%,
;          M 1869      PGE =
;            1870    (MLCS2)<10,1>%,
;          M 1871      MXF =
;            1872    (MLCS2)<9,1>%,
;          M 1873      MDPE =
;            1874    (MLCS2)<8,1>%,
;          M 1875      ORDY =
;            1876    (MLCS2)<7,1>%,
;          M 1877      IRDY =
;            1878    (MLCS2)<6,1>%,
;          M 1879      CLR =
;            1880    (MLCS2)<5,1>%,
;          M 1881      PAT =
;            1882    (MLCS2)<4,1>%,
;          M 1883      BAI =
;            1884    (MLCS2)<3,1>%,
;          M 1885      DRV_NUM =
;            1886    (MLCS2)<0,3>%,
;          M 1887      ATTN =
```

```
436 ;ML4AD
437 ;                    DECLARATION SECTION                    29-Mar-1982 16:23:04   TOPS-20 Bliss-16 V2(212)
438 ;                                                           29-Mar-1982 16:21:03   PA:<NEALE>ML4AD.BLI.4 (3)
439 ;          1888      (MLDS)<15,1>%,                        !MLDS BIT ASSIGNMENTS
440 ;        M 1889        COMP_ERR =
441 ;          1890      (MLDS)<14,1>%,
442 ;        M 1891        MOL =
443 ;          1892      (MLDS)<12,1>%,
444 ;        M 1893        LBT =
445 ;          1894      (MLDS)<10,1>%,
446 ;        M 1895        DPR =
447 ;          1896      (MLDS)<8,1>%,
448 ;        M 1897        DRY =
449 ;          1898      (MLDS)<7,1>%,
450 ;        M 1899        VV =
451 ;          1900      (MLDS)<6,1>%,
452 ;        M 1901        DCK =
453 ;          1902      (MLER)<15,1>%,                        !MLER BIT ASSIGNMENTS
454 ;        M 1903        UNS =
455 ;          1904      (MLER)<14,1>%,
456 ;        M 1905        OPI =
457 ;          1906      (MLER)<13,1>%,
458 ;        M 1907        IAE =
459 ;          1908      (MLER)<10,1>%,
460 ;        M 1909        AOE =
461 ;          1910      (MLER)<9,1>%,
462 ;        M 1911        ECH_ERR =
463 ;          1912      (MLER)<6,1>%,
464 ;        M 1913        DPAR =
465 ;          1914      (MLER)<5,1>%,
466 ;        M 1915        CPAR =
467 ;          1916      (MLER)<3,1>%,
468 ;        M 1917        RMR =
469 ;          1918      (MLER)<2,1>%,
470 ;        M 1919        ILR =
471 ;          1920      (MLER)<1,1>%,
472 ;        M 1921        ILF =
473 ;          1922      (MLER)<0,1>%,
474 ;        M 1923        ARR_TYP =
475 ;          1924      (MLMR)<10,1>%,                        !MLMR BIT ASSIGNMENTS
476 ;        M 1925        ML_NUM_ARR =
477 ;          1926      (MLMR)<11,5>%,
478 ;        M 1927        REF_MAR =
479 ;          1928      (MLMR)<7,1>%,
480 ;        M 1929        PROM_RW =
481 ;          1930      (MLMR)<6,1>%,
482 ;        M 1931        PROM_DIS =
483 ;          1932      (MLMR)<5,1>%,
484 ;        M 1933        DAT_CLK =
485 ;          1934      (MLMR)<4,1>%,
486 ;        M 1935        DAT_DM =
487 ;          1936      (MLMR)<3,1>%,
488 ;        M 1937        DCK_EN =
489 ;          1938      (MLMR)<2,1>%,
490 ;        M 1939        ECC_DIS =
```

```
;ML4AD
;                    DECLARATION SECTION

;          940   (MLMR)<1,1>%,
;        M 941      ECC_DM =
;          942   (MLMR)<0,1>%,
;        M 943      DRV_TYP =
;          944   (MLDT)<0,1>%,                          !MLDT BIT ASSIGNMENTS
;        M 945      CRC_A =
;        M 946                                         !MLE1 BIT ASSIGNMENTS
;          947   (MLE1)<0,6>%,
;        M 948      PAR_CRC_WRD =
;          949   (MLE1)<8,6>%,
;        M 950      CRC_B =
;        M 951                                         !MLE2 BIT ASSIGNMENTS
;          952   (MLE2)<0,6>%,
;        M 953      UNC_ERR =
;          954   (MLEE)<15,1>%,                        !MLEE BIT ASSIGNMENTS
;        M 955      SGL_ERR =
;          956   (MLEE)<14,1>%,
;        M 957      CRC_ERR =
;          958   (MLEE)<13,1>%,
;        M 959      BIT_IN_ERR =
;          960   (MLEE)<0,6>%,
;        M 961      CHAN_IN_ERR =
;          962   (MLEE)<6,6>%,
;          963   !
;          964   ! MISCELLANEOUS MACRO DEFINITIONS:
;          965   !
;        M 966      SN3 =
;          967   (MLSN)<12,4>%,
;        M 968      SN2 =
;          969   (MLSN)<8,4>%,
;        M 970      SN1 =
;          971   (MLSN)<4,4>%,
;        M 972      SN0 =
;          973   (MLSN)<0,4>%,
;        M 974      IS_SET =
;        M 975                                         !TEST IF BIT IS EQUAL 1
;          976   eql 1%,
;        M 977      IS_NOT_SET =
;        M 978                                         !TEST IF BIT IS EQUAL 0
;          979   eql 0%,
;        M 980      HI =
;        M 981                                         !READS REGISTERS FORCED HI BITS FROM PERSONALITY TABLE
;          982   ML_REG[.index,FORCE_HI]%,
;        M 983      LO =
;        M 984                                         !READS REGISTERS FORCED LO BITS FROM PERSONALITY TABLE
;          985   ML_REG[.index,FORCE_LO]%,
;        M 986      IGNORE =
;        M 987                                         !READS REGISTERS DONT_CARE BITS FROM PERSONALITY TABLE
;          988   ML_REG[.index,DONT_CARE]%,
;        M 989      MLE2_MASK =
;        M 990                                         !READS MLE2 DONT CARE MASK EITHER DATA DIAG OR ECC CIE REG
;          991   ML_REG[14,DONT_CARE]%,
```

```
548 ;ML4AD
549 ;                   DECLARATION SECTION                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
550 ;                                                                29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (3)
551 ;        M 1992       WRT_MASK =
552 ;        M 1993                                                  !GENERATE MASK DATA PATTERN USING REGISTER FORCE LO, HI AND IGNORE B
553 ;          1994    .IGNORE or ((not .LO) and (.HI or .TST_PAT))%,
554 ;        M 1995       CLR_MBUS =
555 ;        M 1996                                                  !CLEAR MASS BUS RESTORE DRIVE NUMBER
556 ;          1997    CLR = ONE; DRV_NUM = .ML_DUT%,
557 ;          1998    !<BLF/SYNONYM IS_SET = EQL 1 * >
558 ;          1999    !<BLF/SYNONYM IS_NOT_SET = EQL 0 * >
559 ;          2000    !
560 ;          2001    ! DIAGNOSTIC DATA REGISTER MACROS
561 ;          2002    !
562 ;        M 2003       RD_LNG_WRD =
563 ;        M 2004                                                  !READ DATA DIAG REGS INTO BIND LOCATIONS
564 ;        M 2005       D1_TEMP = .MLD1;
565 ;        M 2006       D2_TEMP = .MLD2;
566 ;          2007       E2_TEMP = .MLE2%,
567 ;        M 2008       WRT_LNG_WRD =
568 ;        M 2009                                                  !LOADS DATA DIAG REG WITH CONTENTS OF BIND LOCATIONS
569 ;        M 2010       MLD1 = .D1_TEMP;
570 ;        M 2011       MLD2 = .D2_TEMP;
571 ;          2012       MLE2 = .E2_TEMP%,
572 ;        M 2013       TIME_OUT_LOOP =
573 ;        M 2014                  !WAIT LOOP DIRING MASS BUS TRANSFER
574 ;        M 2015            do
575 ;        M 2016              0
576 ;          2017            until .DRY;%,
577 ;          2018    !
578 ;          2019    ! ERROR PRINTING THRESHOLD MACROS
579 ;          2020    !
580 ;        M 2021       CLR_THRESHOLD =
581 ;          2022    P_CNT = ZERO;%,
582 ;        M 2023       CMP_THRESHOLD =
583 ;        M 2024    P_CNT = .P_CNT + 1;
584 ;        M 2025
585 ;        M 2026    if .P_CNT gtr .LIMIT
586 ;          2027    then exitloop;%;
587 ;          2028
588 ;          2029    !
589 ;          2030    ! BIND DECLARATIONS
590 ;          2031    !
591 ;          2032
592 ;          2033    bind
593 ;          2034    !
594 ;          2035    ! MANUAL INTERVENTION TEST ERROR AND OPERATOR PROMPT MESSAGES.
595 ;          2036    ! TESTS 21 'POWER FAIL TEST' AND TEST 61 'BATTERY BACK-UP TEST'
596 ;          2037    !
597 ;          2038       T_21 = uplit (%asciz'********* MANUAL INTERVENTION TEST 21 POWER FAIL *********'),
598 ;          2039       T_61 = uplit (%asciz'********* MANUAL INTERVENTION TEST 61 BATTERY BACK-UP *********'),
599 ;          2040       PWR_OFF = uplit (%asciz'WITH BATTERY BACK-UP ON TURN BOX AC CIRCUIT BREAKER OFF'),
6   ;          2041       PWR_ON = uplit (%asciz'WITH BATTERY BACK-UP OFF TURN BOX AC CIRCUIT BREAKER ON'),
    ;          2042       UNS_ERR = uplit (%asciz'UNS BIT NOT SET DURING AC POWER FAIL'),
6.  ;          2043       VV_NOT_SET = uplit (%asciz'VV BIT NOT SET AFTER RD_IN_PRESET FUNCTION'),
```

```
;ML4AD
;                    DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                           29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (3)

;          2044      VV_CLEAR = uplit (%asciz'VV BIT CLEARED AFTER RD-IN-PRESET FUNCTION'),
;          2045      NO_INIT = uplit (%asciz'MEMORY NOT INITIALIZED AFTER POWER UP WITH BAD BATTERY BACK-UP'),
;          2046      MB_DIS_ERR = uplit (%asciz'REGISTERS NOT CLEARED BY MB DIS AFTER POWER FAIL'),
;          2047      WC_ERR = uplit (%asciz'SC BIT SET DURING WRITE CHECK TRANSFER'),
;          2048      W_ERR = uplit (%asciz'SC BIT SET DURING WRITE TRANSFER'),
;          2049      VV_SET = uplit (%asciz'VV BIT STILL SET AFTER POWER LOSS WITH BAD BATTERY BACK-UP'),
;          2050      PUP_BB = uplit (%asciz'WITH BATTERY BACK-UP ON TURN BOX AC CIRCUIT BREAKER ON'),
;          2051      BB_VV_ERR = uplit (%asciz'VV BIT CLEARED AFTER POWER FAIL WITH GOOD BATTERY BACK-UP'),
;          2052      BB_INIT_ERR = uplit (%asciz'MEMORY INITED AFTER POWER FAIL WITH GOOD BATTERY BACK-UP'),
;          2053      BB_BB_ERR = uplit (%asciz'BATTERY BACK-UP FAILED TO MAINTAIN MEMORY INTEGRITY DURING POWER FAIL'),
;          2054  !
;          2055  ! ERROR DATA MAPPING FORMATS
;          2056  !
;          2057      FMT_1 = uplit (%asciz'%AEXPECTED: %O6%A    READ: %O6%N%N'),
;          2058      FMT_2 = uplit (%asciz'%AGOOD DATA: %O6%A    BAD DATA: %O6%A    XOR: %O6%N%N'),
;          2059      FMT_3 = uplit (%asciz'%ADRIVE SN: %O6%N%N'),
;          2060      FMT_4 = uplit (%asciz'%ABIT IN ERROR: %O6%N%N'),
;          2061      FMT_5 = uplit (%asciz'%AGOOD NIB DATA: %O2%A    BAD NIB DATA: %O2%A    NIB POS: %O4%N%N'),
;          2062      FMT_6 = uplit (%asciz'%ANIB IN ERROR: %D4%N%N'),
;          2063      FMT_7 = uplit (%asciz'%AFAILED AT: %O6%N%N'),
;          2064      FMT_8 = uplit (%asciz'%AREPLACE ARR MOD: %D2%N%N'),
;          2065      FMT_9 = uplit (%asciz'%AFAILED AT DSA: %O6%N%N'),
;          2066      FMT_10 = uplit (%asciz'%ABIT<15:10>: %B6%A    BIT<9:0>: %B10%N%N'),
;          2067      FMT_11 = uplit (%asciz'%AFAILING REG ADRS: %O6%N%N'),
;          2068      FMT_12 = uplit (%asciz'%AFAILING FUNC: %O6%N%N'),
;          2069      FMT_13 = uplit (%asciz' %AOFF SET CNT FOR NIB : %D2 %A = %D2 %N%N'),
;          2070      FMT_14 = uplit (%asciz'%AWROTE: %D2%A    READ: %D2%N%N'),
;          2071      FMT_15 = uplit (%asciz'%ANIBBLES XFERED BEFORE ERROR: %D3%N'),
;          2072      FMT_16 = uplit (%asciz'%AFAILING REG: %O6%A GOOD DATA: %O6%A BAD DATA: %O6%N%N'),
;          2073      FMT_17 = uplit (%asciz'%N%ADIAGNOSING UNIT %O1%N%N'),
;          2074      FMT_18 = uplit (%asciz'%ATIMED OUT DURING MBUS %O2%A FUNC%N%N'),
;          2075      FMT_19 = uplit (%asciz'%ACRC GEN = B: %B6%A A: %B6%A P: %B6%N'),
;          2076      FMT_20 = uplit (%asciz'%ACRC CAL = B: %B6%A A: %B6%A P: %B6%N%N'),
;          2077      FMT_21 = uplit (%asciz'%AFAILED AT PLOG: %D2%A CHANNEL: %D2%N%N'),
;          2078      FMT_22 = uplit (%asciz'%AFAILED AT WRD: %D2%A BIT: %D2%N%N'),
;          2079      FMT_23 = uplit (%asciz'%AREGISTER%S3%AADDRESS%S7%ACONTENTS%N%N'),
;          2080      FMT_24 = uplit (%asciz'%S%T%S5%O6%S4%O16%N'),
;          2081      FMT_25 = uplit (%asciz'%S4%T%A:%S%O6%N'),
;          2082      FMT_26 = uplit (%asciz'%S4%T%A:%S%D1%D1%D1%D1%N'),
;          2083  !
;          2084  ! ERROR MESSAGE MAPPING FORMATS
;          2085  !
;          2086      ONE_FMT = uplit (%asciz'%T%N'),
;          2087      TWO_FMT = uplit (%asciz'%T%T%N'),
;          2088      THR_FMT = uplit (%asciz'%T%T%T%N'),
;          2089      FOR_FMT = uplit (%asciz'%T%T%T%T%T%N'),
;          2090      FIV_FMT = uplit (%asciz'%T%T%T%T%T%T%N'),
;          2091      SIX_FMT = uplit (%asciz'%T%T%T%T%T%T%T%N'),
;          2092      SEV_FMT = uplit (%asciz'%T%T%T%T%T%T%T%T%N'),
;          2093      EIG_FMT = uplit (%asciz'%T%T%T%T%T%T%T%T%T%N'),
;          2094      NIN_FMT = uplit (%asciz'%T%T%T%T%T%T%T%T%T%T%N'),
;          2095      TEN_FMT = uplit (%asciz'%T%T%T%T%T%T%T%T%T%T%T%N'),
```

```
;ML4AD
;                    DECLARATION SECTION

;          2096      ELV_FMT = uplit (%asciz'%T%T%T%T%T%T%T%T%T%T%T%T%N'),
;          2097  !
;          2098  ! DIAGNOSTIC VOCABULARY
;          2099  !
;          2100  !
;          2101  ! WORDS
;          2102  !
;          2103      WRD_1 = uplit (%asciz' GO'),
;          2104      WRD_2 = uplit (%asciz' DRV_RDY'),
;          2105      WRD_3 = uplit (%asciz' ILF'),
;          2106      WRD_4 = uplit (%asciz' OPI'),
;          2107      WRD_5 = uplit (%asciz' BAD'),
;          2108      WRD_6 = uplit (%asciz' GOOD'),
;          2109      WRD_7 = uplit (%asciz' PARITY NOT'),
;          2110      WRD_8 = uplit (%asciz' GENERATED'),
;          2111      WRD_9 = uplit (%asciz' DETECTED'),
;          2112      WRD_10 = uplit (%asciz' ERROR'),
;          2113      WRD_11 = uplit (%asciz' AFTER'),
;          2114      WRD_12 = uplit (%asciz' DURING'),
;          2115      WRD_13 = uplit (%asciz' AT'),
;          2116      WRD_14 = uplit (%asciz' FAILURE'),
;          2117      WRD_15 = uplit (%asciz' ATA'),
;          2118      WRD_16 = uplit (%asciz' ATTN'),
;          2119      WRD_17 = uplit (%asciz' WRITING'),
;          2120      WRD_18 = uplit (%asciz' VV'),
;          2121      WRD_19 = uplit (%asciz' FUNC'),
;          2122      WRD_20 = uplit (%asciz' TRE'),
;          2123      WRD_21 = uplit (%asciz' RMR'),
;          2124      WRD_22 = uplit (%asciz' EXCESSIVE'),
;          2125      WRD_23 = uplit (%asciz' MBUS'),
;          2126      WRD_24 = uplit (%asciz' DATA'),
;          2127      WRD_25 = uplit (%asciz' CONTINUITY'),
;          2128      WRD_26 = uplit (%asciz' ADE'),
;          2129      WRD_27 = uplit (%asciz' LBT'),
;          2130      WRD_29 = uplit (%asciz' PREMATURLY'),
;          2131      WRD_30 = uplit (%asciz' IAE'),
;          2132      WRD_31 = uplit (%asciz' INCREMENT'),
;          2133      WRD_32 = uplit (%asciz' WITH'),
;          2134      WRD_33 = uplit (%asciz' UV'),
;          2135      WRD_34 = uplit (%asciz' UNS'),
;          2136      WRD_35 = uplit (%asciz' PROM'),
;          2137      WRD_36 = uplit (%asciz' OR'),
;          2138      WRD_37 = uplit (%asciz' SELECT'),
;          2139      WRD_38 = uplit (%asciz' REG'),
;          2140      WRD_39 = uplit (%asciz' UNIQUE'),
;          2141      WRD_40 = uplit (%asciz' 14'),
;          2142      WRD_41 = uplit (%asciz' NIBBLE CNT'),
;          2143      WRD_42 = uplit (%asciz' GTR'),
;          2144      WRD_43 = uplit (%asciz' WHILE'),
;          2145      WRD_44 = uplit (%asciz' TRE'),
;          2146      WRD_45 = uplit (%asciz' INITIAL'),
;          2147      WRD_46 = uplit (%asciz' OFF_SET'),
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (3)

```
;ML4AD
;                    DECLARATION SECTION

;          2148      WRD_47 = uplit (%asciz' COUNT'),
;          2149      WRD_48 = uplit (%asciz' DELAY'),
;          2150      WRD_49 = uplit (%asciz' TESTS'),
;          2151      WRD_50 = uplit (%asciz' ADRS'),
;          2152      WRD_51 = uplit (%asciz' COUNTER'),
;          2153      WRD_52 = uplit (%asciz' REG'),
;          2154      WRD_53 = uplit (%asciz' TESTED'),
;          2155      WRD_54 = uplit (%asciz' NIBBLE'),
;          2156      WRD_55 = uplit (%asciz' ALL'),
;          2157      WRD_56 = uplit (%asciz' TEST'),
;          2158      WRD_57 = uplit (%asciz' XFERED'),
;          2159      WRD_58 = uplit (%asciz' NIBBLES'),
;          2160      WRD_59 = uplit (%asciz' SC'),
;          2161      WRD_60 = uplit (%asciz' MULTIPLEXER'),
;          2162      WRD_61 = uplit (%asciz' UNEXPECTED'),
;          2163      WRD_62 = uplit (%asciz' NED'),
;          2164      WRD_63 = uplit (%asciz' ILR'),
;          2165      WRD_64 = uplit (%asciz' CRC'),
;          2166      WRD_65 = uplit (%asciz' SGL'),
;          2167      WRD_67 = uplit (%asciz' ECH'),
;          2168      WRD_68 = uplit (%asciz'UNC'),
;          2169      WRD_69 = uplit (%asciz' BIT'),
;          2170      WRD_70 = uplit (%asciz' CHANNEL'),
;          2171      WRD_71 = uplit (%asciz' LATCH'),
;          2172      WRD_72 = uplit (%asciz' DCK'),
;          2173      WRD_73 = uplit (%asciz' BUS'),
;          2174      WRD_74 = uplit (%asciz' SYNDROME'),
;          2175      WRD_75 = uplit (%asciz' DECODE'),
;          2176      WRD_76 = uplit (%asciz' NOT'),
;          2177      WRD_77 = uplit (%asciz' UNCORRECTABLE'),
;          2178      !
;          2179      ! PHRASES
;          2180      !
;          2181      PHR_1 = uplit (%asciz' BIT NOT SET'),
;          2182      PHR_2 = uplit (%asciz' BIT NOT CLR'),
;          2183      PHR_3 = uplit (%asciz' NO RESPONCE AFTER 1.5 US'),
;          2184      PHR_4 = uplit (%asciz' DATA ERRORS'),
;          2185      PHR_5 = uplit (%asciz' BIT SET'),
;          2186      PHR_6 = uplit (%asciz' BIT CLR'),
;          2187      PHR_7 = uplit (%asciz' OF OTHER DRIVES'),
;          2188      PHR_8 = uplit (%asciz' CLASS A'),
;          2189      PHR_9 = uplit (%asciz' CLASS B'),
;          2190      PHR_10 = uplit (%asciz' TO FIND'),
;          2191      PHR_11 = uplit (%asciz' NOT LATCHED'),
;          2192      PHR_12 = uplit (%asciz' SINGLE BIT ERROR'),
;          2193      PHR_13 = uplit (%asciz' MULTIPLE BIT ERROR'),
;          2194      PHR_14 = uplit (%asciz'          REGISTER   DUMP'),
;          2195      PHR_15 = uplit (%asciz'SERIAL #'),
;          2196      !
;          2197      ! FUNCTIONS
;          2198      !
;          2199      FNC_1 = uplit (%asciz' MEM SIZING'),
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (3)

```
;ML4AD
;                    DECLARATION SECTION

;          2200      FNC_2 = uplit (%asciz' NOOP'),
;          2201      FNC_3 = uplit (%asciz' DRV'),
;          2202      FNC_4 = uplit (%asciz' WRITE CHECK'),
;          2203      FNC_5 = uplit (%asciz' WRITE'),
;          2204      FNC_6 = uplit (%asciz' READ'),
;          2205      FNC_7 = uplit (%asciz' CLEAR'),
;          2206      FNC_8 = uplit (%asciz' COMP ERROR'),
;          2207      FNC_9 = uplit (%asciz' SYS CLR'),
;          2208      FNC_10 = uplit (%asciz' SEARCH'),
;          2209      FNC_11 = uplit (%asciz' READ-IN-PRESET'),
;          2210      FNC_12 = uplit (%asciz' ILLEGAL'),
;          2211      FNC_13 = uplit (%asciz' ABORT'),
;          2212      FNC_14 = uplit (%asciz' ARR RD WRT'),
;          2213      FNC_15 = uplit (%asciz' GOOD BLK'),
;          2214      FNC_16 = uplit (%asciz' REFRESH'),
;          2215      FNC_17 = uplit (%asciz' ARRAY'),
;          2216      FNC_18 = uplit (%asciz' RAM-BUS'),
;          2217      FNC_19 = uplit (%asciz' OVERFLOW'),
;          2218      FNC_21 = uplit (%asciz' CHK_SUM'),
;          2219      FNC_22 = uplit (%asciz' LAST BLK'),
;          2220      FNC_23 = uplit (%asciz' INITIALIZE'),
;          2221      !
;          2222      ! REGISTERS
;          2223      !
;          2224      REG_1 = uplit (%asciz' MLCS1'),
;          2225      REG_2 = uplit (%asciz' MLDS'),
;          2226      REG_3 = uplit (%asciz' MLER'),
;          2227      REG_4 = uplit (%asciz' MLMR'),
;          2228      REG_5 = uplit (%asciz' MLAS'),
;          2229      REG_6 = uplit (%asciz' MLDA'),
;          2230      REG_7 = uplit (%asciz' MLDT'),
;          2231      REG_8 = uplit (%asciz' MLPA'),
;          2232      REG_9 = uplit (%asciz' MLSN'),
;          2233      REG_10 = uplit (%asciz' MLE1'),
;          2234      REG_11 = uplit (%asciz' MLE2'),
;          2235      REG_12 = uplit (%asciz' MLD1'),
;          2236      REG_13 = uplit (%asciz' MLD2'),
;          2237      REG_14 = uplit (%asciz' MLEE'),
;          2238      REG_15 = uplit (%asciz' MLEL'),
;          2239      REG_16 = uplit (%asciz' MLPD'),
;          2240      REG_17 = uplit (%asciz' MLCS2'),
;          2241      REG_18 = uplit (%asciz' MLWC'),
;          2242      REG_19 = uplit (%asciz' MLBA'),
;          2243      REG_20 = uplit (%asciz' MLBAJ'),
;          2244      REG_21 = uplit (%asciz' MLCS3'),
;          2245      !
;          2246      ! MODULES IN ERROR MESSAGES
;          2247      !
;          2248      ASYNC = uplit (%asciz'ASYNCHRONOUS FAILURE MODULE 7361'),
;          2249      SYNC = uplit (%asciz'SYNCHRONOUS FAILURE MODULE 7362 '),
;          2250      ARR_DAT = uplit (%asciz'ARRAY DATA FAILURE MODULE 7363 '),
;          2251      MEM_ARR = uplit (%asciz'MEMORY ARRAY FAILURE MODULE 7357'),
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (3)

```
;ML4AD
;                    DECLARATION SECTION

;          2252          INTER = uplit (%asciz'INTERMEDIATE DIAGNOSTIC MESSAGE'),
;          2253          DATA_LATE = uplit (%asciz'DATA LATE ERROR DURING TRANSFER'),
;          2254          SC_SET = uplit (%asciz'SC BIT SET DURING TRANSFER'),
;          2255          TRBLE_LOOP = uplit (%asciz'TROUBLE SHOOT LOOP ERRORS'),
;          2256          RH_ERROR = uplit (%asciz'RH CONTROLLER ERRORS'),
;          2257          TIME_OUT = uplit (%asciz'DRIVE HUNG AFTER MASS BUS TRANSFER'),
;          2258      !
;          2259      ! DATA DIAGNOSTIC REGISTER SAVE LOCATIONS
;          2260      !
;          2261          D1_TEMP = NIB_SAVE,
;          2262          D2_TEMP = NIB_SAVE [1, 0, 16, 0],
;          2263          E2_TEMP = NIB_SAVE [2, 0, 16, 0];
;          2264
;          2265      %SBTTL 'ROUTINE DECLARATION SECTION'
;          2266
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION

;          2267  routine LOAD_STACK (STK_PTR, NIB_PTR) : novalue =
;          2268      begin
;          2269
;          2270  !++
;          2271  ! FUNCTIONAL DESCRIPTION:
;          2272  !       LOAD STACK TAKES GOOD NIBBLE DATA
;          2273  !       FOUND IN THE STRUCTURE 'NIB_SAVE'
;          2274  !       AND STORES IT INTO THE STRUCTURE
;          2275  !       'STACK' REWRITING ANY BAD NIBBLE
;          2276  !       'STACK' LOCATIONS WITH GOOD NIBBLE
;          2277  !       DATA
;          2278  !
;          2279  ! FORMAL PARAMETERS:
;          2280  ! STK_PTR
;          2281  !       POINTS TO PRESENT DEPTH OF THE
;          2282  !       'STACK' WHERE PRESENT GOOD NIBBLE
;          2283  !       DATA IS TO BE STORED.
;          2284  !
;          2285  ! NIB_PTR
;          2286  !       POINTS TO CURRENT NIBBLE POSITION BEING
;          2287  !       MANIPULATED.
;          2288  !
;          2289  ! IMPLICIT INPUTS:
;          2290  ! STACK
;          2291  !       VECTOR OF 198 BYTE LOCATIONS WHERE
;          2292  !       GOOD NIBBLE DATA IS STORED
;          2293  !       DURING DIAGNOSTIC MODE READS, AFTER
;          2294  !       BAD NIBBLE LOCATIONS HAVE BEEN
;          2295  !       STRIPPED AWAY.
;          2296  !
;          2297  ! STK_OFF
;          2298  !
;          2299  !    vector of 9 byte LOCATIONS WHICH
;          2300  !       STORES AWAY A BAD NIBBLE OFF_SET
;          2301  !       COUNT FOR EACH NIBBLE POSITION
;          2302  !
;          2303  ! NIB_SAVE
;          2304  !       BLOCK OF 3 WORDS TO STORE THE
;          2305  !       DATA FOUND IN MLD1, MLD2 AND
;          2306  !       MLE2 AFTER A DIAGNOSTIC MODE READ.
;          2307  !
;          2308  ! IMPLICIT OUTPUTS:
;          2309  !       'STACK' LOADED WITH GOOD NIBBLE
;          2310  !       DATA
;          2311  !
;          2312  ! COMPLETETION CODES:   NONE
;          2313  !
;          2314  ! SIDE EFFECTS:         NONE
;          2315  !
;          2316  !--
;          2317
;          2318      case .NIB_PTR from 0 to 9 of              !SELECT NIBBLE DATA TO BE LOADED INTO THE STACK
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION

;          2319          set
;          2320
;          2321          [0] :
;          2322              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_0];
;          2323                                                  !LOAD NIBBLE DATA 0 INTO SELECTED STACK LOCATION
;          2324
;          2325          [1] :
;          2326              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_1];
;          2327                                                  !LOAD NIBBLE DATA 1 INTO SELECTED STACK LOCATION
;          2328
;          2329          [2] :
;          2330              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_2];
;          2331                                                  !LOAD NIBBLE DATA 2 INTO SELECTED STACK LOCATION
;          2332
;          2333          [3] :
;          2334              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_3];
;          2335                                                  !LOAD NIBBLE DATA 3 INTO SELECTED STACK LOCATION
;          2336
;          2337          [4] :
;          2338              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_4];
;          2339                                                  !LOAD NIBBLE DATA 4 INTO SELECTED STACK LOCATION
;          2340
;          2341          [5] :
;          2342              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_5];
;          2343                                                  !LOAD NIBBLE DATA 5 INTO SELECTED STACK LOCATION
;          2344
;          2345          [6] :
;          2346              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_6];
;          2347                                                  !LOAD NIBBLE DATA 6 INTO SELECTED STACK LOCATION
;          2348
;          2349          [7] :
;          2350              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_7];
;          2351                                                  !LOAD NIBBLE DATA 7 INTO SELECTED STACK LOCATION
;          2352
;          2353          [8] :
;          2354              stack [ .STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_8];
;          2355                                                  !LOAD NIBBLE DATA 8 INTO SELECTED STACK LOCATION
;          2356
;          2357          [9] :
;          2358              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_9];
;          2359                                                  !LOAD NIBBLE DATA 9 INTO SELECTED STACK LOCATION
;          2360          tes;
;          2361
;          2362      end;
;
;
;
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION

004716  052  052  052  P.AAA:  .ASCII  /***/
004721  052  052  052         .ASCII  /***/
004724  052  052  052         .ASCII  /***/
004727  040  115  101         .ASCII  / MA/
004732  116  125  101         .ASCII  /NUA/
004735  114  040  111         .ASCII  /L I/
004740  116  124  105         .ASCII  /NTE/
004743  122  126  105         .ASCII  /RVE/
004746  116  124  111         .ASCII  /NTI/
004751  117  116  040         .ASCII  /ON /
004754  124  105  123         .ASCII  /TES/
004757  124  040  062         .ASCII  /T 2/
004762  061  040  120         .ASCII  /1 P/
004765  117  127  105         .ASCII  /OWE/
004770  122  040  106         .ASCII  /R F/
004773  101  111  114         .ASCII  /AIL/
004776  040  052  052         .ASCII  / **/
005001  052  052  052         .ASCII  /***/
005004  052  052  052         .ASCII  /***/
005007  052  000  000         .ASCII  /*/<00><00>
005012  052  052  052  P.AAB:  .ASCII  /***/
005015  052  052  052         .ASCII  /***/
005020  052  052  052         .ASCII  /***/
005023  040  115  101         .ASCII  / MA/
005026  116  125  101         .ASCII  /NUA/
005031  114  040  111         .ASCII  /L I/
005034  116  124  105         .ASCII  /NTE/
005037  122  126  105         .ASCII  /RVE/
005042  116  124  111         .ASCII  /NTI/
005045  117  116  040         .ASCII  /ON /
005050  124  105  123         .ASCII  /TES/
005053  124  040  066         .ASCII  /T 6/
005056  061  040  102         .ASCII  /1 B/
005061  101  124  124         .ASCII  /ATT/
005064  105  122  131         .ASCII  /ERY/
005067  040  102  101         .ASCII  / BA/
005072  103  113  055         .ASCII  /CK-/
005075  125  120  040         .ASCII  /UP /
005100  052  052  052         .ASCII  /***/
005103  052  052  052         .ASCII  /***/
005106  052  052  052         .ASCII  /***/
005111  000                   .ASCII  <00>
005112  127  111  124  P.AAC:  .ASCII  /WIT/
005115  110  040  102         .ASCII  /H B/
005120  101  124  124         .ASCII  /ATT/
005123  105  122  131         .ASCII  /ERY/
005126  040  102  101         .ASCII  / BA/
005131  103  113  055         .ASCII  /CK-/
005134  125  120  040         .ASCII  /UP /
005137  117  116  040         .ASCII  /ON /
005142  124  125  122         .ASCII  /TUR/
005145  116  040  102         .ASCII  /N B/
```

```
1013                                    ;ML4AD
1014                                    ;               ROUTINE DECLARATION SECTION
1015
1016 005150    117     130     040             .ASCII  /OX /
1017 005153    101     103     040             .ASCII  /AC /
1018 005156    103     111     122             .ASCII  /CIR/
1019 005161    103     125     111             .ASCII  /CUI/
1020 005164    124     040     102             .ASCII  /T B/
1021 005167    122     105     101             .ASCII  /REA/
1022 005172    113     105     122             .ASCII  /KER/
1023 005175    040     117     106             .ASCII  / OF/
1024 005200    106     000                     .ASCII  /F/<00>
1025 005202    127     111     124     P.AAD:  .ASCII  /WIT/
1026 005205    110     040     102             .ASCII  /H B/
1027 005210    101     124     124             .ASCII  /ATT/
1028 005213    105     122     131             .ASCII  /ERY/
1029 005216    040     102     101             .ASCII  / BA/
1030 005221    103     113     055             .ASCII  /CK-/
1031 005224    125     120     040             .ASCII  /UP /
1032 005227    117     106     106             .ASCII  /OFF/
1033 005232    040     124     125             .ASCII  / TU/
1034 005235    122     116     040             .ASCII  /RN /
1035 005240    102     117     130             .ASCII  /BOX/
1036 005243    040     101     103             .ASCII  / AC/
1037 005246    040     103     111             .ASCII  / CI/
1038 005251    122     103     125             .ASCII  /RCU/
1039 005254    111     124     040             .ASCII  /IT /
1040 005257    102     122     105             .ASCII  /BRE/
1041 005262    101     113     105             .ASCII  /AKE/
1042 005265    122     040     117             .ASCII  /R O/
1043 005270    116     000                     .ASCII  /N/<00>
1044 005272    125     116     123     P.AAE:  .ASCII  /UNS/
1045 005275    040     102     111             .ASCII  / BI/
1046 005300    124     040     116             .ASCII  /T N/
1047 005303    117     124     040             .ASCII  /OT /
1048 005306    123     105     124             .ASCII  /SET/
1049 005311    040     104     125             .ASCII  / DU/
1050 005314    122     111     116             .ASCII  /RIN/
1051 005317    107     040     101             .ASCII  /G A/
1052 005322    103     040     120             .ASCII  /C P/
1053 005325    117     127     105             .ASCII  /OWE/
1054 005330    122     040     106             .ASCII  /R F/
1055 005333    101     111     114             .ASCII  /AIL/
1056 005336    000     000                     .ASCII  <00><00>
1057 005340    126     126     040     P.AAF:  .ASCII  /VV /
1058 005343    102     111     124             .ASCII  /BIT/
1059 005346    040     116     117             .ASCII  / NO/
1060 005351    124     040     123             .ASCII  /T S/
1061 005354    105     124     040             .ASCII  /ET /
1062 005357    101     106     124             .ASCII  /AFT/
1063 005362    105     122     040             .ASCII  /ER /
1064 005365    122     104     137             .ASCII  /RD_/
1065 005370    111     116     137             .ASCII  /IN_/
1066 005373    120     122     105             .ASCII  /PRE/
1067 005376    123     105     124             .ASCII  /SET/
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION

005401  040  106  125          .ASCII  / FU/
005404  116  103  124          .ASCII  /NCT/
005407  111  117  116          .ASCII  /ION/
005412  000  000               .ASCII  <00><00>
005414  126  126  040  P.AAG:   .ASCII  /VV /
005417  102  111  124          .ASCII  /BIT/
005422  040  103  114          .ASCII  / CL/
005425  105  101  122          .ASCII  /EAR/
005430  105  104  040          .ASCII  /ED /
005433  101  106  124          .ASCII  /AFT/
005436  105  122  040          .ASCII  /ER /
005441  122  104  055          .ASCII  /RD-/
005444  111  116  055          .ASCII  /IN-/
005447  120  122  105          .ASCII  /PRE/
005452  123  105  124          .ASCII  /SET/
005455  040  106  125          .ASCII  / FU/
005460  116  103  124          .ASCII  /NCT/
005463  111  117  116          .ASCII  /ION/
005466  000  000               .ASCII  <00><00>
005470  115  105  115  P.AAH:   .ASCII  /MEM/
005473  117  122  131          .ASCII  /ORY/
005476  040  116  117          .ASCII  / NO/
005501  124  040  111          .ASCII  /T I/
005504  116  111  124          .ASCII  /NIT/
005507  111  101  114          .ASCII  /IAL/
005512  111  132  105          .ASCII  /IZE/
005515  104  040  101          .ASCII  /D A/
005520  106  124  105          .ASCII  /FTE/
005523  122  040  120          .ASCII  /R P/
005526  117  127  105          .ASCII  /OWE/
005531  122  040  125          .ASCII  /R U/
005534  120  040  127          .ASCII  /P W/
005537  111  124  110          .ASCII  /ITH/
005542  040  102  101          .ASCII  / BA/
005545  104  040  102          .ASCII  /D B/
005550  101  124  124          .ASCII  /ATT/
005553  105  122  131          .ASCII  /ERY/
005556  040  102  101          .ASCII  / BA/
005561  103  113  055          .ASCII  /CK-/
005564  125  120  000          .ASCII  /UP/<00>
005567  000                    .ASCII  <00>
005570  122  105  107  P.AAI:   .ASCII  /REG/
005573  111  123  124          .ASCII  /IST/
005576  105  122  123          .ASCII  /ERS/
005601  040  116  117          .ASCII  / NO/
005604  124  040  103          .ASCII  /T C/
005607  114  105  101          .ASCII  /LEA/
005612  122  105  104          .ASCII  /RED/
005615  040  102  131          .ASCII  / BY/
005620  040  115  102          .ASCII  / MB/
005623  137  104  111          .ASCII  / DI/
005626  123  040  101          .ASCII  /S A/
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION
1128 005631   106   124   105          .ASCII  /FTE/
1129 005634   122   040   120          .ASCII  /R P/
1130 005637   117   127   105          .ASCII  /OWE/
1131 005642   122   040   106          .ASCII  /R F/
1132 005645   101   111   114          .ASCII  /AIL/
1133 005650   000   000                .ASCII  <00><00>
1134 005652   123   103   040  P.AAJ:   .ASCII  /SC /
1135 005655   102   111   124          .ASCII  /BIT/
1136 005660   040   123   105          .ASCII  / SE/
1137 005663   124   040   104          .ASCII  /T D/
1138 005666   125   122   111          .ASCII  /URI/
1139 005671   116   107   040          .ASCII  /NG /
1140 005674   127   122   111          .ASCII  /WRI/
1141 005677   124   105   040          .ASCII  /TE /
1142 005702   103   110   105          .ASCII  /CHE/
1143 005705   103   113   040          .ASCII  /CK /
1144 005710   124   122   101          .ASCII  /TRA/
1145 005713   116   123   106          .ASCII  /NSF/
1146 005716   105   122   000          .ASCII  /ER/<00>
1147 005721   000                      .ASCII  <00>
1148 005722   123   103   040  P.AAK:   .ASCII  /SC /
1149 005725   102   111   124          .ASCII  /BIT/
1150 005730   040   123   105          .ASCII  / SE/
1151 005733   124   040   104          .ASCII  /T D/
1152 005736   125   122   111          .ASCII  /URI/
1153 005741   116   107   040          .ASCII  /NG /
1154 005744   127   122   111          .ASCII  /WRI/
1155 005747   124   105   040          .ASCII  /TE /
1156 005752   124   122   101          .ASCII  /TRA/
1157 005755   116   123   106          .ASCII  /NSF/
1158 005760   105   122   000          .ASCII  /ER/<00>
1159 005763   000                      .ASCII  <00>
1160 005764   126   126   040  P.AAL:   .ASCII  /VV /
1161 005767   102   111   124          .ASCII  /BIT/
1162 005772   040   123   124          .ASCII  / ST/
1163 005775   111   114   114          .ASCII  /ILL/
1164 006000   040   123   105          .ASCII  / SE/
1165 006003   124   040   101          .ASCII  /T A/
1166 006006   106   124   105          .ASCII  /FTE/
1167 006011   122   040   120          .ASCII  /R P/
1168 006014   117   127   105          .ASCII  /OWE/
1169 006017   122   040   114          .ASCII  /R L/
1170 006022   117   123   123          .ASCII  /OSS/
1171 006025   040   127   111          .ASCII  / WI/
1172 006030   124   110   040          .ASCII  /TH /
1173 006033   102   101   104          .ASCII  /BAD/
1174 006036   040   102   101          .ASCII  / BA/
1175 006041   124   124   105          .ASCII  /TTE/
1176 006044   122   131   040          .ASCII  /RY /
1177 006047   102   101   103          .ASCII  /BAC/
1178 006052   113   055   125          .ASCII  /K-U/
1179 006055   120   000   000          .ASCII  /P/<00><00>
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
006060  127  111  124  P.AAM:  .ASCII  /WIT/
006063  110  040  102          .ASCII  /H B/
006066  101  124  124          .ASCII  /ATT/
006071  105  122  131          .ASCII  /ERY/
006074  040  102  101          .ASCII  / BA/
006077  103  113  055          .ASCII  /CK-/
006102  125  120  040          .ASCII  /UP /
006105  117  116  040          .ASCII  /ON /
006110  124  125  122          .ASCII  /TUR/
006113  116  040  102          .ASCII  /N B/
006116  117  130  040          .ASCII  /OX /
006121  101  103  040          .ASCII  /AC /
006124  103  111  122          .ASCII  /CIR/
006127  103  125  111          .ASCII  /CUI/
006132  124  040  102          .ASCII  /T B/
006135  122  105  101          .ASCII  /REA/
006140  113  105  122          .ASCII  /KER/
006143  040  117  116          .ASCII  / ON/
006146  000  000               .ASCII  <00><00>
006150  126  126  040  P.AAN:  .ASCII  /VV /
006153  102  111  124          .ASCII  /BIT/
006156  040  103  114          .ASCII  / CL/
006161  105  101  122          .ASCII  /EAR/
006164  105  104  040          .ASCII  /ED /
006167  101  106  124          .ASCII  /AFT/
006172  105  122  040          .ASCII  /ER /
006175  120  117  127          .ASCII  /POW/
006200  105  122  040          .ASCII  /ER /
006203  106  101  111          .ASCII  /FAI/
006206  114  040  127          .ASCII  /L W/
006211  111  124  110          .ASCII  /ITH/
006214  040  107  117          .ASCII  / GO/
006217  117  104  040          .ASCII  /OD /
006222  102  101  124          .ASCII  /BAT/
006225  124  105  122          .ASCII  /TER/
006230  131  040  102          .ASCII  /Y B/
006233  101  103  113          .ASCII  /ACK/
006236  055  125  120          .ASCII  /-UP/
006241  000                    .ASCII  <00>
006242  115  105  115  P.AAO:  .ASCII  /MEM/
006245  117  122  131          .ASCII  /ORY/
006250  040  111  116          .ASCII  / IN/
006253  111  124  105          .ASCII  /ITE/
006256  104  040  101          .ASCII  /D A/
006261  106  124  105          .ASCII  /FTE/
006264  122  040  120          .ASCII  /R P/
006267  117  127  105          .ASCII  /OWE/
006272  122  040  106          .ASCII  /R F/
006275  101  111  114          .ASCII  /AIL/
006300  040  127  111          .ASCII  / WI/
006303  124  110  040          .ASCII  /TH /
006306  107  117  117          .ASCII  /GOO/
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION
1237
1238
1239
1240 006311  1J4  040  102             .ASCII  /D B/
1241 006314  101  124  124             .ASCII  /ATT/
1242 006317  105  122  131             .ASCII  /ERY/
1243 006322  040  102  101             .ASCII  / BA/
1244 006325  103  113  055             .ASCII  /CK-/
1245 006330  125  120  000             .ASCII  /UP/<00>
1246 006333  000                       .ASCII  <00>
1247 006334  102  101  124  P.AAP:     .ASCII  /BAT/
1248 006337  124  105  122             .ASCII  /TER/
1249 006342  131  040  102             .ASCII  /Y B/
1250 006345  101  103  113             .ASCII  /ACK/
1251 006350  055  125  120             .ASCII  /-UP/
1252 006353  040  106  101             .ASCII  / FA/
1253 006356  111  114  105             .ASCII  /ILE/
1254 006361  104  040  124             .ASCII  /D T/
1255 006364  117  040  115             .ASCII  /O M/
1256 006367  101  111  116             .ASCII  /AIN/
1257 006372  124  101  111             .ASCII  /TAI/
1258 006375  116  040  115             .ASCII  /N M/
1259 006400  105  115  117             .ASCII  /EMO/
1260 006403  122  131  040             .ASCII  /RY /
1261 006406  111  116  124             .ASCII  /INT/
1262 006411  105  107  122             .ASCII  /EGR/
1263 006414  111  124  131             .ASCII  /ITY/
1264 006417  040  104  125             .ASCII  / DU/
1265 006422  122  111  116             .ASCII  /RIN/
1266 006425  107  040  120             .ASCII  /G P/
1267 006430  117  127  105             .ASCII  /OWE/
1268 006433  122  040  106             .ASCII  /R F/
1269 006436  101  111  114             .ASCII  /AIL/
1270 006441  000                       .ASCII  <00>
1271 006442  045  101  105  P.AAQ:     .ASCII  /%AE/
1272 006445  130  120  105             .ASCII  /XPE/
1273 006450  103  124  105             .ASCII  /CTE/
1274 006453  104  072  040             .ASCII  /D: /
1275 006456  045  117  066             .ASCII  /%O6/
1276 006461  045  101  040             .ASCII  /%A /
1277 006464  040  040  040             .ASCII  /   /
1278 006467  122  105  101             .ASCII  /REA/
1279 006472  104  072  040             .ASCII  /D: /
1280 006475  045  117  066             .ASCII  /%O6/
1281 006500  045  116  045             .ASCII  /%N%/
1282 006503  116  000  000             .ASCII  /N/<00><00>
1283 006506  045  101  107  P.AAR:     .ASCII  /%AG/
1284 006511  117  117  104             .ASCII  /OOD/
1285 006514  040  104  101             .ASCII  / DA/
1286 006517  124  101  072             .ASCII  /TA:/
1287 006522  040  045  117             .ASCII  / %O/
1288 006525  066  045  101             .ASCII  /6%A/
1289 006530  040  040  040             .ASCII  /   /
1290 006533  040  102  101             .ASCII  / BA/
1291 006536  104  040  104             .ASCII  /D D/
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
1293
1294                                    ;ML4AD
1295                                    ;               ROUTINE DECLARATION SECTION
1296 006541  101  124  101                    .ASCII  /ATA/
1297 006544  072  040  045                    .ASCII  /: %/
1298 006547  117  066  045                    .ASCII  /06%/
1299 006552  101  040  040                    .ASCII  /A  /
1300 006555  040  040  130                    .ASCII  /  X/
1301 006560  117  122  072                    .ASCII  /OR:/
1302 006563  040  045  117                    .ASCII  / %O/
1303 006566  066  045  116                    .ASCII  /6%N/
1304 006571  045  116  000                    .ASCII  /%N/<00>
1305 006574  045  101  104  P.AAS:           .ASCII  /%AD/
1306 006577  122  111  126                    .ASCII  /RIV/
1307 006602  105  040  123                    .ASCII  /E S/
1308 006605  116  072  040                    .ASCII  /N: /
1309 006610  045  117  066                    .ASCII  /%O6/
1310 006613  045  116  045                    .ASCII  /%N%/
1311 006616  116  000                        .ASCII  /N/<00>
1312 006620  045  101  102  P.AAT:           .ASCII  /%AB/
1313 006623  111  124  040                    .ASCII  /IT /
1314 006626  111  116  040                    .ASCII  /IN /
1315 006631  105  122  122                    .ASCII  /ERR/
1316 006634  117  122  072                    .ASCII  /OR:/
1317 006637  040  045  117                    .ASCII  / %O/
1318 006642  066  045  116                    .ASCII  /6%N/
1319 006645  045  116  000                    .ASCII  /%N/<00>
1320 006650  045  101  107  P.AAU:           .ASCII  /%AG/
1321 006653  117  117  104                    .ASCII  /OOD/
1322 006656  040  116  111                    .ASCII  / NI/
1323 006661  102  040  104                    .ASCII  /B D/
1324 006664  101  124  101                    .ASCII  /ATA/
1325 006667  072  040  045                    .ASCII  /: %/
1326 006672  117  062  045                    .ASCII  /O2%/
1327 006675  101  040  040                    .ASCII  /A  /
1328 006700  040  040  102                    .ASCII  /  B/
1329 006703  101  104  040                    .ASCII  /AD /
1330 006706  116  111  102                    .ASCII  /NIB/
1331 006711  040  104  101                    .ASCII  / DA/
1332 006714  124  101  072                    .ASCII  /TA:/
1333 006717  040  045  117                    .ASCII  / %O/
1334 006722  062  045  101                    .ASCII  /2%A/
1335 006725  040  040  040                    .ASCII  /   /
1336 006730  040  116  111                    .ASCII  / NI/
1337 006733  102  040  120                    .ASCII  /B P/
1338 006736  117  123  072                    .ASCII  /OS:/
1339 006741  040  045  117                    .ASCII  / %O/
1340 006744  064  045  116                    .ASCII  /4%N/
1341 006747  045  116  000                    .ASCII  /%N/<00>
1342 006752  045  101  116  P.AAV:           .ASCII  /%AN/
1343 006755  [illegible]  102  040           .ASCII  /IB /
1344 006760  111  116  040                    .ASCII  /IN /
1345 006763  105  122  122                    .ASCII  /ERR/
1346 006766  117  122  072                    .ASCII  /OR:/
1347 006771  040  045  104                    .ASCII  / %D/
```

```
1349                                    ;ML4AD
1350                                    ;               ROUTINE DECLARATION SECTION
1351
1352 006774  064  045  116             .ASCII  /4%N/
1353 006777  045  116  000             .ASCII  /%N/<00>
1354 007002  045  101  106  P.AAW:     .ASCII  /%AF/
1355 007005  101  111  114             .ASCII  /AIL/
1356 007010  105  104  040             .ASCII  /ED /
1357 007013  101  124  072             .ASCII  /AT:/
1358 007016  040  045  117             .ASCII  / %O/
1359 007021  066  045  116             .ASCII  /6%N/
1360 007024  045  116  000             .ASCII  /%N/<00>
1361 007027  000                       .ASCII  <00>
1362 007030  045  101  122  P.AAX:     .ASCII  /%AR/
1363 007033  105  120  114             .ASCII  /EPL/
1364 007036  101  103  105             .ASCII  /ACE/
1365 007041  040  101  122             .ASCII  / AR/
1366 007044  122  040  115             .ASCII  /R M/
1367 007047  117  104  072             .ASCII  /OD:/
1368 007052  040  045  104             .ASCII  / %D/
1369 007055  062  045  116             .ASCII  /2%N/
1370 007060  045  116  000             .ASCII  /%N/<00>
1371 007063  000                       .ASCII  <00>
1372 007064  045  101  106  P.AAY:     .ASCII  /%AF/
1373 007067  101  111  114             .ASCII  /AIL/
1374 007072  105  104  040             .ASCII  /ED /
1375 007075  101  124  040             .ASCII  /AT /
1376 007100  104  123  101             .ASCII  /DSA/
1377 007103  072  040  045             .ASCII  /: %/
1378 007106  117  066  045             .ASCII  /O6%/
1379 007111  116  045  116             .ASCII  /N%N/
1380 007114  000  000                  .ASCII  <00><00>
1381 007116  045  101  102  P.AAZ:     .ASCII  /%AB/
1382 007121  111  124  074             .ASCII  /IT</
1383 007124  061  065  072             .ASCII  /15:/
1384 007127  061  060  076             .ASCII  /10>/
1385 007132  072  040  045             .ASCII  /: %/
1386 007135  102  066  045             .ASCII  /B6%/
1387 007140  101  040  040             .ASCII  /A  /
1388 007143  040  040  102             .ASCII  /  B/
1389 007146  111  124  074             .ASCII  /IT</
1390 007151  071  072  060             .ASCII  /9:0/
1391 007154  076  072  040             .ASCII  />: /
1392 007157  045  102  061             .ASCII  /%B1/
1393 007162  060  045  116             .ASCII  /0%N/
1394 007165  045  116  000             .ASCII  /%N/<00>
1395 007170  045  101  106  P.ABA:     .ASCII  /%AF/
1396 007173  101  111  114             .ASCII  /AIL/
1397 007176  111  116  107             .ASCII  /ING/
1398 007201  040  122  105             .ASCII  / RE/
1399 007204  107  040  101             .ASCII  /G A/
1400 007207  104  122  123             .ASCII  /DRS/
1401 007212  072  040  045             .ASCII  /: %/
1402 007215  117  066  045             .ASCII  /O6%/
1403 007220  116  045  116             .ASCII  /N%N/
```

```
1405                                    ;ML4AD
1406                                    ;
1407                                    ;       ROUTINE DECLARATION SECTION
1408 007223  000                               .ASCII  <00>
1409 007224  045  101  106  P.ABB:      .ASCII  /%AF/
1410 007227  101  111  114                    .ASCII  /AIL/
1411 007232  111  116  107                    .ASCII  /ING/
1412 007235  040  106  125                    .ASCII  / FU/
1413 007240  116  103  072                    .ASCII  /NC:/
1414 007243  040  045  117                    .ASCII  / %O/
1415 007246  066  045  116                    .ASCII  /6%N/
1416 007251  045  116  000                    .ASCII  /%N/<00>
1417 007254  040  045  101  P.ABC:      .ASCII  / %A/
1418 007257  117  106  106                    .ASCII  /OFF/
1419 007262  137  123  105                    .ASCII  / SE/
1420 007265  124  040  103                    .ASCII  /T C/
1421 007270  116  124  040                    .ASCII  /NT /
1422 007273  106  117  122                    .ASCII  /FOR/
1423 007276  040  116  111                    .ASCII  / NI/
1424 007301  102  040  072                    .ASCII  /B :/
1425 007304  040  045  104                    .ASCII  / %D/
1426 007307  062  040  045                    .ASCII  /2 %/
1427 007312  101  040  075                    .ASCII  /A =/
1428 007315  040  045  104                    .ASCII  / %D/
1429 007320  062  040  045                    .ASCII  /2 %/
1430 007323  116  045  116                    .ASCII  /N%N/
1431 007326  000  000                          .ASCII  <00><00>
1432 007330  045  101  127  P.ABD:      .ASCII  /%AW/
1433 007333  122  117  124                    .ASCII  /ROT/
1434 007336  105  072  040                    .ASCII  /E: /
1435 007341  045  104  062                    .ASCII  /%D2/
1436 007344  045  101  040                    .ASCII  /%A /
1437 007347  040  040  040                    .ASCII  /   /
1438 007352  122  105  101                    .ASCII  /REA/
1439 007355  104  072  040                    .ASCII  /D: /
1440 007360  045  104  062                    .ASCII  /%D2/
1441 007363  045  116  045                    .ASCII  /%N%/
1442 007366  116  000                          .ASCII  /N/<00>
1443 007370  045  101  116  P.ABE:      .ASCII  /%AN/
1444 007373  111  102  102                    .ASCII  /IBB/
1445 007376  114  105  123                    .ASCII  /LES/
1446 007401  040  130  106                    .ASCII  / XF/
1447 007404  105  122  105                    .ASCII  /ERE/
1448 007407  104  040  102                    .ASCII  /D B/
1449 007412  105  106  117                    .ASCII  /EFO/
1450 007415  122  105  040                    .ASCII  /RE /
1451 007420  105  122  122                    .ASCII  /ERR/
1452 007423  117  122  072                    .ASCII  /OR:/
1453 007426  040  045  104                    .ASCII  / %D/
1454 007431  063  045  116                    .ASCII  /3%N/
1455 007434  000  000                          .ASCII  <00><00>
1456 007436  045  101  106  P.ABF:      .ASCII  /%AF/
1457 007441  101  111  114                    .ASCII  /AIL/
1458 007444  111  116  107                    .ASCII  /ING/
1459 007447  040  122  105                    .ASCII  / RE/
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
1461                                    ;ML4AD
1462                                    ;               ROUTINE DECLARATION SECTION
1463
1464 007452  107  072  040             .ASCII  /G: /
1465 007455  045  117  066             .ASCII  /%06/
1466 007460  045  101  040             .ASCII  /%A /
1467 007463  107  117  117             .ASCII  /GOO/
1468 007466  104  040  104             .ASCII  /D D/
1469 007471  101  124  101             .ASCII  /ATA/
1470 007474  072  040  045             .ASCII  /: %/
1471 007477  117  066  045             .ASCII  /06%/
1472 007502  101  040  102             .ASCII  /A B/
1473 007505  101  104  040             .ASCII  /AD /
1474 007510  104  101  124             .ASCII  /DAT/
1475 007513  101  072  040             .ASCII  /A: /
1476 007516  045  117  066             .ASCII  /%06/
1477 007521  045  116  045             .ASCII  /%N%/
1478 007524  116  000                  .ASCII  /N/<00>
1479 007526  045  116  045  P.ABG:     .ASCII  /%N%/
1480 007531  101  104  111             .ASCII  /ADI/
1481 007534  101  107  116             .ASCII  /AGN/
1482 007537  117  123  111             .ASCII  /OSI/
1483 007542  116  107  040             .ASCII  /NG /
1484 007545  125  116  111             .ASCII  /UNI/
1485 007550  124  040  045             .ASCII  /T %/
1486 007553  117  061  045             .ASCII  /01%/
1487 007556  116  045  116             .ASCII  /N%N/
1488 007561  000                       .ASCII  <00>
1489 007562  045  101  124  P.ABH:     .ASCII  /%AT/
1490 007565  111  115  105             .ASCII  /IME/
1491 007570  104  040  117             .ASCII  /D O/
1492 007573  125  124  040             .ASCII  /UT /
1493 007576  104  125  122             .ASCII  /DUR/
1494 007601  111  116  107             .ASCII  /ING/
1495 007604  040  115  102             .ASCII  / MB/
1496 007607  125  123  040             .ASCII  /US /
1497 007612  045  117  062             .ASCII  /%02/
1498 007615  045  101  040             .ASCII  /%A /
1499 007620  106  125  116             .ASCII  /FUN/
1500 007623  103  045  116             .ASCII  /C%N/
1501 007626  045  116  000             .ASCII  /%N/<00>
1502 007631  000                       .ASCII  <00>
1503 007632  045  101  103  P.ABI:     .ASCII  /%AC/
1504 007635  122  103  040             .ASCII  /RC /
1505 007640  107  105  116             .ASCII  /GEN/
1506 007643  040  075  040             .ASCII  / = /
1507 007646  102  072  040             .ASCII  /B: /
1508 007651  045  102  066             .ASCII  /%B6/
1509 007654  045  101  040             .ASCII  /%A /
1510 007657  101  072  040             .ASCII  /A: /
1511 007662  045  102  066             .ASCII  /%B6/
1512 007665  045  101  040             .ASCII  /%A /
1513 007670  120  072  040             .ASCII  /P: /
1514 007673  045  102  066             .ASCII  /%B6/
1515 007676  045  116  000             .ASCII  /%N/<00>
```

```
                                        ;%L4AD                                          29-Mar-1982 16:23:04    TOPS
                                        ;               ROUTINE DECLARATION SECTION     29-Mar-1982 16:21:03    PA:<
 1517
 1518
 1519
 1520 007701  000                                       .ASCII  <00>
 1521 007702  045     101     103        P.ABJ:         .ASCII  /%AC/
 1522 007705  122     103     040                       .ASCII  /RC /
 1523 007710  103     101     114                       .ASCII  /CAL/
 1524 007713  040     075     040                       .ASCII  / = /
 1525 007716  102     072     040                       .ASCII  /B: /
 1526 007721  045     102     066                       .ASCII  /%B6/
 1527 007724  045     101     040                       .ASCII  /%A /
 1528 007727  101     072     040                       .ASCII  /A: /
 1529 007732  045     102     066                       .ASCII  /%B6/
 1530 007735  045     101     040                       .ASCII  /%A /
 1531 007740  120     072     040                       .ASCII  /P: /
 1532 007743  045     102     066                       .ASCII  /%B6/
 1533 007746  045     116     045                       .ASCII  /%N%/
 1534 007751  116     000     000                       .ASCII  /N/<00><00>
 1535 007754  045     101     106        P.ABK:         .ASCII  /%AF/
 1536 007757  101     111     114                       .ASCII  /AIL/
 1537 007762  105     104     040                       .ASCII  /ED /
 1538 007765  101     124     040                       .ASCII  /AT /
 1539 007770  120     114     117                       .ASCII  /PLO/
 1540 007773  107     072     040                       .ASCII  /G: /
 1541 007776  045     104     062                       .ASCII  /%D2/
 1542 010001  045     101     040                       .ASCII  /%A /
 1543 010004  103     110     101                       .ASCII  /CHA/
 1544 010007  116     116     105                       .ASCII  /NNE/
 1545 010012  114     072     040                       .ASCII  /L: /
 1546 010015  045     104     062                       .ASCII  /%D2/
 1547 010020  045     116     045                       .ASCII  /%N%/
 1548 010023  116     000     000                       .ASCII  /N/<00><00>
 1549 010026  045     101     106        P.ABL:         .ASCII  /%AF/
 1550 010031  101     111     114                       .ASCII  /AIL/
 1551 010034  105     104     040                       .ASCII  /ED /
 1552 010037  101     124     040                       .ASCII  /AT /
 1553 010042  127     122     104                       .ASCII  /WRD/
 1554 010045  072     040     045                       .ASCII  /: %/
 1555 010050  104     062     045                       .ASCII  /D2%/
 1556 010053  101     040     102                       .ASCII  /A B/
 1557 010056  111     124     072                       .ASCII  /IT:/
 1558 010061  040     045     104                       .ASCII  / %D/
 1559 010064  062     045     116                       .ASCII  /2%N/
 1560 010067  045     116     000                       .ASCII  /%N/<00>
 1561 010072  045     101     122        P.ABM:         .ASCII  /%AR/
 1562 010075  105     107     111                       .ASCII  /EGI/
 1563 010100  123     124     105                       .ASCII  /STE/
 1564 010103  122     045     123                       .ASCII  /R%S/
 1565 010106  063     045     101                       .ASCII  /3%A/
 1566 010111  101     104     104                       .ASCII  /ADD/
 1567 010114  122     105     123                       .ASCII  /RES/
 1568 010117  123     045     123                       .ASCII  /S%S/
 1569 010122  067     045     101                       .ASCII  /7%A/
 1570 010125  103     117     116                       .ASCII  /CON/
 1571 010130  124     105     116                       .ASCII  /TEN/
```

```
1573                                    ;PL4AD
1574                                    ;               ROUTINE DECLARATION SECTION
1575
1576 010133  124  123  045              .ASCII  /TS%/
1577 010136  116  045  116              .ASCII  /N%N/
1578 010141  000                        .ASCII  <00>
1579 010142  045  123  045  P.ABN:      .ASCII  /%S%/
1580 010145  124  045  123              .ASCII  /T%S/
1581 010150  065  045  117              .ASCII  /5%O/
1582 010153  066  045  123              .ASCII  /6%S/
1583 010156  064  045  117              .ASCII  /4%O/
1584 010161  061  066  045              .ASCII  /16%/
1585 010164  116  000                   .ASCII  /N/<00>
1586 010166  045  123  064  P.ABO:      .ASCII  /%S4/
1587 010171  045  124  045              .ASCII  /%T%/
1588 010174  101  072  045              .ASCII  /A:%/
1589 010177  123  045  117              .ASCII  /S%O/
1590 010202  066  045  116              .ASCII  /6%N/
1591 010205  000                        .ASCII  <00>
1592 010206  045  123  064  P.ABP:      .ASCII  /%S4/
1593 010211  045  124  045              .ASCII  /%T%/
1594 010214  101  072  045              .ASCII  /A:%/
1595 010217  123  045  104              .ASCII  /S%D/
1596 010222  061  045  104              .ASCII  /1%D/
1597 010225  061  045  104              .ASCII  /1%D/
1598 010230  061  045  104              .ASCII  /1%D/
1599 010233  061  045  116              .ASCII  /1%N/
1600 010236  000  000                   .ASCII  <00><00>
1601 010240  045  124  045  P.ABQ:      .ASCII  /%T%/
1602 010243  116  000  000              .ASCII  /N/<00><00>
1603 010246  045  124  045  P.ABR:      .ASCII  /%T%/
1604 010251  124  045  116              .ASCII  /T%N/
1605 010254  000  000                   .ASCII  <00><00>
1606 010256  045  124  045  P.ABS:      .ASCII  /%T%/
1607 010261  124  045  124              .ASCII  /T%T/
1608 010264  045  116  000              .ASCII  /%N/<00>
1609 010267  000                        .ASCII  <00>
1610 010270  045  124  045  P.ABT:      .ASCII  /%T%/
1611 010273  124  045  124              .ASCII  /T%T/
1612 010276  045  124  045              .ASCII  /%T%/
1613 010301  116  000  000              .ASCII  /N/<00><00>
1614 010304  045  124  045  P.ABU:      .ASCII  /%T%/
1615 010307  124  045  124              .ASCII  /T%T/
1616 010312  045  124  045              .ASCII  /%T%/
1617 010315  124  045  116              .ASCII  /T%N/
1618 010320  000  000                   .ASCII  <00><00>
1619 010322  045  124  045  P.ABV:      .ASCII  /%T%/
1620 010325  124  045  124              .ASCII  /T%T/
1621 010330  045  124  045              .ASCII  /%T%/
1622 010333  124  045  124              .ASCII  /T%T/
1623 010336  045  116  000              .ASCII  /%N/<00>
1624 010341  000                        .ASCII  <00>
1625 010342  045  124  045  P.ABW:      .ASCII  /%T%/
1626 010345  124  045  124              .ASCII  /T%T/
1627 010350  045  124  045              .ASCII  /%T%/
```

```
1629                                    ;ML4AD
1630                                    ;               ROUTINE DECLARATION SECTION
1631
1632 010353  124  045  124             .ASCII  /T%T/
1633 010356  045  124  045             .ASCII  /%T%/
1634 010361  116  000  000             .ASCII  /N/<00><00>
1635 010364  045  124  045  P.ABX:     .ASCII  /%T%/
1636 010367  124  045  124             .ASCII  /T%T/
1637 010372  045  124  045             .ASCII  /%T%/
1638 010375  124  045  124             .ASCII  /T%T/
1639 010400  045  124  045             .ASCII  /%T%/
1640 010403  124  045  116             .ASCII  /T%N/
1641 010406  000  000                  .ASCII  <00><00>
1642 010410  045  124  045  P.ABY:     .ASCII  /%T%/
1643 010413  124  045  124             .ASCII  /T%T/
1644 010416  045  124  045             .ASCII  /%T%/
1645 010421  124  045  124             .ASCII  /T%T/
1646 010424  045  124  045             .ASCII  /%T%/
1647 010427  124  045  124             .ASCII  /T%T/
1648 010432  045  116  000             .ASCII  /%N/<00>
1649 010435  000                       .ASCII  <00>
1650 010436  045  124  045  P.ABZ:     .ASCII  /%T%/
1651 010441  124  045  124             .ASCII  /T%T/
1652 010444  045  124  045             .ASCII  /%T%/
1653 010447  124  045  124             .ASCII  /T%T/
1654 010452  045  124  045             .ASCII  /%T%/
1655 010455  124  045  124             .ASCII  /T%T/
1656 010460  045  124  045             .ASCII  /%T%/
1657 010463  116  000  000             .ASCII  /N/<00><00>
1658 010466  045  124  045  P.ACA:     .ASCII  /%T%/
1659 010471  124  045  124             .ASCII  /T%T/
1660 010474  045  124  045             .ASCII  /%T%/
1661 010477  124  045  124             .ASCII  /T%T/
1662 010502  045  124  045             .ASCII  /%T%/
1663 010505  124  045  124             .ASCII  /T%T/
1664 010510  045  124  045             .ASCII  /%T%/
1665 010513  124  045  116             .ASCII  /T%N/
1666 010516  000  000                  .ASCII  <00><00>
1667 010520  040  107  117  P.ACB:     .ASCII  / GO/
1668 010523  000                       .ASCII  <00>
1669 010524  040  104  122  P.ACC:     .ASCII  / DR/
1670 010527  126  137  122             .ASCII  /V_R/
1671 010532  104  131  000             .ASCII  /DY/<00>
1672 010535  000                       .ASCII  <00>
1673 010536  040  111  114  P.ACD:     .ASCII  / IL/
1674 010541  106  000  000             .ASCII  /F/<00><00>
1675 010544  040  117  120  P.ACE:     .ASCII  / OP/
1676 010547  111  000  000             .ASCII  /I/<00><00>
1677 010552  040  102  101  P.ACF:     .ASCII  / BA/
1678 010555  104  000  000             .ASCII  /D/<00><00>
1679 010560  040  107  117  P.ACG:     .ASCII  / GO/
1680 010563  117  104  000             .ASCII  /OD/<00>
1681 010566  040  120  101  P.ACH:     .ASCII  / PA/
1682 010571  122  111  124             .ASCII  /RIT/
1683 010574  131  040  116             .ASCII  /Y N/
```

```
                                        ;ML4AD
                                        ;                ROUTINE DECLARATION SECTION

1688 010577  117  124  000             .ASCII  /OT/<00>
1689 010602  040  107  105  P.ACI:     .ASCII  / GE/
1690 010605  116  105  122             .ASCII  /NER/
1691 010610  101  124  105             .ASCII  /ATE/
1692 010613  104  000  000             .ASCII  /D/<00><00>
1693 010616  040  104  105  P.ACJ:     .ASCII  / DE/
1694 010621  124  105  103             .ASCII  /TEC/
1695 010624  124  105  104             .ASCII  /TED/
1696 010627  000                       .ASCII  <00>
1697 010630  040  105  122  P.ACK:     .ASCII  / ER/
1698 010633  122  117  122             .ASCII  /ROR/
1699 010636  000  000                  .ASCII  <00><00>
1700 010640  040  101  106  P.ACL:     .ASCII  / AF/
1701 010643  124  105  122             .ASCII  /TER/
1702 010646  000  000                  .ASCII  <00><00>
1703 010650  040  104  125  P.ACM:     .ASCII  / DU/
1704 010653  122  111  116             .ASCII  /RIN/
1705 010656  107  000                  .ASCII  /G/<00>
1706 010660  040  101  124  P.ACN:     .ASCII  / AT/
1707 010663  000                       .ASCII  <00>
1708 010664  040  106  101  P.ACO:     .ASCII  / FA/
1709 010667  111  114  125             .ASCII  /ILU/
1710 010672  122  105  000             .ASCII  /RE/<00>
1711 010675  000                       .ASCII  <00>
1712 010676  040  101  124  P.ACP:     .ASCII  / AT/
1713 010701  101  000  000             .ASCII  /A/<00><00>
1714 010704  040  101  124  P.ACQ:     .ASCII  / AT/
1715 010707  124  116  000             .ASCII  /TN/<00>
1716 010712  040  127  122  P.ACR:     .ASCII  / WR/
1717 010715  111  124  111             .ASCII  /ITI/
1718 010720  116  107  000             .ASCII  /NG/<00>
1719 010723  000                       .ASCII  <00>
1720 010724  040  126  126  P.ACS:     .ASCII  / VV/
1721 010727  000                       .ASCII  <00>
1722 010730  040  106  125  P.ACT:     .ASCII  / FU/
1723 010733  116  103  000             .ASCII  /NC/<00>
1724 010736  040  124  122  P.ACU:     .ASCII  / TR/
1725 010741  105  000  000             .ASCII  /E/<00><00>
1726 010744  040  122  115  P.ACV:     .ASCII  / RM/
1727 010747  122  000  000             .ASCII  /R/<00><00>
1728 010752  040  105  130  P.ACW:     .ASCII  / EX/
1729 010755  103  105  123             .ASCII  /CES/
1730 010760  123  111  126             .ASCII  /SIV/
1731 010763  105  000  000             .ASCII  /E/<00><00>
1732 010766  040  115  102  P.ACX:     .ASCII  / MB/
1733 010771  125  123  000             .ASCII  /US/<00>
1734 010774  040  104  101  P.ACY:     .ASCII  / DA/
1735 010777  124  101  000             .ASCII  /TA/<00>
1736 011002  040  103  117  P.ACZ:     .ASCII  / CO/
1737 011005  116  124  111             .ASCII  /NTI/
1738 011010  116  125  111             .ASCII  /NUI/
1739 011013  124  131  000             .ASCII  /TY/<00>
```

```
                                        ;ML4AD
                                        ;                ROUTINE DECLARATION SECTION

1741
1742
1743
1744 011016  040  101  117  P.ADA:  .ASCII  / AO/
1745 011021  105  000  000           .ASCII  /E/<00><00>
1746 011024  040  114  102  P.ADB:  .ASCII  / LB/
1747 011027  124  000  000           .ASCII  /T/<00><00>
1748 011032  040  120  122  P.ADC:  .ASCII  / PR/
1749 011035  105  115  101           .ASCII  /EMA/
1750 011040  124  125  122           .ASCII  /TUR/
1751 011043  114  131  000           .ASCII  /LY/<00>
1752 011046  040  111  101  P.ADD:  .ASCII  / IA/
1753 011051  105  000  000           .ASCII  /E/<00><00>
1754 011054  040  111  116  P.ADE:  .ASCII  / IN/
1755 011057  103  122  105           .ASCII  /CRE/
1756 011062  115  105  116           .ASCII  /MEN/
1757 011065  124  000  000           .ASCII  /T/<00><00>
1758 011070  040  127  111  P.ADF:  .ASCII  / WI/
1759 011073  124  110  000           .ASCII  /TH/<00>
1760 011076  040  125  126  P.ADG:  .ASCII  / UV/
1761 011101  000                    .ASCII  <00>
1762 011102  040  125  116  P.ADH:  .ASCII  / UN/
1763 011105  123  000  000           .ASCII  /S/<00><00>
1764 011110  040  120  122  P.ADI:  .ASCII  / PR/
1765 011113  117  115  000           .ASCII  /OM/<00>
1766 011116  040  117  122  P.ADJ:  .ASCII  / OR/
1767 011121  000                    .ASCII  <00>
1768 011122  040  123  105  P.ADK:  .ASCII  / SE/
1769 011125  114  105  103           .ASCII  /LEC/
1770 011130  124  000                .ASCII  /T/<00>
1771 011132  040  122  105  P.ADL:  .ASCII  / RE/
1772 011135  107  000  000           .ASCII  /G/<00><00>
1773 011140  040  125  116  P.ADM:  .ASCII  / UN/
1774 011143  111  121  125           .ASCII  /IQU/
1775 011146  105  000                .ASCII  /E/<00>
1776 011150  040  061  064  P.ADN:  .ASCII  / 14/
1777 011153  000                    .ASCII  <00>
1778 011154  040  116  111  P.ADO:  .ASCII  / NI/
1779 011157  102  102  114           .ASCII  /BBL/
1780 011162  105  040  103           .ASCII  /E C/
1781 011165  116  124  000           .ASCII  /NT/<00>
1782 011170  040  107  124  P.ADP:  .ASCII  / GT/
1783 011173  122  000  000           .ASCII  /R/<00><00>
1784 011176  040  127  110  P.ADQ:  .ASCII  / WH/
1785 011201  111  114  105           .ASCII  /ILE/
1786 011204  000  000                .ASCII  <00><00>
1787 011206  040  124  122  P.ADR:  .ASCII  / TR/
1788 011211  105  000  000           .ASCII  /E/<00><00>
1789 011214  040  111  116  P.ADS:  .ASCII  / IN/
1790 011217  111  124  111           .ASCII  /ITI/
1791 011222  101  114  000           .ASCII  /AL/<00>
1792 011225  000                    .ASCII  <00>
1793 011226  040  117  106  P.ADT:  .ASCII  / OF/
1794 011231  106  137  123           .ASCII  /F_S/
1795 011234  105  124  000           .ASCII  /ET/<00>
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION

1800 011237  000                                .ASCII  <00>
1801 011240  040  103  117  P.ADU:              .ASCII  / CO/
1802 011243  125  116  124                      .ASCII  /UNT/
1803 011246  000  000                           .ASCII  <00><00>
1804 011250  040  104  105  P.ADV:              .ASCII  / DE/
1805 011253  114  101  131                      .ASCII  /LAY/
1806 011256  000  000                           .ASCII  <00><00>
1807 011260  040  124  105  P.ADW:              .ASCII  / TE/
1808 011263  123  124  123                      .ASCII  /STS/
1809 011266  000  000                           .ASCII  <00><00>
1810 011270  040  101  104  P.ADX:              .ASCII  / AD/
1811 011273  122  123  000                      .ASCII  /RS/<00>
1812 011276  040  103  117  P.ADY:              .ASCII  / CO/
1813 011301  125  116  124                      .ASCII  /UNT/
1814 011304  105  122  000                      .ASCII  /ER/<00>
1815 011307  000                                .ASCII  <00>
1816 011310  040  122  105  P.ADZ:              .ASCII  / RE/
1817 011313  107  000  000                      .ASCII  /G/<00><00>
1818 011316  040  124  105  P.AEA:              .ASCII  / TE/
1819 011321  123  124  105                      .ASCII  /STE/
1820 011324  104  000                           .ASCII  /D/<00>
1821 011326  040  116  111  P.AEB:              .ASCII  / NI/
1822 011331  102  102  114                      .ASCII  /BBL/
1823 011334  105  000                           .ASCII  /E/<00>
1824 011336  040  101  114  P.AEC:              .ASCII  / AL/
1825 011341  114  000  000                      .ASCII  /L/<00><00>
1826 011344  040  124  105  P.AED:              .ASCII  / TE/
1827 011347  123  124  000                      .ASCII  /ST/<00>
1828 011352  040  130  106  P.AEE:              .ASCII  / XF/
1829 011355  105  122  105                      .ASCII  /ERE/
1830 011360  104  000                           .ASCII  /D/<00>
1831 011362  040  116  111  P.AEF:              .ASCII  / NI/
1832 011365  102  102  114                      .ASCII  /BBL/
1833 011370  105  123  000                      .ASCII  /ES/<00>
1834 011373  000                                .ASCII  <00>
1835 011374  040  123  103  P.AEG:              .ASCII  / SC/
1836 011377  000                                .ASCII  <00>
1837 011400  040  115  125  P.AEH:              .ASCII  / MU/
1838 011403  114  124  111                      .ASCII  /LTI/
1839 011406  120  114  105                      .ASCII  /PLE/
1840 011411  130  105  122                      .ASCII  /XER/
1841 011414  000  000                           .ASCII  <00><00>
1842 011416  040  125  116  P.AEI:              .ASCII  / UN/
1843 011421  105  130  120                      .ASCII  /EXP/
1844 011424  105  103  124                      .ASCII  /ECT/
1845 011427  105  104  000                      .ASCII  /ED/<00>
1846 011432  040  116  105  P.AEJ:              .ASCII  / NE/
1847 011435  104  000  000                      .ASCII  /D/<00><00>
1848 011440  040  111  114  P.AEK:              .ASCII  / IL/
1849 011443  122  000  000                      .ASCII  /R/<00><00>
1850 011446  040  103  122  P.AEL:              .ASCII  / CR/
1851 011451  103  000  000                      .ASCII  /C/<00><00>
```

```
1853                                    ;ML4AD
1854                                    ;
1855                                    ;       ROUTINE DECLARATION SECTION
1856 011454  040  123  107  P.AEM:  .ASCII  / SG/
1857 011457  114  000  000           .ASCII  /L/<00><00>
1858 011462  040  105  103  P.AEN:  .ASCII  / EC/
1859 011465  110  000  000           .ASCII  /H/<00><00>
1860 011470  125  116  103  P.AEO:  .ASCII  /UNC/
1861 011473  000                     .ASCII  <00>
1862 011474  040  102  111  P.AEP:  .ASCII  / BI/
1863 011477  124  000  000           .ASCII  /T/<00><00>
1864 011502  040  103  110  P.AEQ:  .ASCII  / CH/
1865 011505  101  116  116           .ASCII  /ANN/
1866 011510  105  114  000           .ASCII  /EL/<00>
1867 011513  000                     .ASCII  <00>
1868 011514  040  114  101  P.AER:  .ASCII  / LA/
1869 011517  124  103  110           .ASCII  /TCH/
1870 011522  000  000                .ASCII  <00><00>
1871 011524  040  104  103  P.AES:  .ASCII  / DC/
1872 011527  113  000  000           .ASCII  /K/<00><00>
1873 011532  040  102  125  P.AET:  .ASCII  / BU/
1874 011535  123  000  000           .ASCII  /S/<00><00>
1875 011540  040  123  131  P.AEU:  .ASCII  / SY/
1876 011543  116  104  122           .ASCII  /NDR/
1877 011546  117  116  105           .ASCII  /ONE/
1878 011551  000                     .ASCII  <00>
1879 011552  040  104  105  P.AEV:  .ASCII  / DE/
1880 011555  103  117  104           .ASCII  /COD/
1881 011560  105  000                .ASCII  /E/<00>
1882 011562  040  116  117  P.AEW:  .ASCII  / NO/
1883 011565  124  000  000           .ASCII  /T/<00><00>
1884 011570  040  125  116  P.AEX:  .ASCII  / UN/
1885 011573  103  117  122           .ASCII  /COR/
1886 011576  122  105  103           .ASCII  /REC/
1887 011601  124  101  102           .ASCII  /TAB/
1888 011604  114  105  000           .ASCII  /LE/<00>
1889 011607  000                     .ASCII  <00>
1890 011610  040  102  111  P.AEY:  .ASCII  / BI/
1891 011613  124  040  116           .ASCII  /T N/
1892 011616  117  124  040           .ASCII  /OT /
1893 011621  123  105  124           .ASCII  /SET/
1894 011624  000  000                .ASCII  <00><00>
1895 011626  040  102  111  P.AEZ:  .ASCII  / BI/
1896 011631  124  040  116           .ASCII  /T N/
1897 011634  117  124  040           .ASCII  /OT /
1898 011637  103  114  122           .ASCII  /CLR/
1899 011642  000  000                .ASCII  <00><00>
1900 011644  040  116  117  P.AFA:  .ASCII  / NO/
1901 011647  040  122  105           .ASCII  / RE/
1902 011652  123  120  117           .ASCII  /SPO/
1903 011655  116  103  105           .ASCII  /NCE/
1904 011660  040  101  106           .ASCII  / AF/
1905 011663  124  105  122           .ASCII  /TER/
1906 011666  040  061  056           .ASCII  / 1./
1907 011671  065  040  125           .ASCII  /5 U/
```

```
1909
1910                                    ;ML4AD
1911                                    :                       ROUTINE DECLARATION SECTION
1912 011674  123  000                                  .ASCII  /S/<00>
1913 011676  040  104  101   P.AFB:                    .ASCII  / DA/
1914 011701  124  101  040                             .ASCII  /TA /
1915 011704  105  122  122                             .ASCII  /ERR/
1916 011707  117  122  123                             .ASCII  /ORS/
1917 011712  000  000                                  .ASCII  <00><00>
1918 011714  040  102  111   P.AFC:                    .ASCII  / BI/
1919 011717  124  040  123                             .ASCII  /T S/
1920 01172_  105  124  000                             .ASCII  /ET/<00>
1921 011725  000                                       .ASCII  <00>
1922 011[illegible]  040  102  111   P.AFD:             .ASCII  / BI/
1923 011731  124  040  103                             .ASCII  /T C/
1924 011734  114  122  000                             .ASCII  /LR/<00>
1925 011737  000                                       .ASCII  <00>
1926 011740  040  117  106   P.AFE:                    .ASCII  / OF/
1927 011743  040  117  124                             .ASCII  / OT/
1928 011746  110  105  122                             .ASCII  /HER/
1929 011751  040  104  122                             .ASCII  / DR/
1930 011754  111  126  105                             .ASCII  /IVE/
1931 011757  123  000  000                             .ASCII  /S/<00><00>
1932 011762  040  103  114   P.AFF:                    .ASCII  / CL/
1933 011765  101  123  123                             .ASCII  /ASS/
1934 011770  040  101  000                             .ASCII  / A/<00>
1935 011773  000                                       .ASCII  <00>
1936 011774  040  103  114   P.AFG:                    .ASCII  / CL/
1937 011777  101  123  123                             .ASCII  /ASS/
1938 012002  040  102  000                             .ASCII  / B/<00>
1939 012005  000                                       .ASCII  <00>
1940 012006  040  124  117   P.AFH:                    .ASCII  / TO/
1941 012011  040  106  111                             .ASCII  / FI/
1942 012014  116  104  000                             .ASCII  /ND/<00>
1943 012017  000                                       .ASCII  <00>
1944 012020  040  116  117   P.AFI:                    .ASCII  / NO/
1945 012023  124  040  114                             .ASCII  /T L/
1946 012026  101  124  103                             .ASCII  /ATC/
1947 012031  110  105  104                             .ASCII  /HED/
1948 012034  000  000                                  .ASCII  <00><00>
1949 012036  040  123  111   P.AFJ:                    .ASCII  / SI/
1950 012041  116  107  114                             .ASCII  /NGL/
1951 012044  105  040  102                             .ASCII  /E B/
1952 012047  111  124  040                             .ASCII  /IT /
1953 012052  105  122  122                             .ASCII  /ERR/
1954 012055  117  122  000                             .ASCII  /OR/<00>
1955 012060  040  115  125   P.AFK:                    .ASCII  / MU/
1956 012063  114  124  111                             .ASCII  /LTI/
1957 012066  120  114  105                             .ASCII  /PLE/
1958 012071  040  102  111                             .ASCII  / BI/
1959 012074  124  040  105                             .ASCII  /T E/
1960 012077  122  122  117                             .ASCII  /RRO/
1961 012102  122  000                                  .ASCII  /R/<00>
1962 012104  040  040  040   P.AFL:                    .ASCII  /   /
1963 012107  040  040  040                             .ASCII  /   /
```

```
1965                                    ;ML4AD
1966                                    ;               ROUTINE DECLARATION SECTION
1967
1968 012112  040  040  040             .ASCII  /   /
1969 012115  122  105  107             .ASCII  /REG/
1970 012120  111  123  124             .ASCII  /IST/
1971 012123  105  122  040             .ASCII  /ER /
1972 012126  040  040  104             .ASCII  /  D/
1973 012131  125  115  120             .ASCII  /UMP/
1974 012134  000  000                  .ASCII  <00><00>
1975 012136  123  105  122  P.AFM:     .ASCII  /SER/
1976 012141  111  101  114             .ASCII  /IAL/
1977 012144  040  043  000             .ASCII  / #/<00>
1978 012147  000                       .ASCII  <00>
1979 012150  040  115  105  P.AFN:     .ASCII  / ME/
1980 012153  115  040  123             .ASCII  /M S/
1981 012156  111  132  111             .ASCII  /IZI/
1982 012161  116  107  000             .ASCII  /NG/<00>
1983 012164  040  116  117  P.AFO:     .ASCII  / NO/
1984 012167  117  120  000             .ASCII  /OP/<00>
1985 012172  040  104  122  P.AFP:     .ASCII  / DR/
1986 012175  126  000  000             .ASCII  /V/<00><00>
1987 012200  040  127  122  P.AFQ:     .ASCII  / WR/
1988 012203  111  124  105             .ASCII  /ITE/
1989 012206  040  103  110             .ASCII  / CH/
1990 012211  105  103  113             .ASCII  /ECK/
1991 012214  000  000                  .ASCII  <00><00>
1992 012216  040  127  122  P.AFR:     .ASCII  / WR/
1993 012221  111  124  105             .ASCII  /ITE/
1994 012224  000  000                  .ASCII  <00><00>
1995 012226  040  122  105  P.AFS:     .ASCII  / RE/
1996 012231  101  104  000             .ASCII  /AD/<00>
1997 012234  040  103  114  P.AFT:     .ASCII  / CL/
1998 012237  105  101  122             .ASCII  /EAR/
1999 012242  000  000                  .ASCII  <00><00>
2000 012244  040  103  117  P.AFU:     .ASCII  / CO/
2001 012247  115  120  040             .ASCII  /MP /
2002 012252  105  122  122             .ASCII  /ERR/
2003 012255  117  122  000             .ASCII  /OR/<00>
2004 012260  040  123  131  P.AFV:     .ASCII  / SY/
2005 012263  123  040  103             .ASCII  /S C/
2006 012266  114  122  000             .ASCII  /LR/<00>
2007 012271  000                       .ASCII  <00>
2008 012272  040  123  105  P.AFW:     .ASCII  / SE/
2009 012275  101  122  103             .ASCII  /ARC/
2010 012300  110  000                  .ASCII  /H/<00>
2011 012302  040  122  105  P.AFX:     .ASCII  / RE/
2012 012305  101  104  055             .ASCII  /AD-/
2013 012310  111  116  055             .ASCII  /IN-/
2014 012313  120  122  105             .ASCII  /PRE/
2015 012316  123  105  124             .ASCII  /SET/
2016 012321  000                       .ASCII  <00>
2017 012322  040  111  114  P.AFY:     .ASCII  / IL/
2018 012325  114  105  107             .ASCII  /LEG/
2019 012330  101  114  000             .ASCII  /AL/<00>
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
2021                                    ;ML4AD
2022                                    ;               ROUTINE DECLARATION SECTION
2023
2024 012333  000                                       .ASCII  <00>
2025 012334  040  101  102  P.AFZ:                     .ASCII  / AB/
2026 012337  117  122  124                             .ASCII  /ORT/
2027 012342  000  000                                  .ASCII  <00><00>
2028 012344  040  101  122  P.AGA:                     .ASCII  / AR/
2029 012347  122  040  122                             .ASCII  /R R/
2030 012352  104  137  127                             .ASCII  /D_W/
2031 012355  122  124  000                             .ASCII  /RT/<00>
2032 012360  040  107  117  P.AGB:                     .ASCII  / GO/
2033 012363  117  104  040                             .ASCII  /OD /
2034 012366  102  114  113                             .ASCII  /BLK/
2035 012371  000                                       .ASCII  <00>
2036 012372  040  122  105  P.AGC:                     .ASCII  / RE/
2037 012375  106  122  105                             .ASCII  /FRE/
2038 012400  123  110  000                             .ASCII  /SH/<00>
2039 012403  000                                       .ASCII  <00>
2040 012404  040  101  122  P.AGD:                     .ASCII  / AR/
2041 012407  122  101  131                             .ASCII  /RAY/
2042 012412  000  000                                  .ASCII  <00><00>
2043 012414  040  122  101  P.AGE:                     .ASCII  / RA/
2044 012417  115  055  102                             .ASCII  /M-B/
2045 012422  125  123  000                             .ASCII  /US/<00>
2046 012425  000                                       .ASCII  <00>
2047 012426  040  117  126  P.AGF:                     .ASCII  / OV/
2048 012431  105  122  106                             .ASCII  /ERF/
2049 012434  114  117  127                             .ASCII  /LOW/
2050 012437  000                                       .ASCII  <00>
2051 012440  040  103  110  P.AGG:                     .ASCII  / CH/
2052 012443  113  137  123                             .ASCII  /K_S/
2053 012446  125  115  000                             .ASCII  /UM/<00>
2054 012451  000                                       .ASCII  <00>
2055 012452  040  114  101  P.AGH:                     .ASCII  / LA/
2056 012455  123  124  040                             .ASCII  /ST /
2057 012460  102  114  113                             .ASCII  /BLK/
2058 012463  000                                       .ASCII  <00>
2059 012464  040  111  116  P.AGI:                     .ASCII  / IN/
2060 012467  111  124  111                             .ASCII  /ITI/
2061 012472  101  114  111                             .ASCII  /ALI/
2062 012475  132  105  000                             .ASCII  /ZE/<00>
2063 012500  040  115  114  P.AGJ:                     .ASCII  / ML/
2064 012503  103  123  061                             .ASCII  /CS1/
2065 012506  000  000                                  .ASCII  <00><00>
2066 012510  040  115  114  P.AGK:                     .ASCII  / ML/
2067 012513  104  123  000                             .ASCII  /DS/<00>
2068 012516  040  115  114  P.AGL:                     .ASCII  / ML/
2069 012521  105  122  000                             .ASCII  /ER/<00>
2070 012524  040  115  114  P.AGM:                     .ASCII  / ML/
2071 012527  115  122  000                             .ASCII  /MR/<00>
2072 012532  040  115  114  P.AGN:                     .ASCII  / ML/
2073 012535  101  123  000                             .ASCII  /AS/<00>
2074 012540  040  115  114  P.AGO:                     .ASCII  / ML/
2075 012543  104  101  000                             .ASCII  /DA/<00>
```

```
2077
2078                                    ;ML4AD
2079                                    ;               ROUTINE DECLARATION SECTION
2080 012546  040  115  114  P.AGP:  .ASCII  / ML/
2081 012551  104  124  000           .ASCII  /DT/<00>
2082 012554  040  115  114  P.AGQ:  .ASCII  / ML/
2083 012557  120  101  000           .ASCII  /PA/<00>
2084 012562  040  115  114  P.AGR:  .ASCII  / ML/
2085 012565  123  116  000           .ASCII  /SN/<00>
2086 012570  040  115  114  P.AGS:  .ASCII  / ML/
2087 012573  105  061  000           .ASCII  /E1/<00>
2088 012576  040  115  114  P.AGT:  .ASCII  / ML/
2089 012601  105  062  000           .ASCII  /E2/<00>
2090 012604  040  115  114  P.AGU:  .ASCII  / ML/
2091 012607  104  061  000           .ASCII  /D1/<00>
2092 012612  040  115  114  P.AGV:  .ASCII  / ML/
2093 012615  104  062  000           .ASCII  /D2/<00>
2094 012620  040  115  114  P.AGW:  .ASCII  / ML/
2095 012623  105  105  000           .ASCII  /EE/<00>
2096 012626  040  115  114  P.AGX:  .ASCII  / ML/
2097 012631  105  114  000           .ASCII  /EL/<00>
2098 012634  040  115  114  P.AGY:  .ASCII  / ML/
2099 012637  120  104  000           .ASCII  /PD/<00>
2100 012642  040  115  114  P.AGZ:  .ASCII  / ML/
2101 012645  103  123  062           .ASCII  /CS2/
2102 012650  000  000                .ASCII  <00><00>
2103 012652  040  115  114  P.AHA:  .ASCII  / ML/
2104 012655  127  103  000           .ASCII  /WC/<00>
2105 012660  040  115  114  P.AHB:  .ASCII  / ML/
2106 012663  102  101  000           .ASCII  /BA/<00>
2107 012666  040  115  114  P.AHC:  .ASCII  / ML/
2108 012671  102  101  111           .ASCII  /BAI/
2109 012674  000  000                .ASCII  <00><00>
2110 012676  040  115  114  P.AHD:  .ASCII  / ML/
2111 012701  103  123  063           .ASCII  /CS3/
2112 012704  000  000                .ASCII  <00><00>
2113 012706  101  123  131  P.AHE:  .ASCII  /ASY/
2114 012711  116  103  110           .ASCII  /NCH/
2115 012714  122  117  116           .ASCII  /RON/
2116 012717  117  125  123           .ASCII  /OUS/
2117 012722  040  106  101           .ASCII  / FA/
2118 012725  111  114  125           .ASCII  /ILU/
2119 012730  122  105  040           .ASCII  /RE /
2120 012733  115  117  104           .ASCII  /MOD/
2121 012736  125  114  105           .ASCII  /ULE/
2122 012741  040  067  063           .ASCII  / 73/
2123 012744  066  061  000           .ASCII  /61/<00>
2124 012747  000                     .ASCII  <00>
2125 012750  123  131  116  P.AHF:  .ASCII  /SYN/
2126 012753  103  110  122           .ASCII  /CHR/
2127 012756  117  116  117           .ASCII  /ONO/
2128 012761  125  123  040           .ASCII  /US /
2129 012764  106  101  111           .ASCII  /FAI/
2130 012767  114  125  122           .ASCII  /LUR/
2131 012772  105  040  115           .ASCII  /E M/
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION

2136 012775  117  104  125             .ASCII  /ODU/
2137 013000  114  105  040             .ASCII  /LE /
2138 013003  067  063  066             .ASCII  /736/
2139 013006  062  040  000             .ASCII  /2 /<00>
2140 013011  000                       .ASCII  <00>
2141 013012  101  122  122  P.AHG:     .ASCII  /ARR/
2142 013015  101  131  040             .ASCII  /AY /
2143 013020  104  101  124             .ASCII  /DAT/
2144 013023  101  040  106             .ASCII  /A F/
2145 013026  101  111  114             .ASCII  /AIL/
2146 013031  125  122  105             .ASCII  /URE/
2147 013034  040  115  117             .ASCII  / MO/
2148 013037  104  125  114             .ASCII  /DUL/
2149 013042  105  040  067             .ASCII  /E 7/
2150 013045  063  066  063             .ASCII  /363/
2151 013050  040  000                  .ASCII  / /<00>
2152 013052  115  105  115  P.AHH:     .ASCII  /MEM/
2153 013055  117  122  131             .ASCII  /ORY/
2154 013060  040  101  122             .ASCII  / AR/
2155 013063  122  101  131             .ASCII  /RAY/
2156 013066  040  106  101             .ASCII  / FA/
2157 013071  111  114  125             .ASCII  /ILU/
2158 013074  122  105  040             .ASCII  /RE /
2159 013077  115  117  104             .ASCII  /MOD/
2160 013102  125  114  105             .ASCII  /ULE/
2161 013105  040  067  063             .ASCII  / 73/
2162 013110  065  067  000             .ASCII  /57/<00>
2163 013113  000                       .ASCII  <00>
2164 013114  111  116  124  P.AHI:     .ASCII  /INT/
2165 013117  105  122  115             .ASCII  /ERM/
2166 013122  105  104  111             .ASCII  /EDI/
2167 013125  101  124  105             .ASCII  /ATE/
2168 013130  040  104  111             .ASCII  / DI/
2169 013133  101  107  116             .ASCII  /AGN/
2170 013136  117  123  124             .ASCII  /OST/
2171 013141  111  103  040             .ASCII  /IC /
2172 013144  115  105  123             .ASCII  /MES/
2173 013147  123  101  107             .ASCII  /SAG/
2174 013152  105  000                  .ASCII  /E/<00>
2175 013154  104  101  124  P.AHJ:     .ASCII  /DAT/
2176 013157  101  040  114             .ASCII  /A L/
2177 013162  101  124  105             .ASCII  /ATE/
2178 013165  040  105  122             .ASCII  / ER/
2179 013170  122  117  122             .ASCII  /ROR/
2180 013173  040  104  125             .ASCII  / DU/
2181 013176  122  111  116             .ASCII  /RIN/
2182 013201  107  040  124             .ASCII  /G T/
2183 013204  122  101  116             .ASCII  /RAN/
2184 013207  123  106  105             .ASCII  /SFE/
2185 013212  122  000                  .ASCII  /R/<00>
2186 013214  123  103  040  P.AHK:     .ASCII  /SC /
2187 013217  102  111  124             .ASCII  /BIT/
```

```
                                        ;ML4AD
                                        ;                ROUTINE DECLARATION SECTION

013222  040  123  105          .ASCII  / SE/
013225  124  040  104          .ASCII  /T D/
013230  125  122  111          .ASCII  /URI/
013233  116  107  040          .ASCII  /NG /
013236  124  122  101          .ASCII  /TRA/
013241  116  123  106          .ASCII  /NSF/
013244  105  122  000          .ASCII  /ER/<00>
013247  000                    .ASCII  <00>
013250  124  122  117  P.AHL:   .ASCII  /TRO/
013253  125  102  114          .ASCII  /UBL/
013256  105  040  123          .ASCII  /E S/
013261  110  117  117          .ASCII  /HOO/
013264  124  040  114          .ASCII  /T L/
013267  117  117  120          .ASCII  /OOP/
013272  040  105  122          .ASCII  / ER/
013275  122  117  122          .ASCII  /ROR/
013300  123  000                .ASCII  /S/<00>
013302  122  110  040  P.AHM:   .ASCII  /RH /
013305  103  117  116          .ASCII  /CON/
013310  124  122  117          .ASCII  /TRO/
013313  114  114  105          .ASCII  /LLE/
013316  122  040  105          .ASCII  /R E/
013321  122  122  117          .ASCII  /RRO/
013324  122  123  000          .ASCII  /RS/<00>
013327  000                    .ASCII  <00>
013330  104  122  111  P.AHN:   .ASCII  /DRI/
013333  126  105  040          .ASCII  /VE /
013336  110  125  116          .ASCII  /HUN/
013341  107  040  101          .ASCII  /G A/
013344  106  124  105          .ASCII  /FTE/
013347  122  040  115          .ASCII  /R M/
013352  101  123  123          .ASCII  /ASS/
013355  040  102  125          .ASCII  / BU/
013360  123  040  124          .ASCII  /S T/
013363  122  101  116          .ASCII  /RAN/
013366  123  106  105          .ASCII  /SFE/
013371  122  000  000          .ASCII  /R/<00><00>


013374                  NIB.SAVE:
013374                          .BLKW   3
013402                  HW.OR.TBL:
013402                          .BLKW   177
014000                  PTBL.PTR:
014000                          .BLKW   1
014002                  OP.NUM.ARR:
014002                          .BLKW   1
014004                  ARR.INC:.BLKW   1
014006                  GOOD.BLK:
014006                          .BLKW   1
```

```
                                ;ML4AD
                                ;                       ROUTINE DECLARATION SECTION

2244
2245
2246
2247 014010                     PAR.DIS:.BLKW   1
2248 014012                     CHIP.SIZ:
2249 014012                             .BLKW   1
2250 014014                     LST.BLK:.BLKW   1
2251 014016                     ARR.16: .BLKW   1
2252 014020                     LST.ARR:.BLKW   1
2253 014022                     IO.BUF: .BLKW   400
2254 015022                     STK.OFF:.BLKW   5
2255 015034                     STACK:  .BLKW   143
2256 015342                     PD.TEMP:.BLKW   1
2257 015344                     W.C.SIZE:
2258 015344                             .BLKW   1
2259 015346                     RAS.INC:.BLKW   1
2260 015350                     WT.DATA:.BLKW   1
2261 015352                     RD.DATA:.BLKW   1
2262 015354                     DRIVE.TYPE:
2263 015354                             .BLKW   1
2264 015356                     LST.DUT.REG:
2265 015356                             .BLKW   1
2266 015360                     REG.INIT.FLG:
2267 015360   000000                    .WORD   0
2268 015362                     A.CAL:  .BLKW   1
2269 015364                     B.CAL:  .BLKW   1
2270 015366                     P.CAL:  .BLKW   1
2271 015370                     A.GEN:  .BLKW   1
2272 015372                     B.GEN:  .BLKW   1
2273 015374                     P.GEN:  .BLKW   1
2274 015376                     P.CNT:  .BLKW   1
2275 015400                     LIMIT:  .BLKW   1
2276 015402                     DROP.CNT:
2277 015402                             .BLKW   1
2278 015404       000           ML.REG: .BYTE   0
2279 015405       000                   .BYTE   0
2280 015406   004000                    .WORD   4000
2281 015410   173701                    .WORD   -4077
2282 015412   160200                    .WORD   -17600
2283 015414       000                   .BYTE   0
2284 015415       000                   .BYTE   0
2285 015416       000                   .BYTE   0
2286 015417       000                   .BYTE   0
2287 015420       000                   .BYTE   0
2288 015421       000                   .BYTE   0
2289 015422       000                   .BYTE   0
2290 015423       000                   .BYTE   0
2291 015424       000                   .BYTE   0
2292 015425       000                   .BYTE   0
2293 015426       000                   .BYTE   0
2294 015427       000                   .BYTE   0
2295 015430       000                   .BYTE   0
2296 015431       000                   .BYTE   0
2297 015432       000                   .BYTE   0
2298 015433       000                   .BYTE   0
```

```
2300
2301
2302
2303 015434   000                ;ML4AD
2304 015435   000                ;                ROUTINE DECLARATION SECTION
2305 015436   000                        .BYTE   0
2306 015437   000                        .BYTE   0
2307 015440   000                        .BYTE   0
2308 015441   000                        .BYTE   0
2309 015442   000                        .BYTE   0
2310 015443   000                        .BYTE   0
2311 015444   000                        .BYTE   0
2312 015445   000                        .BYTE   0
2313 015446   000                        .BYTE   0
2314 015447   000                        .BYTE   0
2315 015450   000                        .BYTE   0
2316 015451   000                        .BYTE   0
2317 015452   000                        .BYTE   0
2318 015453   000                        .BYTE   0
2319 015454   000                        .BYTE   0
2320 015455   000                        .BYTE   0
2321 015456   010600                     .WORD   10600
2322 015460   025077                     .WORD   25077
2323 015462   000100                     .WORD   100
2324 015464   000                        .BYTE   0
2325 015465   000                        .BYTE   0
2326 015466   000                        .BYTE   0
2327 015467   000                        .BYTE   0
2328 015470   014620                     .WORD   14620
2329 015472   000                        .BYTE   0
2330 015473   000                        .BYTE   0
2331 015474   000                        .BYTE   0
2332 015475   000                        .BYTE   0
2333 015476   000                        .BYTE   0
2334 015477   000                        .BYTE   0
2335 015500   000                        .BYTE   0
2336 015501   000                        .BYTE   0
2337 015502   177400                     .WORD   -400
2338 015504   000                        .BYTE   0
2339 015505   000                        .BYTE   0
2340 015506   000                        .BYTE   0
2341 015507   000                        .BYTE   0
2342 015510   100000                     .WORD   -100000
2343 015512   000                        .BYTE   0
2344 015513   000                        .BYTE   0
2345 015514   000                        .BYTE   0
2346 015515   000                        .BYTE   0
2347 015516   000                        .BYTE   0
2348 015517   000                        .BYTE   0
2349 015520   000                        .BYTE   0
2350 015521   000                        .BYTE   0
2351 015522   000                        .BYTE   0
2352 015523   000                        .BYTE   0
2353 015524   000                        .BYTE   0
2354 015525   000                        .BYTE   0
```

```
                                ;ML4AD
                                ;                ROUTINE DECLARATION SECTION

2356
2357
2358
2359 015526     000                    .BYTE   0
2360 015527     000                    .BYTE   0
2361 015530  000020                    .WORD   20
2362 015532  177400                    .WORD   -400
2363 015534     000                    .BYTE   0
2364 015535     000                    .BYTE   0
2365 015536  000110                    .WORD   110
2366 015540  177666                    .WORD   -112
2367 015542  000001                    .WORD   1
2368 015544     000                    .BYTE   0
2369 015545     000                    .BYTE   0
2370 015546     000                    .BYTE   0
2371 015547     000                    .BYTE   0
2372 015550     000                    .BYTE   0
2373 015551     000                    .BYTE   0
2374 015552     000                    .BYTE   0
2375 015553     000                    .BYTE   0
2376 015554     000                    .BYTE   0
2377 015555     000                    .BYTE   0
2378 015556     000                    .BYTE   0
2379 015557     000                    .BYTE   0
2380 015560  140300                    .WORD   -37500
2381 015562     000                    .BYTE   0
2382 015563     000                    .BYTE   0
2383 015564     000                    .BYTE   0
2384 015565     000                    .BYTE   0
2385 015566     000                    .BYTE   0
2386 015567     000                    .BYTE   0
2387 015570  100300                    .WORD   -77500
2388 015572     000                    .BYTE   0
2389 015573     000                    .BYTE   0
239C 015574     000                    .BYTE   0
2391 015575     000                    .BYTE   0
2392 015576     000                    .BYTF   0
2393 015577     000                    .BYTE   0
2394 015600     000                    .BYTE   0
2395 015601     000                    .BYTE   0
2396 015602     000                    .BYTE   0
2397 015603     000                    .BYTE   0
2398 015604     000                    .BYTE   0
2399 015605     000                    .BYTE   0
2400 015606     000                    .BYTE   0
2401 015607     000                    .BYTE   0
2402 015610     000                    .BYTE   0
2403 015611     000                    .BYTE   0
2404 015612     000                    .BYTE   0
2405 015613     000                    .BYTE   0
2406 015614     000                    .BYTE   0
2407 015615     000                    .BYTE   0
2408 015616     000                    .BYTE   0
2409 015617     000                    .BYTE   0
2410 015620  010000                    .WORD   10000
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION

2412
2413
2414
2415 015622  000                                .BYTE   0
2416 015623  000                                .BYTE   0
2417 015624  000                                .BYTE   0
2418 015625  000                                .BYTE   0
2419 015626  000                                .BYTE   0
2420 015627  000                                .BYTE   0
2421 015630  000                                .BYTE   0
2422 015631  000                                .BYTE   0
2423 015632  000                                .BYTE   0
2424 015633  000                                .BYTE   0
2425 015634  000                                .BYTE   0
2426 015635  000                                .BYTE   0
2427 015636  000                                .BYTE   0
2428 015637  000                                .BYTE   0
2429 015640  000                                .BYTE   0
2430 015641  000                                .BYTE   0
2431 015642  000                                .BYTE   0
2432 015643  000                                .BYTE   0
2433 015644  000                                .BYTE   0
2434 015645  000                                .BYTE   0
2435 015646  000                                .BYTE   0
2436 015647  000                                .BYTE   0
2437 015650  000                                .BYTE   0
2438 015651  000                                .BYTE   0
2439 015652  000                                .BYTE   0
2440 015653  000                                .BYTE   0
2441 015654  000                                .BYTE   0
2442 015655  000                                .BYTE   0
2443 015656  000                                .BYTE   0
2444 015657  000                                .BYTE   0
2445 015660  000                                .BYTE   0
2446 015661  000                                .BYTE   0
2447 015662  000000                             .WORD   0
2448 015664  001001                 REM.TBL:    .WORD   1001
2449 015666  004004                             .WORD   4004
2450 015670  020020                             .WORD   20020
2451 015672  003003                             .WORD   3003
2452 015674  014014                             .WORD   14014
2453 015676  021460                             .WORD   21460
2454 015700  005005                             .WORD   5005
2455 015702  024024                             .WORD   24024
2456 015704  023023                             .WORD   23023
2457 015706  017017                             .WORD   17017
2458 015710  035474                             .WORD   35474
2459 015712  024465                             .WORD   24465
2460 015714  021021                             .WORD   21021
2461 015716  007007                             .WORD   7007
2462 015720  034034                             .WORD   34034
2463 015722  022463                             .WORD   22463
2464 015724  011011                             .WORD   11011
2465 015726  005444                             .WORD   5444
2466 015730  026026                             .WORD   26026
```

```
                                ;ML4A.D
                                ;                ROUTINE DECLARATION SECTION

                                                 .WORD   33033
                                                 .WORD   16457
                                                 .WORD   33472
                                                 .WORD   14455
                                                 .WORD   23462
                                                 .WORD   15015
                                                 .WORD   25464
                                                 .WORD   25025
                                                 .WORD   27027
                                                 .WORD   37037
                                                 .WORD   36477
                                                 .WORD   30471
                                                 .BYTE   41
                                                 .EVEN
                                DT.1:            .WORD   64477
                                                 .WORD   7700
                                                 .BYTE   0
                                                 .BYTE   0
                                                 .WORD   64477
                                                 .WORD   77
                                                 .BYTE   0
                                                 .BYTE   0
                                                 .WORD   -77201
                                                 .WORD   17700
                                                 .BYTE   77
                                                 .BYTE   0
                                                 .WORD   -77201
                                                 .WORD   0
                                                 .BYTE   77
                                                 .BYTE   0
                                                 .WORD   -77201
                                                 .WORD   10077
                                                 .BYTE   77
                                                 .EVEN
                                RH.ADD:          .BLKW   1
                                RH.TYP:          .BLKW   1
                                RH.VEC:          .BLKW   1
                                ML.LUN:          .BLKW   1
                                ML.DUT:          .BLKW   1


                                                 .GLOBL  REGDMP, ONEPAS, INTERVEN, PRSN
                                                 .GLOBL  ERRTHR


                        100000  BIT15==                  -100000
                        040000  BIT14==                  40000
                        020000  BIT13==                  20000
                        010000  BIT12==                  10000
                        004000  BIT11==                  4000
                        002000  BIT10==                  2000
                        001000  BIT09==                  1000
```

```
2524                                ;ML4AD
2525                                :                ROUTINE DECLARATION SECTION
2526
2527    000400                      BIT08==          400
2528    000200                      BIT07==          200
2529    000100                      BIT06==          100
2530    000040                      BIT05==          40
2531    000020                      BIT04==          20
2532    000010                      BIT03==          10
2533    000004                      BIT02==          4
2534    000002                      BIT01==          2
2535    000001                      BIT00==          1
2536    001000                      BIT9==           1000
2537    000400                      BIT8==           400
2538    000200                      BIT7==           200
2539    000100                      BIT6==           100
2540    000040                      BIT5==           40
2541    000020                      BIT4==           20
2542    000010                      BIT3==           10
2543    000004                      BIT2==           4
2544    000002                      BIT1==           2
2545    000001                      BIT0==           1
2546    000040                      EF.START==       40
2547    000037                      EF.RESTART==     37
2548    000036                      EF.CONTINUE==    36
2549    000035                      EF.NEW==         35
2550    000034                      EF.PWR==         34
2551    000340                      PRI07==          340
2552    000300                      PRI06==          300
2553    000240                      PRI05==          240
2554    000200                      PRI04==          200
2555    000140                      PRI03==          140
2556    000100                      PRI02==          100
2557    000040                      PRI01==          40
2558    000000                      PRI00==          0
2559    000004                      EVL==            4
2560    000010                      LOT==            10
2561    000020                      ADR==            20
2562    000040                      IDU==            40
2563    000100                      ISR==            100
2564    000200                      UAM==            200
2565    000400                      BOE==            400
2566    001000                      PNT==            1000
2567    002000                      PRI==            2000
2568    004000                      IXE==            4000
2569    010000                      IBE==            10000
2570    020000                      IER==            20000
2571    040000                      LOE==            40000
2572    100000                      HOE==            -100000
2573    004716                      T.21=            P.AAA
2574    005012                      T.61=            P.AAB
2575    005112                      PWR.OFF=         P.AAC
2576    005202                      PWR.ON=          P.AAD
2577    005272                      UNS.ERR=         P.AAE
2578    005340                      VV.NOT.SET=      P.AAF
```

```
2580                                ;ML4AD
2581                                ;                ROUTINE DECLARATION SECTION
2582
2583    005414                      VV.CLEAR=        P.AAG
2584    005470                      NO.INIT=         P.AAH
2585    005570                      MB.DIS.ERR=      P.AAI
2586    005652                      WC.ERR=          P.AAJ
2587    005722                      W.ERR=           P.AAK
2588    005764                      VV.SET=          P.AAL
2589    006060                      PUP.BB=          P.AAM
2590    006150                      BB.VV.ERR=       P.AAN
2591    006242                      BB.INIT.ERR=     P.AAO
2592    006334                      BB.BB.ERR=       P.AAP
2593    006442                      FMT.1=           P.AAQ
2594    006506                      FMT.2=           P.AAR
2595    006574                      FMT.3=           P.AAS
2596    006620                      FMT.4=           P.AAT
2597    006650                      FMT.5=           P.AAU
2598    006752                      FMT.6=           P.AAV
2599    007002                      FMT.7=           P.AAW
2600    007030                      FMT.8=           P.AAX
2601    007064                      FMT.9=           P.AAY
2602    007116                      FMT.10=          P.AAZ
2603    007170                      FMT.11=          P.ABA
2604    007224                      FMT.12=          P.ABB
2605    007254                      FMT.13=          P.ABC
2606    007330                      FMT.14=          P.ABD
2607    007370                      FMT.15=          P.ABE
2608    007436                      FMT.16=          P.ABF
2609    007526                      FMT.17=          P.ABG
2610    007562                      FMT.18=          P.ABH
2611    007632                      FMT.19=          P.ABI
2612    007702                      FMT.20=          P.ABJ
2613    007754                      FMT.21=          P.ABK
2614    010026                      FMT.22=          P.ABL
2615    010072                      FMT.23=          P.ABM
2616    010142                      FMT.24=          P.ABN
2617    010166                      FMT.25=          P.ABO
2618    010206                      FMT.26=          P.ABP
2619    010240                      ONE.FMT=         P.ABQ
2620    010246                      TWO.FMT=         P.ABR
2621    010256                      THR.FMT=         P.ABS
2622    010270                      FOR.FMT=         P.ABT
2623    010304                      FIV.FMT=         P.ABU
2624    010322                      SIX.FMT=         P.ABV
2625    010342                      SEV.FMT=         P.ABW
2626    010364                      EIG.FMT=         P.ABX
2627    010410                      NIN.FMT=         P.ABY
2628    010436                      TEN.FMT=         P.ABZ
2629    010466                      ELV.FMT=         P.ACA
2630    010520                      WRD.1=           P.ACB
2631    010524                      WRD.2=           P.ACC
2632    010536                      WRD.3=           P.ACD
2633    010544                      WRD.4=           P.ACE
2634    010552                      WRD.5=           P.ACF
```

```
2636                                ;ML4AD
2637                                ;               ROUTINE DECLARATION SECTION
2638
2639    010560                      WRD.6=          P.ACG
2640    010566                      WRD.7=          P.ACH
2641    010602                      WRD.8=          P.ACI
2642    010616                      WRD.9=          P.ACJ
2643    010630                      WRD.10=         P.ACK
2644    010640                      WRD.11=         P.ACL
2645    010650                      WRD.12=         P.ACM
2646    010660                      WRD.13=         P.ACN
2647    010664                      WRD.14=         P.ACO
2648    010676                      WRD.15=         P.ACP
2649    010704                      WRD.16=         P.ACQ
2650    010712                      WRD.17=         P.ACR
2651    010724                      WRD.18=         P.ACS
2652    010730                      WRD.19=         P.ACT
2653    010736                      WRD.20=         P.ACU
2654    010744                      WRD.21=         P.ACV
2655    010752                      WRD.22=         P.ACW
2656    010766                      WRD.23=         P.ACX
2657    010774                      WRD.24=         P.ACY
2658    011002                      WRD.25=         P.ACZ
2659    011016                      WRD.26=         P.ADA
2660    011024                      WRD.27=         P.ADB
2661    011032                      WRD.29=         P.ADC
2662    011046                      WRD.30=         P.ADD
2663    011054                      WRD.31=         P.ADE
2664    011070                      WRD.32=         P.ADF
2665    011076                      WRD.33=         P.ADG
2666    011102                      WRD.34=         P.ADH
2667    011110                      WRD.35=         P.ADI
2668    011116                      WRD.36=         P.ADJ
2669    011122                      WRD.37=         P.ADK
2670    011132                      WRD.38=         P.ADL
2671    011140                      WRD.39=         P.ADM
2672    011150                      WRD.40=         P.ADN
2673    011154                      WRD.41=         P.ADO
2674    011170                      WRD.42=         P.ADP
2675    011176                      WRD.43=         P.ADQ
2676    011206                      WRD.44=         P.ADR
2677    011214                      WRD.45=         P.ADS
2678    011226                      WRD.46=         P.ADT
2679    011240                      WRD.47=         P.ADU
2680    011250                      WRD.48=         P.ADV
2681    011260                      WRD.49=         P.ADW
2682    011270                      WRD.50=         P.ADX
2683    011276                      WRD.51=         P.ADY
2684    011310                      WRD.52=         P.ADZ
2685    011316                      WRD.53=         P.AEA
2686    011326                      WRD.54=         P.AEB
2687    011336                      WRD.55=         P.AEC
2688    011344                      WRD.56=         P.AED
2689    011352                      WRD.57=         P.AEE
2690    011362                      WRD.58=         P.AEF
```

```
2692                                ;ML4AD
2693                                ;           ROUTINE DECLARATION SECTION
2694
2695        011374                  WRD.59=         P.AEG
2696        011400                  WRD.60=         P.AEH
2697        011416                  WRD.61=         P.AEI
2698        011432                  WRD.62=         P.AEJ
2699        011440                  WRD.63=         P.AEK
2700        011446                  WRD.64=         P.AEL
2701        011454                  WRD.65=         P.AEM
2702        011462                  WRD.67=         P.AEN
2703        011470                  WRD.68=         P.AEO
2704        011474                  WRD.69=         P.AEP
2705        011502                  WRD.70=         P.AEQ
2706        011514                  WRD.71=         P.AER
2707        011524                  WRD.72=         P.AES
2708        011532                  WRD.73=         P.AET
2709        011540                  WRD.74=         P.AEU
2710        011552                  WRD.75=         P.AEV
2711        011562                  WRD.76=         P.AEW
2712        011570                  WRD.77=         P.AEX
2713        011610                  PHR.1=          P.AEY
2714        011626                  PHR.2=          P.AEZ
2715        011644                  PHR.3=          P.AFA
2716        011676                  PHR.4=          P.AFB
2717        011714                  PHR.5=          P.AFC
2718        011726                  PHR.6=          P.AFD
2719        011740                  PHR.7=          P.AFE
2720        011762                  PHR.8=          P.AFF
2721        011774                  PHR.9=          P.AFG
2722        012006                  PHR.10=         P.AFH
2723        012020                  PHR.11=         P.AFI
2724        012036                  PHR.12=         P.AFJ
2725        012060                  PHR.13=         P.AFK
2726        012104                  PHR.14=         P.AFL
2727        012136                  PHR.15=         P.AFM
2728        012150                  FNC.1=          P.AFN
2729        012164                  FNC.2=          P.AFO
2730        012172                  FNC.3=          P.AFP
2731        012200                  FNC.4=          P.AFQ
2732        012216                  FNC.5=          P.AFR
2733        012226                  FNC.6=          P.AFS
2734        012234                  FNC.7=          P.AFT
2735        012244                  FNC.8=          P.AFU
2736        012260                  FNC.9=          P.AFV
2737        012272                  FNC.10=         P.AFW
2738        012302                  FNC.11=         P.AFX
2739        012322                  FNC.12=         P.AFY
2740        012334                  FNC.13=         P.AFZ
2741        012344                  FNC.14=         P.AGA
2742        012360                  FNC.15=         P.AGB
2743        012372                  FNC.16=         P.AGC
2744        012404                  FNC.17=         P.AGD
2745        012414                  FNC.18=         P.AGE
2746        012426                  FNC.19=         P.AGF
```

```
                                ;ML4AD
                                ;                ROUTINE DECLARATION SECTION

012440                          FNC.21=          P.AGG
012452                          FNC.22=          P.AGH
012464                          FNC.23=          P.AGI
012500                          REG.1=           P.AGJ
012510                          REG.2=           P.AGK
012516                          REG.3=           P.AGL
012524                          REG.4=           P.AGM
012532                          REG.5=           P.AGN
012540                          REG.6=           P.AGO
012546                          REG.7=           P.AGP
012554                          REG.8=           P.AGQ
012562                          REG.9=           P.AGR
012570                          REG.10=          P.AGS
012576                          REG.11=          P.AGT
012604                          REG.12=          P.AGU
012612                          REG.13=          P.AGV
012620                          REG.14=          P.AGW
012626                          REG.15=          P.AGX
012634                          REG.16=          P.AGY
012642                          REG.17=          P.AGZ
012652                          REG.18=          P.AHA
012660                          REG.19=          P.AHB
012666                          REG.20=          P.AHC
012676                          REG.21=          P.AHD
012706                          ASYNC=           P.AHE
012750                          SYNC=            P.AHF
013012                          ARR.DAT=         P.AHG
013052                          MEM.ARR=         P.AHH
013114                          INTER=           P.AHI
013154                          DATA.LATE=       P.AHJ
013214                          SC.SET=          P.AHK
013250                          TRBLE.LOOP=      P.AHL
013302                          RH.ERROR=        P.AHM
013330                          TIME.OUT=        P.AHN
013374                          D1.TEMP=         NIB.SAVE
013376                          D2.TEMP=         NIB.SAVE+2
013400                          E2.TEMP=         NIB.SAVE+4


                                        .SBTTL   LOAD.STACK ROUTINE DECLARATION SECTION

016034                          LOAD.STACK:
016034  004167  166514                  JSR      R1,$SAVE3                ;                2267
016040  016601  000012                  MOV      12(SP),R1                ; NIB.PTR,*      2322
016044  012702  015022                  MOV      #STK.OFF,R2
016050  060102                          ADD      R1,R2
016052  016601  000014                  MOV      14(SP),R1                ; STK.PTR,*
016056  016600  000012                  MOV      12(SP),R0                ; NIB.PTR,*      2318
```

```
                                ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
                                ;               ROUTINE DECLARATION SECTION             29-Mar-1982 16:21:03    PA:<
016062  006300                          ASL     R0
016064  066007  016070                  ADD     1$(R0),PC
016070  000024          1$:     .WORD   2$-1$
016072  000042                  .WORD   3$-1$
016074  000060                  .WORD   4$-1$
016076  000076                  .WORD   5$-1$
016100  000114                  .WORD   6$-1$
016102  000132                  .WORD   7$-1$
016104  000160                  .WORD   9$-1$
016106  000176                  .WORD   10$-1$
016110  000224                  .WORD   12$-1$
016112  000260                  .WORD   15$-1$
016114  005003          2$:     CLR     R3                      ;                       2322
016116  151203                  BISB    (R2),R3
016120  010100                  MOV     R1,R0
016122  160300                  SUB     R3,R0
016124  016703  175244          MOV     NIB.SAVE,R3
016130  000500                  BR      14$
016132  005003          3$:     CLR     R3                      ;                       2326
016134  151203                  BISB    (R2),R3
016136  010100                  MOV     R1,R0
016140  160300                  SUB     R3,R0
016142  016703  175226          MOV     NIB.SAVE,R3
016146  000433                  BR      8$
016150  005003          4$:     CLR     R3                      ;                       2330
016152  151203                  BISB    (R2),R3
016154  010100                  MOV     R1,R0
016156  160300                  SUB     R3,R0
016160  016703  175210          MOV     NIB.SAVE,R3
016164  000461                  BR      13$
016166  005003          5$:     CLR     R3                      ;                       2334
016170  151203                  BISB    (R2),R3
016172  010100                  MOV     R1,R0
016174  160300                  SUB     R3,R0
016176  016703  175172          MOV     NIB.SAVE,R3
016202  000437                  BR      11$
016204  005003          6$:     CLR     R3                      ;                       2338
016206  151203                  BISB    (R2),R3
016210  010100                  MOV     R1,R0
016212  160300                  SUB     R3,R0
016214  016703  175156          MOV     NIB.SAVE+2,R3
016220  000444                  BR      14$
016222  005003          7$:     CLR     R3                      ;                       2342
016224  151203                  BISB    (R2),R3
016226  010100                  MOV     R1,R0
016230  160300                  SUB     R3,R0
016232  016703  175140          MOV     NIB.SAVE+2,R3
016236  006203          8$:     ASR     R3
016240  006203                  ASR     R3
016242  006203                  ASR     R3
016244  006203                  ASR     R3
016246  000431                  BR      14$
```

```
                                ;ML4AD
                                ;                       ROUTINE DECLARATION SECTION                     29-Mar-1982 16:27:04  TOPS
                                ;                                                                       29-Mar-1982 16:21:03  PA:<
016250  005003                  9$:     CLR     R3                                      ;                                     2346
016252  151203                          BISB    (R2),R3
016254  010100                          MOV     R1,R0
016256  160300                          SUB     R3,R0
016260  016703  175112                  MOV     NIB.SAVE+2,R3
016264  000421                          BR      13$
016266  005003                  10$:    CLR     R3                                      ;                                     2350
016270  151203                          BISB    (R2),R3
016272  010100                          MOV     R1,R0
016274  160300                          SUB     R3,R0
016276  016703  175074                  MOV     NIB.SAVE+2,R3
016302  006203                  11$:    ASR     R3
016304  006203                          ASR     R3
016306  006203                          ASR     R3
016310  006203                          ASR     R3
016312  000406                          BR      13$
016314  005003                  12$:    CLR     R3                                      ;                                     2354
016316  151203                          BISB    (R2),R3
016320  010100                          MOV     R1,R0
016322  160300                          SUB     R3,R0
016324  016703  175050                  MOV     NIB.SAVE+4,R3
016330  000303                  13$:    SWAB    R3
016332  042703  177760          14$:    BIC     #177760,R3
016336  105060  015034                  CLRB    STACK(R0)
016342  150360  015034                  BISB    R3,STACK(R0)
016346  000207                          RTS     PC                                      ;                                     2318
016350  005003                  15$:    CLR     R3                                      ;                                     2358
016352  151203                          BISB    (R2),R3
016354  160301                          SUB     R3,R1
016356  016703  175016                  MOV     NIB.SAVE+4,R3
016362  006203                          ASR     R3
016364  006203                          ASR     R3
016366  006203                          ASR     R3
016370  006203                          ASR     R3
016372  000303                          SWAB    R3
016374  042703  177770                  BIC     #177770,R3
016400  105061  015034                  CLRB    STACK(R1)
016404  150361  015034                  BISB    R3,STACK(R1)
016410  000207                          RTS     PC                                      ;                                     2267

                                ; Routine Size:  119 words
                                ; Maximum stack depth per invocation:  4 words
```

```
;ML4AD
;                       ROUTINE DECLARATION SECTION                     29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (5)

;               2363    routine DAT_DM_XFER : novalue =
;               2364
;               2365    !++
;               2366    ! FUNCTIONAL DESCRIPTION:
;               2367    !
;               2368    !       A REPEATEDLY CALLED SEQUENCE
;               2369    !       OF ASSIGNMENT EXPRESSIONS TO
;               2370    !       LOAD THE DSA, BUS ADRS  AND WORD
;               2371    !       COUNT REGISTERS WITH APPROPRIATE
;               2372    !       INFORMATION BEFORE A MASS BUS
;               2373    !       TRANSFER CAN COMMENCE
;               2374    !
;               2375    !       LOADS A MASS BUS BLOCK TRANSFER,
;               2376    !       IN DIAGNOSTIC MODE, AT THE GOOD
;               2377    !       BLOCK ADRS.
;               2378    !
;               2379    !--
;               2380
;               2381        begin
;               2382        DAT_DM = ONE;                       !SET DATA DIAG MODE
;               2383        MLDA = .GOOD_BLK;                   !LOAD DSA REG WITH THE GOOD BLOCK ADRS
;               2384        MLBA = IO_BUF;                      !LOAD BUS ADRS REG WITH THE IO BUF ADRS
;               2385        MLWC = not 255;                     !LOAD WORD COUNT REG WITH COMPLIMENT 256
;               2386        end;
;
;
;                                       .SBTTL  DAT.DM.XFER ROUTINE DECLARATION SECTION
; 016412                        DAT.DM.XFER:
; 016412  152777  000010  177104          BISB    #10,@ML.REG+120         ;                       2382
; 016420  016777  175362  177006          MOV     GOOD.BLK,@ML.REG+30     ;                       2383
; 016426  012777  014022  176770          MOV     #IO.BUF,@ML.REG+20      ;                       2384
; 016434  012777  177400  176752          MOV     #-400,@ML.REG+10        ;                       2385
; 016442  000207                          RTS     PC                      ;                       2363
;
;                                       ; Routine Size:  13 words
;                                       ; Maximum stack depth per invocation:  0 words
;
;
```

```
                routine STRIPPER (WRD_CNT, NIB_CNT) : novalue =
                    begin

                !++
                ! FUNCTIONAL DESCRIPTION:
                !       STRIPPER RUNS IN DATA DIAG MODE AND STRIPS OUT
                !       A VARIABLE NUMBER OF GOOD NIBBLES FROM THE
                !       GOOD BLOCK AND STORES THEM INTO A CONTIGIOUS
                !       STACK FOR SEQUENTIAL ACCESSING
                !
                ! FORMAL PARAMETERS:
                !       WRD_CNT
                !       TELLS STRIPPER HOW MANY WORDS IN THE GOOD
                !       BLOCK TO READ OUT
                !
                !       NIB_CNT
                !       TELLS STRIPPER HOW MANY NIBBLES TO
                !       STRIP OUT OF EACH WORD.
                !
                ! IMPLICIT INPUTS:
                !--

                    local
                        STK_PTR;                                !STACK POINTER

                    incr CNT from 0 to .NIB_CNT do              !CLEAR OUT THE STACK OFFSETS
                        STK_OFF [.CNT] = ZEROES;

                    CLR_MBUS;
                    STK_PTR = -1;                               !RESET THE STACK POINTER
                    DAT_DM_XFER ();                             !SET UP A DATA XFER AT THE GOOD BLOCK
                    MLCS1 = read;                               !DO A READ XFER
                    DELAY (ONE_US);                             !ALLOW PROM DATA TO GET INTO THE MLPD REG

                    incr CNT from 0 to .WRD_CNT do              !LOAD THE STACK WITH ALL GOOD NIBBLE DATA
                        begin
                        PD_TEMP = .MLPD;                        !GET THE PROM DATA
                        DAT_CLK = ONE;                          !CLOCK OUT THE DATA WORD
                        DELAY (ONE_US);                         !
                        RD_LNG_WRD;                             !READ DATA DIAG REGS INTO NIBBLE SAVE

                        incr NIB_PTR from 0 to .NIB_CNT do      !STRIP OUT X NUMBER OF NIBBLES
                            begin
                            STK_PTR = .STK_PTR + 1;             !INCREMENT THE STACK POINTER

                            if .PD_TEMP [.NIB_PTR] IS_SET       !SEE IF THIS A GOOD NIBBLE
                            then
                                STK_OFF [.NIB_PTR] = (.STK_OFF [.NIB_PTR]) + (.NIB_CNT + 1)
                                                                !THEN INCREMENT IT'S STACK OFFSET
                            else
                                LOAD_STACK (.STK_PTR, .NIB_PTR);        !ELSE LOAD THE NIBBLE IN THE STACK
```

```
3018 ;ML4AD
3019 ;                  ROUTINE DECLARATION SECTION
3020
3021 ;          2439            end;
3022 ;          2440
3023 ;          2441        end;
3024 ;          2442
3025 ;          2443    CLR_MBUS;
3026 ;          2444    end;
3030
3031                                    .GLOBL  L$DLY
3032
3033
3034                                    .SBTTL  STRIPPER ROUTINE DECLARATION SECTION
3038 016444                        STRIPPER:
3039 016444  004167  166142              JSR     R1,$SAVE5               ;                               2387
3040 016450  005746                       TST     -(SP)
3041 016452  016604  000020              MOV     20(SP),R4               ; NIB.CNT,*                     2412
3042 016456  005001                       CLR     R1                      ; CNT
3043 016460  000403                       BR      2$
3044 016462  105061  015022        1$:   CLRB    STK.OFF(R1)             ; *(CNT)                        2413
3045 016466  005201                       INC     R1                      ; CNT                           2412
3046 016470  020104                2$:   CMP     R1,R4                   ; CNT,*
3047 016472  003773                       BLE     1$
3048 016474  152777  000040  176742      BISB    #40,@ML.REG+40          ;                               2413
3049 016502  016705  177324              MOV     ML.DUT,R5
3050 016506  042705  177770              BIC     #177770,R5
3051 016512  142777  000007  176724      BICB    #7,@ML.REG+40
3052 016520  150577  176720              BISB    R5,@ML.REG+40
3053 016524  012705  177777              MOV     #-1,R5
3054 016530  004767  177656              JSR     PC,DAT.DM.XFER          ; *,STK.PTR                     2416
3055 016534  012777  000071  176642      MOV     #71,@ML.REG             ;                               2417
3056 016542  012701  000001              MOV     #1,R1                   ;                               2418
3057 016546  001410                3$:   BEQ     6$                      ; *,$STMP2                      2419
3058 016550  016702  163342              MOV     L$DLY,R2                ; *,$STMP1
3059 016554  001403                       BEQ     5$
3060 016556  005016                4$:   CLR     (SP)                    ; $STMP
3061 016560  005302                       DEC     R2                      ; $STMP1
3062 016562  001375                       BNE     4$
3063 016564  005301                5$:   DEC     R1                      ; $STMP2
3064 016566  000767                       BR      3$
3065 016570  005003                6$:   CLR     R3                      ; CNT
3066 016572  000502                       BR      16$                                                     2421
3067 016574  017767  177034  176540  7$: MOV     @ML.REG+230,PD.TEMP     ;                               2423
3068 016602  152777  000020  176714      BISB    #20,@ML.REG+120         ;                               2424
3069 016610  012701  000001              MOV     #1,R1                   ; *,$STMP2                      2425
3070 016614  001410                8$:   BEQ     11$
3071 016616  016702  163274              MOV     L$DLY,R2                ; *,$STMP1
3072 016622  001403                       BEQ     10$
```

```
                                        ;ML4AD
                                        ;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04   TOPS
                                                                                                        29-Mar-1982 16:21:03   PA:<
016624  005016                  9$:     CLR     (SP)                    ; $STMP
016626  005302                          DEC     R2                      ; $STMP1
016630  001375                          BNE     9$
016632  005301                  10$:    DEC     R1                      ; $STMP2
016634  000767                          BR      8$
016636  017767  176732  174530  11$:    MOV     @ML.REG+170,D1.TEMP
016644  017767  176734  174524          MOV     @ML.REG+200,D2.TEMP
016652  017767  176706  174520          MOV     @ML.REG+160,E2.TEMP
016660  005002                          CLR     R2                      ; NIB.PTR                      2428
016662  000443                          BR      15$
016664  005205                  12$:    INC     R5                      ; STK.PTR                      2430
016666  010201                          MOV     R2,R1                   ; NIB.PTR,*                    2432
016670  006201                          ASR     R1
016672  006201                          ASR     R1
016674  006201                          ASR     R1
016676  062701  015342                  ADD     #PD.TEMP,R1
016702  010146                          MOV     R1,-(SP)
016704  010246                          MOV     R2,-(SP)                ; NIB.PTR,*
016706  042716  177770                  BIC     #177770,(SP)
016712  012746  000001                  MOV     #1,-(SP)
016716  005046                          CLR     -(SP)
016720  004767  164710                  JSR     PC,BL$GT2
016724  062706  000010                  ADD     #10,SP
016730  005300                          DEC     R0
016732  001011                          BNE     13$
016734  005001                          CLR     R1                      ;                              2434
016736  156201  015022                  BISB    STK.OFF(R2),R1          ; *(NIB.PTR),*
016742  060401                          ADD     R4,R1
016744  010100                          MOV     R1,R0
016746  005200                          INC     R0
016750  110062  015022                  MOVB    R0,STK.OFF(R2)          ; *,*(NIB.PTR)
016754  000405                          BR      14$                     ;
016756  010546                  13$:    MOV     R5,-(SP)                ; STK.PTR,*                    2432
016760  010246                          MOV     R2,-(SP)                ; NIB.PTR,*                    2437
016762  004767  177046                  JSR     PC,LOAD.STACK
016766  022626                          CMP     (SP)+,(SP)+
016770  005202                  14$:    INC     R2                      ; NIB.PTR                      2428
016772  020204                  15$:    CMP     R2,R4                   ; NIB.PTR,*
016774  003733                          BLE     12$
016776  005203                          INC     R3                      ; CNT                          2421
017000  020366  000022                  CMP     R3,22(SP)               ; CNT,WRD.CNT
017004  003673                          BLE     7$
017006  152777  000040  176430          BISB    #40,@ML.REG+40          ;                              2441
017014  016705  177012                  MOV     ML.DUT,R5
017020  042705  177770                  BIC     #177770,R5
017024  142777  000007  176412          BICB    #7,@ML.REG+40
017032  150577  176406                  BISB    R5,@ML.REG+40
017036  005726                          TST     (SP)+                   ;                              2387
017040  000207                          RTS     PC

                        ; Routine Size:  127 words
                        ; Maximum stack depth per invocation:  11 words
```

```
3130                                    ;ML4AD
3131                                    ;               ROUTINE DECLARATION SECTION
3132
3137
3138
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (7)

```
;ML4AD
;                ROUTINE DECLARATION SECTION
;
;          2445  routine CAL_CRC : novalue =
;          2446      begin
;          2447
;          2448  !++
;          2449  ! FUNCTIONAL DESCRIPTION:
;          2450  !       CALCULATE THE CRC CODE FOR THE GOOD BLOCKS
;          2451  !       FIRST CRC GROUP (60 NIBBLES) BY STRIPPING
;          2452  !       OUT GOOD NIBBLES AND MODULO 2 ADDING EACH
;          2453  !       BITS WEIGHT RESULTING IN A_CAL,B_CAL,P_CAL.
;          2454  !
;          2455  ! IMPLICIT INPUTS:
;          2456  !       P_CAL,A_CAL,B_CAL
;          2457  !       GOLBAL STORAGE LOCATION WHERE CALCULATED
;          2458  !       CRC CODES ARE STORED AND EXAMINED FROM
;          2459  !
;          2460  !       REM_TBL
;          2461  !       TABLE CONTAINING PRECALCULATED BIT POSITION
;          2462  !       WEIGHTS USED IN THE MODULO 2 CALCULATION OF THE
;          2463  !       CRC CODE
;          2464  !--
;          2465
;          2466      local
;          2467          CHANNEL,                          !CHANNEL POINTS TO THE BITS IN A WORD
;          2468          STK_PTR,                          !STACK POINTER
;          2469          NIB_SAV : bitvector [4],          !NIBBLE SAVE LOCATION
;          2470          ALOG,                             !INDEX INTO REMAINDER TABLE
;          2471          BLOG;                             !INDEX INTO REMAINDER TABLE
;          2472
;          2473      STRIPPER (12, 9);                     !STRIP OUT ALL 10 NIBBLE FROM 13 ARRAY WORDS
;          2474      STK_PTR = -1;                         !RESET STACK POINTER
;          2475
;          2476      incr PLOG from 0 to 5 do              !PLOG POINTS TO ONE CRC GROUP '6 WORDS'
;          2477          begin
;          2478          CHANNEL = -1;                     !RESET CHANNEL FOR EACH WORD
;          2479
;          2480          incr NIB_CNT from 0 to 8 do       !READ DATA NIBBLE 0-8 AND CALCULATE CRC
;          2481              begin
;          2482              STK_PTR = .STK_PTR + 1;       !INCREMENT THE STACK POINTER
;          2483              NIB_SAV = .stack [.STK_PTR];  !LOAD NIV_SAV WITH A STACK NIBBLE
;          2484
;          2485              incr BIT_TST from 0 to 3 do   !TEST THE BIT FOR BEING SET
;          2486                  begin
;          2487                  CHANNEL = .CHANNEL + 1;   !CHANNEL POINTS TO THIS BIT
;          2488
;          2489                  if .NIB_SAV [.BIT_TST] IS_SET   !SEE IF THIS BIT IS SET
;          2490                  then
;          2491                      begin                 !IF SET THEN MOD 2 ADD THE REMAINDER
;          2492                      ALOG = .PLOG + .CHANNEL;    !CALCULATE THE ALOG
;          2493                      BLOG = .PLOG + .CHANNEL*2;  !CALCULATE THE BLOG
;          2494
;          2495                      while .ALOG geq 63 do       !REDUCE ALOG UNTIL < 64
;          2496                          ALOG = .ALOG - 63;
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (7)

;          2497
;          2498                    while .BLOG geq 63 do        !REDUCE BLOG UNTIL <64
;          2499                        BLOG = .BLOG - 63;
;          2500
;          2501                    P_CAL = (.P_CAL) xor (.REM_TBL [.PLOG]);     !MOD 2 P_CAL WITH REM_TBL
;          2502                    A_CAL = (.A_CAL) xor (.REM_TBL [.ALOG]);     !MOD 2 A_CAL WITH REM_TBL
;          2503                    B_CAL = (.B_CAL) xor (.REM_TBL [.BLOG]);     !MOD 2 B_CAL WITH REM_TBL
;          2504                    end;
;          2505
;          2506                end;
;          2507
;          2508            end;
;          2509
;          2510        STK_PTR = .STK_PTR + 1;              !SKIP OVER THE CRC NIB NT THE STACK
;          2511        end;
;          2512
;          2513    end;

                                .SBTTL  CAL.CRC ROUTINE DFCLARATION SECTION
017042 004167 165544        CAL.CRC:JSR     R1,$SAVE5                       ;                       2445
017046 162706 000012                SUB     #12,SP
017052 012746 000014                MOV     #14,-(SP)                       ;                       2473
017056 012746 000011                MOV     #11,-(SP)
017062 004767 177356                JSR     PC,STRIPPER
017066 012766 177777 000006         MOV     #-1,6(SP)                       ; *,STK.PTR             2474
017074 005066 000004                CLR     4(SP)                           ; PLOG                  2476
017100 012766 177777 000010 1$:     MOV     #-1,10(SP)                      ; *,CHANNEL             2478
017106 005066 000012                CLR     12(SP)                          ; NIB.CNT               2480
017112 005266 000006        2$:     INC     6(SP)                           ; STK.PTR               2482
017116 016605 000006                MOV     6(SP),R5                        ; STK.PTR,*             2483
017122 116566 015034 000014         MOVB    STACK(R5),14(SP)                ; *,NIB.SAV
017130 105066 000015                CLRB    15(SP)                          ; NIB.SAV
017134 005003                       CLR     R3                              ; BIT.TST               2485
017136 005266 000010        3$:     INC     10(SP)                          ; CHANNEL               2487
017142 010305                       MOV     R3,R5                           ; BIT.TST,*             2489
017144 006205                       ASR     R5
017146 006205                       ASR     R5
017150 006205                       ASR     R5
017152 012704 000014                MOV     #14,R4
017156 060604                       ADD     SP,R4                           ; NIB.SAV,*
017160 060405                       ADD     R4,R5
017162 010546                       MOV     R5,-(SP)
017164 010346                       MOV     R3,-(SP)                        ; BIT.TST,*
017166 042716 177770                BIC     #177770,(SP)
017172 012746 000001                MOV     #1,-(SP)
017176 005046                       CLR     -(SP)
```

```
                                        ;ML4AD
                                        ;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04   TOPS
                                                                                                        29-Mar-1982 16:21:03   PA:<
017200  0047o7  164430                          JSR     PC,BL$GT2
017204  062706  000010                          ADD     #10,SP
017210  005300                                  DEC     R0
017212  001066                                  BNE     7$
017214  016601  000004                          MOV     4(SP),R1        ; PLOG,ALOG                    2492
017220  066601  000010                          ADD     10(SP),R1       ; CHANNEL,ALOG
017224  016605  000010                          MOV     10(SP),R5       ; CHANNEL,*                    2493
017230  006305                                  ASL     R5
017232  066605  000004                          ADD     4(SP),R5        ; PLOG,*
017236  010502                                  MOV     R5,R2           ; *,BLOG
017240  020127  000077            4$:           CMP     R1,#77          ; ALOG,*                       2495
017244  002403                                  BLT     5$
017246  162701  000077                          SUB     #77,R1          ; *,ALOG                       2496
017252  000772                                  BR      4$              ;                              2495
017254  020227  000077            5$:           CMP     R2,#77          ; BLOG,*                       2498
017260  002403                                  BLT     6$
017262  162702  000077                          SUB     #77,R2          ; *,BLOG                       2499
017266  000772                                  BR      5$              ;                              2498
017270  005005                    6$:           CLR     R5              ;                              2501
017272  016604  000004                          MOV     4(SP),R4        ; PLOG,*
017276  156405  015664                          BISB    REM.TBL(R4),R5
017302  010546                                  MOV     R5,-(SP)
017304  046716  176056                          BIC     P.CAL,(SP)
017310  040567  176052                          BIC     R5,P.CAL
017314  052667  176046                          BIS     (SP)+,P.CAL
017320  005005                                  CLR     R5              ;                              2502
017322  156105  015664                          BISB    REM.TBL(R1),R5  ; *(ALOG),*
017326  010504                                  MOV     R5,R4
017330  046704  176026                          BIC     A.CAL,R4
017334  040567  176022                          BIC     R5,A.CAL
017340  050467  176016                          BIS     R4,A.CAL
017344  005005                                  CLR     R5              ;                              2503
017346  156205  015664                          BISB    REM.TBL(R2),R5  ; *(BLOG),*
017352  010504                                  MOV     R5,R4
017354  046704  176004                          BIC     B.CAL,R4
017360  040567  176000                          BIC     R5,B.CAL
017364  050467  175774                          BIS     R4,B.CAL
017370  005203                    7$:           INC     R3              ; BIT.TST                      2485
017372  020327  000003                          CMP     R3,#3           ; BIT.TST,*
017376  003657                                  BLE     3$
017400  005266  000012                          INC     12(SP)          ; NIB.CNT                      2480
017404  026627  000012  000010                  CMP     12(SP),#10      ; NIB.CNT,*
017412  003637                                  BLE     2$
017414  005266  000006                          INC     6(SP)           ; STK.PTR                      2510
017420  005266  000004                          INC     4(SP)           ; PLOG                         2476
017424  026627  000004  000005                  CMP     4(SP),#5        ; PLOG,*
017432  003622                                  BLE     1$
017434  062706  000016                          ADD     #16,SP          ;                              2445
017440  000207                                  RTS     PC

                                        ; Routine Size:  128 words
                                        ; Maximum stack depth per invocation:  17 words
```

```
                                ;ML4AD
                                ;               ROUTINE DECLARATION SECTION
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (8)
;
;          2514  routine ERR_CHK_CRC =
;          2515      begin
;          2516
;          2517  !++
;          2518  ! FUNCTIONAL DESCRIPTION:
;          2519  !       'ERROR CHECK CRC' COMPARES HARDWARE GENERATED CRC
;          2520  !       CODES TO SOFTWARE CALCULATED CRC CODE FOR ONE
;          2521  !       CRC GROUP AND RETURNS A ONE ON DETECTION OF ERRORS
;          2522  !
;          2523  ! IMPLICIT INPUTS:
;          2524  !       P_CAL,A_CAL,B_CAL
;          2525  !       GLOBAL STORAGE LOCATIONS WHERE CALCULATED
;          2526  !       CRC CODES ARE STORED AND EXAMINED FROM
;          2527  !
;          2528  !       P_GEN,A_GEN,B_GEN
;          2529  !       GLOBAL STROAGE LOCATIONS WHERE HARDWARE GENERATED
;          2530  !       CRC CODES ARE STORED AND EXAMINED FROM
;          2531  !--
;          2532
;          2533      local
;          2534          POS,                                    !VARIABLE FOR FIELD SELECTOR
;          2535          TEMP;                                   !TEMPORARY STORAGE LOCATION
;          2536
;          2537      POS = -1;                                   !RESET THE POS
;          2538
;          2539      incr STK_PTR from 9 to 59 by 10 do          !CONVERT VERTICAL STACK CRC TO HORIZONTAL REPRESENTATION
;          2540          begin
;          2541          POS = .POS + 1;                         !INCREMENT POS
;          2542          TEMP = .stack [.STK_PTR];               !READ CRC NIB FROM THE STACK
;          2543          (P_GEN)<.POS, 1> = .TEMP<0, 1>;         !CONVERT 1ST BIT
;          2544          (A_GEN)<.POS, 1> = not .TEMP<1, 1>;     !CONVERT 2ND BIT
;          2545          (B_GEN)<.POS, 1> = not .TEMP<2, 1>;     !CONVERT 3ED BIT
;          2546          end;
;          2547
;          2548      if (.P_GEN neq .P_CAL) or (.A_GEN neq .A_CAL) or (.B_GEN neq .B_CAL) then return ONE else return ZERO;
;          2549
;          2550                                                  !COMPARE GENERATED CRC CODE TO CALCULATED CRC CODE
;          2551      end;
;
;                                .SBTTL  ERR.CHK.CRC ROUTINE DECLARATION SECTION
017442                  ERR.CHK.CRC:
017442  004167  165106          JSR     R1,$SAVE3                   ;                                       2514
017446  012701  177777          MOV     #-1,R1                      ; *.POS                                 2537
017452  012702  000011          MOV     #11,R2                      ; *.STK.PTR                             2539
017456  005201                  INC     R1                      1$: ; POS                                   2541
017460  005003                  CLR     R3                          ; TEMP                                  2542
```

```
                                        ;ML4AD
                                        ;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS
                                                                                                        29-Mar-1982 16:21:03    PA:<
017462  156203  015034                        BISB    STACK(R2),R3            ; *(STK.PTR),TEMP
017466  012746  015374                        MOV     #P.GEN,-(SP)            ;                                                2545
017472  010146                                MOV     R1,-(SP)                ; POS,*
017474  012746  000001                        MOV     #1,-(SP)
017500  010346                                MOV     R3,-(SP)                ; TEMP,*
017502  042716  177776                        BIC     #177776,(SP)
017506  004767  164360                        JSR     PC,BL$PU2
017512  012716  015370                        MOV     #A.GEN,(SP)             ;                                                2544
017516  010146                                MOV     R1,-(SP)                ; POS,*
017520  012746  000001                        MOV     #1,-(SP)
017524  005046                                CLR     -(SP)
017526  032703  000002                        BIT     #2,R3                   ; *,TEMP
017532  001401                                BEQ     2$
017534  005216                                INC     (SP)
017536  005116                        2$:     COM     (SP)
017540  004767  164326                        JSR     PC,BL$PU2
017544  012716  015372                        MOV     #B.GEN,(SP)             ;                                                2545
017550  010146                                MOV     R1,-(SP)                ; POS,*
017552  012746  000001                        MOV     #1,-(SP)
017556  005046                                CLR     -(SP)
017560  032703  000004                        BIT     #4,R3                   ; *,TEMP
017564  001401                                BEQ     3$
017566  005216                                INC     (SP)
017570  005116                        3$:     COM     (SP)
017572  004767  164274                        JSR     PC,BL$PU2
017576  062706  000024                        ADD     #24,SP                  ;                                                2540
017602  062702  000012                        ADD     #12,R2                  ; *,STK.PTR                                      2539
017606  020227  000073                        CMP     R2,#73                  ; STK.PTR,*
017612  003721                                BLE     1$
017614  026767  175554  175544              CMP     P.GEN,P.CAL             ;                                                2548
017622  001010                                BNE     4$
017624  026767  175540  175530              CMP     A.GEN,A.CAL
017632  001004                                BNE     4$
017634  026767  175532  175522              CMP     B.GEN,B.CAL
017642  001403                                BEQ     5$
017644  012701  000001                4$:     MOV     #1,R1
017650  000401                                BR      6$
017652  005001                        5$:     CLR     R1
017654  010100                        6$:     MOV     R1,R0
017656  000207                                RTS     PC                      ;                                                2515
                                                                              ;                                                2514

                                        ; Routine Size:  71 words
                                        ; Maximum stack depth per invocation:  14 words
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (9)

;          2552  routine FIND_COMP_BIT (BITS_XFERED) =
;          2553      begin
;          2554
;          2555  !++
;          2556  ! FUNCTIONAL DESCRIPTION:
;          2557  !       'FIND COMPLIMENT'  BIT IS CALLED BY THE
;          2558  !       SYNDROME DECODE TESTS AND INDEXES INTO
;          2559  !       THE IO_BUF LOOKING FOR COMPLIMENTED
;          2560  !       BITS.  A COMPLIMENTED BIT IS BY DEF A ONE
;          2561  !
;          2562  !       A ONE IS RETURNED IF THE INDEXED BIT IS
;          2563  !       NOT SET
;          2564  !
;          2565  ! FORMAL PARAMETERS:
;          2566  !       BITS_XFERED
;          2567  !       REPRESENTS THE BIT BEING TESTED
;          2568  !       IN THE DRIVE AND FROM THIS THE
;          2569  !       WORD AND BIT INDEX INTO THE
;          2570  !       IO_BUF CAN BE CALCULATED
;          2571  !
;          2572  ! IMPLICIT INPUTS:
;          2573  !--
;          2574
;          2575      local
;          2576          COMP_WRD,                          !WORD WHERE COMPLINEMT BIT IS LOCATED
;          2577          COMP_BIT,                          !BIT THAT IS COMPLINEMTED
;          2578          BUF_SAV : bitvector [16];          !STORES THE WORD IN WHICH THE BIT IS COMPLIMENTED
;          2579
;          2580      COMP_WRD = .BITS_XFERED/16;            !CALCULATE THE COMP WORD
;          2581      COMP_BIT = .BITS_XFERED mod 16;        !CALCULATE THE COMP BIT
;          2582      BUF_SAV = .IO_BUF [.COMP_WRD];         !LOAD THE COMP WORD INTO BUF_SAV
;          2583      IO_BUF [.COMP_WRD] = ZEROES;           !ZERO THE COMP WORD
;          2584
;          2585      if .BUF_SAV [.COMP_BIT] IS_NOT_SET then return ZERO else return ONE;
;          2586
;          2587                                             !FIND COMPLIMENTED BIT AND RETURN ERROR STATUS
;          2588      end;
;
;                                  .SBTTL  FIND.COMP.BIT ROUTINE DECLARATION SECTION
017660                       FIND.COMP.BIT:
017660  004167  164654              JSR     R1,$SAVE2                  ;                          2552
017664  005746                      TST     -(SP)
017666  016646  000012              MOV     12(SP),-(SP)               ; BITS.XFERED,*            2580
017672  012746  000020              MOV     #20,-(SP)
017676  004767  164544              JSR     PC,BL$DIV
017702  010001                      MOV     R0,R1                      ; *,COMP.WRD
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;       ROUTINE DECLARATION SECTION                             29-Mar-1982 16:21:03    PA:<
017704  016616  000016                        MOV     16(SP),(SP)             ; BITS.XFERED,*                                 2581
017710  012746  000020                        MOV     #20,-(SP)
017714  004767  164540                        JSR     PC,BL$MOD
017720  010002                                MOV     R0,R2                   ; *,COMP.BIT
017722  010100                                MOV     R1,R0                   ; COMP.WRD,*                                    2582
017724  006300                                ASL     R0
017726  016066  014022  000006                MOV     IO.BUF(R0),6(SP)        ; *,BUF.SAV
017734  005060  014022                        CLR     IO.BUF(R0)              ;                                               2583
017740  010200                                MOV     R2,R0                   ; COMP.BIT,*                                    2585
017742  006200                                ASR     R0
017744  006200                                ASR     R0
017746  006200                                ASR     R0
017750  012701  000006                        MOV     #6,R1
017754  060601                                ADD     SP,R1                   ; BUF.SAV,*
017756  060100                                ADD     R1,R0
017760  010046                                MOV     R0,-(SP)
017762  010246                                MOV     R2,-(SP)                ; COMP.BIT,*
017764  042716  177770                        BIC     #177770,(SP)
017770  012746  000001                        MOV     #1,-(SP)
017774  005046                                CLR     -(SP)
017776  004767  163632                        JSR     PC,BL$GT2
020002  062706  000010                        ADD     #10,SP
020006  005700                                TST     R0
020010  001002                                BNE     1$
020012  005001                                CLR     R1
020014  000402                                BR      2$
020016  012701  000001                1$:     MOV     #1,R1
020022  010100                        2$:     MOV     R1,R0                   ;                                               2553
020024  062706  000010                        ADD     #10,SP                  ;                                               2552
020030  000207                                RTS     PC

                                        ; Routine Size:  53 words
                                        ; Maximum stack depth per invocation:  11 words
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (10)

;          2589  routine FORCE_REM (PLOG, CHAN) : novalue =
;          2590      begin
;          2591
;          2592  !++
;          2593  ! FUNCTIONAL DESCRIPTION:
;          2594  !       VIA ECC_DM AND THE PRECALCULATED MODULO 2
;          2595  !       REMAINDERS STORED IN REM_TBL[], THE
;          2596  !       CRC_A,CRC_B AND PAR_CRC_WRD
;          2597  !       ARE FORCED WITH ERRONEOUS CRC DATA TO
;          2598  !       FORCE PERDICTABLE ECC ERRORS
;          2599  !
;          2600  ! FORMAL PARAMETERS:
;          2601  !       PLOG
;          2602  !       POINTS TO ONE OF THE SIX WORDS OF
;          2603  !       A CRC GROUP
;          2604  !
;          2605  !       CHAN
;          2606  !       POINTS TO ONE OF 36 CHANNELS IN
;          2607  !       A CRC GROUP
;          2608  !
;          2609  ! IMPLICIT INPUTS:
;          2610  !--
;          2611
;          2612      local
;          2613          TMP_E1,                         !TEMP STORAGE FOR E1 DATA WORD
;          2614          ALOG,                           !INDEX INTO REM_TBL
;          2615          BLOG;                           !INDEX INTO REM_TBL
;          2616
;          2617      ALOG = .PLOG + .CHAN;               !CALCULATE A_LOG
;          2618      BLOG = .PLOG + .CHAN*2;             !CALCULATE B_LOG
;          2619
;          2620      while .ALOG geq 63 do               !REDUCE A_LOG UNTIL < 64
;          2621          ALOG = .ALOG - 63;
;          2622
;          2623      while .BLOG geq 63 do               !REDUCE B_LOG UNTIL < 64
;          2624          BLOG = .BLOG - 63;
;          2625
;          2626      ECC_DM = ONE;                       !SET ECC DIAG MODE
;          2627  !+
;          2628  ! THE FOLLOWING ASSIGNMENT:
;          2629  !
;          2630  !       MLE1 = .TMP_E1
;          2631  !
;          2632  ! IS EQUIVALENT TO THE FOLLOWING
;          2633  ! TWO ASSINGMENTS:
;          2634  !
;          2635  !   PAR_CRC_WRD = .REM_TBL [.PLOG];     !LOAD PAR_CRC_WRD WITH REM_TBL
;          2636  !   CRC_A = .REM_TBL [.ALOG];           !LOAD CRC_A WITH REM_TBL
;          2637  !
;          2638  ! THIS IS NECESSARY DUE TO THE FACT THAT
;          2639  ! THE E1 REGISTER IS WORD ORIENTATED AND
;          2640  ! THE BLISS COMPILER GENERATES BYTE INST
```

```
;ML4AD
;                       ROUTINE DECLARATION SECTION                     29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (10)
;
;               2641    ! TO ACCESS THE REGISTER.
;               2642    !-
;               2643
;               2644        TMP_E1 = ZEROES;                            !CLEAR OUT THE TEMP WORD
;               2645        TMP_E1<0, 6> = .REM_TBL [.ALOG];            !LOAD CRC_A WITH REM_TBL
;               2646        TMP_E1<8, 6> = .REM_TBL [.PLOG];            !LOAD PAR_CRC_WRD WITH REM_TBL
;               2647        MLET = .TMP_E1;                             !LOAD THE E1 REGISTER WITH DATA
;               2648
;               2649        CRC_B = .REM_TBL [.BLOG];                   !LOAD CRC_B WITH REM_TBL
;               2650        end;

                                        .SBTTL  FORCE.REM ROUTINE DECLARATION SECTION
                                FORCE.REM:
020032  004167  164534                    JSR     R1,$SAVE4               ;                                       2589
020036  016601  000016                    MOV     16(SP),R1               ; PLOG,*                                2617
020042  010100                            MOV     R1,R0                   ; *,ALOG
020044  066600  000014                    ADD     14(SP),R0               ; CHAN,ALOG
020050  016602  000014                    MOV     14(SP),R2               ; CHAN,*                                2618
020054  006302                            ASL     R2
020056  060102                            ADD     R1,R2
020060  010203                            MOV     R2,R3                   ; *,BLOG
020062  020027  000077            1$:     CMP     R0,#77                  ; ALOG,*                                2620
020066  002403                            BLT     2$
020070  162700  000077                    SUB     #77,R0                  ; *,ALOG                                2621
020074  000772                            BR      1$                      ;                                       2620
020076  020327  000077            2$:     CMP     R3,#77                  ; BLOG,*                                2623
020102  002403                            BLT     3$
020104  162703  000077                    SUB     #77,R3                  ; *,BLOG                                2624
020110  000772                            BR      2$                      ;                                       2623
020112  152777  000001  175404    3$:     BISB    #1,@ML.REG+120          ;                                       2626
020120  005002                            CLR     R2                      ; TMP.E1                                2644
020122  116004  015664                    MOVB    REM.TBL(R0),R4          ; *(ALOG),*                             2645
020126  042704  177700                    BIC     #177700,R4
020132  142702  000077                    BICB    #77,R2                  ; *,TMP.E1
020136  050402                            BIS     R4,R2                   ; *,TMP.E1
020140  116104  015664                    MOVB    REM.TBL(R1),R4          ;                                       2646
020144  000304                            SWAB    R4
020146  042704  140377                    BIC     #140377,R4
020152  042702  037400                    BIC     #37400,R2               ; *,TMP.E1
020156  050402                            BIS     R4,R2                   ; *,TMP.E1
020160  010277  175370                    MOV     R2,@ML.REG+150          ; TMP.E1,*
020164  116304  015664                    MOVB    REM.TBL(R3),R4          ; *(BLOG),*                             2647
020170  042704  177700                    BIC     #177700,R4                                                      2649
020174  142777  000077  175362            BICB    #77,@ML.REG+160
020202  150477  175356                    BISB    R4,@ML.REG+160
020206  000207                            RTS     PC                      ;                                       2589
```

```
3638                                    ;ML4AD
3639                                    ;               ROUTINE DECLARATION SECTION
3640
3641
3642                                    ; Routine Size:  55 words
3643                                    ; Maximum stack depth per invocation:  5 words
3648
3649
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;               ROUTINE DECLARATION SECTION               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (11)

;          2651  routine FOR_ECC_ERR (SGL, UNC) : novalue =
;          2652      begin
;          2653
;          2654  !++
;          2655  ! FUNCTIONAL DESCRIPTION:
;          2656  !       'FORCE ECC ERROR' WHEN CALLED WILL FORCE
;          2657  !       VIA ECC DIAG MODE, SINGLE BIT ECC ERRORS
;          2658  !       OR UNCORRECTABLE ECC ERRORS INTO THE
;          2659  !       ECC DIAGNOSTIC REGISTERS
;          2660  !
;          2661  ! FORMAL PARAMETERS:
;          2662  !       SGL
;          2663  !       WHEN SET TO A ONE THIS
;          2664  !       ROUTINE WILL FORCE SINGLE BIT ERRORS
;          2665  !
;          2666  !       UNC
;          2667  !       WHEN SET TO A ONE THIS ROUTINE WILL
;          2668  !       FORCE UNCORRECTABLE ECC ERRORS
;          2669  !
;          2670  ! IMPLICIT INPUTS:
;          2671  !--
;          2672
;          2673      local
;          2674          TMP_E1;                          !TEMP STORAGE FOR E1 DATA WORD
;          2675
;          2676      ECC_DM = ONE;                        !SET ECC DIAG MODE
;          2677
;          2678      if .SGL                              !IF SGL IS TRUE
;          2679      then
;          2680          begin                            !THEN FORCE A SINGLE BIT ERROR
;          2681  !+
;          2682  ! THE FOLLOWING ASSIGNMENT:
;          2683  !
;          2684  !       MLE1 = .TMP_E1
;          2685  !
;          2686  ! IS EQUIVALENT TO THE FOLLOWING
;          2687  ! TWO ASSINGMENTS:
;          2688  !
;          2689  !   PAR_CRC_WRD = %b'111111';          !LOAD PAR_CRC_WRD WITH REM_TBL
;          2690  !   CRC_A = %b'000000';                !LOAD CRC_A WITH REM_TBL
;          2691  !
;          2692  ! THIS IS NECESSARY DUE TO THE FACT THAT
;          2693  ! THE E1 REGISTER IS WORD ORIENTATED AND
;          2694  ! THE BLISS COMPILER GENERATES BYTE INST
;          2695  ! TO ACCESS THE REGISTER.
;          2696  !-
;          2697
;          2698          TMP_E1 = ZEROES;                     !CLEAR OUT THE TEMP WORD
;          2699          TMP_E1<0, 6> = %b'000000';           !LOAD CRC_A WITH REM_TBL
;          2700          TMP_E1<8, 6> = %b'111111';           !LOAD PAR_CRC_WRD WITH REM_TBL
;          2701          MLE1 = .TMP_E1;                      !LOAD THE E1 REGISTER WITH DATA
;          2702
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (11)
;
;          2703          CRC_B = %b'000000';
;          2704          end
;          2705      else
;          2706          begin                                 !ELSE FORCE A UNCORRECTABLE ERROR
;          2707  !+
;          2708  ! THE FOLLOWING ASSIGNMENT:
;          2709  !
;          2710  !       MLE1 = .TMP_E1
;          2711  !
;          2712  ! IS EQUIVALENT TO THE FOLLOWING
;          2713  ! TWO ASSINGMENTS:
;          2714  !
;          2715  !   PAR_CRC_WRD = %b'111111';          !LOAD PAR_CRC_WRD WITH REM_TBL
;          2716  !   CRC_A = %b'111111';                !LOAD CRC_A WITH REM_TBL
;          2717  !
;          2718  ! THIS IS NECESSARY DUE TO THE FACT THAT
;          2719  ! THE E1 REGISTER IS WORD ORIENTATED AND
;          2720  ! THE BLISS COMPILER GENERATES BYTE INST
;          2721  ! TO ACCESS THE REGISTER.
;          2722  !-
;          2723
;          2724          TMP_E1 = ZEROES;                          !CLEAR OUT THE TEMP WORD
;          2725          TMP_E1<0, 6> = %b'111111';        !LOAD CRC_A WITH REM_TBL
;          2726          TMP_E1<8, 6> = %b'111111';        !LOAD PAR_CRC_WRD WITH REM_TBL
;          2727          MLE1 = .TMP_E1;                   !LOAD THE E1 REGISTER WITH DATA
;          2728
;          2729          CRC_B = %b'000000';
;          2730          end;
;          2731
;          2732      end;
;
;                                .SBTTL  FOR.ECC.ERR ROUTINE DECLARATION SECTION
;                        FOR.ECC.ERR:
020210                          BISB    #1,@ML.REG+120          ;                                       2676
020216  005000                  CLR     R0                      ; TMP.E1                                2698
020220  032766  000001  000004  BIT     #1,4(SP)                ; *,SGL                                 2678
020226  001403                  BEQ     1$
020230  142700  000077          BICB    #77,R0                  ; *,TMP.E1                              2699
020234  000402                  BR      2$                      ;                                       2700
020236  152700  000077  1$:     BISB    #77,R0                  ; *,TMP.E1                              2725
020242  052700  037400  2$:     BIS     #37400,R0               ; *,TMP.E1                              2726
020246  010077  175302          MOV     R0,@ML.REG+150          ; TMP.E1,*                              2727
020252  142777  000077  175304  BICB    #77,@ML.REG+160         ;                                       2729
020260  000207                  RTS     PC                      ;                                       2651

                        ; Routine Size:  21 words
```

```
3763                                    ;ML4AD
3764                                    ;                ROUTINE DECLARATION SECTION
3765
3766                                    ; Maximum stack depth per invocation:  0 words
3771
3772
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (12)

;          2733  routine FIRST_BLK_XFER : novalue =
;          2734
;          2735  !++
;          2736  ! FUNCTIONAL DESCRIPTION:
;          2737  !       A REPEATEDLY CALLED SEQUENCE OF
;          2738  !       ASSIGNMENT EXPRESSION TO LOAD
;          2739  !       THE DSA, BUS ADRS AND WORD COUNT
;          2740  !       REGISTERS WITH APPROPRIATE INFORMATION
;          2741  !       BEFORE MASS BUS TRANSFERS CAN
;          2742  !       COMMENCE.
;          2743  !
;          2744  !       LOADS A MASS_BUS BLOCK XFERR AT
;          2745  !       BLOCK ZERO.
;          2746  !
;          2747  !--
;          2748
;          2749      begin
;          2750      ECC_DIS = ONE;                    !DISABLE ERROR CORRECTION
;          2751      MLDA = ZEROES;                    !LOAD THE DSA REG WITH SECTOR ZERO
;          2752      MLBA = IO_BUF;                    !LOAD THE BUS ADDRESS REG WITH IO_BUF ADRS
;          2753      MLWC = not 255;                   !LOAD WORD COUNT REG WITH COMPLIMENT 256
;          2754      end;
3802
3803                                        .SBTTL  FIRST.BLK.XFER ROUTINE DECLARATION SECTION
3807 020262                         FIRST.BLK.XFER:
3808 020262  152777  000002  175234        BISB    #2,@ML.REG+120                  ;  2750
3809 020270  005077  175140                CLR     @ML.REG+30                      ;  2751
3810 020274  012777  014022  175122        MOV     #IO.BUF,@ML.REG+20              ;  2752
3811 020302  012777  177400  175104        MOV     #-400,@ML.REG+10                ;  2753
3812 020310  000207                        RTS     PC                              ;  2733
3813
3814                                ; Routine Size:  12 words
3815                                ; Maximum stack depth per invocation:  0 words
3820
3821
```

```
;ML4AD                                                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                    ROUTINE DECLARATION SECTION                29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (13)

;          2755  routine GD_BLK_XFER : novalue =
;          2756
;          2757  !++
;          2758  ! FUNCTIONAL DESCRIPTION:
;          2759  !       A REPEATEDLY CALLED SEQUENCE OF
;          2760  !       ASSIGNMENT EXPRESSIONS TO LOAD
;          2761  !       THE DSA, BUS ADRS AND WORD
;          2762  !       COUNT REGISTERS WITH APPROPRIATE
;          2763  !       INFORMATION BEFORE A MASS BUS
;          2764  !       TRANSFERS CAN COMMENCE
;          2765  !
;          2766  !       LOADS A MASS BUS BLOCK XFERR
;          2767  !       AT THE GOOD BLOCK ADRS FOUND
;          2768  !       BY THE READ WRITE ARRAYS WITH
;          2769  !       PROM DATA TEST
;          2770  !
;          2771  !--
;          2772
;          2773      begin
;          2774      ECC_DIS = ONE;                            !DISABLE ERROR CORRECTION
;          2775      MLDA = .GOOD_BLK;                         !LOAD DSA REG WITH THE GOOD BLOCK ADRS
;          2776      MLBA = IO_BUF;                            !LOAD BUS ADRS REG WITH IO BUF ADRS
;          2777      MLWC = not 255;                           !LOAD WORD COUNT REG WITH COMPLIMENT 256
;          2778      end;

                                .SBTTL  GD.BLK.XFER ROUTINE DECLARATION SECTION
020312                  GD.BLK.XFER:
020312  152777  000002  175204          BISB    #2,@ML.REG+120                  ;  2774
020320  016777  173462  175106          MOV     GOOD.BLK,@ML.REG+30             ;  2775
020326  012777  014022  175070          MOV     #IO.BUF,@ML.REG+20              ;  2776
020334  012777  177400  175052          MOV     #-400,@ML.REG+10                ;  2777
020342  000207                          RTS     PC                              ;  2755

                                ; Routine Size:  13 words
                                ; Maximum stack depth per invocation:  0 words
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION

;          2779  routine LAST_BLK_XFER : novalue =
;          2780
;          2781  !++
;          2782  ! FUNCTIONAL DESCRIPTION:
;          2783  !       A REPEATEDLY CALLED SEQUENCE OF
;          2784  !       ASSIGNMENT EXPRESSIONS TO LOAD
;          2785  !       THE DSA, BUS ADRS AND WORD
;          2786  !       COUNT REGISTERS WITH APPROPRIATE
;          2787  !       INFORMATION BEFORE A MBUS
;          2788  !       TRANSFER CAN COMMENCE
;          2789  !
;          2790  !       LOADS A MASS BUS BLOCK
;          2791  !       TRANSFER AT THE LAST BLOCK
;          2792  !       ADDRESS
;          2793  !
;          2794  !--
;          2795
;          2796      begin
;          2797      ECC_DIS = ONE;                  !DISABLE ERROR CORRECTION
;          2798      MLDA = .LST_BLK;                !LOAD DSA REG WITH THE LAST BLOCK ADRS
;          2799      MLBA = IO_BUF;                  !LOAD BUS ADRS REG WITH THE IO_BUF ADRS
;          2800      MLWC = not 255;                 !LOAD WORD COUNT REG WITH COMPLIMENT 256
;          2801      end;
;
;                          .SBTTL  LAST.BLK.XFER ROUTINE DECLARATION SECTION
020344                  LAST.BLK.XFER:
020344  152777  000002  175152          BISB    #2,@ML.REG+120          ;2797
020352  016777  173436  175054          MOV     LST.BLK,@ML.REG+30      ;2798
020360  012777  014022  175036          MOV     #IO.BUF,@ML.REG+20      ;2799
020366  012777  177400  175020          MOV     #-400,@ML.REG+10        ;2800
020374  000207                          RTS     PC                      ;2779
;
;                  ; Routine Size:  13 words
;                  ; Maximum stack depth per invocation:  0 words
;
;
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION

routine TST_LNG_WRD (NIB_NUM, NIB_PAT, ERR_FLG) : novalue =
    begin

!++
! FUNCTIONAL DESCRIPTION:
!       COMPARES THE CURRENT NIBBLE
!       POSITION IN 'NIB_SAVE' WITH THE
!       CURRENT TEST PATTERN.  IF THE
!       TWO VALUES ARE NOT EQUAL AN
!       ERROR FLG IS SET WHICH THE
!       CALLER CAN INTERIGATE
!
! FORMAL PARAMETERS:
!       NIB_NUM
!       CASE SELECT EXPRESSION TO SELECT THE
!       CURRENT NIBBLE TO BE EXAMINED
!
!       NIB_PAT
!       CURRENT NIBBLE PATTERN TO BE
!       COMPARED
!
!       ERR_FLG
!       CONTAINS THE ADDRESS (PASSED BY REF)
!       OF THE CALLERS ERROR FLG
!       TO ENABLE THE CALLER TO EXAMINE
!       THE ERROR STATUS OF THE ROUTINE CALL
!
! IMPLICIT INPUTS:
!       NIB_SAVE
!       BLOCK OF 3 WORDS TO STORE
!       THE DATA FOUND IN MLD1, MLD2
!       AND MLE2 AFTER A DIAGNOSTIC MODE
!       READ
!
! IMPLICIT OUTPUTS:     NONE
!--

    .ERR_FLG = ZERO;                                !CLEAR THE ERROR FLAG BACK IN THE CALLING ROUTINE

    case .(NIB_NUM) from 0 to 9 of                  !SELECT THE NIBBLE TO BE TESTED
        set

        [0] :

            if .NIB_SAVE [NIB_0] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
                                                    !TEST NIBBLE 0 AND SET ERR FLG IF NEQ

        [1] :

            if .NIB_SAVE [NIB_1] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
```

```
;          2854                                                    !TEST NIBBLE 1 AND SET ERR FLG IF NEQ
;          2855
;          2856          [2] :
;          2857
;          2858              if .NIB_SAVE [NIB_2] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
;          2859
;          2860                                                    !TEST NIBBLE 2 AND SET ERR FLG IF NEQ
;          2861
;          2862          [3] :
;          2863
;          2864              if .NIB_SAVE [NIB_3] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
;          2865
;          2866                                                    !TEST NIBBLE 3 AND SET ERR FLG IF NEQ
;          2867
;          2868          [4] :
;          2869
;          2870              if .NIB_SAVE [NIB_4] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
;          2871
;          2872                                                    !TEST NIBBLE 4 AND SET ERR FLG IF NEQ
;          2873
;          2874          [5] :
;          2875
;          2876              if .NIB_SAVE [NIB_5] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
;          2877
;          2878                                                    !TEST NIBBLE 5 AND SET ERR FLG IF NEQ
;          2879
;          2880          [6] :
;          2881
;          2882              if .NIB_SAVE [NIB_6] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
;          2883
;          2884                                                    !TEST NIBBLE 6 AND SET ERR FLG IF NEQ
;          2885
;          2886          [7] :
;          2887
;          2888              if .NIB_SAVE [NIB_7] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
;          2889
;          2890                                                    !TEST NIBBLE 7 AND SET ERR FLG IF NEQ
;          2891
;          2892          [8] :
;          2893
;          2894              if .NIB_SAVE [NIB_8] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
;          2895
;          2896                                                    !TEST NIBBLE 8 AND SET ERR FLG IF NEQ
;          2897
;          2898          [9] :
;          2899
;          2900              if .NIB_SAVE [NIB_9] neq .(NIB_PAT)<0, 3>
;          2901              then
;          2902                  .ERR_FLG = ONE                    !TEST NIBBLE 9 AND SET ERR FLG IF NEQ
;          2903          tes;
;          2904
;          2905      end;
```

```
                                        .SBTTL  TST.LNG.WRD ROUTINE DECLARATION SECTION
020376                          TST.LNG.WRD:
020376  004167  164136                JSR     R1,$SAVE2                       ;                                       2802
020402  016600  000010                MOV     10(SP),R0                       ; ERR.FLG,*                             2839
020406  005010                        CLR     (R0)
020410  016601  000014                MOV     14(SP),R1                       ; NIB.NUM,*                             2841
020414  006301                        ASL     R1
020416  066107  020422                ADD     1$(R1),PC
020422  000024                  1$:   .WORD   2$-1$
020424  000052                        .WORD   3$-1$
020426  000110                        .WORD   4$-1$
020430  000140                        .WORD   5$-1$
020432  000200                        .WORD   6$-1$
020434  000226                        .WORD   7$-1$
020436  000264                        .WORD   8$-1$
020440  000314                        .WORD   9$-1$
020442  000354                        .WORD   10$-1$
020444  000404                        .WORD   11$-1$
020446  016602  000012          2$:   MOV     12(SP),R2                       ; NIB.PAT,*                             2846
020452  042702  177760                BIC     #177760,R2
020456  016701  172712                MOV     NIB.SAVE,R1
020462  042701  177760                BIC     #177760,R1
020466  020102                        CMP     R1,R2
020470  001577                        BEQ     13$
020472  000574                        BR      12$
020474  016601  000012          3$:   MOV     12(SP),R1                       ; NIB.PAT,*                             2852
020500  042701  177760                BIC     #177760,R1
020504  016702  172664                MOV     NIB.SAVE,R2
020510  006202                        ASR     R2
020512  006202                        ASR     R2
020514  006202                        ASR     R2
020516  006202                        ASR     R2
020520  042702  177760                BIC     #177760,R2
020524  020201                        CMP     R2,R1
020526  001560                        BEQ     13$
020530  000555                        BR      12$
020532  016601  000012          4$:   MOV     12(SP),R1                       ; NIB.PAT,*                             2858
020536  042701  177760                BIC     #177760,R1
020542  016702  172626                MOV     NIB.SAVE,R2
020546  000302                        SWAB    R2
020550  042702  177760                BIC     #177760,R2
020554  020201                        CMP     R2,R1
020556  001544                        BEQ     13$
020560  000541                        BR      12$
020562  016601  000012          5$:   MOV     12(SP),R1                       ; NIB.PAT,*                             2864
```

```
                                        ;ML4AD                                                                  29-Mar-1982 16:23:04    TOPS
                                        ;               ROUTINE DECLARATION SECTION                             29-Mar-1982 16:21:03    PA:<
                                        
020566  042701  177760                        BIC     #177760,R1
020572  016702  172576                        MOV     NIB.SAVE,R2
020576  006202                              ASR     R2
020600  006202                              ASR     R2
020602  006202                              ASR     R2
020604  006202                              ASR     R2
020606  000302                              SWAB    R2
020610  042702  177760                        BIC     #177760,R2
020614  020201                              CMP     R2,R1
020616  001524                              BEQ     13$
020620  000521                              BR      12$
020622  016601  000012              6$:     MOV     12(SP),R1               ; NIB.PAT,*                     2870
020626  042701  177760                        BIC     #177760,R1
020632  016702  172540                        MOV     NIB.SAVE+2,R2
020636  042702  177760                        BIC     #177760,R2
020642  020201                              CMP     R2,R1
020644  001511                              BEQ     13$
020646  000506                              BR      12$
020650  016601  000012              7$:     MOV     12(SP),R1               ; NIB.PAT,*                     2876
020654  042701  177760                        BIC     #177760,R1
020660  016702  172512                        MOV     NIB.SAVE+2,R2
020664  006202                              ASR     R2
020666  006202                              ASR     R2
020670  006202                              ASR     R2
020672  006202                              ASR     R2
020674  042702  177760                        BIC     #177760,R2
020700  020201                              CMP     R2,R1
020702  001472                              BEQ     13$
020704  000467                              BR      12$
020706  016601  000012              8$:     MOV     12(SP),R1               ; NIB.PAT,*                     2882
020712  042701  177760                        BIC     #177760,R1
020716  016702  172454                        MOV     NIB.SAVE+2,R2
020722  000302                              SWAB    R2
020724  042702  177760                        BIC     #177760,R2
020730  020201                              CMP     R2,R1
020732  001456                              BEQ     13$
020734  000453                              BR      12$
020736  016601  000012              9$:     MOV     12(SP),R1               ; NIB.PAT,*                     2888
020742  042701  177760                        BIC     #177760,R1
020746  016702  172424                        MOV     NIB.SAVE+2,R2
020752  006202                              ASR     R2
020754  006202                              ASR     R2
020756  006202                              ASR     R2
020760  006202                              ASR     R2
020762  000302                              SWAB    R2
020764  042702  177760                        BIC     #177760,R2
020770  020201                              CMP     R2,R1
020772  001436                              BEQ     13$
020774  000433                              BR      12$
020776  016601  000012              10$:    MOV     12(SP),R1               ; NIB.PAT,*                     2894
021002  042701  177760                        BIC     #177760,R1
021006  016702  172366                        MOV     NIB.SAVE+4,R2
```

```
4148                                    ;ML4AD
4149                                    ;               ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS
4150                                                                                                   29-Mar-1982 16:21:03    PA:<
4151 021012  000302                            SWAB    R2
4152 021014  042702  177760                   BIC     #177760,R2
4153 021020  020201                            CMP     R2,R1
4154 021022  001422                            BEQ     13$
4155 021024  000417                            BR      12$
4156 021026  016601  000012            11$:    MOV     12(SP),R1               ; NIB.PAT,*                    2900
4157 021032  042701  177770                   BIC     #177770,R1
4158 021036  016702  172336                   MOV     NIB.SAVE+4,R2
4159 021042  006202                            ASR     R2
4160 021044  006202                            ASR     R2
4161 021046  006202                            ASR     R2
4162 021050  006202                            ASR     R2
4163 021052  000302                            SWAB    R2
4164 021054  042702  177770                   BIC     #177770,R2
4165 021060  020201                            CMP     R2,R1
4166 021062  001402                            BEQ     13$
4167 021064  012710  000001            12$:    MOV     #1,(R0)                 ;                              2902
4168 021070  000207                    13$:    RTS     PC                      ;                              2802
4169
4170                                    ; Routine Size:  158 words
4171                                    ; Maximum stack depth per invocation:  3 words
4176
4177
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION

;          2906  routine XOR_LNG_WRD (NIB_NUM, NIB_PAT, RESULT) : novalue =
;          2907      begin
;          2908
;          2909  !++
;          2910  ! FUNCTIONAL DESCRIPTION:
;          2911  !       EXCLUSIVE ORS THE CURRENT
;          2912  !       NIBBLE POSITION IN 'NIB_SAVE'
;          2913  !       WITH THE CURRENT TEST PATTERN
;          2914  !       AND ASSIGNS THE RESULTS TO THE
;          2915  !       FORMAL PARAMETER 'RESULT'.
;          2916  !
;          2917  ! FORMAL PARAMETERS:
;          2918  !       NIB_NUM
;          2919  !       CASE SELECT EXPRESSION TO
;          2920  !       SELECT THE CURRENT NIBBLE TO BE
;          2921  !        EXAMINED
;          2922  !
;          2923  !       NIB_PAT
;          2924  !       CURRENT NIBBLE PATTERN TO BE
;          2925  !       XOR'ED
;          2926  !
;          2927  !       RESULT
;          2928  !       CONTAINS THE ADDRESS (PASSED BY REF)
;          2929  !       OF AN OWN STORAGE LOCATION TO
;          2930  !       ENABLE THE CALLER TO EXAMINE THE XOR RESULTS.
;          2931  !
;          2932  ! IMPLICIT INPUTS:
;          2933  !       NIB_SAVE
;          2934  !       BLOCK OF 3 WORDS TO STORE
;          2935  !       THE DATA FOUND IN MLD1
;          2936  !       MLD2 AND MLE2 AFTER A
;          2937  !       DIAGNOSTIC MODE READ.
;          2938  !
;          2939  ! IMPLICIT OUTPUTS:    NONE
;          2940  !--
;          2941
;          2942      case .(NIB_NUM) from 0 to 9 of              !SELECT THE NIBBLE TO BE XOR'ED
;          2943          set
;          2944
;          2945          [0] :
;          2946              .RESULT = .NIB_SAVE [NIB_0] xor .NIB_PAT;
;          2947                                          !XOR NIBBLE 0 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2948
;          2949          [1] :
;          2950              .RESULT = .NIB_SAVE [NIB_1] xor .NIB_PAT;
;          2951                                          !XOR NIBBLE 1 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2952
;          2953          [2] :
;          2954              .RESULT = .NIB_SAVE [NIB_2] xor .NIB_PAT;
;          2955                                          !XOR NIBBLE 2 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2956
;          2957          [3] :
```

```
;ML4AD
;
;                ROUTINE DECLARATION SECTION
;
;          2958              .RESULT = .NIB_SAVE [NIB_3] xor .NIB_PAT;
;          2959                                        !XOR NIBBLE 3 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2960
;          2961          [4] :
;          2962              .RESULT = .NIB_SAVE [NIB_4] xor .NIB_PAT;
;          2963                                        !XOR NIBBLE 4 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2964
;          2965          [5] :
;          2966              .RESULT = .NIB_SAVE [NIB_5] xor .NIB_PAT;
;          2967                                        !XOR NIBBLE 5 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2968
;          2969          [6] :
;          2970              .RESULT = .NIB_SAVE [NIB_6] xor .NIB_PAT;
;          2971                                        !XOR NIBBLE 6 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2972
;          2973          [7] :
;          2974              .RESULT = .NIB_SAVE [NIB_7] xor .NIB_PAT;
;          2975                                        !XOR NIBBLE 7 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2976
;          2977          [8] :
;          2978              .RESULT = .NIB_SAVE [NIB_8] xor .NIB_PAT;
;          2979                                        !XOR NIBBLE 8 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2980
;          2981          [9] :
;          2982              .RESULT = .NIB_SAVE [NIB_9] xor .NIB_PAT
;          2983                                        !XOR NIBBLE 9 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2984          tes;
;          2985
;          2986      end;
;
;
                        .SBTTL  XOR.LNG.WRD ROUTINE DECLARATION SECTION
000000 021072           XOR.LNG.WRD:
000000 021072 004167 163474     JSR     R1,$SAVE4                       ;
000004 021076 016601 000016     MOV     16(SP),R1                       ; NIB.PAT,*     2906
000010 021102 016602 000014     MOV     14(SP),R2                       ; RESULT,*      2946
000014 021106 016600 000020     MOV     20(SP),R0                       ; NIB.NUM,*     2942
000020 021112 006300            ASL     R0
000022 021114 066007 021120     ADD     1$(R0),PC
000026 021120 000024    1$:     .WORD   2$-1$
000030 021122 000036            .WORD   3$-1$
000032 021124 000062            .WORD   4$-1$
000034 021126 000070            .WORD   5$-1$
000036 021130 000076            .WORD   6$-1$
000040 021132 000120            .WORD   8$-1$
000042 021134 000136            .WORD   9$-1$
000044 021136 000144            .WORD   10$-1$
```

```
                                ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                ;                       ROUTINE DECLARATION SECTION             29-Mar-1982 16:21:03    PA:<
                                        .WORD   12$-1$
                                        .WORD   15$-1$
021140  000162
021142  000176
021144  016712  172224        2$:     MOV     NIB.SAVE,(R2)           ;                       2946
021150  042712  177760                BIC     #177760,(R2)
021154  000472                        BR      17$
021156  016703  172212        3$:     MOV     NIB.SAVE,R3             ;                       2950
021162  006203                        ASR     R3
021164  006203                        ASR     R3
021166  006203                        ASR     R3
021170  006203                        ASR     R3
021172  042703  177760                BIC     #177760,R3
021176  010312                        MOV     R3,(R2)
021200  000412                        BR      7$
021202  016704  172166        4$:     MOV     NIB.SAVE,R4             ;                       2954
021206  000437                        BR      13$
021210  016704  172160        5$:     MOV     NIB.SAVE,R4             ;                       2958
021214  000425                        BR      11$
021216  016712  172154        6$:     MOV     NIB.SAVE+2,(R2)         ;                       2962
021222  042712  177760                BIC     #177760,(R2)
021226  010104        7$:     MOV     R1,R4
021230  041204                        BIC     (R2),R4
021232  040112                        BIC     R1,(R2)
021234  050412                        BIS     R4,(R2)
021236  000207                        RTS     PC
021240  016704  172132        8$:     MOV     NIB.SAVE+2,R4           ;                       2942
021244  006204                        ASR     R4                      ;                       2966
021246  006204                        ASR     R4
021250  006204                        ASR     R4
021252  006204                        ASR     R4
021254  000415                        BR      14$
021256  016704  172114        9$:     MOV     NIB.SAVE+2,R4           ;                       2970
021262  000411                        BR      13$
021264  016704  172106        10$:    MOV     NIB.SAVE+2,R4           ;                       2974
021270  006204        11$:    ASR     R4
021272  006204                        ASR     R4
021274  006204                        ASR     R4
021276  006204                        ASR     R4
021300  000402                        BR      13$
021302  016704  172072        12$:    MOV     NIB.SAVE+4,R4           ;                       2978
021306  000304        13$:    SWAB    R4
021310  042704  177760        14$:    BIC     #177760,R4
021314  000411                        BR      16$
021316  016704  172056        15$:    MOV     NIB.SAVE+4,R4           ;                       2982
021322  006204                        ASR     R4
021324  006204                        ASR     R4
021326  006204                        ASR     R4
021330  006204                        ASR     R4
021332  000304                        SWAB    R4
021334  042704  177770                BIC     #177770,R4
021340  010412        16$:    MOV     R4,(R2)
021342  010103        17$:    MOV     R1,R3
021344  041203                        BIC     (R2),R3
```

SEQ 0114

```
                                ;ML4AD
                                ;               ROUTINE DECLARATION SECTION
4350 021346  040112                     BIC     R1,(R2)
4351 021350  050312                     BIS     R3,(R2)
4352 021352  000207                     RTS     PC                              ;
4353
4354                            ; Routine Size:  89 words
4355                            ; Maximum stack depth per invocation:  5 words
4360
4361
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (17)

;        2987  routine LD_LNG_WRD (NIB_NUM, NIB_PAT) : novalue =
;        2988      begin
;        2989
;        2990  !++
;        2991  ! FUNCTIONAL DESCRIPTION:
;        2992  !       LOADS 'NIB_SAVE' WITH UNIQUE
;        2993  !       NIBBLE PATTERNS PRIOR TO WRITING
;        2994  !       TO MLD1, MLD2 AND MLE2
;        2995  !       DATA DIAGNOSTIC REGISTERS.
;        2996  !
;        2997  ! FORMAL PARAMETERS:
;        2998  !       NIB_NUM
;        2999  !       CASE SELECT EXPRESSION TO SELECT
;        3000  !       THE CURRENT NIBBLE TO BE LOADED
;        3001  !
;        3002  !       NIB_PAT
;        3003  !       CURRENT NIBBLE PATTERN TO BE
;        3004  !       LOADED
;        3005  !
;        3006  ! IMPLICIT INPUTS:
;        3007  !       NIB_SAVE
;        3008  !       BLOCK OF 3 WORDS TO STORE
;        3009  !       THE DATA TO BE WRITTEN
;        3010  !       INTO MLD1 MLD2 MLE2
;        3011  !
;        3012  ! IMPLICIT OUTPUTS:
;        3013  !       NIB_SAVE IS LOADED WIH
;        3014  !       THE CURRENT NIBBLE PATTERN
;        3015  !--
;        3016
;        3017      case .(NIB_NUM) from 0 to 9 of          !SELECT THE NIBBLE LOCATION IN NIB_SAVE TO BE LOADED
;        3018          set
;        3019
;        3020          [0] :
;        3021              NIB_SAVE [NIB_0] = .NIB_PAT;    !LOAD NIBBLE 0 WITH NIB_PAT
;        3022
;        3023          [1] :
;        3024              NIB_SAVE [NIB_1] = .NIB_PAT;    !LOAD NIBBLE 1 WITH NIB_PAT
;        3025
;        3026          [2] :
;        3027              NIB_SAVE [NIB_2] = .NIB_PAT;    !LOAD NIBBLE 2 WITH NIB_PAT
;        3028
;        3029          [3] :
;        3030              NIB_SAVE [NIB_3] = .NIB_PAT;    !LOAD NIBBLE 3 WITH NIB_PAT
;        3031
;        3032          [4] :
;        3033              NIB_SAVE [NIB_4] = .NIB_PAT;    !LOAD NIBBLE 4 WITH NIB_PAT
;        3034
;        3035          [5] :
;        3036              NIB_SAVE [NIB_5] = .NIB_PAT;    !LOAD NIBBLE 5 WITH NIB_PAT
;        3037
;        3038          [6] :
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (17)
;          3039                  NIB_SAVE [NIB_6] = .NIB_PAT;        !LOAD NIBBLE 6 WITH NIB_PAT
;          3040
;          3041              [7] :
;          3042                  NIB_SAVE [NIB_7] = .NIB_PAT;        !LOAD NIBBLE 7 WITH NIB_PAT
;          3043
;          3044              [8] :
;          3045                  NIB_SAVE [NIB_8] = .NIB_PAT;        !LOAD NIBBLE 8 WITH NIB_PAT
;          3046
;          3047              [9] :
;          3048                  NIB_SAVE [NIB_9] = .NIB_PAT         !LOAD NIBBLE 9 WITH NIB_PAT
;          3049              tes;
;          3050
;          3051          end;

                                        .SBTTL  LD.LNG.WRD ROUTINE DECLARATION SECTION
021354                          LD.LNG.WRD:
021354  010146                          MOV     R1,-(SP)                ;
021356  016600  000004                  MOV     4(SP),R0                ; NIB.PAT,*                     2987
021362  016601  000006                  MOV     6(SP),R1                ; NIB.NUM,*                     3017
021366  006301                          ASL     R1
021370  066107  021374                  ADD     1$(R1),PC
021374  000024                  1$:     .WORD   2$-1$
021376  000040                          .WORD   3$-1$
021400  000064                          .WORD   4$-1$
021402  000102                          .WORD   5$-1$
021404  000134                          .WORD   7$-1$
021406  000150                          .WORD   8$-1$
021410  000174                          .WORD   9$-1$
021412  000212                          .WORD   10$-1$
021414  000244                          .WORD   12$-1$
021416  000262                          .WORD   13$-1$
021420  042700  177760          2$:     BIC     #177760,R0              ;                               3021
021424  142767  000017  171742          BICB    #17,NIB.SAVE
021432  000433                          BR      6$
021434  006300                  3$:     ASL     R0                      ;                               3024
021436  006300                          ASL     R0
021440  006300                          ASL     R0
021442  006300                          ASL     R0
021444  042700  177417                  BIC     #177417,R0
021450  142767  000360  171716          BICB    #360,NIB.SAVE
021456  000421                          BR      6$
021460  000300                  4$:     SWAB    R0                      ;                               3027
021462  042700  170377                  BIC     #170377,R0
021466  042767  007400  171700          BIC     #7400,NIB.SAVE
021474  000412                          BR      6$
021476  000300                  5$:     SWAB    R0                      ;                               3030
```

```
                                        ;ML4AD
                                        ;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS
                                                                                                        29-Mar-1982 16:21:03    PA:<
021500  006300                                    ASL     R0
021502  006300                                    ASL     R0
021504  006300                                    ASL     R0
021506  006300                                    ASL     R0
021510  042700  007777                            BIC     #7777,R0
021514  042767  170000  171652                    BIC     #170000,NIB.SAVE
021522  050067  171646                6$:         BIS     R0,NIB.SAVE
021526  000467                                    BR      15$
021530  042700  177760                7$:         BIC     #177760,R0                      ;               3017
021534  142767  000017  171634                    BICB    #17,NIB.SAVE+2                  ;               3033
021542  000433                                    BR      11$
021544  006300                        8$:         ASL     R0                              ;               3036
021546  006300                                    ASL     R0
021550  006300                                    ASL     R0
021552  006300                                    ASL     R0
021554  042700  177417                            BIC     #177417,R0
021560  142767  000360  171610                    BICB    #360,NIB.SAVE+2
021566  000421                                    BR      11$
021570  000300                        9$:         SWAB    R0                              ;               3039
021572  042700  170377                            BIC     #170377,R0
021576  042767  007400  171572                    BIC     #7400,NIB.SAVE+2
021604  000412                                    BR      11$
021606  000300                        10$:        SWAB    R0                              ;               3042
021610  006300                                    ASL     R0
021612  006300                                    ASL     R0
021614  006300                                    ASL     R0
021616  006300                                    ASL     R0
021620  042700  007777                            BIC     #7777,R0
021624  042767  170000  171544                    BIC     #170000,NIB.SAVE+2
021632  050067  171540                11$:        BIS     R0,NIB.SAVE+2
021636  000423                                    BR      15$
021640  000300                        12$:        SWAB    R0                              ;               3017
021642  042700  170377                            BIC     #170377,R0                      ;               3045
021646  042767  007400  171524                    BIC     #7400,NIB.SAVE+4
021654  000412                                    BR      14$
021656  000300                        13$:        SWAB    R0                              ;               3048
021660  006300                                    ASL     R0
021662  006300                                    ASL     R0
021664  006300                                    ASL     R0
021666  006300                                    ASL     R0
021670  042700  107777                            BIC     #107777,R0
021674  042767  070000  171476                    BIC     #70000,NIB.SAVE+4
021702  050067  171472                14$:        BIS     R0,NIB.SAVE+4
021706  012601                        15$:        MOV     (SP)+,R1                        ;               2987
021710  000207                                    RTS     PC

                                        ; Routine Size:  111 words
                                        ; Maximum stack depth per invocation:  2 words
```

```
 4531 ;ML4AD
 4532 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 4533 ;                                                                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (18)
 4534 ;          3052    routine WRT_CS1 (TST_PAT, index) : novalue =
 4535 ;          3053        begin
 4536 ;          3054
 4537 ;          3055    !++
 4538 ;          3056    ! FUNCTIONAL DESCRIPTION:
 4539 ;          3057    !
 4540 ;          3058    !       LOADS THE CONTROL & STATUS REGISTER 1 WITH A DATA PATTERN
 4541 ;          3059    !       GENERATED BY THE MACRO WRT_MASK.
 4542 ;          3060    !
 4543 ;          3061    ! FORMAL PARAMETERS:
 4544 ;          3062    !       TST_PAT
 4545 ;          3063    !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
 4546 ;          3064    !
 4547 ;          3065    !       INDEX
 4548 ;          3066    !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
 4549 ;          3067    !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
 4550 ;          3068    !
 4551 ;          3069    !--
 4552 ;          3070
 4553 ;          3071        MLCS1 = WRT_MASK;                          !LOAD GENERATED WRT_MASK PATTERN INTO MLCS1
 4554 ;          3072        end;
 4558
 4559                                    .SBTTL  WRT.CS1 ROUTINE DECLARATION SECTION
 4563 021712  004167  162622        WRT.CS1:JSR     R1,$SAVE2                       ;                       3052
 4564 021716  016600  000010                MOV     10(SP),R0                       ; INDEX,*               3071
 4565 021722  006300                        ASL     R0
 4566 021724  006300                        ASL     R0
 4567 021726  006300                        ASL     R0
 4568 021730  010001                        MOV     R0,R1
 4569 021732  016100  015406                MOV     ML.REG+2(R1),R0
 4570 021736  056600  000012                BIS     12(SP),R0                       ; TST.PAT,*
 4571 021742  046100  015410                BIC     ML.REG+4(R1),R0
 4572 021746  016102  015412                MOV     ML.REG+6(R1),R2
 4573 021752  050002                        BIS     R0,R2
 4574 021754  010277  173424                MOV     R2,@ML.REG
 4575 021760  000207                        RTS     PC                              ;                       3052
 4576
 4577                           ; Routine Size:  20 words
 4578                           ; Maximum stack depth per invocation:  3 words
 4583
 4584
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (19)

;        3073  routine RD_CS1 (TST_PAT, index, ERR_FLG) : novalue =
;        3074      begin
;        3075
;        3076  !++
;        3077  ! FUNCTIONAL DESCRIPTION:
;        3078  !
;        3079  !       COMPARES THE CONTENTS OF THE
;        3080  !       CONTROL & STATUS REGISTER 1
;        3081  !       WITH THE MASKED DATA PATTERN
;        3082  !       GENERATED BY THE MACRO 'WRT_MASK'
;        3083  !
;        3084  !       IF THE COMPARE IS NOT EQUAL THEN
;        3085  !       THE FORMAL PARAMETER 'ERR_FLG' IS
;        3086  !       ASSIGNED A ONE TO INDICATE THE
;        3087  !       ERROR.
;        3088  !
;        3089  ! FORMAL PARAMETERS:
;        3090  !
;        3091  !       TST_PAT
;        3092  !       DATA PATTERN TO BE MASKED AND
;        3093  !       COMPARED AGAINST THE CONTENTS
;        3094  !       OF THE REGISTER UNDER TEST.
;        3095  !
;        3096  !       INDEX
;        3097  !        USED BY THE MACRO WRT_MASK TO
;        3098  !       SELECT THE CURRENT REGISTERS ADDRESS,
;        3099  !       FORCED HI, FORCED LO AND DON'T CARE
;        3100  !       MASK INFORMATION.
;        3101  !
;        3102  !       ERR_FLG
;        3103  !       CONTAINS THE ADDRESS (PASSED BY REF)
;        3104  !       OF THE CALLERS ERROR FLG TO ENABLE THE
;        3105  !       CALLER TO EXAMINE THE ERROR STATUS
;        3106  !       OF THE ROUTINE CALL.
;        3107  !
;        3108  ! IMPLICIT INPUTS:
;        3109  !       WT_DATA
;        3110  !         GETS LOADED WITH THE GENERATED
;        3111  !         WRT_MASK DATA PATTERN THUS ALLOWING
;        3112  !         CALLER TO PRINT FAILING GOOD DATA.
;        3113  !
;        3114  !       RD_DATA
;        3115  !         GETS LOADED WITH DATA READ FROM THE
;        3116  !         REGISTER THUS ALLOWING CALLER
;        3117  !         TO PRINT FAILING BAD DATA.
;        3118  !
;        3119  ! IMPLICIT OUTPUTS:
;        3120  !       GLOBAL LOCATION WR_DATA
;        3121  !       AND RD_DATA LOADED WITH GOOD
;        3122  !       AND BAD REGISTER DATA
;        3123  !--
;        3124
```

```
;ML4AD
;                       ROUTINE DECLARATION SECTION                     29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (19)
;
;               3125        .ERR_FLG = ZERO;                            !CLEAR ERROR FLAG
;               3126        WT_DATA = WRT_MASK;                         !SAVE THE DATA WRITTEN TO THE REGISTER
;               3127        RD_DATA = .ML[CS1 or .IGNORE;               !READ AND SAVE THE REGISTER
;               3128
;               3129        if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
;               3130
;               3131                                                    !SET ERROR FLAG IF NEQ
;               3132        end;
;
;                                       .SBTTL  RD.CS1 ROUTINE DECLARATION SECTION
;
4661 021762  004167  162552     RD.CS1: JSR     R1,$SAVE2                       ;                       3073
4662 021766  005076  000010             CLR     @10(SP)                         ; ERR.FLG               3125
4663 021772  016600  000012             MOV     12(SP),R0                       ; INDEX,*               3126
4664 021776  006300                     ASL     R0
4665 022000  006300                     ASL     R0
4666 022002  006300                     ASL     R0
4667 022004  010001                     MOV     R0,R1
4668 022006  016100  015406             MOV     ML.REG+2(R1),R0
4669 022012  056600  000014             BIS     14(SP),R0                       ; TST.PAT,*
4670 022016  046100  015410             BIC     ML.REG+4(R1),R0
4671 022022  016102  015412             MOV     ML.REG+6(R1),R2
4672 022026  050002                     BIS     R0,R2
4673 022030  010267  173314             MOV     R2,WT.DATA
4674 022034  017702  173344             MOV     @ML.REG,R2                      ;                       3127
4675 022040  056102  015412             BIS     ML.REG+6(R1),R2
4676 022044  010267  173302             MOV     R2,RD.DATA
4677 022050  026767  173274  173274     CMP     WT.DATA,RD.DATA                 ;                       3129
4678 022056  001403                     BEQ     1$
4679 022060  012776  000001  000010     MOV     #1,@10(SP)                      ; *.ERR.FLG
4680 022066  000207             1$:     RTS     PC                              ;                       3073
4681
4682                                    ; Routine Size:  35 words
4683                                    ; Maximum stack depth per invocation:  3 words
4688
4689
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (20)
;          3133  routine WRT_ER (TST_PAT, index) : novalue =
;          3134      begin
;          3135
;          3136  !++
;          3137  ! FUNCTIONAL DESCRIPTION:
;          3138  !
;          3139  !       LOADS THE ERROR REGISTER WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
;          3140  !
;          3141  ! FORMAL PARAMETERS:
;          3142  !
;          3143  !       TST_PAT
;          3144  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
;          3145  !
;          3146  !       INDEX
;          3147  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
;          3148  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
;          3149  !--
;          3150
;          3151      MLER = WRT_MASK;                                !LOAD GENERATE WRT_MASK PATTERN INTO MLER
;          3152      end;

                                        .SBTTL  WRT.ER ROUTINE DECLARATION SECTION
022070  004167  162444          WRT.ER: JSR     R1,$SAVE2                       ;
022074  016600  000010                  MOV     10(SP),R0                       ; INDEX,*                       3133
022100  006300                          ASL     R0                                                              3151
022102  006300                          ASL     R0
022104  006300                          ASL     R0
022106  010001                          MOV     R0,R1
022110  016100  015406                  MOV     ML.REG+2(R1),R0
022114  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
022120  046100  015410                  BIC     ML.REG+4(R1),R0
022124  016102  015412                  MOV     ML.REG+6(R1),R2
022130  050002                          BIS     R0,R2
022132  010277  173326                  MOV     R2,@ML.REG+60
022136  000207                          RTS     PC                              ;                               3133

                        ; Routine Size:  20 words
                        ; Maximum stack depth per invocation:  3 words
```

```
4745 ;ML4AD
4746 :                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
4747                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (21)
4748 ;          3153   routine RD_ER (TST_PAT, index, ERR_FLG) : novalue =
4749 ;          3154       begin
4750 ;          3155
4751 ;          3156   !++
4752 ;          3157   ! FUNCTIONAL DESCRIPTION:
4753 ;          3158   !
4754 ;          3159   !       COMPARES THE CONTENTS OF THE
4755 ;          3160   !       ERROR REGISTER WITH
4756 ;          3161   !       THE MASKED DATA PATTERN
4757 ;          3162   !       GENERATED BY THE MACRO 'WRT_MASK'.
4758 ;          3163   !
4759 ;          3164   !       IF THE COMPARE IS NOT EQUAL THEN
4760 ;          3165   !       THE FORMAL PARAMETER 'ERR_FLG' IS
4761 ;          3166   !       ASSIGNED A ONE TO INDICATE THE
4762 ;          3167   !       ERROR.
4763 ;          3168   !
4764 ;          3169   ! FORMAL PARAMETERS:
4765 ;          3170   !
4766 ;          3171   !       TST_PAT
4767 ;          3172   !       DATA PATTERN TO BE MASKED AND
4768 ;          3173   !       COMPARED AGAINST THE CONTENTS
4769 ;          3174   !       OF THE REGISTER UNDER TEST.
4770 ;          3175   !
4771 ;          3176   !       INDEX
4772 ;          3177   !       USED BY THE MACRO WRT_MASK TO
4773 ;          3178   !       SELECT THE CURRENT REGISTERS ADDRESS,
4774 ;          3179   !       FORCED HI, FORCED LO AND DON'T CARE
4775 ;          3180   !       MASK INFORMATION.
4776 ;          3181   !
4777 ;          3182   !       ERR_FLG
4778 ;          3183   !       CONTAINS THE ADDRESS (PASSED BY REF)
4779 ;          3184   !       OF THE CALLERS ERROR FLG TO ENABLE THE
4780 ;          3185   !       CALLER TO EXAMINE THE ERROR STATUS
4781 ;          3186   !       OF THE ROUTINE CALL.
4782 ;          3187   !
4783 ;          3188   ! IMPLICIT INPUTS:
4784 ;          3189   !       WT_DATA
4785 ;          3190   !         GETS LOADED WITH THE GENERATED
4786 ;          3191   !         WRT_MASK DATA PATTERN THUS ALLOWING
4787 ;          3192   !         CALLER TO PRINT FAILING GOOD DATA.
4788 ;          3193   !
4789 ;          3194   !       RD_DATA
4790 ;          3195   !         GETS LOADED WITH DATA READ FROM THE
4791 ;          3196   !         REGISTER THUS ALLOWING CALLER
4792 ;          3197   !         TO PRINT FAILING BAD DATA.
4793 ;          3198   !
4794 ;          3199   ! IMPLICIT OUTPUTS:
4795 ;          3200   !       GLOBAL LOCATION WR_DATA
4796 ;          3201   !       AND RD_DATA LOADED WITH GOOD
4797 ;          3202   !       AND BAD REGISTER DATA
4798 ;          3203   !
4799 ;          3204   !--
```

```
4801  :ML4AD
4802  :                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
4803                                                                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (21)
4804  :           3205
4805  :           3206        .ERR_FLG = ZERO;                           !CLEAR THE ERROR FLAG
4806  :           3207        WT_DATA = WRT_MASK;                        !SAVE THE DATA WRITTEN TO THE REGISTER
4807  :           3208        RD_DATA = .MLER or .IGNORE;                !READ AND SAVE THE REGISTER
4808  :           3209
4809  :           3210        if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
4810  :           3211
4811  :           3212                                                   !SET ERROR FLAG IF NEQ
4812  :           3213        end;
4816
4817                                        .SBTTL  RD.ER ROUTINE DECLARATION SECTION
4821  022140  004167  162374          RD.ER: JSR     R1,$SAVE2                    ;                                    3153
4822  022144  005076  000010                 CLR     @10(SP)                      ; ERR.FLG                            3206
4823  022150  016600  000012                 MOV     12(SP),R0                    ; INDEX,*                            3207
4824  022154  006300                         ASL     R0
4825  022156  006300                         ASL     R0
4826  022160  006300                         ASL     R0
4827  022162  010001                         MOV     R0,R1
4828  022164  016100  015406                 MOV     ML.REG+2(R1),R0
4829  022170  056600  000014                 BIS     14(SP),R0                    ; TST.PAT,*
4830  022174  046100  015410                 BIC     ML.REG+4(R1),R0
4831  022200  016102  015412                 MOV     ML.REG+6(R1),R2
4832  022204  050002                         BIS     R0,R2
4833  022206  010267  173136                 MOV     R2,WT.DATA
4834  022212  017702  173246                 MOV     @ML.REG+60,R2                ;                                    3208
4835  022216  056102  015412                 BIS     ML.REG+6(R1),R2
4836  022222  010267  173124                 MOV     R2,RD.DATA
4837  022226  026767  173116  173116         CMP     WT.DATA,RD.DATA              ;                                    3210
4838  022234  001403                         BEQ     1$
4839  022236  012776  000001  000010         MOV     #1,@10(SP)                   ; *,ERR.FLG
4840  022244  000207                 1$:     RTS     PC                           ;                                    3153
4841
4842                                        ; Routine Size:  35 words
4843                                        ; Maximum stack depth per invocation:  3 words
4848
4849
```

```
4851 :ML4AD
4852 :                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
4853                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (22)
4854 :          3214    routine WRT_DA (TST_PAT, index) : novalue =
4855 :          3215        begin
4856 :          3216
4857 :          3217    !++
4858 :          3218    ! FUNCTIONAL DESCRIPTION:
4859 :          3219    !
4860 :          3220    !       LOADS THE DESIRED SECTOR WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
4861 :          3221    !
4862 :          3222    ! FORMAL PARAMETERS:
4863 :          3223    !
4864 :          3224    !       TST_PAT
4865 :          3225    !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
4866 :          3226    !
4867 :          3227    !       INDEX
4868 :          3228    !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
4869 :          3229    !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
4870 :          3230    !--
4871 :          3231
4872 :          3232        MLDA = WRT_MASK;                               !LOAD MLDA WITH GENERATED WRT_MASK PATTERN
4873 :          3233        end;
4877
4878                                        .SBTTL  WRT.DA ROUTINE DECLARATION SECTION
4882 022246  004167  162266          WRT.DA: JSR     R1,$SAVE2                       ;
4883 022252  016600  000010                  MOV     10(SP),R0                       ; INDEX,*          3214
4884 022256  006300                          ASL     R0                                                 3232
4885 022260  006300                          ASL     R0
4886 022262  006300                          ASL     R0
4887 022264  010001                          MOV     R0,R1
4888 022266  016100  015406                  MOV     ML.REG+2(R1),R0
4889 022272  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
4890 022276  046100  015410                  BIC     ML.REG+4(R1),R0
4891 022302  016102  015412                  MOV     ML.REG+6(R1),R2
4892 022306  050002                          BIS     R0,R2
4893 022310  010277  173120                  MOV     R2,@ML.REG+30
4894 022314  000207                          RTS     PC                              ;                  3214
4895
4896                                ; Routine Size:  20 words
4897                                ; Maximum stack depth per invocation:  3 words
4902
4903
```

```
                3234  routine RD_DA (TST_PAT, index, ERR_FLG) : novalue =
                3235      begin
                3236
                3237  !++
                3238  ! FUNCTIONAL DESCRIPTION:
                3239  !
                3240  !       COMPARES THE CONTENTS OF THE
                3241  !       DESIRED SECTOR ADDRESS REGISTER
                3242  !       WITH THE MASKED DATA PATTERN
                3243  !       GENERATED BY THE MACRO 'WRT_MASK'
                3244  !
                3245  !       IF THE COMPARE IS NOT EQUAL THEN
                3246  !       THE FORMAL PARAMETER 'ERR_FLG' IS
                3247  !       ASSIGNED A ONE TO INDICATE THE
                3248  !       ERROR
                3249  !
                3250  ! FORMAL PARAMETERS:
                3251  !
                3252  !       TST_PAT
                3253  !       DATA PATTERN TO BE MASKED AND
                3254  !       COMPARED AGAINST THE CONTENTS
                3255  !       OF THE REGISTER UNDER TEST
                3256  !
                3257  !       INDEX
                3258  !       USED BY THE MACRO WRT_MASK TO
                3259  !       SELECT THE CURRENT REGISTERS ADDRESS,
                3260  !       FORCED HI, FORCED LO AND DON'T CARE
                3261  !       MASK INFORMATION
                3262  !
                3263  !       ERR_FLG
                3264  !       CONTAINS THE ADDRESS (PASSED BY REF)
                3265  !       OF THE CALLERS ERROR FLG TO ENABLE THE
                3266  !       CALLER TO EXAMINE THE ERROR STATUS
                3267  !       OF THE ROUTINE CALL.
                3268  ! IMPLICIT INPUTS:
                3269  !       WT_DATA
                3270  !         GETS LOADED WITH THE GENERATED
                3271  !         WRT_MASK DATA PATTERN THUS ALLOWING
                3272  !         CALLER TO PRINT FAILING GOOD DATA.
                3273  !
                3274  !       RD_DATA
                3275  !         GETS LOADED WITH DATA READ FROM THE
                3276  !         REGISTER THUS ALLOWING CALLER
                3277  !         TO PRINT FAILING BAD DATA.
                3278  !
                3279  ! IMPLICIT OUTPUTS:
                3280  !       GLOBAL LOCATION WR_DATA
                3281  !       AND RD_DATA LOADED WITH GOOD
                3282  !       AND BAD REGISTER DATA
                3283  !
                3284  !--
                3285
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (23)
;          3286      .ERR_FLG = ZERO;                               !CLEAR THE ERROR FLAG
;          3287      WT_DATA = WRT_MASK;                            !SAVE THE DATA WRITTEN TO THE REGISTER
;          3288      RD_DATA = .MLDA _r .IGNORE;                    !READ AND SAVE THE REGISTER
;          3289
;          3290      if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;      !READ THE REG FOR WRT_MASK
;          3291
;          3292                                                     !SET ERROR FLAG IF NEQ
;          3293      end;
```

```
                                .SBTTL  RD.DA ROUTINE DECLARATION SECTION
022316  004167  162216      RD.DA:  JSR     R1,$SAVE2                       ;
022322  005076  000010              CLR     @10(SP)                         ; ERR.FLG                       3234
022326  016600  000012              MOV     12(SP),R0                       ; INDEX,*                       3286
022332  006300                      ASL     R0                                                              3287
022334  006300                      ASL     R0
022336  006300                      ASL     R0
022340  010001                      MOV     R0,R1
022342  016100  015406              MOV     ML.REG+2(R1),R0
022346  056600  000014              BIS     14(SP),R0                       ; TST.PAT,*
022352  046100  015410              BIC     ML.REG+4(R1),R0
022356  016102  015412              MOV     ML.REG+6(R1),R2
022362  050002                      BIS     R0,R2
022364  010267  172760              MOV     R2,WT.DATA
022370  017702  173040              MOV     @ML.REG+30,R2                   ;                               3288
022374  056102  015412              BIS     ML.REG+6(R1),R2
022400  010267  172746              MOV     R2,RD.DATA
022404  026767  172740  172740      CMP     WT.DATA,RD.DATA                 ;                               3290
022412  001403                      BEQ     1$
022414  012776  000001  000010      MOV     #1,@10(SP)                      ; *,ERR.FLG
022422  000207              1$:     RTS     PC                              ;                               3234

                        ; Routine Size:  35 words
                        ; Maximum stack depth per invocation:  3 words
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (24)

;          3294  routine WRT_MR (TST_PAT, index) : novalue =
;          3295      begin
;          3296
;          3297  !++
;          3298  ! FUNCTIONAL DESCRIPTION:
;          3299  !
;          3300  !       LOADS THE MAINTENANCE REGISTER WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
;          3301  !
;          3302  ! FORMAL PARAMETERS:
;          3303  !
;          3304  !       TST_PAT
;          3305  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
;          3306  !
;          3307  !       INDEX
;          3308  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
;          3309  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
;          3310  !--
;          3311
;          3312      MLMR = WRT_MASK;                          !LOAD MLMR WITH GENERATED WRT_MASK PATTERN
;          3313      end;

                                    .SBTTL  WRT.MR ROUTINE DECLARATION SECTION
5041 022424  004167  162110  WRT.MR: JSR     R1,$SAVE2                       ;
5042 022430  016600  000010          MOV     10(SP),R0                       ; INDEX,*          3312
5043 022434  006300                  ASL     R0
5044 022436  006300                  ASL     R0
5045 022440  006300                  ASL     R0
5046 022442  010001                  MOV     R0,R1
5047 022444  016100  015406          MOV     ML.REG+2(R1),R0
5048 022450  056600  000012          BIS     12(SP),R0                       ; TST.PAT,*
5049 022454  046100  015410          BIC     ML.REG+4(R1),R0
5050 022460  016102  015412          MOV     ML.REG+6(R1),R2
5051 022464  050002                  BIS     R0,R2
5052 022466  010277  173032          MOV     R2,@ML.REG+120
5053 022472  000207                  RTS     PC                              ;          3294
5054
5055                        ; Routine Size:  20 words
5056                        ; Maximum stack depth per invocation:  3 words
5061
5062
```

```
5064 ;ML4AD
5065 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
5066 ;                                                                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (25)
5067 ;          3314    routine RD_MR (TST_PAT, index, ERR_FLG) : novalue =
5068 ;          3315        begin
5069 ;          3316
5070 ;          3317    !++
5071 ;          3318    ! FUNCTIONAL DESCRIPTION:
5072 ;          3319    !
5073 ;          3320    !       COMPARES THE CONTENTS OF THE
5074 ;          3321    !       MAINTENANCE REGISTER WITH THE
5075 ;          3322    !       MASKED DATA PATTERN
5076 ;          3323    !       GENERATED BY THE MACRO 'WRT_MASK'.
5077 ;          3324    !
5078 ;          3325    !       IF THE COMPARE IS NOT EQUAL THEN
5079 ;          3326    !       THE FORMAL PARAMETER 'ERR_FLG' IS
5080 ;          3327    !       ASSIGNED A ONE TO INDICATE THE
5081 ;          3328    !       ERROR.
5082 ;          3329    !
5083 ;          3330    ! FORMAL PARAMETERS:
5084 ;          3331    !
5085 ;          3332    !       TST_PAT
5086 ;          3333    !       DATA PATTERN TO BE MASKED AND
5087 ;          3334    !       COMPARED AGAINST THE CONTENTS
5088 ;          3335    !       OF THE REGISTER UNDER TEST.
5089 ;          3336    !
5090 ;          3337    !       INDEX
5091 ;          3338    !       USED BY THE MACRO WRT_MASK TO
5092 ;          3339    !       SELECT THE CURRENT REGISTERS ADDRESS,
5093 ;          3340    !       FORCED HI, FORCED LO AND DON'T CARE
5094 ;          3341    !       MASK INFORMATION.
5095 ;          3342    !
5096 ;          3343    !       ERR_FLG
5097 ;          3344    !       CONTAINS THE ADDRESS (PASSED BY REF)
5098 ;          3345    !       OF THE CALLERS ERROR FLG TO ENABLE THE
5099 ;          3346    !       CALLER TO EXAMINE THE ERROR STATUS
5100 ;          3347    !       OF THE ROUTINE CALL.
5101 ;          3348    !
5102 ;          3349    ! IMPLICIT INPUTS:
5103 ;          3350    !       WT_DATA
5104 ;          3351    !         GETS LOADED WITH THE GENERATED
5105 ;          3352    !         WRT_MASK DATA PATTERN THUS ALLOWING
5106 ;          3353    !         CALLER TO PRINT FAILING GOOD DATA.
5107 ;          3354    !
5108 ;          3355    !       RD_DATA
5109 ;          3356    !         GETS LOADED WITH DATA READ FROM THE
5110 ;          3357    !         REGISTER THUS ALLOWING CALLER
5111 ;          3358    !         TO PRINT FAILING BAD DATA.
5112 ;          3359    !
5113 ;          3360    ! IMPLICIT OUTPUTS:
5114 ;          3361    !       GLOBAL LOCATION WR_DATA
5115 ;          3362    !       AND RD_DATA LOADED WITH GOOD
5116 ;          3363    !       AND BAD REGISTER DATA
5117 ;          3364    !
5118 ;          3365    !--
```

```
5120 ;ML4AD
5121 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
5122                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (25)
5123 ;          3366
5124 ;          3367        .ERR_FLG = ZERO;                           !CLEAR THE ERROR FLAG
5125 ;          3368        WT_DATA = WRT_MASK;                        !SAVE THE DATA WRITTEN TO THE REGISTER
5126 ;          3369        RD_DATA = .MLMR or .IGNORE;                !READ AND SAVE THE REGISTER
5127 ;          3370
5128 ;          3371        if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
5129 ;          3372
5130 ;          3373                                                   !SET ERROR FLAG IF NEQ
5131 ;          3374        end;
5135
5136                                        .SBTTL  RD.MR ROUTINE DECLARATION SECTION
5140 022474  004167  162040          RD.MR: JSR     R1,$SAVE2                    ;                              3314
5141 022500  005076  000010                 CLR     @10(SP)                      ; ERR.FLG                      3367
5142 022504  016600  000012                 MOV     12(SP),R0                    ; INDEX,*                      3368
5143 022510  006300                         ASL     R0
5144 022512  006300                         ASL     R0
5145 022514  006300                         ASL     R0
5146 022516  010001                         MOV     R0,R1
5147 022520  016100  015406                 MOV     ML.REG+2(R1),R0
5148 022524  056600  000014                 BIS     14(SP),R0                    ; TST.PAT,*
5149 022530  046100  015410                 BIC     ML.REG+4(R1),R0
5150 022534  016102  015412                 MOV     ML.REG+6(R1),R2
5151 022540  050002                         BIS     R0,R2
5152 022542  010267  172602                 MOV     R2,WT.DATA
5153 022546  017702  172752                 MOV     @ML.REG+120,R2               ;                              3369
5154 022552  056102  015412                 BIS     ML.REG+6(R1),R2
5155 022556  010267  172570                 MOV     R2,RD.DATA
5156 022562  026767  172562  172562         CMP     WT.DATA,RD.DATA              ;                              3371
5157 022570  001403                         BEQ     1$
5158 022572  012776  000001  000010         MOV     #1,@10(SP)                   ; *,ERR.FLG
5159 022600  000207                 1$:     RTS     PC                           ;                              3314
5160
5161                                        ; Routine Size:  35 words
5162                                        ; Maximum stack depth per invocation:  3 words
5167
5168
```

```
5170 ;ML4AD
5171 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
5172                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (26)
5173 ;          3375   routine WRT_PA (TST_PAT, index) : novalue =
5174 ;          3376       begin
5175 ;          3377
5176 ;          3378   !++
5177 ;          3379   ! FUNCTIONAL DESCRIPTION:
5178 ;          3380   !       LOADS THE PROM ADDRESS REGISTER WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
5179 ;          3381   ! FORMAL PARAMETERS:
5180 ;          3382   !       TST_PAT
5181 ;          3383   !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
5182 ;          3384   !       INDEX
5183 ;          3385   !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
5184 ;          3386   !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
5185 ;          3387   !--
5186 ;          3388
5187 ;          3389       PROM_DIS = ONE;                            !SET PROM DISABLE BIT
5188 ;          3390       MLPA = WRT_MASK;                           !LOAD MLPA WITH GENERATED WRT_MASK PATTERN
5189 ;          3391       PROM_DIS = ZERO;                           !CLEAR PROM DISABLE BIT
5190 ;          3392       end;
5194
5195                                        .SBTTL  WRT.PA ROUTINE DECLARATION SECTION
5199 022602  004167  161732          WRT.PA: JSR     R1,$SAVE2                       ;                          3375
5200 022606  152777  000040  172710          BISB    #40,@ML.REG+120                 ;                          3389
5201 022614  016600  000010                  MOV     10(SP),R0                       ; INDEX,*                  3390
5202 022620  006300                          ASL     R0
5203 022622  006300                          ASL     R0
5204 022624  006300                          ASL     R0
5205 022626  010001                          MOV     R0,R1
5206 022630  016100  015406                  MOV     ML.REG+2(R1),R0
5207 022634  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
5208 022640  046100  015410                  BIC     ML.REG+4(R1),R0
5209 022644  016102  015412                  MOV     ML.REG+6(R1),R2
5210 022650  050002                          BIS     R0,R2
5211 022652  010277  172626                  MOV     R2,@ML.REG+100
5212 022656  142777  000040  172640          BICB    #40,@ML.REG+120                 ;                          3391
5213 022664  000207                          RTS     PC                              ;                          3375
5214
5215                                ; Routine Size:  26 words
5216                                ; Maximum stack depth per invocation:  3 words
5221
5222
```

```
5224 ;ML4AD
5225 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
5226                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (27)
5227 ;          3393   routine RD_PA (TST_PAT, index, ERR_FLG) : novalue =
5228 ;          3394       begin
5229 ;          3395
5230 ;          3396   !++
5231 ;          3397   ! FUNCTIONAL DESCRIPTION:
5232 ;          3398   !
5233 ;          3399   !       COMPARES THE CONTENTS OF THE
5234 ;          3400   !       PROM ADDRESS REGISTER
5235 ;          3401   !       WITH THE MASKED DATA PATTERN
5236 ;          3402   !       GENERATED BY THE MACRO 'WRT_MASK'.
5237 ;          3403   !
5238 ;          3404   !       IF THE COMPARE IS NOT EQUAL THEN
5239 ;          3405   !       THE FORMAL PARAMETER 'ERR_FLG' IS
5240 ;          3406   !       ASSIGNED A ONE TO INDICATE THE
5241 ;          3407   !       ERROR.
5242 ;          3408   !
5243 ;          3409   ! FORMAL PARAMETERS:
5244 ;          3410   !       TST_PAT
5245 ;          3411   !       DATA PATTERN TO BE MASKED AND
5246 ;          3412   !       COMPARED AGAINST THE CONTENTS
5247 ;          3413   !       OF THE REGISTER UNDER TEST.
5248 ;          3414   !
5249 ;          3415   !       INDEX
5250 ;          3416   !       USED BY THE MACRO WRT_MASK TO
5251 ;          3417   !       SELECT THE CURRENT REGISTERS ADDRESS,
5252 ;          3418   !       FORCED HI, FORCED LO AND DON'T CARE
5253 ;          3419   !       MASK INFORMATION.
5254 ;          3420   !
5255 ;          3421   !       ERR_FLG
5256 ;          3422   !       CONTAINS THE ADDRESS (PASSED BY REF)
5257 ;          3423   !       OF THE CALLERS ERROR FLG TO ENABLE THE
5258 ;          3424   !       CALLER TO EXAMINE THE ERROR STATUS
5259 ;          3425   !       OF THE ROUTINE CALL.
5260 ;          3426   !
5261 ;          3427   ! IMPLICIT INPUTS:
5262 ;          3428   !       WT_DATA
5263 ;          3429   !         GETS LOADED WITH THE GENERATED
5264 ;          3430   !         WRT_MASK DATA PATTERN THUS ALLOWING
5265 ;          3431   !         CALLER TO PRINT FAILING GOOD DATA.
5266 ;          3432   !
5267 ;          3433   !       RD_DATA
5268 ;          3434   !         GETS LOADED WITH DATA READ FROM THE
5269 ;          3435   !         REGISTER THUS ALLOWING CALLER
5270 ;          3436   !         TO PRINT FAILING BAD DATA.
5271 ;          3437   !
5272 ;          3438   ! IMPLICIT OUTPUTS:
5273 ;          3439   !       GLOBAL LOCATION WR_DATA
5274 ;          3440   !       AND RD_DATA LOADED WITH GOOD
5275 ;          3441   !       AND BAD REGISTER DATA
5276 ;          3442   !
5277 ;          3443   !--
5278 ;          3444
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (27)
;
;          3445      .ERR_FLG = ZERO;                               !CLEAR THE ERROR FLAG
;          3446      PROM_DIS = ONE;                                !SET THE PROM DISABLE BIT
;          3447      WT_DATA = WRT_MASK;                            !SAVE THE DATA WRITTEN TO THE REGISTER
;          3448      RD_DATA = .MLPA or .IGNORE;                    !READ AND SAVE THE REGISTER
;          3449
;          3450      if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
;          3451
;          3452      PROM_DIS = ZERO;                               !SET ERROR IF NEQ
;          3453      end;                                           !CLEAR THE PROM DISABLE BIT
;
                                    .SBTTL  RD.PA ROUTINE DECLARATION SECTION
022666  004167  161646        RD.PA:  JSR     R1,$SAVE2                       ;                       3393
022672  005076  000010                CLR     @10(SP)                         ; ERR.FLG               3445
022676  152777  000040  172620        BISB    #40,@ML.REG+120                 ;                       3446
022704  016600  000012                MOV     12(SP),R0                       ; INDEX,*               3447
022710  006300                        ASL     R0
022712  006300                        ASL     R0
022714  006300                        ASL     R0
022716  010001                        MOV     R0,R1
022720  016100  015406                MOV     ML.REG+2(R1),R0
022724  056600  000014                BIS     14(SP),R0                       ; TST.PAT,*
022730  046100  015410                BIC     ML.REG+4(R1),R0
022734  016102  015412                MOV     ML.REG+6(R1),R2
022740  050002                        BIS     R0,R2
022742  010267  172402                MOV     R2,WT.DATA
022746  017702  172532                MOV     @ML.REG+100,R2                  ;                       3448
022752  056102  015412                BIS     ML.REG+6(R1),R2
022756  010267  172370                MOV     R2,RD.DATA
022762  026767  172362  172362        CMP     WT.DATA,RD.DATA                 ;                       3450
023000  001403                        BEQ     1$
023002  012776  000001  000010        MOV     #1,@10(SP)                      ; *,ERR.FLG
023010  142777  000040  172516  1$:   BICB    #40,@ML.REG+120                 ;                       3452
023016  000207                        RTS     PC                              ;                       3393

                        ; Routine Size:  41 words
                        ; Maximum stack depth per invocation:  3 words
```

```
;ML4AD
;                       ROUTINE DECLARATION SECTION                     29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (28)

;               3454    routine WRT_E1 (TST_PAT, index) : novalue =
;               3455        begin
;               3456
;               3457    !++
;               3458    ! FUNCTIONAL DESCRIPTION:
;               3459    !       LOADS THE ECC CRC WORD REG 1 WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
;               3460    ! FORMAL PARAMETERS:
;               3461    !       TST_PAT
;               3462    !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
;               3463    !       INDEX
;               3464    !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
;               3465    !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
;               3466    !--
;               3467
;               3468        ECC_DM = ONE;                                   !SET ECC DIAG MODE
;               3469        MLE1 = WRT_MASK;                                !LOAD MLE1 WITH GENERATED WRT_MASK PATTERN
;               3470        ECC_DM = ZERO;                                  !CLEAR ECC DIAG MODE
;               3471        end;

                                        .SBTTL  WRT.E1 ROUTINE DECLARATION SECTION
023010  004167  161524          WRT.E1: JSR     R1,$SAVE2                       ;                               3454
023014  152777  000001  172502          BISB    #1,@ML.REG+120                  ;                               3468
023022  016600  000010                  MOV     10(SP),R0                       ; INDEX,*                       3469
023026  006300                          ASL     R0
023030  006300                          ASL     R0
023032  006300                          ASL     R0
023034  010001                          MOV     R0,R1
023036  016100  015406                  MOV     ML.REG+2(R1),R0
023042  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
023046  046100  015410                  BIC     ML.REG+4(R1),R0
023052  016102  015412                  MOV     ML.REG+6(R1),R2
023056  050002                          BIS     R0,R2
023060  010277  172470                  MOV     R2,@ML.REG+150
023064  142777  000001  172432          BICB    #1,@ML.REG+120                  ;                               3470
023072  000207                          RTS     PC                              ;                               3454

                        ; Routine Size:  26 words
                        ; Maximum stack depth per invocation:  3 words
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (29)

;        3472    routine RD_E1 (TST_PAT, index, ERR_FLG) : novalue =
;        3473        begin
;        3474
;        3475    !++
;        3476    ! FUNCTIONAL DESCRIPTION:
;        3477    !
;        3478    !       COMPARES THE CONTENTS OF THE
;        3479    !       ECC CRC WORD REGISTER 1
;        3480    !       WITH THE MASKED DATA PATTERN
;        3481    !       GENERATED BY THE MACRO 'WRT_MASK'
;        3482    !
;        3483    !       IF THE COMPARE IS NOT EQUAL THEN
;        3484    !        THE FORMAL PARAMETER 'ERR_FLG' IS
;        3485    !       ASSIGNED A ONE TO INDICATE THE ERROR.
;        3486    !
;        3487    ! FORMAL PARAMETERS:
;        3488    !       TST_PAT
;        3489    !       DATA PATTERN TO BE MASKED AND
;        3490    !       COMPARED AGAINST THE CONTENTS
;        3491    !       OF THE REGISTER UNDER TEST.
;        3492    !
;        3493    !       INDEX
;        3494    !       USED BY THE MACRO WRT_MASK TO
;        3495    !       SELECT THE CURRENT REGISTERS ADDRESS,
;        3496    !       FORCED HI, FORCED LO AND DON'T CARE
;        3497    !       MASK INFORMATION.
;        3498    !
;        3499    !       ERR_FLG
;        3500    !       CONTAINS THE ADDRESS (PASSED BY REF)
;        3501    !       OF THE CALLERS ERROR FLG TO ENABLE THE
;        3502    !       CALLER TO EXAMINE THE ERROR STATUS
;        3503    !       OF THE ROUTINE CALL.
;        3504    !
;        3505    ! IMPLICIT INPUTS:
;        3506    !       WT_DATA
;        3507    !         GETS LOADED WITH THE GENERATED
;        3508    !         WRT_MASK DATA PATTERN THUS ALLOWING
;        3509    !         CALLER TO PRINT FAILING GOOD DATA.
;        3510    !
;        3511    !       RD_DATA
;        3512    !         GETS LOADED WITH DATA READ FROM THE
;        3513    !         REGISTER THUS ALLOWING CALLER
;        3514    !         TO PRINT FAILING BAD DATA.
;        3515    !
;        3516    ! IMPLICIT OUTPUTS:
;        3517    !       GLOBAL LOCATION WR_DATA
;        3518    !       AND RD_DATA LOADED WITH GOOD
;        3519    !       AND BAD REGISTER DATA
;        3520    !
;        3521    !--
;        3522
;        3523        .ERR_FLG = ZERO;                        !CLEAR THE ERROR FLAG
```

```
5442 ;ML4AD
5443 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
5444                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (29)
5445 ;          3524        ECC_DM = ONE;                             !SET ECC DIAG MODE
5446 ;          3525        WT_DATA = WRT_MASK;                       !SAVE THE DATA WRITTEN TO THE REGISTER
5447 ;          3526        RD_DATA = .ML$E1 or .IGNORE;              !READ AND SAVE THE REGISTER
5448 ;          3527
5449 ;          3528        if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
5450                3529
5451 .          3530        ECC_DM = ZERO;                            !SET ERROR FLAG IF NEQ
5452 ;          3531                                                  !CLEAR ECC DIAG MODE
5453 ;          3532        end;
5457
5458                                        .SBTTL  RD.E1 ROUTINE DECLARATION SECTION
5462 023074  004167  161440          RD.E1: JSR     R1,$SAVE2                    ;                              3472
5463 023100  005076  000010                 CLR     @10(SP)                      ; ERR.FLG                      3523
5464 023104  152777  000001  172412        BISB    #1,@ML.REG+120               ;                              3524
5465 023112  016600  000012                 MOV     12(SP),R0                    ; INDEX,*                      3525
5466 023116  006300                         ASL     R0
5467 023120  006300                         ASL     R0
5468 023122  006300                         ASL     R0
5469 023124  010001                         MOV     R0,R1
5470 023126  016100  015406                 MOV     ML.REG+2(R1),R0
5471 023132  056600  000014                 BIS     14(SP),R0                    ; TST.PAT,*
5472 023136  046100  015410                 BIC     ML.REG+4(R1),R0
5473 023142  016102  015412                 MOV     ML.REG+6(R1),R2
5474 023146  050002                         BIS     R0,R2
5475 023150  010267  172174                 MOV     R2,WT.DATA
5476 023154  017702  172374                 MOV     @ML.REG+150,R2               ;                              3526
5477 023160  056102  015412                 BIS     ML.REG+6(R1),R2
5478 023164  010267  172162                 MOV     R2,RD.DATA
5479 023170  026767  172154  172154        CMP     WT.DATA,RD.DATA              ;                              3528
5480 023176  001403                         BEQ     1$
5481 023200  012776  000001  000010        MOV     #1,@10(SP)                   ; *,ERR.FLG
5482 023206  142777  000001  172310  1$:   BICB    #1,@ML.REG+120               ;                              3530
5483 023214  000207                         RTS     PC                           ;                              3472
5484
5485                                ; Routine Size:  41 words
5486                                ; Maximum stack depth per invocation:  3 words
5491
5492
```

```
5494 ;ML4AD
5495 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
5496                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (30)
5497 ;          3533   routine WRT_E2 (TST_PAT, index) : novalue =
5498 ;          3534       begin
5499 ;          3535
5500 ;          3536   !++
5501 ;          3537   ! FUNCTIONAL DESCRIPTION:
5502 ;          3538   !
5503 ;          3539   !       LOADS THE ECC CRC WORD
5504 ;          3540   !       REGISTER 2 WITH A DATA PATTERN
5505 ;          3541   !       GENERATED BY THE MACRO
5506 ;          3542   !       WRT_MASK
5507 ;          3543   !
5508 ;          3544   ! FORMAL PARAMETERS:
5509 ;          3545   !
5510 ;          3546   !       TST_PAT
5511 ;          3547   !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
5512 ;          3548   !
5513 ;          3549   !       INDEX
5514 ;          3550   !       USED BY THE MACRO WRT_MASK
5515 ;          3551   !       TO SELECT THE CURRENT REGISTERS
5516 ;          3552   !       ADDRESS, FORCED HI, FORCED
5517 ;          3553   !       LO AND DON'T CARE MASK
5518 ;          3554   !       INFORMATION.
5519 ;          3555   !
5520 ;          3556   !--
5521 ;          3557
5522 ;          3558       ECC_DM = ONE;                              !SET ECC DIAG MODE
5523 ;          3559       MLE2_MASK = %o'177700';                    !MASK OUT DATA DIAG BITS
5524 ;          3560       MLE2 = WRT_MASK;                           !LOAD MLE2 WITH GENERATED WRT_MASK PATTERN
5525 ;          3561       MLE2_MASK = %o'000000';                    !RESTORE MLE2_MASK
5526 ;          3562       ECC_DM = ZERO;                             !CLEAR ECC DIAG MODE
5527 ;          3563       end;
5531
5532                                        .SBTTL  WRT.E2 ROUTINE DECLARATION SECTION
5536 023216  004167  161316          WRT.E2: JSR     R1,$SAVE2                       ;                              3533
5537 023222  152777  000001  172274          BISB    #1,@#ML.REG+120                 ;                              3558
5538 023230  012767  177700  172334          MOV     #-100,ML.REG+166                ;                              3559
5539 023236  016600  000010                  MOV     10(SP),R0                       ; INDEX,*                      3560
5540 023242  006300                          ASL     R0
5541 023244  006300                          ASL     R0
5542 023246  006300                          ASL     R0
5543 023250  010001                          MOV     R0,R1
5544 023252  016100  015406                  MOV     ML.REG+2(R1),R0
5545 023256  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
5546 023262  046100  015410                  BIC     ML.REG+4(R1),R0
5547 023266  016102  015412                  MOV     ML.REG+6(R1),R2
5548 023272  050002                          BIS     R0,R2
```

```
5550                                    ;ML4AD
5551                                    ;               ROUTINE DECLARATION SECTION
5552
5553 023274  010277  172264                    MOV     R2,@ML.REG+160
5554 023300  005067  172266                    CLR     ML.REG+166
5555 023304  142777  000001  172212          BICB    #1,@ML.REG+120
5556 023312  000207                            RTS     PC
5557
5558                                    ; Routine Size:  31 words
5559                                    ; Maximum stack depth per invocation:  3 words
5564
5565
```


3561
3562
3533

```
                3564  routine RD_E2 (TST_PAT, index, ERR_FLG) : novalue =
                3565      begin
                3566
                3567  !++
                3568  ! FUNCTIONAL DESCRIPTION:
                3569  !
                3570  !       COMPARES THE CONTENTS OF THE
                3571  !       ECC CRC WORD REGISTER 2
                3572  !       WITH THE MASKED DATA PATTERN
                3573  !       GENERATED BY THE MACRO 'WRT_MASK'
                3574  !
                3575  !       IF THE COMPARE IS NOT EQUAL THEN
                3576  !       THE FORMAL PARAMETER 'ERR_FLG' IS
                3577  !       ASSIGNED A ONE TO INDICATE THE
                3578  !       ERROR.
                3579  !
                3580  ! FORMAL PARAMETERS:
                3581  !
                3582  !       TST_PAT
                3583  !       DATA PATTERN TO BE MASKED AND
                3584  !       COMPARED AGAINST THE CONTENTS
                3585  !       OF THE REGISTER UNDER TEST.
                3586  !
                3587  !       INDEX
                3588  !       USED BY THE MACRO WRT_MASK TO
                3589  !       SELECT THE CURRENT REGISTERS ADDRESS,
                3590  !       FORCED HI, FORCED LO AND DON'T CARE
                3591  !       MASK INFORMATION.
                3592  !
                3593  !       ERR_FLG
                3594  !       CONTAINS THE ADDRESS (PASSED BY REF)
                3595  !       OF THE CALLERS ERROR FLG TO ENABLE THE
                3596  !       CALLER TO EXAMINE THE ERROR STATUS
                3597  !       OF THE ROUTINE CALL.
                3598  !
                3599  ! IMPLICIT INPUTS:
                3600  !       WT_DATA
                3601  !         GETS LOADED WITH THE GENERATED
                3602  !         WRT_MASK DATA PATTERN THUS ALLOWING
                3603  !         CALLER TO PRINT FAILING GOOD DATA.
                3604  !
                3605  !       RD_DATA
                3606  !         GETS LOADED WITH DATA READ FROM THE
                3607  !         REGISTER THUS ALLOWING CALLER
                3608  !         TO PRINT FAILING BAD DATA.
                3609  !
                3610  ! IMPLICIT OUTPUTS:
                3611  !       GLOBAL LOCATION WR_DATA
                3612  !       AND RD_DATA LOADED WITH GOOD
                3613  !       AND BAD REGISTER DATA
                3614  !
                3615  !-
```

```
5623 ;ML4AD
5624 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
5625 ;                                                                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (31)
5626 ;          3616
5627 ;          3617        .ERR_FLG = ZERO;                           !CLEAR ERROR FLAG
5628 ;          3618        ECC_DM = ONE;                              !SET ECC DIAG MODE
5629 ;          3619        ML£2_MASK = %o'177700';                    !MASK OUT DATA DIAG BITS
5630 ;          3620        WT_DATA = WRT_MASK;                        !SAVE THE DATA WRITTEN TO THE REGISTER
5631 ;          3621        RD_DATA = .ML£2 or .IGNORE;                !READ AND SAVE THE REGISTER
5632 ;          3622
5633 ;          3623        if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
5634 ;          3624
5635 ;          3625                                                   !SET ERROR FLG IF NEQ
5636 ;          3626        ML£2_MASK = %o'000000';                    !RESTORE ML£2 MASK
5637 ;          3627        ECC_DM = ZERO;                             !CLEAR ECC DIAG MODE
5638 ;          3628        end;
5642
5643                                        .SBTTL  RD.E2 ROUTINE DECLARATION SECTION
5647 023314  004167  161220          RD.E2: JSR     R1,$SAVE2                    ;                          3564
5648 023320  005076  000010                 CLR     @10(SP)                      ; ERR.FLG                  3617
5649 023324  152777  000001  172172        BISB    #1,@ML.REG+120               ;                          3618
5650 023332  012767  177700  172232        MOV     #-100,ML.REG+166             ;                          3619
5651 023340  016600  000012                 MOV     12(SP),R0                    ; INDEX,*                  3620
5652 023344  006300                         ASL     R0
5653 023346  006300                         ASL     R0
5654 023350  006300                         ASL     R0
5655 023352  010001                         MOV     R0,R1
5656 023354  016100  015406                 MOV     ML.REG+2(R1),R0
5657 023360  056600  000014                 BIS     14(SP),R0                    ; TST.PAT,*
5658 023364  046100  015410                 BIC     ML.REG+4(R1),R0
5659 023370  016102  015412                 MOV     ML.REG+6(R1),R2
5660 023374  050002                         BIS     R0,R2
5661 023376  010267  171746                 MOV     R2,WT.DATA
5662 023402  017702  172156                 MOV     @ML.REG+160,R2               ;                          3621
5663 023406  056102  015412                 BIS     ML.REG+6(R1),R2
5664 023412  010267  171734                 MOV     R2,RD.DATA
5665 023416  026767  171726  171726        CMP     WT.DATA,RD.DATA              ;                          3623
5666 023424  001403                         BEQ     1$
5667 023426  012776  000001  000010        MOV     #1,@10(SP)                   ; *,ERR.FLG
5668 023434  005067  172132          1$:    CLR     ML.REG+166                   ;                          3626
5669 023440  142777  000001  172056        BICB    #1,@ML.REG+120               ;                          3627
5670 023446  000207                         RTS     PC                           ;                          3564
5671
5672
5673                                ; Routine Size:  46 words
                                    ; Maximum stack depth per invocation:  3 words
```

```
5679 ;ML4AD
5680 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
5681                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (32)
5682 ;          3629   routine WRT_PD (TST_PAT, index) : novalue =
5683 ;          3630       begin
5684 ;          3631
5685 ;          3632   !++
5686 ;          3633   ! FUNCTIONAL DESCRIPTION:
5687 ;          3634   !
5688 ;          3635   !       LOADS THE PROM DATA
5689 ;          3636   !       REGISTER WITH A DATA PATTERN
5690 ;          3637   !       GENERATED BY THE MACRO
5691 ;          3638   !       WRT_MASK
5692 ;          3639   !
5693 ;          3640   ! FORMAL PARAMETERS:
5694 ;          3641   !
5695 ;          3642   !       TST_PAT
5696 ;          3643   !       CURRENT DATA PATTERN TO BE
5697 ;          3644   !       LOADED IN THE REGISTER.
5698 ;          3645   !
5699 ;          3646   !       INDEX
5700 ;          3647   !       USED BY THE MACRO WRT_MASK
5701 ;          3648   !       TO SELECT THE CURRENT REGISTERS
5702 ;          3649   !       ADDRESS, FORCED HI, FORCED
5703 ;          3650   !       LO AND DON'T CARE MASK
5704 ;          3651   !       INFORMATION
5705 ;          3652   !
5706 ;          3653   !--
5707 ;          3654
5708 ;          3655       PROM_RW = ONE;                  !SET PROM READ WRITE
5709 ;          3656       DAT_DM = ONE;                   !SET DATA DIAG MODE
5710 ;          3657       MLPD = WRT_MASK;                !LOAD MLPD WITH GENERATED WRT_MASK PATTERN
5711 ;          3658       DAT_CLK = ONE;                  !DO A DATA CLK
5712 ;          3659       PROM_RW = ZERO;                 !CLEAR PROM READ WRITE
5713 ;          3660       DAT_DM = ZERO;                  !CLEAR DATA DIAG MODE
5714 ;          3661       end;
5718
5719                                       .SBTTL  WRT.PD ROUTINE DECLARATION SECTION
5723 023450 004167 161064          WRT.PD: JSR     R1,$SAVE2                          ;                    3629
5724 023454 152777 000100 172042          BISB    #100,@ML.REG+120                   ;                    3655
5725 023462 152777 000010 172034          BISB    #10,@ML.REG+120                    ;                    3656
5726 023470 016600 000010                 MOV     10(SP),R0                          ; INDEX,*            3657
5727 023474 006300                        ASL     R0
5728 023476 006300                        ASL     R0
5729 023500 006300                        ASL     R0
5730 023502 010001                        MOV     R0,R1
5731 023504 016100 015406                 MOV     ML.REG+2(R1),R0
5732 023510 056600 000012                 BIS     12(SP),R0                          ; TST.PAT,*
5733 023514 046100 015410                 BIC     ML.REG+4(R1),R0
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION

5738 023520  016102  015412                    MOV     ML.REG+6(R1),R2
5739 023524  050002                            BIS     R0,R2
5740 023526  010277  172102                    MOV     R2,@ML.REG+230
5741 023532  152777  000020  171764            BISB    #20,@ML.REG+120
5742 023540  142777  000100  171756            BICB    #100,@ML.REG+120
5743 023546  142777  000010  171750            BICB    #10,@ML.REG+120
5744 023554  000207                            RTS     PC

                                        ; Routine Size:  35 words
                                        ; Maximum stack depth per invocation:  3 words
```

```
5755 ;ML4AD
5756 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
5757 ;                                                                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (33)
5758 ;          3662    routine RD_PD (TST_PAT, index, ERR_FLG) : novalue =
5759 ;          3663        begin
5760 ;          3664
5761 ;          3665    !++
5762 ;          3666    ! FUNCTIONAL DESCRIPTION:
5763 ;          3667    !
5764 ;          3668    !       COMPARES THE CONTENTS OF THE
5765 ;          3669    !       PROM DATA REGISTER
5766 ;          3670    !       WITH THE MASKED DATA PATTERN
5767 ;          3671    !       GENERATED BY THE MACRO 'WRT_MASK'
5768 ;          3672    !
5769 ;          3673    !       IF THE COMPARE IS NOT EQUAL THEN
5770 ;          3674    !       THE FORMAL PARAMETER 'ERR FLG' IS
5771 ;          3675    !       ASSIGNED A ONE TO INDICATE THE
5772 ;          3676    !       ERROR
5773 ;          3677    !
5774 ;          3678    ! FORMAL PARAMETERS:
5775 ;          3679    !
5776 ;          3680    !       TST_PAT
5777 ;          3681    !       DATA PATTERN TO BE MASKED AND
5778 ;          3682    !       COMPARED AGAINST THE CONTENTS
5779 ;          3683    !       OF THE REGISTER UNDER TEST
5780 ;          3684    !
5781 ;          3685    !       INDEX
5782 ;          3686    !       USED BY THE MACRO WRT_MASK TO
5783 ;          3687    !       SELECT THE CURRENT REGISTERS ADDRESS,
5784 ;          3688    !       FORCED HI, FORCED LO AND DON'T CARE
5785 ;          3689    !       MASK INFORMATION.
5786 ;          3690    !
5787 ;          3691    !       ERR_FLG
5788 ;          3692    !       CONTAINS THE ADDRESS (PASSED BY REF)
5789 ;          3693    !       OF THE CALLERS ERROR FLG TO ENABLE THE
5790 ;          3694    !       CALLER TO EXAMINE THE ERROR STATUS
5791 ;          3695    !       OF THE ROUTINE CALL.
5792 ;          3696    !
5793 ;          3697    ! IMPLICIT INPUTS:
5794 ;          3698    !       WT_DATA
5795 ;          3699    !         GETS LOADED WITH THE GENERATED
5796 ;          3700    !         WRT_MASK DATA PATTERN THUS ALLOWING
5797 ;          3701    !         CALLER TO PRINT FAILING GOOD DATA.
5798 ;          3702    !
5799 ;          3703    !       RD_DATA
5800 ;          3704    !         GETS LOADED WITH DATA READ FROM THE
5801 ;          3705    !         REGISTER THUS ALLOWING CALLER
5802 ;          3706    !         TO PRINT FAILING BAD DATA.
5803 ;          3707    !
5804 ;          3708    ! IMPLICIT OUTPUTS:
5805 ;          3709    !       GLOBAL LOCATION WR_DATA
5806 ;          3710    !       AND RD_DATA LOADED WITH GOOD
5807 ;          3711    !       AND BAD REGISTER DATA
5808 ;          3712    !
5809 ;          3713    !--
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (33)

;          3714
;          3715      .ERR_FLG = ZERO;                          !CLEAR ERROR FLG
;          3716      PROM_RW = ONE;                            !SET PROM READ WRITE
;          3717      DAT_DM = ONE;                             !SET DATA DIAG MODE
;          3718      WT_DATA = WRT_MASK;                       !SAVE THE DATA WRITTEN TO THE REGISTER
;          3719      RD_DATA = .MLPD or .IGNORE;               !READ AND SAVE THE REGISTER
;          3720
;          3721      if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
;          3722
;          3723                                                !SET ERROR FLAG IF NEQ
;          3724      PROM_RW = ZERO;                           !CLEAR PROM READ WRTE
;          3725      DAT_DM = ZERO;                            !CLEAR DATA DIAG MODE
;          3726      end;

                                        .SBTTL  RD.PD ROUTINE DECLARATION SECTION
023556  004167  160756          RD.PD:  JSR     R1,$SAVE2                                                       ;
023562  005076  000010                  CLR     @10(SP)                         ; ERR.FLG                       3662
023566  152777  000100  171730          BISB    #100,@ML.REG+120                ;                               3715
023574  152777  000010  171722          BISB    #10,@ML.REG+120                 ;                               3716
023602  016600  000012                  MOV     12(SP),R0                       ; INDEX,*                       3717
023606  006300                          ASL     R0                                                              3718
023610  006300                          ASL     R0
023612  006300                          ASL     R0
023614  010001                          MOV     R0,R1
023616  016100  015406                  MOV     ML.REG+2(R1),R0
023622  056600  000014                  BIS     14(SP),R0                       ; TST.PAT,*
023626  046100  015410                  BIC     ML.REG+4(R1),R0
023632  016102  015412                  MOV     ML.REG+6(R1),R2
023636  050002                          BIS     R0,R2
023640  010267  171504                  MOV     R2,WT.DATA
023644  017702  171764                  MOV     @ML.REG+230,R2                  ;                               3719
023650  056102  015412                  BIS     ML.REG+6(R1),R2
023654  010267  171472                  MOV     R2,RD.DATA
023660  026767  171464  171464          CMP     WT.DATA,RD.DATA                 ;                               3721
023666  001403                          BEQ     1$
023670  012776  000001  000010          MOV     #1,@10(SP)                      ; *,ERR.FLG
023676  142777  000100  171620  1$:     BICB    #100,@ML.REG+120                ;                               3724
023704  142777  000010  171612          BICB    #10,@ML.REG+120                 ;                               3725
023712  000207                          RTS     PC                              ;                               3662

                        ; Routine Size:  47 words
                        ; Maximum stack depth per invocation:  3 words
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (34)

;          3727  routine WRT_EL (TST_PAT, index) : novalue =
;          3728      begin
;          3729
;          3730  !++
;          3731  ! FUNCTIONAL DESCRIPTION:
;          3732  !       DUMMY ROUTINE CALL TO ASSIST IN THE READ
;          3733  !       WRITE REGISTER ALGORITHEM
;          3734  !
;          3735  ! FORMAL PARAMETERS:
;          3736  !       TST_PAT
;          3737  !       DATA PATTERN TO BE MASKED AND
;          3738  !       COMPARED AGAINST THE CONTENTS
;          3739  !       OF THE REGISTER UNDER TEST
;          3740  !
;          3741  !       INDEX
;          3742  !       USED BY THE MACRO WRT_MASK TO
;          3743  !       SELECT THE CURRENT REGISTERS ADDRESS,
;          3744  !       FORCED HI, FORCED LO AND DON'T CARE
;          3745  !       MASK INFORMATION.
;          3746  !
;          3747  !--
;          3748
;          3749  !
;          3750  !  ERROR LOCATION REG IS READ ONLY
;          3751      return;
;          3752      end;
;
;                                  .SBTTL  WRT.EL ROUTINE DECLARATION SECTION
;023714  000207  WRT.EL: RTS     PC                              ;  3727
;
;                ; Routine Size:  1 word
;                ; Maximum stack depth per invocation:  0 words
;
;
```

```
;ML4AD                                                      29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;              ROUTINE DECLARATION SECTION                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (35)

;              routine RD_EL (TST_PAT, index, ERR_FLG) : novalue =
;                  begin
;
;              !++
;              ! FUNCTIONAL DESCRIPTION:
;              !
;              !        COMPARES THE CONTENTS OF THE
;              !        ERROR LOCATION REGISTER
;              !        WITH THE CONTENTS OF TST_PAT.
;              !
;              !        IF THE COMPARE IS NOT EQUAL THEN
;              !        THE FORMAL PARAMETER 'ERR FLG' IS
;              !        ASSIGNED A ONE TO INDICATE THE
;              !        ERROR
;              !
;              ! FORMAL PARAMETERS:
;              !        TST_PAT
;              !        DATA PATTERN TO BE MASKED AND
;              !        COMPARED AGAINST THE CONTENTS
;              !        OF THE REGISTER UNDER TEST
;              !
;              !        INDEX
;              !        USED BY THE MACRO WRT_MASK TO
;              !        SELECT THE CURRENT REGISTERS ADDRESS,
;              !        FORCED HI, FORCED LO AND DON'T CARE
;              !        MASK INFORMATION.
;              !
;              !        ERR_FLG
;              !        CONTAINS THE ADDRESS (PASSED BY REF)
;              !        OF THE CALLERS ERROR FLG TO ENABLE THE
;              !        CALLER TO EXAMINE THE ERROR STATUS
;              !        OF THE ROUTINE CALL.
;              !
;              ! IMPLICIT INPUTS:
;              !        WT_DATA
;              !          GETS LOADED WITH THE GENERATED
;              !          WRT_MASK DATA PATTERN THUS ALLOWING
;              !          CALLER TO PRINT FAILING GOOD DATA.
;              !
;              !        RD_DATA
;              !          GETS LOADED WITH DATA READ FROM THE
;              !          REGISTER THUS ALLOWING CALLER
;              !          TO PRINT FAILING BAD DATA.
;              !
;              ! IMPLICIT OUTPUTS:
;              !        GLOBAL LOCATION WR_DATA
;              !        AND RD_DATA LOADED WITH GOOD
;              !        AND BAD REGISTER DATA
;              !
;              !--
;
;                .ERR_FLG = ZERO;
```

SEQ 0146

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (35)
;          3805      WT_DATA = .TST_PAT;                            !SAVE THE DATA WRITTEN TO THE REGISTER
;          3806      RD_DATA = .MLEC;                               !READ AND SAVE THE REGISTER
;          3807
;          3808      if .RD_DATA neq .WT_DATA then .ERR_FLG = ONE;       !READ REGISTER FOR WT_DATA
;          3809
;          3810      end;

                                    .SBTTL  RD.EL ROUTINE DECLARATION SECTION
023716  005076  000002          RD.EL:  CLR     @2(SP)                          ; ERR.FLG                       3804
023722  016667  000006  171420          MOV     6(SP),WT.DATA                   ; TST.PAT,*                     3805
023730  017767  171670  171414          MOV     @ML.REG+220,RD.DATA             ;                               3806
023736  026767  171410  171404          CMP     RD.DATA,WT.DATA                 ;                               3808
023744  001403                          BEQ     1$
023746  012776  000001  000002          MOV     #1,@2(SP)                       ; *,ERR.FLG
023754  000207                  1$:     RTS     PC                              ;                               3753

                                    ; Routine Size:  16 words
                                    ; Maximum stack depth per invocation:  0 words
```

```
6005 :ML4AD
6006 :                  ROUTINE DECLARATION SECTION                     29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
6007                                                                    29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (36)
6008 :          3811    routine WRT_EE (TST_PAT, index) : novalue =
6009 :          3812        begin
6010 :          3813
6011 :          3814    !++
6012 :          3815    ! FUNCTIONAL DESCRIPTION:
6013 :          3816    !       DUMMY ROUTINE CALL TO ASSIST IN THE READ
6014 :          3817    !       WRITE REGISTER ALGORITHEM
6015 :          3818    !
6016 :          3819    ! FORMAL PARAMETERS:
6017 :          3820    !       TST_PAT
6018 :          3821    !       DATA PATTERN TO BE MASKED AND
6019 :          3822    !       COMPARED AGAINST THE CONTENTS
6020 :          3823    !       OF THE REGISTER UNDER TEST
6021 :          3824    !
6022 :          3825    !       INDEX
6023 :          3826    !       USED BY THE MACRO WRT_MASK TO
6024 :          3827    !       SELECT THE CURRENT REGISTERS ADDRESS,
6025 :          3828    !       FORCED HI, FORCED LO AND DON'T CARE
6026 :          3829    !       MASK INFORMATION.
6027 :          3830    !
6028 :          3831    !--
6029 :          3832
6030 :          3833    ! ECC ERROR REGISTER IS READ ONLY
6031 :          3834        return;
6032 :          3835        end;
6036
6037
                                .SBTTL  WRT.EE ROUTINE DECLARATION SECTION
6041 023756  000207   WRT.EE: RTS     PC                                  ;
6042                                                                                                    3811
6043                    ; Routine Size:  1 word
6044                    ; Maximum stack depth per invocation:  0 words
6049
6050
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (37)

;          3836  routine RD_EE (TST_PAT, index, ERR_FLG) : novalue =
;          3837      begin
;          3838
;          3839  !++
;          3840  ! FUNCTIONAL DESCRIPTION:
;          3841  !
;          3842  !       COMPARES THE CONTENTS OF THE
;          3843  !       ECC ERROR REGISTER WITH TST_PAT.
;          3844  !
;          3845  !       IF THE COMPARE IS NOT EQUAL THEN
;          3846  !       THE FORMAL PARAMETER 'ERR_FLG' IS
;          3847  !       ASSIGNED A ONE TO INDICATE THE
;          3848  !       ERROR.
;          3849  !
;          3850  ! FORMAL PARAMETERS:
;          3851  !       TST_PAT
;          3852  !       DATA PATTERN TO BE
;          3853  !       COMPARED AGAINST THE CONTENTS
;          3854  !       OF THE REGISTER UNDER TEST.
;          3855  !
;          3856  !       INDEX
;          3857  !       USED BY THE MACRO MLEE TO
;          3858  !       SELECT THE CURRENT REGISTERS ADDRESS.
;          3859  !
;          3860  !       ERR_FLG
;          3861  !       CONTAINS THE ADDRESS (PASSED
;          3862  !       BY REF) OF THE CALLERS ERROR FLG TO ENABLE
;          3863  !       THE CALLER TO EXAMINE THE ERROR STATUS
;          3864  !       OF THE ROUTINE CALL.
;          3865  !
;          3866  ! IMPLICIT INPUTS:
;          3867  !       WT_DATA
;          3868  !         GETS LOADED WITH THE TST_PAT
;          3869  !         THUS ALLOWING CALLER TO PRINT
;          3870  !         THE FAILING DATA.
;          3871  !
;          3872  !       RD_DATA
;          3873  !         GETS LOADED WITH DATA READ FROM THE
;          3874  !         REGISTER THUS ALLOWING CALLER
;          3875  !         TO PRINT FAILING BAD DATA.
;          3876  !
;          3877  ! IMPLICIT OUTPUTS:
;          3878  !       GLOBAL LOCATION WR_DATA
;          3879  !       AND RD_DATA LOADED WITH GOOD
;          3880  !       AND BAD REGISTER DATA
;          3881  !
;          3882  !--
;          3883
;          3884      .ERR_FLG = ZERO;                          !CLEAR THE ERROR FLAG
;          3885      WT_DATA = .TST_PAT;                       !SAVE DATA WRITTEN TO THE REGISTER
;          3886      RD_DATA = .MLEE;                          !READ AND SAVE THE REGISTER
;          3887
```

```
6108 ;ML4AD
6109 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
6110                                                                    29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (37)
6111 ;          3888        if .RD_DATA neq .WT_DATA then .ERR_FLG = ONE;       !READ MLEE FOR TST_PAT
6112 ;          3889
6113 ;          3890                                                    !SET THE ERROR FLAG IF NEQ
6114 ;          3891        end;
6118
6119                                    .SBTTL  RD.EE ROUTINE DECLARATION SECTION
6123 023760  005076  000002        RD.EE:  CLR     @2(SP)                          ; ERR.FLG                       3884
6124 023764  016667  000006  171356        MOV     6(SP),WT.DATA                   ; TST.PAT,*                     3885
6125 023772  017767  171616  171352        MOV     @ML.REG+210,RD.DATA             ;                               3886
6126 024000  026767  171346  171342        CMP     RD.DATA,WT.DATA                 ;                               3888
6127 024006  001403                       BEQ     1$
6128 024010  012776  000001  000002        MOV     #1,@2(SP)                       ; *.ERR.FLG
6129 024016  000207                 1$:    RTS     PC                              ;                               3836
6130
6131                                    ; Routine Size:  16 words
6132                                    ; Maximum stack depth per invocation:  0 words
6137
6138
```

```
:ML4AD
:                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
:                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (38)
:          3892  routine WRT_D1 (TST_PAT, index) : novalue =
:          3893      begin
:          3894
:          3895  !++
:          3896  ! FUNCTIONAL DESCRIPTION:
:          3897  !       LOADS THE DATA DIAG REG 1 WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
:          3898  ! FORMAL PARAMETERS:
:          3899  !       TST_PAT
:          3900  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
:          3901  !       INDEX
:          3902  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
:          3903  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
:          3904  !--
:          3905
:          3906      DAT_DM = ONE;                                 !SET DATA DIAG MODE
:          3907      MLDT = WRT_MASK;                              !LOAD MLD1 WITH GENERATED WRT_MASK PATTERN
:          3908      DAT_DM = ZERO;                                !CLEAR DATA DIAG MODE
:          3909      end;

                                    .SBTTL  WRT.D1 ROUTINE DECLARATION SECTION
024020  004167  160514        WRT.D1: JSR     R1,$SAVE2                       ;                       3892
024024  152777  000010  171472        BISB    #10,@ML.REG+120                 ;                       3906
024032  016600  000010                MOV     10(SP),R0                       ; INDEX,*               39C7
024036  006300                        ASL     R0
024040  006300                        ASL     R0
024042  006300                        ASL     R0
024044  010001                        MOV     R0,R1
024046  016100  015406                MOV     ML.REG+2(R1),R0
024052  056600  000012                BIS     12(SP),R0                       ; TST.PAT,*
024056  046100  015410                BIC     ML.REG+4(R1),R0
024062  016102  015412                MOV     ML.REG+6(R1),R2
024066  050002                        BIS     R0,R2
024070  010277  171500                MOV     R2,@ML.REG+170
024074  142777  000010  171422        BICB    #10,@ML.REG+120                 ;                       3908
024102  000207                        RTS     PC                              ;                       3892

                        ; Routine Size:  26 words
                        ; Maximum stack depth per invocation:  3 words
```

```
6194 ;ML4AD
6195 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
6196 ;                                                                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (39)
6197 ;          3910    routine RD_D1 (TST_PAT, index, ERR_FLG) : novalue =
6198 ;          3911        begin
6199 ;          3912
6200 ;          3913    !++
6201 ;          3914    ! FUNCTIONAL DESCRIPTION:
6202 ;          3915    !
6203 ;          3916    !       COMPARES THE CONTENTS OF THE
6204 ;          3917    !       DATA DIAG REGISTER 1
6205 ;          3918    !       WITH THE MASKED DATA PATTERN
6206 ;          3919    !       GENERATED BY THE MACRO 'WRT_MASK''.
6207 ;          3920    !
6208 ;          3921    !       IF THE COMPARE IS NOT EQUAL THEN
6209 ;          3922    !       THE FORMAL PARAMETER 'ERR_FLG'' IS
6210 ;          3923    !       ASSIGNED A ONE TO INDICATE THE
6211 ;          3924    !       ERROR.
6212 ;          3925    !
6213 ;          3926    ! FORMAL PARAMETERS:
6214 ;          3927    !
6215 ;          3928    !       TST_PAT
6216 ;          3929    !       DATA PATTERN TO BE MASKED AND
6217 ;          3930    !       COMPARED AGAINST THE CONTENTS
6218 ;          3931    !       OF THE REGISTER UNDER TEST.
6219 ;          3932    !
6220 ;          3933    !       INDEX
6221 ;          3934    !       USED BY THE MACRO WRT_MASK TO
6222 ;          3935    !       SELECT THE CURRENT REGISTERS ADDRESS,
6223 ;          3936    !       FORCED HI, FORCED LO AND DON'T CARE
6224 ;          3937    !       MASK INFORMATION.
6225 ;          3938    !
6226 ;          3939    !       ERR_FLG
6227 ;          3940    !       CONTAINS THE ADDRESS (PASSED BY REF)
6228 ;          3941    !       OF THE CALLERS ERROR FLG TO ENABLE THE
6229 ;          3942    !       CALLER TO EXAMINE THE ERROR STATUS
6230 ;          3943    !       OF THE ROUTINE CALL.
6231 ;          3944    !
6232 ;          3945    ! IMPLICIT INPUTS:
6233 ;          3946    !       WT_DATA
6234 ;          3947    !         GETS LOADED WITH THE GENERATED
6235 ;          3948    !         WRT_MASK DATA PATTERN THUS ALLOWING
6236 ;          3949    !         CALLER TO PRINT FAILING GOOD DATA.
6237 ;          3950    !
6238 ;          3951    !       RD_DATA
6239 ;          3952    !         GETS LOADED WITH DATA READ FROM THE
6240 ;          3953    !         REGISTER THUS ALLOWING CALLER
6241 ;          3954    !         TO PRINT FAILING BAD DATA.
6242 ;          3955    !
6243 ;          3956    ! IMPLICIT OUTPUTS:
6244 ;          3957    !       GLOBAL LOCATION WR_DATA
6245 ;          3958    !       AND RD_DATA LOADED WITH GOOD
6246 ;          3959    !       AND BAD REGISTER DATA
6247 ;          3960    !
6248 ;          3961    !--
```

```
6250 :ML4AD
6251 :                  ROUTINE DECLARATION SECTION                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
6252 :                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (39)
6253 :          3962
6254 :          3963        .ERR_FLG = ZERO;                                 !CLEAR THE ERROR FLG
6255 :          3964        DAT_DM = ONE;                                    !SET DATA DIAG MODE
6256 :          3965        ML_FUNC = write;                                 !LOAD MLCS1 WITH WRITE FUNCTION
6257 :          3966        DAT_CLK = ONE;                                   !DO A DATA CLK
6258 :          3967
6259 :          3968        if .REG_INIT_FLG IS_SET                          !SEE IF CALLER IS REG INIT TEST
6260 :          3969                                                         !SET ERROR FLAG IF NEQ
6261 :          3970        then
6262 :          3971            begin
6263 :          3972            CLR_MBUS;                                    !CLEAR MBUS TO GENERATE INIT DATA
6264 :          3973            DAT_DM = ONE;
6265 :          3974            end;
6266 :          3975
6267 :          3976        WT_DATA = WRT_MASK;                              !SAVE THE DATA WRITTEN TO THE REGISTER
6268 :          3977        RD_DATA = .MLD1;                                 !READ AND SAVE THE REGISTER
6269 :          3978
6270 :          3979        if .RD_DATA neq .WT_DATA then .ERR_FLG = ONE;        !READ MLD1 FOR WRT_MASK
6271 :          3980
6272 :          3981                                                         !SET ERR_FLG IF NEQ
6273 :          3982        CLR_MBUS;                                        !CLEAR THE MBUS
6274 :          3983        end;
6278
6279                                        .SBTTL  RD.D1 ROUTINE DECLARATION SECTION
6283 024104  004167  160430          RD.D1:  JSR     R1,$SAVE2                        ;
6284 024110  005076  000010                  CLR     @10(SP)                          ; ERR.FLG                    3910
6285 024114  152777  000010  171402         BISB    #10,@ML.REG+120                  ;                            3963
6286 024122  152777  000077  171254         BICB    #77,@ML.REG                      ;                            3964
6287 024130  152777  000061  171246         BISB    #61,@ML.REG                      ;                            3965
6288 024136  152777  000020  171360         BISB    #20,@ML.REG+120                  ;
6289 024144  026727  171210  000001         CMP     REG.INIT.FLG,#1                  ;                            3966
6290 024152  001017                         BNE     1$                               ;                            3968
6291 024154  152777  000040  171262         BISB    #40,@ML.REG+40                   ;                            3971
6292 024162  016701  171644                 MOV     ML.DUT,R1
6293 024166  042701  177770                 BIC     #177770,R1
6294 024172  142777  000007  171244         BICB    #7,@ML.REG+40
6295 024200  150177  171240                 BISB    R1,@ML.REG+40
6296 024204  152777  000010  171312         BISB    #10,@ML.REG+120                  ;
6297 024212  016600  000012          1$:     MOV     12(SP),R0                        ; INDEX,*                    3973
6298 024216  006300                         ASL     R0                                                            3976
6299 024220  006300                         ASL     R0
6300 024222  006300                         ASL     R0
6301 024224  010001                         MOV     R0,R1
6302 024226  016100  015406                 MOV     ML.REG+2(R1),R0
6303 024232  056600  000014                 BIS     14(SP),R0                        ; TST.PAT,*
6304 024236  046100  015410                 BIC     ML.REG+4(R1),R0
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS
                                                                                                       29-Mar-1982 16:21:03    PA:<

 6309 024242  016102  015412                   MOV     ML.REG+6(R1),R2
 6310 024246  050002                           BIS     R0,R2
 6311 024250  010267  171074                   MOV     R2,WT.DATA
 6312 024254  017767  171074  171070           MOV     @ML.REG+170,RD.DATA             ;                3977
 6313 024262  026767  171064  171060           CMP     RD.DATA,WT.DATA                 ;                3979
 6314 024270  001403                           BEQ     2$
 6315 024272  012776  000001  000010           MOV     #1,@10(SP)                      ; *.ERR.FLG
 6316 024300  152777  000040  171136  2$:      BISB    #40,@ML.REG+40
 6317 024306  016702  171520                   MOV     ML.DUT,R2
 6318 024312  042702  177770                   BIC     #177770,R2
 6319 024316  142777  000007  171120           BICB    #7,@ML.REG+40
 6320 024324  150277  171114                   BISB    R2,@ML.REG+40
 6321 024330  000207                           RTS     PC                              ;                3910
 6322
 6323                                   ; Routine Size:  75 words
 6324                                   ; Maximum stack depth per invocation:  3 words
 6329
 6330
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (40)

;          3984  routine WRT_D2 (TST_PAT, index) : novalue =
;          3985      begin
;          3986
;          3987  !++
;          3988  ! FUNCTIONAL DESCRIPTION:
;          3989  !       LOADS THE DATA DIAG REG 2 WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
;          3990  ! FORMAL PARAMETERS:
;          3991  !       TST_PAT
;          3992  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
;          3993  !       INDEX
;          3994  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
;          3995  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
;          3996  !--
;          3997
;          3998      DAT_DM = ONE;                                  !SET DATA DIAG MODE
;          3999      MLD2 = WRT_MASK;                               !LOAD MLD2 WITH GENERATED WRT_MASK PATTERN
;          4000      DAT_DM = ZERO;                                 !CLEAR DATA DIAG MODE
;          4001      end;

                                    .SBTTL  WRT.D2 ROUTINE DECLARATION SECTION
024332  004167  160202        WRT.D2: JSR     R1,$SAVE2                       ;                       3984
024336  152777  000010  171160        BISB    #10,@ML.REG+120                 ;                       3998
024344  016600  000010                MOV     10(SP),R0                       ; INDEX,*               3999
024350  006300                        ASL     R0
024352  006300                        ASL     R0
024354  006300                        ASL     R0
024356  010001                        MOV     R0,R1
024360  016100  015406                MOV     ML.REG+2(R1),R0
024364  056600  000012                BIS     12(SP),R0                       ; TST.PAT,*
024370  046100  015410                BIC     ML.REG+4(R1),R0
024374  016102  015412                MOV     ML.REG+6(R1),R2
024400  050002                        BIS     R0,R2
024402  010277  171176                MOV     R2,@ML.REG+200
024406  142777  000010  171110        BICB    #10,@ML.REG+120                 ;                       4000
024414  000207                        RTS     PC                              ;                       3984

                        ; Routine Size:  26 words
                        ; Maximum stack depth per invocation:  3 words
```

```
6386 ;ML4AD
6387 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
6388 ;                                                                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (41)
6389 ;          4002    routine RD_D2 (TST_PAT, index, ERR_FLG) : novalue =
6390 ;          4003        begin
6391 ;          4004
6392 ;          4005    !++
6393 ;          4006    ! FUNCTIONAL DESCRIPTION:
6394 ;          4007    !
6395 ;          4008    !       COMPARE THE CONTENTS OF THE
6396 ;          4009    !       DATA DIAG REGISTER 2
6397 ;          4010    !       WITH THE MASKED DATA PATTERN
6398 ;          4011    !       GENERATED BY THE MACRO 'WRT_MASK".
6399 ;          4012    !       IF THE COMPARE IS NOT EQUAL THEN THE
6400 ;          4013    !       FORMAL PARAMETER 'ERR_FLG" IS
6401 ;          4014    !       ASSIGNED A ONE TO INDICATE THE
6402 ;          4015    !       ERROR.
6403 ;          4016    !
6404 ;          4017    ! FORMAL PARAMETERS:
6405 ;          4018    !
6406 ;          4019    !       TST_PAT
6407 ;          4020    !       DATA PATTERN TO BE MASKED AND
6408 ;          4021    !       COMPARED AGAINST THE CONTENTS
6409 ;          4022    !       OF THE REGISTER UNDER TEST.
6410 ;          4023    !
6411 ;          4024    !       INDEX
6412 ;          4025    !       USED BY THE MACRO WRT_MASK TO
6413 ;          4026    !       SELECT THE CURRENT REGISTERS ADDRESS,
6414 ;          4027    !       FORCED HI, FORCED LO AND DON'T CARE
6415 ;          4028    !       MASK INFOMATION.
6416 ;          4029    !
6417 ;          4030    !       ERR_FLG
6418 ;          4031    !       CONTAINS THE ADDRESS (PASSED BY REF)
6419 ;          4032    !       OF THE CALLERS ERROR FLG TO ENABLE THE
6420 ;          4033    !       CALLER TO EXAMINE THE ERROR STATUS
6421 ;          4034    !       OF THE ROUTINE CALL.
6422 ;          4035    !
6423 ;          4036    ! IMPLICIT INPUTS:
6424 ;          4037    !       WT_DATA
6425 ;          4038    !         GETS LOADED WITH THE GENERATED
6426 ;          4039    !         WRT_MASK DATA PATTERN THUS ALLOWING
6427 ;          4040    !         CALLER TO PRINT FAILING GOOD DATA.
6428 ;          4041    !
6429 ;          4042    !       RD_DATA
6430 ;          4043    !         GETS LOADED WITH DATA READ FROM THE
6431 ;          4044    !         REGISTER THUS ALLOWING CALLER
6432 ;          4045    !         TO PRINT FAILING BAD DATA.
6433 ;          4046    !
6434 ;          4047    ! IMPLICIT OUTPUTS:
6435 ;          4048    !       GLOBAL LOCATION WR_DATA
6436 ;          4049    !       AND RD_DATA LOADED WITH GOOD
6437 ;          4050    !       AND BAD REGISTER DATA
6438 ;          4051    !
6439 ;          4052    !--
6440 ;          4053
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (41)

;          4054      .ERR_FLG = ZERO;                               !CLEAR ERROR FLAG
;          4055      DAT_DM = ONE;                                  !SET DATA DIAG MODE
;          4056      ML_FUNC = write;                               !LOAD WRITE FUNC TO CS1
;          4057      DAT_CLK = ONE;                                 !DO A DATA CLOCK
;          4058
;          4059      if .REG_INIT_FLG IS_SET                        !SEE IF CALLER IS REG INIT TEST
;          4060      then                                           !CLEAR MBUS TO GENERATE INIT DATA
;          4061          begin
;          4062          CLR_MBUS;
;          4063          DAT_DM = ONE;
;          4064          end;
;          4065
;          4066      WT_DATA = WRT_MASK;                            !SAVE THE DATA WRITTEN TO THE REGISTER
;          4067      RD_DATA = .MLD2;                               !READ AND SAVE THE REGISTER
;          4068
;          4069      if .RD_DATA neq .WT_DATA then .ERR_FLG = ONE;       !READ MLD2 FR WRT_MASK PATTERN
;          4070
;          4071                                                     !SET ERROR FLAG IF NEQ
;          4072      CLR_MBUS;                                      !CLR MASS BUSS
;          4073      end;

                                    .SBTTL  RD.D2 ROUTINE DECLARATION SECTION
024416 004167 160116          RD.D2:JSR     R1,$SAVE2                ;                                  4002
024422 005076 000010                CLR     @10(SP)                  ; ERR.FLG                        4054
024426 152777 000010 171070         BISB    #10,@ML.REG+120          ;                                4055
024434 142777 000077 170742         BICB    #77,@ML.REG              ;                                4056
024442 152777 000061 170734         BISB    #61,@ML.REG
024450 152777 000020 171046         BISB    #20,@ML.REG+120          ;                                4057
024456 026727 170676 000001         CMP     REG.INIT.FLG,#1          ;                                4059
024464 001017                       BNE     1$
024466 152777 000040 170750         BISB    #40,@ML.REG+40           ;                                4061
024474 016701 171332                MOV     ML.DUT,R1
024500 042701 177770                BIC     #177770,R1
024504 142777 000007 170732         BICB    #7,@ML.REG+40
024512 150177 170726                BISB    R1,@ML.REG+40
024516 152777 000010 171000         BISB    #10,@ML.REG+120          ;
024524 016600 000012          1$:   MOV     12(SP),R0                ; INDEX,*                        4063
024530 006300                       ASL     R0                                                        4066
024532 006300                       ASL     R0
024534 006300                       ASL     R0
024536 010001                       MOV     R0,R1
024540 016100 015406                MOV     ML.REG+2(R1),R0
024544 056600 000014                BIS     14(SP),R0                ; TST.PAT,*
024550 046100 015410                BIC     ML.REG+4(R1),R0
024554 016102 015412                MOV     ML.REG+6(R1),R2
024560 050002                       BIS     R0,R2
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS
                                        ;                                                              29-Mar-1982 16:21:03    PA:<
 6501 024562  010267  170562                    MOV     R2,WT.DATA
 6502 024566  017767  171012  170556           MOV     @ML.REG+200,RD.DATA
 6503 024574  026767  170552  170546           CMP     RD.DATA,WT.DATA                ;               4067
 6504 024602  001403                           BEQ     2$                             ;               4069
 6505 024604  012776  000001  000010           MOV     #1,@10(SP)                     ; *,ERR.FLG
 6506 024612  152777  000040  170624  2$:      BISB    #40,@ML.REG+40
 6507 024620  016702  171206                    MOV     ML.DUT,R2
 6508 024624  042702  177770                    BIC     #177770,R2
 6509 024630  142777  000007  170606           BICB    #7,@ML.REG+40
 6510 024636  150277  170602                    BISB    R2,@ML.REG+40
 6511 024642  000207                           RTS     PC                             ;               4002
 6512
 6513                                   ; Routine Size:  75 words
 6514                                   ; Maximum stack depth per invocation:  3 words
 6519
 6520
```

```
6522 ;ML4AD
6523 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
6524 ;                                                                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (42)
6525 ;          4074    routine WRT_D3 (TST_PAT, index) : novalue =
6526 ;          4075        begin
6527 ;          4076
6528 ;          4077    !++
6529 ;          4078    ! FUNCTIONAL DESCRIPTION:
6530 ;          4079    !
6531 ;          4080    !       LOADS THE DATA DIAG
6532 ;          4081    !       REGISTER 3 WITH A DATA PATTERN
6533 ;          4082    !       GENERATED BY THE MACRO
6534 ;          4083    !       WRT_MASK.
6535 ;          4084    !
6536 ;          4085    ! FORMAL PARAMETERS:
6537 ;          4086    !
6538 ;          4087    !       TST_PAT
6539 ;          4088    !       CURRENT DATA PATTERN TO BE
6540 ;          4089    !       LOADED IN THE REGISTER.
6541 ;          4090    !
6542 ;          4091    !       INDEX
6543 ;          4092    !       USE BY THE MACRO WRT_MASK
6544 ;          4093    !       TO SELECT THE CURRENT REGISTERS
6545 ;          4094    !       ADDRESS, FORCED HI, FORCED
6546 ;          4095    !       LO AND DON'T CARE MASK
6547 ;          4096    !       INFORMATION.
6548 ;          4097    !
6549 ;          4098    !--
6550 ;          4099
6551 ;          4100        DAT_DM = ONE;                   !SET DATA DIAG MODE
6552 ;          4101        MLE2_MASK = %o'000377';         !MASK OUT ECC CRC WORD BITS
6553 ;          4102        MLE2 = WRT_MASK;                !LOAD MLE2 WITH GENERATED WRT_MASK PATTERN
6554 ;          4103        MLE2_MASK = %o'100300';         !RESTORE MASK
6555 ;          4104        DAT_DM = ZERO;                  !CLEAR DATA DIAG MODE.
6556 ;          4105        end;
6560
6561                                        .SBTTL  WRT.D3 ROUTINE DECLARATION SECTION
6565 024644  004167  157670          WRT.D3: JSR     R1,$SAVE2                       ;                       4074
6566 024650  152777  000010  170646          BISB    #10,@ML.REG+120                 ;                       4100
6567 024656  012767  000377  170706          MOV     #3.7,ML.REG+166                 ;                       4101
6568 024664  016600  000010                  MOV     10(SP),R0                       ; INDEX,*               4102
6569 024670  006300                          ASL     R0
6570 024672  006300                          ASL     R0
6571 024674  006300                          ASL     R0
6572 024676  010001                          MOV     R0,R1
6573 024700  016100  015406                  MOV     ML.REG+2(R1),R0
6574 024704  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
6575 024710  046100  015410                  BIC     ML.REG+4(R1),R0
6576 024714  016102  015412                  MOV     ML.REG+6(R1),R2
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION
                                                                                                29-Mar-1982 16:23:04    TOPS
                                                                                                29-Mar-1982 16:21:03    PA:<
                                                BIS     R0,R2
                                                MOV     R2,@ML.REG+160
                                                MOV     #-77500,ML.REG+166                                              4103
                                                BICB    #10,@ML.REG+120                                                 4104
                                                RTS     PC                                                              4074

                                        ; Routine Size:  32 words
                                        ; Maximum stack depth per invocation:  3 words
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (43)
;
;        routine RD_D3 (TST_PAT, index, FRR_FLG) : novalue =
;            begin
;
;        !++
;        ! FUNCTIONAL DESCRIPTION:
;        !
;        !       COMPARES THE CONTENTS OF THE
;        !       DATA DIAG REGISTER 3
;        !       WITH THE MASKED DATA PATTERN
;        !       GENERATED BY THE MACRO "WRT_MASK".
;        !
;        !       IF THE COMPARE IS NOT EQUAL THE
;        !       FORMAL PARAMETER "ERR_FLG" IS
;        !       ASSIGNED A ONE TO INDICATE THE
;        !       ERROR.
;        !
;        ! FORMAL PARAMETERS:
;        !
;        !       TST_PAT
;        !       DATA PATTERN TO BE MASKED AND
;        !       COMPARED AGAINST THE CONTENTS
;        !       OF THE REGISTER UNDER TEST.
;        !
;        !       INDEX
;        !       USED BY THE MACRO WRT_MASK TO
;        !       SELECT THE CURRENT REGISTERS ADDRESS,
;        !       FORCED HI, FORCED LO AND DON'T CARE
;        !       MASK INFORMATION
;        !
;        !       ERR_FLG
;        !       CONTAINS THE ADDRESS (PASSED BY REF)
;        !       OF THE CALLERS ERROR FLG TO ENABLE THE
;        !       CALLER TO EXAMINE THE ERROR STATUS
;        !       OF THE ROUTINE CALL.
;        !
;        ! IMPLICIT INPUTS:
;        !       WT_DATA
;        !         GETS LOADED WITH THE GENERATED
;        !         WRT_MASK DATA PATTERN THUS ALLOWING
;        !         CALLER TO PRINT FAILING GOOD DATA.
;        !
;        !       RD_DATA
;        !         GETS LOADED WITH DATA READ FROM THE
;        !         REGISTER THUS ALLOWING CALLER
;        !         TO PRINT FAILING BAD DATA.
;        !
;        ! IMPLICIT OUTPUTS:
;        !       GLOBAL LOCATION WR_DATA
;        !       AND RD_DATA LOADED WITH GOOD
;        !       AND BAD REGISTER DATA
;        !
;        !--
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (43)
;
;          4158
;          4159      .ERR_FLG = ZERO;                          !CLEAR ERROR FLAG
;          4160      MLE2_MASK = %o'000377';                   !SET DATA DIAG MODE
;          4161      DAT_DM = ONE;
;          4162      ML_FUNC = write;                          !LOAD WRITE FUNCTION TO MLCS1
;          4163      DAT_CLK = ONE;                            !DO A DATA CLOCK
;          4164
;          4165      if .REG_INIT_FLG IS_SET                   !SEE IF CALLER IS  REG INIT TEST
;          4166      then                                      !CLEAR  MBUS TO GENERATE INIT DATA
;          4167          begin
;          4168          CLR_MBUS;
;          4169          DAT_DM = ONE;
;          4170          end;
;          4171
;          4172      WT_DATA = WRT_MASK;                       !SAVE THE DATA WRITTEN TO THE REGISTER
;          4173      RD_DATA = .MLE2 or .IGNORE;               !READ AND SAVE THE REGISTER
;          4174
;          4175      if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
;          4176
;          4177      MLE2_MASK = %o'100300';
;          4178      CLR_MBUS;                                 !CLEAR THE MASS BUS
;          4179      end;
;
;                                .SBTTL  RD.D3 ROUTINE DECLARATION SECTION
6685 024744  004167  157570      RD.D3:  JSR     R1,$SAVE2                       ;                       4106
6686 024750  005076  000010              CLR     @10(SP)                         ; ERR.FLG               4159
6687 024754  012767  000377  170610      MOV     #377,ML.REG+166                 ;                       4160
6688 024762  152777  000010  170534      BISB    #10,@ML.REG+120                 ;                       4161
6689 024770  142777  000077  170406      BICB    #77,@ML.REG                     ;                       4162
6690 024776  152777  000061  170400      BISB    #61,@ML.REG
6691 025004  152777  000020  170512      BISB    #20,@ML.REG+120                 ;                       4163
6692 025012  026727  170342  000001      CMP     REG.INIT.FLG,#1                 ;                       4165
6693 025020  001017                      BNE     1$
6694 025022  152777  000040  170414      BISB    #40,@ML.REG+40                  ;                       4167
6695 025030  016701  170776              MOV     ML.OUT,R1
6696 025034  042701  177770              BIC     #177770,R1
6697 025040  142777  000007  170376      BICB    #7,@ML.REG+40
6698 025046  150177  170372              BISB    R1,@ML.REG+40
6699 025052  152777  000010  170444      BISB    #10,@ML.REG+120                 ;                       4169
6700 025060  016600  000012      1$:     MOV     12(SP),R0                       ; INDEX,*               4172
6701 025064  006300                      ASL     R0
6702 025066  006300                      ASL     R0
6703 025070  006300                      ASL     R0
6704 025072  010001                      MOV     R0,R1
6705 025074  016100  015406              MOV     ML.REG+2(R1),R0
6706 025100  056600  000014              BIS     14(SP),R0                       ; TST.PAT,*
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS
                                                                                                       29-Mar-1982 16:21:03    PA:<
 6711 025104  046100  015410                    BIC     ML.REG+4(R1),R0
 6712 025110  016102  015412                    MOV     ML.REG+6(R1),R2
 6713 025114  050002                            BIS     R0,R2
 6714 025116  010267  170226                    MOV     R2,WT.DATA
 6715 025122  017702  170436                    MOV     @ML.REG+160,R2                                 ;
 6716 025126  056102  015412                    BIS     ML.REG+6(R1),R2                                ;                                       4173
 6717 025132  010267  170214                    MOV     R2,RD.DATA
 6718 025136  026767  170206  170206            CMP     WT.DATA,RD.DATA                                ;
 6719 025144  001403                            BEQ     2$                                             ;                                       4175
 6720 025146  012776  000001  000010            MOV     #1,@10(SP)                                     ; *,ERR.FLG
 6721 025154  012767  100300  170410  2$:       MOV     #-77500,ML.REG+166                             ;
 6722 025162  152777  000040  170254            BISB    #40,@ML.REG+40                                 ;                                       4177
 6723 025170  016702  170636                    MOV     ML.DUT,R2
 6724 025174  042702  177770                    BIC     #177770,R2
 6725 025200  142777  000007  170236            BICB    #7,@ML.REG+40
 6726 025206  150277  170232                    BISB    R2,@ML.REG+40
 6727 025212  000207                            RTS     PC                                             ;                                       4106
 6728
 6729                                   ; Routine Size:  84 words
 6730                                   ; Maximum stack depth per invocation:  3 words
 6735
 6736
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (44)

;          4180  routine WRT_DS (TST_PAT, index) : novalue =
;          4181      begin
;          4182
;          4183  !++
;          4184  ! FUNCTIONAL DESCRIPTION:
;          4185  !       DUMMY ROUTINE CALL TO ASSIST IN THE READ
;          4186  !       WRITE REGISTER ALGORITHEM
;          4187  !
;          4188  ! FORMAL PARAMETERS:
;          4189  !       TST_PAT
;          4190  !       DATA PATTERN TO BE MASKED AND
;          4191  !       COMPARED AGAINST THE CONTENTS
;          4192  !       OF THE REGISTER UNDER TEST
;          4193  !
;          4194  !       INDEX
;          4195  !       USED BY THE MACRO WRT_MASK TO
;          4196  !       SELECT THE CURRENT REGISTERS ADDRESS,
;          4197  !       FORCED HI, FORCED LO AND DON'T CARE
;          4198  !       MASK INFORMATION.
;          4199  !
;          4200  !--
;          4201
;          4202  !
;          4203  !  DRIVE STATUS REG IS READ ONLY
;          4204      return;
;          4205      end;
;
;
;                                .SBTTL  WRT.DS ROUTINE DECLARATION SECTION
6775 025214  000207  WRT.DS: RTS     PC                                      ;  4180
;
;                  ; Routine Size:  1 word
;                  ; Maximum stack depth per invocation:  0 words
;
;
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (45)
;
;          4206  routine RD_DS (TST_PAT, index, ERR_FLG) : novalue =
;          4207      begin
;          4208
;          4209  !++
;          4210  ! FUNCTIONAL DESCRIPTION:
;          4211  !
;          4212  !       COMPARES THE CONTENTS OF THE
;          4213  !       DRIVE STATUS REGISTER WITH THE
;          4214  !       MASKED DATA PATTERN
;          4215  !       GENERATED BY THE MACRO 'WRT_MASK'.
;          4216  !
;          4217  !       IF THE COMPARE IS NOT EQUAL THEN
;          4218  !       THE FORMAL PARAMETER 'ERR_FLG' IS
;          4219  !       ASSIGNED A ONE TO INDICATE THE
;          4220  !       ERROR.
;          4221  !
;          4222  ! FORMAL PARAMETERS:
;          4223  !
;          4224  !       TST_PAT
;          4225  !       DATA PATTERN TO BE MASKED AND
;          4226  !       COMPARED AGAINST THE CONTENTS
;          4227  !       OF THE REGISTER UNDER TEST.
;          4228  !
;          4229  !       INDEX
;          4230  !       USED BY THE MACRO WRT_MASK TO
;          4231  !       SELECT THE CURRENT REGISTERS ADDRESS.
;          4232  !       FORCED HI, FORCED LO AND DON'T CARE
;          4233  !       MASK INFORMATION.
;          4234  !
;          4235  !       ERR_FLG
;          4236  !       CONTAINS THE ADDRESS (PASSED BY REF)
;          4237  !       OF THE CALLERS ERROR FLG TO ENABLE THE
;          4238  !       CALLER TO EXAMINE THE ERROR STATUS
;          4239  !       OF THE ROUTINE CALL.
;          4240  !
;          4241  ! IMPLICIT INPUTS:
;          4242  !       WT_DATA
;          4243  !         GETS LOADED WITH THE GENERATED
;          4244  !         WRT_MASK DATA PATTERN THUS ALLOWING
;          4245  !         CALLER TO PRINT FAILING GOOD DATA.
;          4246  !
;          4247  !       RD_DATA
;          4248  !         GETS LOADED WITH DATA READ FROM THE
;          4249  !         REGISTER THUS ALLOWING CALLER
;          4250  !         TO PRINT FAILING BAD DATA.
;          4251  !
;          4252  ! IMPLICIT OUTPUTS:
;          4253  !       GLOBAL LOCATION WR_DATA
;          4254  !       AND RD_DATA LOADED WITH GOOD
;          4255  !       AND BAD REGISTER DATA
;          4256  !
;          4257  !--
```

```
6842 ;ML4AD
6843 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
6844                                                                    29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (45)
6845 ;          4258
6846 ;          4259       .ERR_FLG = ZERO;                            !CLEAR THE ERROR FLAG
6847 ;          4260       WT_DATA = WRT_MASK;                         !SAVE THE DATA WRITTEN TO THE REGISTER
6848 ;          4261       RD_DATA = .MLDS or .IGNORE;                 !READ AND SAVE THE REGISTER
6849 ;          4262
6850 ;          4263       if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
6851 ;          4264
6852 ;          4265                                                   !SET ERROR FLAG IF NEQ
6853 ;          4266       end;
6857
6858                                        .SBTTL  RD.DS ROUTINE DECLARATION SECTION
6862 025216  004167  157316          RD.DS: JSR     R1,$SAVE2                    ;                                    4206
6863 025222  005076  000010                 CLR     @10(SP)                      ; ERR.FLG                            4259
6864 025226  016600  000012                 MOV     12(SP),R0                    ; INDEX,*                            4260
6865 025232  006300                         ASL     R0
6866 025234  006300                         ASL     R0
6867 025236  006300                         ASL     R0
6868 025240  010001                         MOV     R0,R1
6869 025242  016100  015406                 MOV     ML.REG+2(R1),R0
6870 025246  056600  000014                 BIS     14(SP),R0                    ; TST.PAT,*
6871 025252  046100  015410                 BIC     ML.REG+4(R1),R0
6872 025256  016102  015412                 MOV     ML.REG+6(R1),R2
6873 025262  050002                         BIS     R0,R2
6874 025264  010267  170060                 MOV     R2,WT.DATA
6875 025270  017702  170160                 MOV     @ML.REG+50,R2                ;                                    4261
6876 025274  056102  015412                 BIS     ML.REG+6(R1),R2
6877 025300  010267  170046                 MOV     R2,RD.DATA
6878 025304  026767  170040  170040        CMP     WT.DATA,RD.DATA              ;                                    4263
6879 025312  001403                         BEQ     1$
6880 025314  012776  000001  000010        MOV     #1,@10(SP)                   ; *,ERR.FLG
6881 025322  000207                 1$:     RTS     PC                           ;                                    4206
6882
6883                                        ; Routine Size:  35 words
6884                                        ; Maximum stack depth per invocation:  3 words
6889
6890
```

```
;ML4AD
;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (46)

;          4267  routine WRT_REG (TST_PAT, REG_SEL, index) : novalue =
;          4268      begin
;          4269
;          4270  !++
;          4271  ! FUNCTIONAL DESCRIPTION:
;          4272  !       A ROUTINE TO SELECTIVELY
;          4273  !       CALLED ROUTINES WHICH
;          4274  !       WRITE TO INDIVIDUAL ML11
;          4275  !       REGISTERS
;          4276  !
;          4277  ! FORMAL PARAMETERS:
;          4278  !       REG_SEL
;          4279  !       CASE SELECT EXPRESSION TO
;          4280  !       SELECT THE WRITE REGISTER
;          4281  !       ROUTINE TO CALLED
;          4282  !
;          4283  !       TST_PAT
;          4284  !       DATA PATTERN WHICH THE SELECTED
;          4285  !       REGISTER WILL BE TESTED AGAINST
;          4286  !
;          4287  !       INDEX
;          4288  !       LOADED WITH THE ML_REG INDEX
;          4289  !       SELECT NUMBER OF THE REGISTER
;          4290  !       BEING TESTED
;          4291  !
;          4292  ! SIDE EFFECTS:
;          4293  !       WHEN A WRITE REGISTER ROUTINE IS CALLED
;          4294  !       THE VARIABLE 'INDEX' FROM THE CALLING
;          4295  !       TEST IS LOADED WITH THE REGISTERS
;          4296  !       ML_REG INDEX NUMBER.
;          4297  !
;          4298  !       THIS ENABLES THE CALLING TEST TO FIND
;          4299  !       THE FAILING REGISTER ADDRESS.
;          4300  !
;          4301  !--
;          4302
;          4303      case .REG_SEL from 0 to 13 of                !SELECT THE WRITE REGISTER ROUTINE CALL
;          4304          set
;          4305
;          4306          [0] :
;          4307              WRT_CS1 (.TST_PAT, .index = 0);      !CALL ROUTINE TO LOAD MLCS1
;          4308
;          4309          [1] :
;          4310              WRT_ER (.TST_PAT, .index = 6);       !CALL ROUTINE TO LOAD MLER
;          4311
;          4312          [2] :
;          4313              WRT_DA (.TST_PAT, .index = 3);       !CALL ROUTINE TO LOAD MLDA
;          4314
;          4315          [3] :
;          4316              WRT_MR (.TST_PAT, .index = 10);      !CALL ROUTINE TO LOAD MLMR
;          4317
;          4318          [4] :
```

```
6948 ;ML4AD
6949 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
6950 ;                                                                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (46)
6951 ;          4319            WRT_E1 (.TST_PAT, .index = 13);     !CALL ROUTINE TO LOAD MLE1
6952 ;          4320
6953 ;          4321        [5] :
6954 ;          4322            WRT_E2 (.TST_PAT, .index = 14);     !CALL ROUTINE TO LOAD MLE2
6955 ;          4323
6956 ;          4324        [6] :
6957 ;          4325            WRT_PA (.TST_PAT, .index = 8);      !CALL ROUTINE TO LOAD MLPA
6958 ;          4326
6959 ;          4327        [7] :
6960 ;          4328            WRT_PD (.TST_PAT, .index = 19);     !CALL ROUTINE TO LOAD MLPD
6961 ;          4329
6962 ;          4330        [8] :
6963 ;          4331            WRT_EE (.TST_PAT, .index = 17);     !CALL ROUTINE TO LOAD MLEE
6964 ;          4332
6965 ;          4333        [9] :
6966 ;          4334            WRT_EL (.TST_PAT, .index = 18);     !CALL ROUTINE TO LOAD MLEL
6967 ;          4335
6968 ;          4336       [10] :
6969 ;          4337            WRT_DS (.TST_PAT, .index = 5);      !CALL ROUTINE TO LOAD MLDS
6970 ;          4338
6971 ;          4339       [11] :
6972 ;          4340            WRT_D1 (.TST_PAT, .index = 15);     !CALL ROUTINE TO LOAD MLD1
6973 ;          4341
6974 ;          4342       [12] :
6975 ;          4343            WRT_D2 (.TST_PAT, .index = 16);     !CALL ROUTINE TO LOAD MLD2
6976 ;          4344
6977 ;          4345       [13] :
6978 ;          4346            WRT_D3 (.TST_PAT, .index = 14);     !CALL ROUTINE TO LOAD MLE2
6979 ;          4347            tes;
6980 ;          4348
6981 ;          4349        end;
6985
6986                                    .SBTTL  WRT.REG ROUTINE DECLARATION SECTION
6990 025324 004167 157210      WRT.REG:JSR     R1,$SAVE2                       ;
6991 025330 016600 000010              MOV     10(SP),R0                       ; INDEX,*          4267
6992 025334 016601 000014              MOV     14(SP),R1                       ; TST.PAT,*        4307
6993 025340 016602 000012              MOV     12(SP),R2                       ; REG.SEL,*
6994 025344 006302                     ASL     R2                                                 4303
6995 025346 066207 025352              ADD     1$(R2),PC
6996 025352 000034             1$:     .WORD   2$-1$
6997 025354 000050                     .WORD   3$-1$
6998 025356 000066                     .WORD   4$-1$
6999 025360 000104                     .WORD   5$-1$
7000 025362 000122                     .WORD   6$-1$
7001 025364 000140                     .WORD   7$-1$
7002 025366 000156                     .WORD   8$-1$
```

SEQ 0168

```
                                        ;ML4AD
                                        ;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS
                                                                                                        29-Mar-1982 16:21:03    PA:<
 7004
 7005
 7006
 7007 025370  000174                            .WORD   9$-1$
 7008 025372  000212                            .WORD   10$-1$
 7009 025374  000230                            .WORD   11$-1$
 7010 025376  000246                            .WORD   12$-1$
 7011 025400  000264                            .WORD   13$-1$
 7012 025402  000302                            .WORD   14$-1$
 7013 025404  000320                            .WORD   15$-1$
 7014 025406  010146                    2$:     MOV     R1,-(SP)                ;                       4307
 7015 025410  005010                            CLR     (R0)
 7016 025412  005046                            CLR     -(SP)
 7017 025414  004767  174272                    JSR     PC,WRT.CS1
 7018 025420  000532                            BR      16$
 7019 025422  010146                    3$:     MOV     R1,-(SP)                ;                       4303
 7020 025424  012710  000006                    MOV     #6,(R0)                 ;                       4310
 7021 025430  011046                            MOV     (R0),-(SP)
 7022 025432  004767  174432                    JSR     PC,WRT.ER
 7023 025436  000523                            BR      16$
 7024 025440  010146                    4$:     MOV     R1,-(SP)                ;                       4303
 7025 025442  012710  000003                    MOV     #3,(R0)                 ;                       4313
 7026 025446  011046                            MOV     (R0),-(SP)
 7027 025450  004767  174572                    JSR     PC,WRT.DA
 7028 025454  000514                            BR      16$
 7029 025456  010146                    5$:     MOV     R1,-(SP)                ;                       4303
 7030 025460  012710  000012                    MOV     #12,(R0)                ;                       4316
 7031 025464  011046                            MOV     (R0),-(SP)
 7032 025466  004767  174732                    JSR     PC,WRT.MR
 7033 025472  000505                            BR      16$
 7034 025474  010146                    6$:     MOV     R1,-(SP)                ;                       4303
 7035 025476  012710  000015                    MOV     #15,(R0)                ;                       4319
 7036 025502  011046                            MOV     (R0),-(SP)
 7037 025504  004767  175300                    JSR     PC,WRT.E1
 7038 025510  000476                            BR      16$
 7039 025512  010146                    7$:     MOV     R1,-(SP)                ;                       4303
 7040 025514  012710  000016                    MOV     #16,(R0)                ;                       4322
 7041 025520  011046                            MOV     (R0),-(SP)
 7042 025522  004767  175470                    JSR     PC,WRT.E2
 7043 025526  000467                            BR      16$
 7044 025530  010146                    8$:     MOV     R1,-(SP)                ;                       4303
 7045 025532  012710  000010                    MOV     #10,(R0)                ;                       4325
 7046 025536  011046                            MOV     (R0),-(SP)
 7047 025540  004767  175036                    JSR     PC,WRT.PA
 7048 025544  000460                            BR      16$
 7049 025546  010146                    9$:     MOV     R1,-(SP)                ;                       4303
 7050 025550  012710  000023                    MOV     #23,(R0)                ;                       4328
 7051 025554  011046                            MOV     (R0),-(SP)
 7052 025556  004767  175666                    JSR     PC,WRT.PD
 7053 025562  000451                            BR      16$
 7054 025564  010146                    10$:    MOV     R1,-(SP)                ;                       4303
 7055 025566  012710  000021                    MOV     #21,(R0)                ;                       4331
 7056 025572  011046                            MOV     (R0),-(SP)
 7057 025574  004767  176156                    JSR     PC,WRT.EE
 7058 025600  000442                            BR      16$                     ;                       4303
```

```
                              ;ML4AD                                                       29-Mar-1982 16:23:04   TOPS
                              ;                 ROUTINE DECLARATION SECTION                29-Mar-1982 16:21:03   PA:<
025602  010146                11$:    MOV     R1,-(SP)                 ;
025604  012710  000022                MOV     #22,(R0)                 ;                                          4334
025610  011046                        MOV     (R0),-(SP)
025612  004767  176076                JSR     PC,WRT.EL
025616  000433                        BR      16$
025620  010146                12$:    MOV     R1,-(SP)                 ;                                          4303
025622  012710  000005                MOV     #5,(R0)                  ;                                          4337
025626  011046                        MOV     (R0),-(SP)
025630  004767  177360                JSR     PC,WRT.DS
025634  000424                        BR      16$
025636  010146                13$:    MOV     R1,-(SP)                 ;                                          4303
025640  012710  000017                MOV     #17,(R0)                 ;                                          4340
025644  011046                        MOV     (R0),-(SP)
025646  004767  176146                JSR     PC,WRT.D1
025652  000415                        BR      16$
025654  010146                14$:    MOV     R1,-(SP)                 ;                                          4303
025656  012710  000020                MOV     #20,(R0)                 ;                                          4343
025662  011046                        MOV     (R0),-(SP)
025664  004767  176442                JSR     PC,WRT.D2
025670  000406                        BR      16$
025672  010146                15$:    MOV     R1,-(SP)                 ;                                          4303
025674  012710  000016                MOV     #16,(R0)                 ;                                          4346
025700  011046                        MOV     (R0),-(SP)
025702  004767  176736                JSR     PC,WRT.D3
025706  022626                16$:    CMP     (SP)+,(SP)+
025710  000207                        RTS     PC                       ;                                          4268
                                                                       ;                                          4267

                              ; Routine Size:  123 words
                              ; Maximum stack depth per invocation:  5 words
```

```
7099 ;ML4AD
7100 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
7101 ;                                                                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (47)
7102 ;          4350    routine RD_REG (TST_PAT, REG_SEL, ERR_FLG) : novalue =
7103 ;          4351        begin
7104 ;          4352
7105 ;          4353    !++
7106 ;          4354    ! FUNCTIONAL DESCRIPTION:
7107 ;          4355    !       A ROUTINE TO SELECTIVELY
7108 ;          4356    !       CALLED ROUTINES WHICH
7109 ;          4357    !       READ TO INDIVIDUAL ML11
7110 ;          4358    !       REGISTERS.
7111 ;          4359    !
7112 ;          4360    ! FORMAL PARAMETERS:
7113 ;          4361    !       ERR_FLG
7114 ;          4362    !       CONTAINS THE ADDRESS (PASSED BY REF)
7115 ;          4363    !       OF THE CALLERS ERROR FLG TO ENABLE THE
7116 ;          4364    !       CALLER TO EXAMINE THE ERROR STATUS
7117 ;          4365    !       OF THE ROUTINE CALL.
7118 ;          4366    !
7119 ;          4367    !       REG_SEL
7120 ;          4368    !       CASE SELECT EXPRESSION TO
7121 ;          4369    !       SELECT THE WRITE REGISTER
7122 ;          4370    !       ROUTINE TO CALLED
7123 ;          4371    !
7124 ;          4372    !       TST_PAT
7125 ;          4373    !       DATA PATTERN WHICH THE SELECTED
7126 ;          4374    !       REGISTER WILL BE TESTED AGAINST
7127 ;          4375    !
7128 ;          4376    !--
7129 ;          4377
7130 ;          4378        case .REG_SEL from 0 to 13 of              !SELECT THE READ REGISTER ROUTINE CALL
7131 ;          4379            set
7132 ;          4380
7133 ;          4381            [0] :
7134 ;          4382                RD_CS1 (.TST_PAT, 0, .ERR_FLG);    !CALL ROOUTINE TO READ MLCS1
7135 ;          4383
7136 ;          4384            [1] :
7137 ;          4385                RD_ER (.TST_PAT, 6, .ERR_FLG);     !CALL ROUTINE TO READ MLER
7138 ;          4386
7139 ;          4387            [2] :
7140 ;          4388                RD_DA (.TST_PAT, 3, .ERR_FLG);     !CALL ROUTINE TO READ MLDA
7141 ;          4389
7142 ;          4390            [3] :
7143 ;          4391                RD_MR (.TST_PAT, 10, .ERR_FLG);    !CALL ROUTINE TO READ MLMR
7144 ;          4392
7145 ;          4393            [4] :
7146 ;          4394                RD_E1 (.TST_PAT, 13, .ERR_FLG);    !CALL ROUTINE TO READ MLE1
7147 ;          4395
7148 ;          4396            [5] :
7149 ;          4397                RD_E2 (.TST_PAT, 14, .ERR_FLG);    !CALL ROUTINE TO READ MLE2
7150 ;          4398
7151 ;          4399            [6] :
7152 ;          4400                RD_PA (.TST_PAT, 8, .ERR_FLG);     !CALL ROUTINE TO READ MLPA
7153 ;          4401
```

```
 7155 ;ML4AD
 7156 ;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 7157                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (47)
 7158 ;          4402          [7] :
 7159 ;          4403              RD_PD (.TST_PAT, 19, .ERR_FLG);     !CALL ROUTINE TO READ MLPD
 7160 ;          4404
 7161 ;          4405          [8] :
 7162 ;          4406              RD_EE (.TST_PAT, 17, .ERR_FLG);     !CALL ROUTINE TO READ MLEE
 7163 ;          4407
 7164 ;          4408          [9] :
 7165 ;          4409              RD_EL (.TST_PAT, 18, .ERR_FLG);     !CALL ROUTINE TO READ MLEL
 7166 ;          4410
 7167 ;          4411          [10] :
 7168 ;          4412              RD_DS (.TST_PAT, 5, .ERR_FLG);      !CALL ROUTINE TO READ MLDS
 7169 ;          4413
 7170 ;          4414          [11] :
 7171 ;          4415              RD_D1 (.TST_PAT, 15, .ERR_FLG);     !CALL ROUTINE TO READ MLD1
 7172 ;          4416
 7173 ;          4417          [12] :
 7174 ;          4418              RD_D2 (.TST_PAT, 16, .ERR_FLG);     !CALL ROUTINE TO READ MLD2
 7175 ;          4419
 7176 ;          4420          [13] :
 7177 ;          4421              RD_D3 (.TST_PAT, 14, .ERR_FLG);     !CALL ROUTINE TO READ MLE2
 7178 ;          4422              tes;
 7179 ;          4423
 7180 ;          4424      end;
 7184
 7185                                        .SBTTL  RD.REG ROUTINE DECLARATION SECTION
 7189 025712  004167  156622          RD.REG: JSR     R1,$SAVE2                       ;                                    4350
 7190 025716  016600  000010                  MOV     10(SP),R0                       ; ERR.FLG,*                          4382
 7191 025722  016601  000014                  MOV     14(SP),R1                       ; TST.PAT,*
 7192 025726  016602  000012                  MOV     12(SP),R2                       ; REG.SEL,*                          4378
 7193 025732  006302                          ASL     R2
 7194 025734  066207  025740                  ADD     1$(R2),PC
 7195 025740  000034                    1$:   .WORD   2$-1$
 7196 025742  000050                          .WORD   3$-1$
 7197 025744  000066                          .WORD   4$-1$
 7198 025746  000104                          .WORD   5$-1$
 7199 025750  000122                          .WORD   6$-1$
 7200 025752  000140                          .WORD   7$-1$
 7201 025754  000156                          .WORD   8$-1$
 7202 025756  000174                          .WORD   9$-1$
 7203 025760  000212                          .WORD   10$-1$
 7204 025762  000230                          .WORD   11$-1$
 7205 025764  000246                          .WORD   12$-1$
 7206 025766  000264                          .WORD   13$-1$
 7207 025770  000302                          .WORD   14$-1$
 7208 025772  000320                          .WORD   15$-1$
 7209 025774  010146                    2$:   MOV     R1,-(SP)                        ;                                    4382
```

SEQ 0172

```
7211                                    ;ML4AD
7212                                    ;
7213                                    ;       ROUTINE DECLARATION SECTION
7214 025776  005046                             CLR     -(SP)
7215 026000  010046                             MOV     R0,-(SP)
7216 026002  004767  173754                     JSR     PC,RD.CS1
7217 026006  000532                             BR      16$
7218 026010  010146                     3$:     MOV     R1,-(SP)
7219 026012  012746  000006                     MOV     #6,-(SP)
7220 026016  010046                             MOV     R0,-(SP)
7221 026020  004767  174114                     JSR     PC,RD.ER
7222 026024  000523                             BR      16$
7223 026026  010146                     4$:     MOV     R1,-(SP)
7224 026030  012746  000003                     MOV     #3,-(SP)
7225 026034  010046                             MOV     R0,-(SP)
7226 026036  004767  174254                     JSR     PC,RD.DA
7227 026042  000514                             BR      16$
7228 026044  010146                     5$:     MOV     R1,-(SP)
7229 026046  012746  000012                     MOV     #12,-(SP)
7230 026052  010046                             MOV     R0,-(SP)
7231 026054  004767  174414                     JSR     PC,RD.MR
7232 026060  000505                             BR      16$
7233 026062  010146                     6$:     MOV     R1,-(SP)
7234 026064  012746  000015                     MOV     #15,-(SP)
7235 026070  010046                             MOV     R0,-(SP)
7236 026072  004767  174776                     JSR     PC,RD.E1
7237 026076  000476                             BR      16$
7238 026100  010146                     7$:     MOV     R1,-(SP)
7239 026102  012746  000016                     MOV     #16,-(SP)
7240 026106  010046                             MOV     R0,-(SP)
7241 026110  004767  175200                     JSR     PC,RD.E2
7242 026114  000467                             BR      16$
7243 026116  010146                     8$:     MOV     R1,-(SP)
7244 026120  012746  000010                     MOV     #10,-(SP)
7245 026124  010046                             MOV     R0,-(SP)
7246 026126  004767  174534                     JSR     PC,RD.PA
7247 026132  000460                             BR      16$
7248 026134  010146                     9$:     MOV     R1,-(SP)
7249 026136  012746  000023                     MOV     #23,-(SP)
7250 026142  010046                             MOV     R0,-(SP)
7251 026144  004767  175406                     JSR     PC,RD.PD
7252 026150  000451                             BR      16$
7253 026152  010146                     10$:    MOV     R1,-(SP)
7254 026154  012746  000021                     MOV     #21,-(SP)
7255 026160  010046                             MOV     R0,-(SP)
7256 026162  004767  175572                     JSR     PC,RD.EE
7257 026166  000442                             BR      16$
7258 026170  010146                     11$:    MOV     R1,-(SP)
7259 026172  012746  000022                     MOV     #22,-(SP)
7260 026176  010046                             MOV     R0,-(SP)
7261 026200  004767  175512                     JSR     PC,RD.EL
7262 026204  000433                             BR      16$
7263 026206  010146                     12$:    MOV     R1,-(SP)
7264 026210  012746  000005                     MOV     #5,-(SP)
7265 026214  010046                             MOV     R0,-(SP)
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

4378 4385 4378 4388 4378 4391 4378 4394 4378 4397 4378 4400 4378 4403 4378 4406 4378 4409 4378 4412

```
7267                                    ;ML4AD
7268                                    ;                ROUTINE DECLARATION SECTION
7269
7270 026216  004767  176774                      JSR     PC,RD.DS
7271 026222  000424                              BR      16$
7272 026224  010146                      13$:    MOV     R1,-(SP)
7273 026226  012746  000017                      MOV     #17,-(SP)
7274 026232  010046                              MOV     R0,-(SP)
7275 026234  004767  175644                      JSR     PC,RD.D1
7276 026240  000415                              BR      16$
7277 026242  010146                      14$:    MOV     R1,-(SP)
7278 026244  012746  000020                      MOV     #20,-(SP)
7279 026250  010046                              MOV     R0,-(SP)
7280 026252  004767  176140                      JSR     PC,RD.D2
7281 026256  000406                              BR      16$
7282 026260  010146                      15$:    MOV     R1,-(SP)
7283 026262  012746  000016                      MOV     #16,-(SP)
7284 026266  010046                              MOV     R0,-(SP)
7285 026270  004767  176450                      JSR     PC,RD.D3
7286 026274  062706  000006              16$:    ADD     #6,SP
7287 026300  000207                              RTS     PC
7288
7289                                    ; Routine Size:  124 words
7290                                    ; Maximum stack depth per invocation:  6 words
7295
7296
7297 ;          4425  !<BLF/PAGE>
```

```
;ML4AD
;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (48)
;          4426                                                 !
;          4427  !
;          4428  BGNMSG (DUMPER);

                                        .SBTTL  DUMPER ROUTINE DECLARATION SECTION
   302  004767  000004          DUMPER::JSR     PC,MSDUMPER                     ;                              4428
026306  104423                          TRAP    23
026310  000207                          RTS     PC

                                ; Routine Size:  4 words
                                ; Maximum stack depth per invocation:  0 words

;          4429
;          4430  !++
;          4431  !       FUNCTIONAL DESCRIPTION
;          4432  !               UPON COMPLETION OF ERROR MESSAGES
;          4433  !               DUMP OUT ALL PERTINENT DRIVE AND
;          4434  !               RH REGISTERS
;          4435  !--
;          4436
;          4437  if .REGDMP
;          4438  then
;          4439      begin
;          4440      PRINTB (ONE_FMT, PHR_14);                 !PRINT REGISTER DUMP MESSAGE
;          4441      PRINTB (FMT_23);                          !PRINT DUMPER COLUMN HEADINGS
;          4442      PRINTB (FMT_24, REG_1, MLCS1, .MLCS1);    !PRINT OUT THE RH & ML11 REGISTER CONTENTS
;          4443      PRINTB (FMT_24, REG_18, MLWC, .MLWC);
;          4444      PRINTB (FMT_24, REG_19, MLBA, .MLBA);
;          4445      PRINTB (FMT_24, REG_6, MLDA, .MLDA);
;          4446      PRINTB (FMT_24, REG_17, MLCS2, .MLCS2);
;          4447      PRINTB (FMT_24, REG_2, MLDS, .MLDS);
;          4448      PRINTB (FMT_24, REG_3, MLER, .MLER);
;          4449      PRINTB (FMT_24, REG_5, MLAS, .MLAS);
;          4450      PRINTB (FMT_24, REG_4, MLMR, .MLMR);
;          4451      PRINTB (FMT_24, REG_7, MLDT, .MLDT);
;          4452      PRINTB (FMT_24, REG_9, MLSN, .MLSN);
;          4453      PRINTB (FMT_24, REG_14, MLEE, .MLEE);
;          4454      PRINTB (FMT_24, REG_15, MLEL, .MLEL);
;          4455
;          4456      if .LST_DUT_REG eql 21                    !SEE IF THIS IS A RH70
;          4457      then                                      !IF YES THEN PRINT RH70 REGISTERS
```

```
7355 ;ML4AD
7356 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
7357                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (48)
7358 ;          4458           begin
7359 ;          4459           PRINTB (FMT_24, REG_20, MLBAE, .MLBAE);
7360 ;          4460           PRINTB (FMT_24, REG_21, MLCS3, .MLCS3);
7361 ;          4461           end;
7362 ;          4462
7363 ;          4463       end;
7364 ;          4464
7365 ;          4465   ENDMSG;
7369
7370                                        .SBTTL  MSDUMPER ROUTINE DECLARATION SECTION
7374 026312                          MSDUMPER:
7375 026312  032767  000001  154064          BIT     #1,REGDMP                       ;                       4437
7376 026320  001001                          BNE     1$
7377 026322  000207                          RTS     PC
7378 026324  012746  012104          1$:     MOV     #PHR.14,-(SP)                   ;                       4440
7379 026330  012746  010240                  MOV     #ONE.FMT,-(SP)
7380 026334  012746  000002                  MOV     #2,-(SP)
7381 026340  010600                          MOV     SP,R0                           ; SP,*
7382 026342  104414                          TRAP    14
7383 026344  012716  010072                  MOV     #FMT.23,(SP)                    ;                       4441
7384 026350  012746  000C01                  MOV     #1,-(SP)
7385 026354  010600                          MOV     SP,R0                           ; SP,*
7386 026356  104414                          TRAP    14
7387 026360  017716  167020                  MOV     @ML.REG,(SP)                    ;                       4442
7388 026364  016746  167014                  MOV     ML.REG,-(SP)
7389 026370  012746  012500                  MOV     #REG.1,-(SP)
7390 026374  012746  010142                  MOV     #FMT.24,-(SP)
7391 026400  012746  000004                  MOV     #4,-(SP)
7392 026404  010600                          MOV     SP,R0                           ; SP,*
7393 026406  104414                          TRAP    14
7394 026410  017716  167000                  MOV     @ML.REG+10,(SP)                 ;                       4443
7395 026414  016746  166774                  MOV     ML.REG+10,-(SP)
7396 026420  012746  012652                  MOV     #REG.18,-(SP)
7397 026424  012746  010142                  MOV     #FMT.24,-(SP)
7398 026430  012746  000004                  MOV     #4,-(SP)
7399 026434  010600                          MOV     SP,R0                           ; SP,*
7400 026436  104414                          TRAP    14
7401 026440  017716  166760                  MOV     @ML.REG+20,(SP)                 ;                       4444
7402 026444  016746  166754                  MOV     ML.REG+20,-(SP)
7403 026450  012746  012660                  MOV     #REG.19,-(SP)
7404 026454  012746  010142                  MOV     #FMT.24,-(SP)
7405 026460  012746  000004                  MOV     #4,-(SP)
7406 026464  010600                          MOV     SP,R0                           ; SP,*
7407 026466  104414                          TRAP    14
7408 026470  017716  166740                  MOV     @ML.REG+30,(SP)                 ;                       4445
7409 026474  016746  166734                  MOV     ML.REG+30,-(SP)
```

```
7411                                    ;ML4AD
7412                                    ;               ROUTINE DECLARATION SECTION
7413
7414 026500  012746  012540                    MOV     #REG.6,-(SP)
7415 026504  012746  010142                    MOV     #FMT.24,-(SP)
7416 026510  012746  000004                    MOV     #4,-(SP)
7417 026514  010600                            MOV     SP,R0                   ; SP,*
7418 026516  104414                            TRAP    14
7419 026520  017716  166720                    MOV     @ML.REG+40,(SP)         ;                       4446
7420 026524  016746  166714                    MOV     ML.REG+40,-(SP)
7421 026530  012746  012642                    MOV     #REG.17,-(SP)
7422 026534  012746  010142                    MOV     #FMT.24,-(SP)
7423 026540  012746  000004                    MOV     #4,-(SP)
7424 026544  010600                            MOV     SP,R0                   ; SP,*
7425 026546  104414                            TRAP    14
7426 026550  017716  166700                    MOV     @ML.REG+50,(SP)         ;                       4447
7427 026554  016746  166674                    MOV     ML.REG+50,-(SP)
7428 026560  012746  012510                    MOV     #REG.2,-(SP)
7429 026564  012746  010142                    MOV     #FMT.24,-(SP)
7430 026570  012746  000004                    MOV     #4,-(SP)
7431 026574  010600                            MOV     SP,R0                   ; SP,*
7432 026576  104414                            TRAP    14
7433 026600  017716  166660                    MOV     @ML.REG+60,(SP)         ;                       4448
7434 026604  016746  166654                    MOV     ML.REG+60,-(SP)
7435 026610  012746  012516                    MOV     #REG.3,-(SP)
7436 026614  012746  010142                    MOV     #FMT.24,-(SP)
7437 026620  012746  000004                    MOV     #4,-(SP)
7438 026624  010600                            MOV     SP,R0                   ; SP,*
7439 026626  104414                            TRAP    14
7440 026630  017716  166640                    MOV     @ML.REG+70,(SP)         ;                       4449
7441 026634  016746  166634                    MOV     ML.REG+70,-(SP)
7442 026640  012746  012532                    MOV     #REG.5,-(SP)
7443 026644  012746  010142                    MOV     #FMT.24,-(SP)
7444 026650  012746  000004                    MOV     #4,-(SP)
7445 026654  010600                            MOV     SP,R0                   ; SP,*
7446 026656  104414                            TRAP    14
7447 026660  017716  166640                    MOV     @ML.REG+120,(SP)        ;                       4450
7448 026664  016746  166634                    MOV     ML.REG+120,-(SP)
7449 026670  012746  012524                    MOV     #REG.4,-(SP)
7450 026674  012746  010142                    MOV     #FMT.24,-(SP)
7451 026700  012746  000004                    MOV     #4,-(SP)
7452 026704  010600                            MOV     SP,R0                   ; SP,*
7453 026706  104414                            TRAP    14
7454 026710  017716  166620                    MOV     @ML.REG+130,(SP)        ;                       4451
7455 026714  016746  166614                    MOV     ML.REG+130,-(SP)
7456 026720  012746  012546                    MOV     #REG.7,-(SP)
7457 026724  012746  010142                    MOV     #FMT.24,-(SP)
7458 026730  012746  000004                    MOV     #4,-(SP)
7459 026734  010600                            MOV     SP,R0                   ; SP,*
7460 026736  104414                            TRAP    14
7461 026740  017716  166600                    MOV     @ML.REG+140,(SP)        ;                       4452
7462 026744  016746  166574                    MOV     ML.REG+140,-(SP)
7463 026750  012746  012562                    MOV     #REG.9,-(SP)
7464 026754  012746  010142                    MOV     #FMT.24,-(SP)
7465 026760  012746  000004                    MOV     #4,-(SP)
```

```
7467                                    ;ML4AD
7468                                    ;                ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS
7469                                                                                                    29-Mar-1982 16:21:03    PA:<
7470 026764  010600                             MOV     SP,R0                           ; SP,*
7471 026766  104414                             TRAP    14
7472 026770  017716  166620                     MOV     @ML.REG+210,(SP)                ;                                       4453
7473 026774  016746  166614                     MOV     ML.REG+210,-(SP)
7474 027000  012746  012620                     MOV     #REG.14,-(SP)
7475 027004  012746  010142                     MOV     #FMT.24,-(SP)
7476 027010  012746  000004                     MOV     #4,-(SP)
7477 027014  010600                             MOV     SP,R0                           ; SP,*
7478 027016  104414                             TRAP    14
7479 027020  062706  000146                     ADD     #146,SP
7480 027024  017716  166574                     MOV     @ML.REG+220,(SP)                ;                                       4454
7481 027030  016746  166570                     MOV     ML.REG+220,-(SP)
7482 027034  012746  012626                     MOV     #REG.15,-(SP)
7483 027040  012746  010142                     MOV     #FMT.24,-(SP)
7484 027044  012746  000004                     MOV     #4,-(SP)
7485 027050  010600                             MOV     SP,R0                           ; SP,*
7486 027052  104414                             TRAP    14
7487 027054  026727  166276  000025             CMP     LST.DUT.REG,#25                 ;                                       4456
7488 027062  001032                             BNE     2$
7489 027064  017746  166554                     MOV     @ML.REG+240,-(SP)               ;                                       4459
7490 027070  016746  166550                     MOV     ML.REG+240,-(SP)
7491 027074  012746  012666                     MOV     #REG.20,-(SP)
7492 027100  012746  010142                     MOV     #FMT.24,-(SP)
7493 027104  012746  000004                     MOV     #4,-(SP)
7494 027110  010600                             MOV     SP,R0                           ; SP,*
7495 027112  104414                             TRAP    14
7496 027114  017716  166534                     MOV     @ML.REG+250,(SP)                ;                                       4460
7497 027120  016746  166530                     MOV     ML.REG+250,-(SP)
7498 027124  012746  012676                     MOV     #REG.21,-(SP)
7499 027130  012746  010142                     MOV     #FMT.24,-(SP)
7500 027134  012746  000004                     MOV     #4,-(SP)
7501 027140  010600                             MOV     SP,R0                           ; SP,*
7502 027142  104414                             TRAP    14
7503 027144  062706  000022                     ADD     #22,SP                          ;                                       4458
7504 027150  062706  000012             2$:     ADD     #12,SP                          ;                                       4439
7505 027154  000207                             RTS     PC                              ;                                       4428
7506
7507                                    ; Routine Size:  210 words
7508                                    ; Maximum stack depth per invocation:  52 words
7513
7514
```

```
7516 ;ML4AD
7517 ;                  ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
7518 ;                                                                 29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (49)
7519 ;          4466    routine WRT_CHK_TRANSFER (SIZE, DST, SRC) : novalue =
7520 ;          4467        begin
7521 ;          4468
7522 ;          4469    !++
7523 ;          4470    !  FUNCTIONAL DESCRIPTION:
7524 ;          4471    !       THIS ROUTINE WHEN CALLED WILL PERFORM
7525 ;          4472    !       A MASS BUS WRITE CHECK TRANSFER TO THE REQUESTED
7526 ;          4473    !       DESTINATION AND SOURCE ADDRESS OF LENGTH
7527 ;          4474    !       'SIZE'.
7528 ;          4475    !
7529 ;          4476    !  FORMAL PARAMETERS:
7530 ;          4477    !       SIZE:   DETERMINES THE NUMBER OF WORDS TO
7531 ;          4478    !               TRANSFER TO THE DESTINATION ADRS.
7532 ;          4479    !
7533 ;          4480    !       DST:    DETERMINES THE DESTINATION ADRS OF
7534 ;          4481    !               TRANSFER.
7535 ;          4482    !
7536 ;          4483    !       SRC:    DETERMINES THE SOURCE ADRS OF THE
7537 ;          4484    !               TRANSFER.
7538 ;          4485    !
7539 ;          4486    !--
7540 ;          4487
7541 ;          4488        CLR_MBUS;                           !CLEAR THE MASS BUS BEFORE WE START
7542 ;          4489        ECC_DIS = ONE;                      !DISABLE ERROR CORRECTION
7543 ;          4490        BAI = ONE;                          !MAKE THE TRANSFER STAY ON ONE BUS ADRS
7544 ;          4491        MLWC = .SIZE;                       !LOAD THE SIZE OF THE TRANSFER
7545 ;          4492        MLDA = .DST;                        !LOAD THE DESTINATION ADRS OF THE TRANSFER
7546 ;          4493        MLBA = .SRC;                        !LOAD THE SOURCE ADRS OF THE TRANSFER
7547 ;          4494        MLCS1 = WRT_CHK;                    !LOAD THE WRITE CHECK FUNCTION INTO CS1
7548 ;          4495
7549 ;          4496        if (.SC) and ( not .WCE)            !DID LOADING THE FUNCTION CAUSE A SC ERROR
7550 ;          4497        then
7551 ;          4498            begin
7552 ;          4499            ERRDF (148, INTER, DUMPER);     !REPORT THE ERROR TO OPERATOR
7553 ;          4500            PRINTB (ONE_FMT, WC_ERR);       !TELL WHAT THE ERROR IS
7554 ;          4501            DODU (.ML_LUN);                 !DROP THE UNIT
7555 ;          4502            DOCLN;                          !EXIT THE PROG
7556 ;          4503            end;
7557 ;          4504
7558 ;          4505        do                                  !DO NOTHING
7559 ;          4506            0                               !
7560 ;          4507        until .DRY;                         !UNTIL THE DRIVE IS READY
7561 ;          4508
7562 ;          4509        if (.SC) and ( not .WCE)            !DID THE SC BIT SET DURING THE XFER
7563 ;          4510        then
7564 ;          4511            begin
7565 ;          4512            ERRDF (148, INTER, DUMPER);     !REPORT THE ERROR TO OPERATOR
7566 ;          4513            PRINTB (ONE_FMT, WC_ERR);       !TELL WHAT THE ERROR IS
7567 ;          4514            DODU (.ML_LUN);                 !DROP THE UNIT
7568 ;          4515            DOCLN;                          !EXIT THE PROGRAM
7569 ;          4516            end;
7570 ;          4517
```

```
7572 ;ML4AD
7573 ;                    ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
7574                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (49)
7575 ;          4518      end;
7579
7580                                                    .SBTTL  WRT.CHK.TRANSFE ROUTINE DECLARATION SECTION
7584 027156                                     WRT.CHK.TRANSFE:
7585 027156  152777  000040  166260                    BISB    #40,@ML.REG+40                  ;                       4467
7586 027164  016700  166642                            MOV     ML.DUT,R0
7587 027170  042700  177770                            BIC     #177770,R0
7588 027174  142777  000007  166242                    BICB    #7,@ML.REG+40
7589 027202  150077  166236                            BISB    R0,@ML.REG+40
7590 027206  152777  000002  166310                    BISB    #2,@ML.REG+120                  ;                       4489
7591 027214  152777  000010  166222                    BISB    #10,@ML.REG+40                  ;                       4490
7592 027222  016677  000006  166164                    MOV     6(SP),@ML.REG+10                ; SIZE,*                4491
7593 027230  016677  000004  166176                    MOV     4(SP),@ML.REG+30                ; DST,*                 4492
7594 027236  016677  000002  166160                    MOV     2(SP),@ML.REG+20                ; SRC,*                 4493
7595 027244  012777  000051  166132                    MOV     #51,@ML.REG                     ;                       4494
7596 027252  005777  166126                            TST     @ML.REG                         ;                       4496
7597 027256  100026                                    BPL     1$
7598 027260  032777  040000  166156                    BIT     #40000,@ML.REG+40
7599 027266  001022                                    BNE     1$
7600 027270  104455                                    TRAP    55                              ;                       4499
7601 027272  000224                                    .WORD   224
7602 027274  013114                                    .WORD   INTER
7603 027276  026302                                    .WORD   DUMPER
7604 027300  012746  005652                            MOV     #WC.ERR,-(SP)                   ;                       4500
7605 027304  012746  010240                            MOV     #ONE.FMT,-(SP)
7606 027310  012746  000002                            MOV     #2,-(SP)
7607 027314  010600                                    MOV     SP,R0                           ; SP,*
7608 027316  104414                                    TRAP    14
7609 027320  016700  166504                            MOV     ML.LUN,R0                       ;                       4501
7610 027324  104451                                    TRAP    51
7611 027326  104444                                    TRAP    44
7612 027330  062706  000006                            ADD     #6,SP
7613 027334  105777  166114            1$:             TSTB    @ML.REG+50                      ;                       4498
7614 027340  100375                                    BPL     1$                              ;                       4507
7615 027342  005777  166036                            TST     @ML.REG                         ;                       4509
7616 027346  100026                                    BPL     2$
7617 027350  032777  040000  166066                    BIT     #40000,@ML.REG+40
7618 027356  001022                                    BNE     2$
7619 027360  104455                                    TRAP    55                              ;                       4512
7620 027362  000224                                    .WORD   224
7621 027364  013114                                    .WORD   INTER
7622 027366  026302                                    .WORD   DUMPER
7623 027370  012746  005652                            MOV     #WC.ERR,-(SP)                   ;                       4513
7624 027374  012746  010240                            MOV     #ONE.FMT,-(SP)
7625 027400  012746  000002                            MOV     #2,-(SP)
7626 027404  010600                                    MOV     SP,R0                           ; SP,*
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS
                                                                                                       29-Mar-1982 16:21:03    PA:<
027406  104414                                TRAP    14
027410  016700  166414                        MOV     ML.LUN,R0               ;                                               4514
027414  104451                                TRAP    51
027416  104444                                TRAP    44
027420  062706  000006                        ADD     #6,SP                   ;                                               4511
027424  000207                        2$:     RTS     PC                      ;                                               4466

                                        ; Routine Size:  84 words
                                        ; Maximum stack depth per invocation:  3 words
```

```
;          4519  routine WRT_TRANSFER (SIZE, DST, SRC) : novalue =
;          4520      begin
;          4521
;          4522  !++
;          4523  !  FUNCTIONAL DESCRIPTION
;          4524  !       THIS ROUTINE WHEN CALLED WILL PERFORM
;          4525  !       A WRITE TRANSFER TO THE REQUESTED
;          4526  !       DESTINATION AND SOURCE ADDRESSES OF
;          4527  !       LENGTH 'SIZE'.
;          4528  !
;          4529  !  FORMAL PARAMETERS:
;          4530  !       SIZE:   DETERMINES THE NUMBER OF WORDS
;          4531  !               TO TRANSFER.
;          4532  !       DST:    DETERMINES THE DESTINATION ADRS
;          4533  !               OF THE TRANSFER.
;          4534  !       SRC:    DETERMINES THE SOURCE ADRS OF THE
;          4535  !               TRANSFER.
;          4536  !--
;          4537
;          4538      CLR_MBUS;                       !CLEAR THE MASS BUS BEFORE WE START
;          4539      BAI = ONE;                      !FORCE THE TRANSFER TO STAY ON ONE BUS ADRS
;          4540      MLWC = .SIZE;                   !LOAD THE WORD COUNT SIZE
;          4541      MLDA = .DST;                    !LOAD THE DESTINATION ADRS
;          4542      MLBA = .SRC;                    !LOAD THE SOURCE ADRS
;          4543      MLCS1 = write;                  !LOAD THE FUNCTION INTO CS1
;          4544
;          4545      if .SC                          !DID LOADING THE FUNCTION CAUSE AN SC ERROR
;          4546      then
;          4547          begin
;          4548          ERRDF (148, INTER, DUMPER); !REPORT THE ERROR
;          4549          PRINTB (ONE_FMT, W_ERR);    !TELL WHAT THE ERROR IS
;          4550          DODU (.ML_LUN);             !DROP THE UNIT
;          4551          DOCLN;                      !EXIT THE PROGRAM
;          4552          end;
;          4553
;          4554      do                              !DO NOTHING
;          4555          0                           !
;          4556      until .DRY;                     !UNTIL THE DRIVE IS READY
;          4557
;          4558      if .SC                          !DID SC ERROR SET DURING THE TRANSFER
;          4559      then
;          4560          begin
;          4561          ERRDF (148, INTER, DUMPER); !REPORT THE ERORR
;          4562          PRINTB (ONE_FMT, W_ERR);    !TELL WHAT THE ERROR IS
;          4563          DODU (.ML_LUN);             !DROP THE UNIT
;          4564          DOCLN;                      !EXIT THE PROGRAM
;          4565          end;
;          4566
;          4567      end;
```

```
                                        ;ML4AD
                                        ;               ROUTINE DECLARATION SECTION                    29-Mar-1982 16:23:04    TOPS
                                        ;                                                              29-Mar-1982 16:21:03    PA<
                                                .SBTTL  WRT.TRANSFER ROUTINE DECLARATION SECTION
027426                                  WRT.TRANSFER:
027426  152777  000040  166010                BISB    #40,@ML.REG+40                  ;                       4520
027434  016700  166372                        MOV     ML.DUT,R0
027440  042700  177770                        BIC     #177770,R0
027444  142777  000007  165772                BICB    #7,@ML.REG+40
027452  150077  165766                        BISB    R0,@ML.REG+40
027456  152777  000010  165760                BISB    #10,@ML.REG+40                  ;                       4539
027464  016677  000006  165722                MOV     6(SP),@ML.REG+10                ; SIZE,*                4540
027472  016677  000004  165734                MOV     4(SP),@ML.REG+30                ; DST,*                 4541
027500  016677  000002  165716                MOV     2(SP),@ML.REG+20                ; SRC,*                 4542
027506  012777  000061  165670                MOV     #61,@ML.REG                     ;                       4543
027514  005777  165664                        TST     @ML.REG                         ;                       4545
027520  100022                                BPL     1$
027522  104455                                TRAP    55                              ;                       4548
027524  000224                                .WORD   224
027526  013114                                .WORD   INTER
027530  026302                                .WORD   DUMPER
027532  012746  005722                        MOV     #W.ERR,-(SP)                    ;                       4549
027536  012746  010240                        MOV     #ONE.FMT,-(SP)
027542  012746  000002                        MOV     #2,-(SP)
027546  010600                                MOV     SP,R0                           ; SP,*
027550  104414                                TRAP    14
027552  016700  166252                        MOV     ML.LUN,R0                       ;                       4550
027556  104451                                TRAP    51
027560  104444                                TRAP    44
027562  062706  000006                        ADD     #6,SP
027566  105777  165662          1$:           TSTB    @ML.REG+50                      ;                       4547
027572  100375                                BPL     1$                              ;                       4556
027574  005777  165604                        TST     @ML.REG                         ;                       4558
027600  100022                                BPL     2$
027602  104455                                TRAP    55                              ;                       4561
027604  000224                                .WORD   224
027606  013114                                .WORD   INTER
027610  026302                                .WORD   DUMPER
027612  012746  005722                        MOV     #W.ERR,-(SP)                    ;                       4562
027616  012746  010240                        MOV     #ONE.FMT,-(SP)
027622  012746  000002                        MOV     #2,-(SP)
027626  010600                                MOV     SP,R0                           ; SP,*
027630  104414                                TRAP    14
027632  016700  166172                        MOV     ML.LUN,R0                       ;                       4563
027636  104451                                TRAP    51
027640  104444                                TRAP    44
027642  062706  000006                        ADD     #6,SP
027646  000207          2$:           RTS     PC                              ;                       4560
                                                                                ;                       4519

                                        ; Routine Size:  73 words
                                        ; Maximum stack depth per invocation:  3 words
```

SEQ 0183

```
                                ;ML4AD
                                ;               ROUTINE DECLARATION SECTION
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
7759
7760
7761
7766
7767
7768 ;          4568  !<BLF/PAGE>
```

```
;ML4AD
;                    INITIALIZATION CODE SECTION          29-Mar-1982 16:23:04   TOPS-20 Bliss-16 V2(212)
;                                                         29-Mar-1982 16:21:03   PA:<NEALE>ML4AD.BLI.4 (52)
;
;          4569  %sbttl 'INITIALIZATION CODE SECTION'
;          4570  !
;          4571  !
;          4572  BGNINIT;
;          4573  !       INITIALIZATION CODE IS EXECUTED AT THE BEGINNING OF EACH
;          4574  !       PASS, WHEN POWER DOWN/POWER UP HAS OCCURRED, OR WHEN THE
;          4575  !       OPERATOR HAS ISSUED A START, RESTART OR CONTINUE COMMAND.
;          4576  !       DURING INITIALIZATION, THE 'GPHARD' MACRO IS USED TO GET
;          4577  !       P-TABLE INFORMATION FOR THE LOGICAL UNIT UNDER TEST. THE
;          4578  !       NUMBER OF UNITS  AVAILABLE FOR TESTING IS CONTAINED IN A
;          4579  !       HEADER LOCATION ('LSUNIT').
;          4580
;          4581  local
;          4582      OFFSET;
;          4583
;          4584  external
;          4585      LSUNIT;
;          4586
;          4587  if not READEF (EF_CONTINUE)                    !SKIP INIT CODE IF CONTINUE
;          4588  then
;          4589      begin                                      !START GPHARDS AT LUN 0 AND LOAD 'ML_REG'
;          4590
;          4591      if READEF (EF_START)                       !SEE IF THIS IS THE VERY FIRST PASS
;          4592      then
;          4593          begin                                  !THIS IS CATEGORY 1 CODE
;          4594          ML_LUN = -1;
;          4595          DROP_CNT = ZERO;                       !CLEAR THE PASS COUNTER
;          4596
;          4597          do
;          4598              begin
;          4599              ML_LUN = .ML_LUN + 1;              !INCREMENT LOGICAL UNIT NUMBER
;          4600
;          4601              if .ML_LUN geq .LSUNIT then DOCLN; !START OVER IF ALL UNITS HAVE BEEN TESTED
;          4602
;          4603              end
;          4604          until (GPHARD (.ML_LUN, PTBL_PTR)) neq 0;       !REPEAT THE GPHARD UNTIL A 0 IS RETURNED
;          4605
;          4606          RH_ADD = .((.PTBL_PTR) + 0);           !GET BASE RH ADDRESS FOR THIS UNIT
;          4607          RH_TYP = .((.PTBL_PTR) + 2);           !GET RH TYPE FOR THIS UNIT
;          4608          RH_VEC = .((.PTBL_PTR) + 4);           !GET RH VECTOR FOR THIS UNIT
;          4609          OFFSET = 0;                            !INIT OFF SET COUNT
;          4610
;          4611          incr COUNT from 0 to 21 do             !LOAD THE REGISTER ADDRESS FOR THIS UNIT INTO ML_REG
;          4612              begin
;          4613              ML_REG [.COUNT, REGISTER_ADD] = .RH_ADD + .OFFSET;
;          4614              OFFSET = .OFFSET + 2;
;          4615              end;
;          4616
;          4617      end
;          4618  else                                           !IS THIS A NEW PASS
;          4619      begin
;          4620
```

```
7826 ;ML4AD
7827 ;                  INITIALIZATION CODE SECTION
7828
7829 ;          4621           if READEF (EF_RESTART) then DROP_CNT = -1;      !RESET THE PASS COUNTER
7830 ;          4622
7831 ;          4623           if READEF (EF_NEW)                              !IS THIS A NEW PASS
7832 ;          4624           then
7833 ;          4625               begin                                       !THIS IS A NEW PASS
7834 ;          4626               DROP_CNT = .DROP_CNT + 1;                   !INCREMENT THE PASS COUNT
7835 ;          4627
7836 ;          4628               if (.ONEPAS) and (.DROP_CNT eql 1)          !HAVE WE DONE ONE PASS YET
7837 ;          4629               then
7838 ;          4630                   begin                                   !DROP THE UNIT ON THE SECOND PASS
7839 ;          4631                   DODU (.ML_LUN);                         !DROP THE UNIT
7840 ;          4632                   DOCLN;                                  !JUMP TO THE CLEAN UP CODE
7841 ;          4633                   end
7842 ;          4634               else                                        !THE FIRST PASS IS NOT COMPLETED YET
7843 ;          4635                   ML_LUN = -1;                            !RESET THE LUN POINTER
7844 ;          4636
7845 ;          4637               end;
7846 ;          4638
7847 ;          4639           do
7848 ;          4640               begin
7849 ;          4641               ML_LUN = .ML_LUN + 1;                       !IF NOT GET NEXT LUN PTABLE
7850 ;          4642
7851 ;          4643               if .ML_LUN geq .LSUNIT then DOCLN;          !START OVER IF ALL UNITS ARE TESTED
7852 ;          4644
7853 ;          4645               end
7854 ;          4646           until (GPHARD (.ML_LUN, PTBL_PTR)) neq 0;       !REPEAT UNTIL A 0 IS RETURNED
7855 ;          4647
7856 ;          4648           end;
7857 ;          4649
7858 ;          4650       PAR_DIS = .((.PTBL_PTR) + 12);                      !GET PARITY DISABLE FLAG
7859 ;          4651       ML_DUT = .((.PTBL_PTR) + 10);                       !GET DRIVE NUMBER
7860 ;          4652       OP_NUM_ARR = .((.PTBL_PTR) + 6) - 1;                !GET OPERATOR NUMBER OF ARRAYS
7861 ;          4653       GOOD_BLK = ZEROES;                                  !INIT GOOD BLOCK TO BLOCK ZERO
7862 ;          4654       ARR_16 = ZEROES;                                    !INIT ARRAY 16 TO ZERO
7863 ;          4655       LST_ARR = ZEROES;                                   !INIT LAST ARRAY TO ZERO
7864 ;          4656       LST_BLK = ZEROES;                                   !INIT LAST BLOCK TO ZERO
7865 ;          4657 !
7866 ;          4658 ! DEFINE ERROR PRINTING THRESHOLD LIMIT
7867 ;          4659 !
7868 ;          4660
7869 ;          4661       if .ERRTHR then LIMIT = 10 else LIMIT = %o'077777';
7870 ;          4662
7871 ;          4663 !
7872 ;          4664 ! CALCULATE THE MOS RAM SIZE RUN TIME
7873 ;          4665 ! PARAMETERS. 'EITHER 16K OR 64K''
7874 ;          4666 !
7875 ;          4667
7876 ;          4668       if .((.PTBL_PTR) + 8) IS_SET                        !CALCULATE ML11 16K MOS RAM PARAMETERS
7877 ;          4669       then
7878 ;          4670           begin
7879 ;          4671           DRIVE_TYPE = %o'000110';                        !EXPECTED DRIVE TYPE VALUE
7880 ;          4672           W_C_SIZE = %o'140000';                          !WORD COUNT SIZE FOR 16K WORD XFER
```

```
7882 ;ML4AD
7883 ;                  INITIALIZATION CODE SECTION
7884
7885 ;          4673           RAS_INC = %o'200';                  !RAS INCREMENT FOR 16K RAMS
7886 ;          4674           CHIP_SIZ = 16;                      !CHIP SIZE
7887 ;          4675           ARR_INC = %o'1000';                 !ARRAY INCREMENT
7888 ;          4676           ARR_16<9, 4> = %o'17';              !ARRAY 16
7889 ;          4677           LST_ARR<9, 4> = .OP_NUM_ARR;        !LAST ARRAY
7890 ;          4678           LST_BLK<9, 4> = .OP_NUM_ARR;        !LAST BLOCK
7891 ;          4679           LST_BLK = .LST_BLK or %o'777';
7892 ;          4680           end
7893 ;          4681       else                                    !CALCULATE ML11 64K MOS RAM PARAMETERS
7894 ;          4682           begin
7895 ;          4683           DRIVE_TYPE = %o'000111';            !EXPECTED DRIVE TYPE VALUE
7896 ;          4684           W_C_SIZE = %o'000000';              !WORD COUNT SIZE FOR 64K WORD XFER
7897 ;          4685           RAS_INC = %o'1000';                 !RAS INCREMENT FOR 64K RAMS
7898 ;          4686           CHIP_SIZ = 64;                      !CHIP SIZE
7899 ;          4687           ARR_INC = %o'4000';                 !ARRAY INCREMENT
7900 ;          4688 !
7901 ;          4689 ! VERSION CZMLAD CHANGED %O'37' TO %O'17'
7902 ;          4690 !
7903 ;          4691           ARR_16<11, 4> = %o'17';             !ARRAY 16
7904 ;          4692           LST_ARR<11, 4> = .OP_NUM_ARR;       !LAST ARRAY
7905 ;          4693           LST_BLK<11, 4> = .OP_NUM_ARR;       !LAST BLOCK
7906 ;          4694           LST_BLK = .LST_BLK or %o'3777';
7907 ;          4695           end;
7908 ;          4696
7909 ;          4697       if .((.PTBL_PTR) + 2) eql %o'70' then LST_DUT_REG = 21 else LST_DUT_REG = 19;
7910 ;          4698
7911 ;          4699       PRINTB (FMT_17, .ML_LUN);               !TELL OPERATOR WHICH UNIT IS BEING TESTED
7912 ;          4700       CLR_MBUS;                               !CLEAR MASS BUS
7913 ;          4701       end;
7914 ;          4702
7915 ;          4703   ENDINIT;
7919
7920                                   .GLOBL  L$UNIT
7921
7922
7923                                   .SBTTL  LINIT INITIALIZATION CODE SECTION
7927 027650 004167 154700          LINIT: JSR     R1,$SAVE3                ;                4567
7928 027654 012700 000036                 MOV     #36,R0                   ;                4587
7929 027660 104447                        TRAP    47
7930 027662 103001                        BHIS    1$
7931 027664 000207                        RTS     PC
7932 027666 012700 000040          1$:    MOV     #40,R0                   ;                4591
7933 027672 104447                        TRAP    47
7934 027674 103063                        BHIS    5$
7935 027676 012767 177777 166124          MOV     #-1,ML.LUN               ;                4594
7936 027704 005067 165472                 CLR     DROP.CNT                 ;                4595
```

```
                                        ;ML4AD
                                        ;               INITIALIZATION CODE SECTION                    29-Mar-1982 16:23:04    TOPS
                                                                                                       29-Mar-1982 16:21:03    PA:<
7941 027710 005267 166114               2$:     INC     ML.LUN
7942 027714 026767 166110 152070                CMP     ML.LUN,L$UNIT           ;                                       4599
7943 027722 002401                              BLT     3$                      ;                                       4601
7944 027724 104444                              TRAP    44
7945 027726 016700 166076               3$:     MOV     ML.LUN,R0
7946 027732 104442                              TRAP    42                      ;                                       4604
7947 027734 010067 164040                       MOV     R0,PTBL.PTR
7948 027740 005767 164034                       TST     PTBL.PTR
7949 027744 001761                              BEQ     2$
7950 027746 017767 164026 166046                MOV     @PTBL.PTR,RH.ADD
7951 027754 016701 164020                       MOV     PTBL.PTR,R1             ;                                       4606
7952 027760 016167 000002 166036                MOV     2(R1),RH.TYP            ;                                       4607
7953 027766 016701 164006                       MOV     PTBL.PTR,R1
7954 027772 016167 000004 166026                MOV     4(R1),RH.VEC            ;                                       4608
7955 030000 005002                              CLR     R2                      ; OFFSET                                4609
7956 030002 005001                              CLR     R1                      ; COUNT                                 4611
7957 030004 010100                      4$:     MOV     R1,R0                   ; COUNT,*                               4613
7958 030006 006300                              ASL     R0
7959 030010 006300                              ASL     R0
7960 030012 006300                              ASL     R0
7961 030014 016703 166002                       MOV     RH.ADD,R3
7962 030020 060203                              ADD     R2,R3                   ; OFFSET,*
7963 030022 010360 015404                       MOV     R3,ML.REG(R0)
7964 030026 062702 000002                       ADD     #2,R2                   ; *,OFFSET                              4614
7965 030032 005201                              INC     R1                      ; COUNT                                 4611
7966 030034 020127 000025                       CMP     R1,#25                  ; COUNT,*
7967 030040 003761                              BLE     4$
7968 030042 000454                              BR      10$
7969 030044 012700 000037               5$:     MOV     #37,R0                  ;                                       4591
7970 030050 104447                              TRAP    47                      ;                                       4621
7971 030052 103003                              BHIS    6$
7972 030054 012767 177777 165320                MOV     #-1,DROP.CNT
7973 030062 012700 000035               6$:     MOV     #35,R0
7974 030066 104447                              TRAP    47                      ;                                       4623
7975 030070 103022                              BHIS    8$
7976 030072 005267 165304                       INC     DROP.CNT
7977 030076 032767 000001 152302                BIT     #1,ONEPAS               ;                                       4626
7978 030104 001411                              BEQ     7$                      ;                                       4628
7979 030106 026727 165270 000001                CMP     DROP.CNT,#1
7980 030114 001005                              BNE     7$
7981 030116 016700 165706                       MOV     ML.LUN,R0
7982 030122 104451                              TRAP    51                      ;                                       4631
7983 030124 104444                              TRAP    44
7984 030126 000403                              BR      8$
7985 030130 012767 177777 165672        7$:     MOV     #-1,ML.LUN              ;                                       4628
7986 030136 005267 165666               8$:     INC     ML.LUN                  ;                                       4635
7987 030142 026767 165662 151642                CMP     ML.LUN,L$UNIT           ;                                       4641
7988 030150 002401                              BLT     9$                      ;                                       4643
7989 030152 104444                              TRAP    44
7990 030154 016700 165650               9$:     MOV     ML.LUN,R0
7991 030160 104442                              TRAP    42                      ;                                       4646
7992 030162 010067 163612                       MOV     R0,PTBL.PTR
```

```
                                        :ML4AD
                                        ;       INITIALIZATION CODE SECTION                    29-Mar-1982 16:23:04    TOPS
                                                                                               29-Mar-1982 16:21:03    PA:<
030166  005767  163606                        TST     PTBL.PTR
030172  001761                                BEQ     8$
030174  016701  163600                10$:    MOV     PTBL.PTR,R1
030200  016167  000014  163602                MOV     14(R1),PAR.DIS          ;                                       4650
030206  016701  163566                        MOV     PTBL.PTR,R1
030212  016167  000012  165612                MOV     12(R1),ML.DUT           ;                                       4651
030220  016701  163554                        MOV     PTBL.PTR,R1
030224  016103  000006                        MOV     6(R1),R3                ;                                       4652
030230  005303                                DEC     R3
030232  010367  163544                        MOV     R3,OP.NUM.ARR
030236  005067  163544                        CLR     GOOD.BLK
030242  005067  163550                        CLR     ARR.16                  ;                                       4653
030246  005067  163546                        CLR     LST.ARR                 ;                                       4654
030252  005067  163536                        CLR     LST.BLK                 ;                                       4655
030256  032767  000001  152116                BIT     #1,ERRTHR               ;                                       4656
030264  001404                                BEQ     11$                     ;                                       4661
030266  012767  000012  165104                MOV     #12,LIMIT
030274  000403                                BR      12$
030276  012767  077777  165074        11$:    MOV     #77777,LIMIT
030304  016701  163470                12$:    MOV     PTBL.PTR,R1
030310  026127  000010  000001                CMP     10(R1),#1               ;                                       4668
030316  001054                                BNE     13$
030320  012767  000110  165026                MOV     #110,DRIVE.TYPE
030326  012767  140000  165010                MOV     #-40000,W.C.SIZE        ;                                       4671
030334  012767  000200  165004                MOV     #200,RAS.INC            ;                                       4672
030342  012767  000020  163442                MOV     #20,CHIP.SIZ            ;                                       4673
030350  012767  001000  163426                MOV     #1000,ARR.INC           ;                                       4674
030356  052767  017000  163432                BIS     #17000,ARR.16           ;                                       4675
030364  016703  163412                        MOV     OP.NUM.ARR,R3           ;                                       4676
030370  000303                                SWAB    R3                      ;                                       4677
030372  006303                                ASL     R3
030374  042703  160777                        BIC     #160777,R3
030400  042767  017000  163412                BIC     #17000,LST.ARR
030406  050367  163406                        BIS     R3,LST.ARR
030412  016703  163364                        MOV     OP.NUM.ARR,R3           ;                                       4678
030416  000303                                SWAB    R3
030420  006303                                ASL     R3
030422  042703  160777                        BIC     #160777,R3
030426  042767  017000  163360                BIC     #17000,LST.BLK
030434  050367  163354                        BIS     R3,LST.BLK
030440  052767  000777  163346                BIS     #777,LST.BLK            ;                                       4679
030446  000456                                BR      14$                     ;                                       4668
030450  012767  000111  164676        13$:    MOV     #111,DRIVE.TYPE         ;                                       4683
030456  005067  164662                        CLR     W.C.SIZE                ;                                       4684
030462  012767  001000  164656                MOV     #1000,RAS.INC           ;                                       4685
030470  012767  000100  163314                MOV     #100,CHIP.SIZ           ;                                       4686
030476  012767  004000  163300                MOV     #4000,ARR.INC           ;                                       4687
030504  052767  074000  163304                BIS     #74000,ARR.16           ;                                       4691
030512  016703  163264                        MOV     OP.NUM.ARR,R3           ;                                       4692
030516  000303                                SWAB    R3
030520  006303                                ASL     R3
030522  006303                                ASL     R3
```

SEQ 0189

```
                        ;ML4AD
                        ;                INITIALIZATION CODE SECTION                    29-Mar-1982 16:23:04    TOPS
                        ;                                                               29-Mar-1982 16:21:03    PA:<
030524 006303                    ASL     R3
030526 042703 103777             BIC     #103777,R3
030532 042767 074000 163260      BIC     #74000,LST.ARR
030540 050367 163254             BIS     R3,LST.ARR
030544 016703 163232             MOV     OP.NUM.ARR,R3          ;                       4693
030550 000303                    SWAB    R3
030552 006303                    ASL     R3
030554 006303                    ASL     R3
030556 006303                    ASL     R3
030560 042703 103777             BIC     #103777,R3
030564 042767 074000 163222      BIC     #74000,LST.BLK
030572 050367 163216             BIS     R3,LST.BLK
030576 052767 003777 163210      BIS     #3777,LST.BLK          ;                       4694
030604 016701 163170     14$:    MOV     PTBL.PTR,R1            ;                       4697
030610 026127 000002 000070      CMP     2(R1),#70
030616 001004                    BNE     15$
030620 012767 000025 164530      MOV     #25,LST.DUT.REG
030626 000403                    BR      16$
030630 012767 000023 164520 15$: MOV     #23,LST.DUT.REG
030636 016746 165166        16$: MOV     ML.LUN,-(SP)           ;                       4699
030642 012746 007526             MOV     #FMT.17,-(SP)
030646 012746 000002             MOV     #2,-(SP)
030652 010600                    MOV     SP,R0                  ; SP,*
030654 104414                    TRAP    14
030656 152777 000040 164560      BISB    #40,@ML.REG+40
030664 016703 165142             MOV     ML.DUT,R3
030670 042703 177770             BIC     #177770,R3
030674 142777 000007 164542      BICB    #7,@ML.REG+40
030702 150377 164536             BISB    R3,@ML.REG+40
030706 062706 000006             ADD     #6,SP                  ;                       4589
030712 000207                    RTS     PC                     ;                       4567

                        ; Routine Size:  274 words
                        ; Maximum stack depth per invocation:  7 words


                                .SBTTL  L$INIT INITIALIZATION CODE SECTION
030714 004767 176730    L$INIT::JSR     PC,LINIT               ;                       4701
030720 104411                   TRAP    11
030722 000207                   RTS     PC
```

```
8106                                    ;ML4AD
8107                                    ;               INITIALIZATION CODE SECTION
8108
8109                                    ; Routine Size:  4 words
8110                                    ; Maximum stack depth per invocation:  0 words
8115
8116
8117 ;          4704   !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (53)

```
;ML4AD
;                TEST CODE SECTION

;          4705  %SBTTL 'TEST CODE SECTION'
;          4706
;          4707  !
;          4708  BGNTST;
;          4709
;          4710  !++
;          4711  ! TEST NUMBER:  TST 1
;          4712  !
;          4713  ! TEST NAME:              MASS BUS READY TEST
;          4714  !
;          4715  ! TEST DESCRIPTION:
;          4716  !       TEST THE RH CONTROLLER FOR EXISTANCE
;          4717  !       BY:
;          4718  !
;          4719  !       1.  WRITTING ONES TO THE RH CS2 REGISTER
;          4720  !           RESULTING IN A MASS BUS CLEAR
;          4721  !
;          4722  !       2.  THEN READ THE CS2 REGISTER FOR CLEAR DATA
;          4723  !
;          4724  !--
;          4725
;          4726  local
;          4727      DODU_FLG;                                   !DROP UNIT FLAG
;          4728
;          4729  DODU_FLG = ZERO;                               !CLEAR THE DROP UNIT FLAG
;          4730  BGNSUB;
;          4731  CLR_MBUS;
;          4732  MLCS1 = ZEROES;                               !CLEAR OUT CS1
;          4733  MLCS2 = %o'177770';                           !LOAD CS2 WITH ONES  AND FORCE A MBUS CLEAR
;          4734  DELAY (ONE_US);
;          4735
;          4736  if ((.MLCS2) and (%o'177670')) neq ZERO       !SEE IF CS2 GOT CLEARED
;          4737  then
;          4738      begin                                     !REPORT ERROR IF NOT CLEARED
;          4739      ERRDF (119, RH_ERROR, DUMPER);
;          4740      PRINTB (THR_FMT, REG_17, FNC_23, PHR_4);
;          4741      DODU_FLG = ONE;
;          4742      end;
;          4743
;          4744  ENDSUB;
;          4745
;          4746  if .DODU_FLG IS_SET                           !DROP THE UNIT IF DODU_FLG IS SET
;          4747  then
;          4748      begin
;          4749      DODU (.ML_LUN);
;          4750      DOCLN;
;          4751      end;
;          4752
;          4753  ENDTST;
```

SEQ 0192


```
                                        ;ML4AD
                                        ;               TEST CODE SECTION

                                                .SBTTL  $T1 TEST CODE SECTION
030724  004167  153610          $T1:  JSR     R1,$SAVE2                       ;                       4703
030730  005746                        TST     -(SP)
030732  005002                        CLR     R2                              ; DODU.FLG              4729
030734  104402                  1$:   TRAP    2
030736  152777  000040  164500        BISB    #40,@ML.REG+40                  ;                       4730
030744  016701  165062                MOV     ML.DUT,R1
030750  042701  177770                BIC     #177770,R1
030754  142777  000007  164462        BICB    #7,@ML.REG+40
030762  150177  164456                BISB    R1,@ML.REG+40
030766  005077  164412                CLR     @ML.REG                         ;                       4732
030772  012777  177770  164444        MOV     #-10,@ML.REG+40                 ;                       4733
031000  012700  000001                MOV     #1,R0                           ; *,$$TMP2              4734
031004  001410                  2$:   BEQ     5$                              ; *,$$TMP1
031006  016701  151104                MOV     L$DLY,R1
031012  001403                        BEQ     4$
031014  005016                  3$:   CLR     (SP)                            ; $$TMP
031016  005301                        DEC     R1                              ; $$TMP1
031020  001375                        BNE     3$
031022  005300                  4$:   DEC     R0                              ; $$TMP2
031024  000767                        BR      2$
031026  032777  177670  164410  5$:   BIT     #-110,@ML.REG+40                ;                       4736
031034  001424                        BEQ     6$
031036  104455                        TRAP    55                              ;                       4739
031040  000167                        .WORD   167
031042  013302                        .WORD   RH.ERROR
031044  026302                        .WORD   DUMPER
031046  012746  011676                MOV     #PHR.4,-(SP)                    ;                       4740
031052  012746  012464                MOV     #FNC.23,-(SP)
031056  012746  012642                MOV     #REG.17,-(SP)
031062  012746  010256                MOV     #THR.FMT,-(SP)
031066  012746  000004                MOV     #4,-(SP)
031072  010600                        MOV     SP,R0                           ; SP,*
031074  104414                        TRAP    14
031076  012702  000001                MOV     #1,R2                           ; *,DODU.FLG            4741
031102  062706  000012                ADD     #12,SP                          ;                       4738
031106  104467            6$:   TRAP    67                              ;                       4742
031110  006000                        ROR     R0
031112  103710                        BLO     1$
031114  005302                        DEC     R2                              ; DODU.FLG              4746
031116  001004                        BNE     7$
031120  016700  164704                MOV     ML.LUN,R0                       ;                       4749
031124  104451                        TRAP    51
031126  104444                        TRAP    44
031130  005726            7$:   TST     (SP)+                           ;                       4703
031132  000207                        RTS     PC

                        ; Routine Size:  68 words
```

```
                                ;ML4AD
                                ;                TEST CODE SECTION

                                ; Maximum stack depth per invocation:  9 words


                                        .SBTTL  T1 TEST CODE SECTION
031134                          T1::
031134  004767  177564          1$:     JSR     PC,$T1                          ;                4751
031140  104466                          TRAP    66
031142  006000                          ROR     R0
031144  103773                          BLO     1$
031146  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


;               4754  !<BLF/PAGE>
```

```
8266 ;ML4AD
8267 ;                  TEST CODE SECTION
8268
8269 ;          4755   !
8270 ;          4756   BGNTST;
8271 ;          4757
8272 ;          4758   !++
8273 ;          4759   ! TEST NUMBER:  TST 2
8274 ;          4760   !
8275 ;          4761   ! TEST NAME:    MASS BUS HANDSHAKE TEST
8276 ;          4762   !
8277 ;          4763   ! TEST DESCRIPTION:
8278 ;          4764   !       TEST MASS BUS TO UNIBUS COMMUMICATIONS
8279 ;          4765   !       VIA THE CONTROL BUS BY:
8280 ;          4766   !
8281 ;          4767   !       1.  READING RH AND DRIVE REGISTERS AND
8282 ;          4768   !           TEST THE NON EXISTANT DRIVE 'NED'
8283 ;          4769   !           BIT
8284 ;          4770   !
8285 ;          4771   !--
8286 ;          4772
8287 ;          4773   local
8288 ;          4774       SAVE,
8289 ;          4775       DODU_FLG;                          !TEMPORARY SAVE LOCATION
8290 ;          4776                                          !DROP UNIT FLAG
8291 ;          4777   CLR_THRESHOLD;                         !CLEAR ERROR PRINT THRESHOLD
8292 ;          4778   DODU_FLG = ZERO;
8293 ;          4779
8294 ;          4780   incr REG_SEL from 0 to .LST_DUT_REG do     !TEST ALL PRESENT RH REGISTERS
8295 ;          4781       begin
8296 ;          4782       BGNSUB;                            !START OF SCOPE LOOP
8297 ;          4783       CLR_MBUS;
8298 ;          4784       SAVE = ..ML_REG [.REG_SEL, REGISTER_ADD];   !READ THE REGISTER
8299 ;          4785       DELAY (ONE_US);                    !DELAY ONE MICRO SECOND
8300 ;          4786
8301 ;          4787       if .NED IS_SET                     !DID READ CAUSE THE NED BIT TO SET
8302 ;          4788       then
8303 ;          4789           begin                          !REPORT AN ERROR IF SET
8304 ;          4790           CMP_THRESHOLD;                 !COMPARE ERROR PRINT THRESHOLD
8305 ;          4791           ERRDF (120, RH_ERROR, DUMPER);
8306 ;          4792           PRINTB (FIV_FMT, WRD_62, PHR_5, WRD_12, WRD_52, FNC_6);
8307 ;          4793           PRINTB (FMT_11, .ML_REG [.REG_SEL, REGISTER_ADD]);
8308 ;          4794           DODU_FLG = ONE;
8309 ;          4795           end;
8310 ;          4796
8311 ;          4797       ENDSUB;                            !END OF SCOPE LOOP
8312 ;          4798       end;
8313 ;          4799
8314 ;          4800   if .DODU_FLG IS_SET
8315 ;          4801   then
8316 ;          4802       begin
8317 ;          4803       DODU (.ML_LUN);
8318 ;          4804       DOCLN;
8319 ;          4805       end;
8320 ;          4806
```

```
:ML4AD                                                        29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;               TEST CODE SECTION                             29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (54)

;        4807   ENDTST;

                                             .SBTTL  $T2 TEST CODE SECTION
031150  004167  153436           $T2:        JSR     R1,$SAVE5
031154  024646                               CMP     -(SP),-(SP)                  ;                    4753
031156  005067  164214                       CLR     P.CNT                        ;                    4775
031162  005001                               CLR     R1                           ; DODU.FLG           4778
031164  016746  164166                       MOV     LST.DUT.REG,-(SP)            ;                    4780
031170  005002                               CLR     R2                           ; REG.SEL
031172  000517                               BR      9$
031174  010200                   1$:         MOV     R2,R0                        ; REG.SEL,*          4784
031176  006300                               ASL     R0
031200  006300                               ASL     R0
031202  006300                               ASL     R0
031204  010003                               MOV     R0,R3
031206  104402                   2$:         TRAP    2                            ;                    4781
031210  152777  000040  164226               BISB    #40,@ML.REG+40               ;                    4782
031216  016705  164610                       MOV     ML.DUT,R5
031222  042705  177770                       BIC     #177770,R5
031226  142777  000007  164210               BICB    #7,@ML.REG+40
031234  150577  164204                       BISB    R5,@ML.REG+40
031240  017366  015404  000002               MOV     @ML.REG(R3),2(SP)            ; *,SAVE             4784
031246  012704  000001                       MOV     #1,R4                        ; *,$$TMP2           4785
031252  001411                   3$:         BEQ     6$
031254  016705  150636                       MOV     L$DLY,R5                     ; *,$$TMP1
031260  001404                               BEQ     5$
031262  005066  000004           4$:         CLR     4(SP)                        ; $$TMP
031266  005305                               DEC     R5                           ; $$TMP1
031270  001374                               BNE     4$
031272  005304                   5$:         DEC     R4                           ; $$TMP2
031274  000766                               BR      3$
031276  032777  010000  164140   6$:         BIT     #10000,@ML.REG+40            ;                    4787
031304  001446                               BEQ     7$
031306  005267  164064                       INC     P.CNT                        ;                    4789
031312  026767  164060  164060               CMP     P.CNT,LIMIT
031320  003043                               BGT     8$
031322  104455                               TRAP    55                           ;                    4791
031324  000170                               .WORD   170
031326  013302                               .WORD   RH.ERROR
031330  026302                               .WORD   DUMPER
031332  012746  012226                       MOV     #FNC.6,-(SP)                 ;                    4792
031336  012746  011310                       MOV     #WRD.52,-(SP)
031342  012746  010650                       MOV     #WRD.12,-(SP)
031346  012746  011714                       MOV     #PHR.5,-(SP)
031352  012746  011432                       MOV     #WRD.62,-(SP)
031356  012746  010304                       MOV     #FIV.FMT,-(SP)
```

```
8378                                    ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
8379                                    ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
8380
8381 031362  012746  000006                    MOV     #6,-(SP)
8382 031366  010600                            MOV     SP,R0                   ; SP,*
8383 031370  104414                            TRAP    14
8384 031372  016316  015404                    MOV     ML.REG(R3),(SP)         ;                                               4793
8385 031376  012746  007170                    MOV     #FMT.11,-(SP)
8386 031402  012746  000002                    MOV     #2,-(SP)
8387 031406  010600                            MOV     SP,R0                   ; SP,*
8388 031410  104414                            TRAP    14
8389 031412  012701  000001                    MOV     #1,R1                   ; *,DODU.FLG                                    4794
8390 031416  062706  000022                    ADD     #22,SP                  ;                                               4789
8391 031422  104467                    7$:     TRAP    67                      ;                                               4795
8392 031424  006000                            ROR     R0
8393 031426  103667                            BLO     2$
8394 031430  005202                    8$:     INC     R2                      ; REG.SEL
8395 031432  020216                    9$:     CMP     R2,(SP)                 ; REG.SEL,*                                     4780
8396 031434  003657                            BLE     1$
8397 031436  005301                            DEC     R1                      ; DODU.FLG                                      4800
8398 031440  001004                            BNE     10$
8399 031442  016700  164362                    MOV     ML.LUN,R0               ;                                               4803
8400 031446  104451                            TRAP    51
8401 031450  104444                            TRAP    44
8402 031452  062706  000006            10$:    ADD     #6,SP                   ;                                               4753
8403 031456  000207                            RTS     PC
8404
8405                                    ; Routine Size:  100 words
8406                                    ; Maximum stack depth per invocation:  18 words
8411
8412
8416
8417                                            .SBTTL  T2 TEST CODE SECTION
8421 031460                             T2::
8422 031460  004767  177464             1$:     JSR     PC,$T2                  ;                                               4805
8423 031464  104466                             TRAP    66
8424 031466  006000                             ROR     R0
8425 031470  103773                             BLO     1$
8426 031472  000207                             RTS     PC
8427
8428
8429                                    ; Routine Size:  6 words
                                        ; Maximum stack depth per invocation:  0 words
```

SEQ 0197

```
 8438
 8439
 8440 ;          4808   !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (55)

```
;ML4AD
;                TEST CODE SECTION
;
;          4809
;          4810  BGNTST;                                                    !
;          4811
;          4812  !++
;          4813  ! TEST NUMBER:  TST 3
;          4814  !
;          4815  ! TEST NAME:    DRIVE PRESENT TEST
;          4816  !
;          4817  ! TEST DESCRIPTION:
;          4818  !       THIS TEST READS THE DESIRED SECTOR
;          4819  !       ADDRESS REGISTER OF THE DRIVE UNDER
;          4820  !       TEST, DELAYS 100 US, THEN
;          4821  !       READS THE  NED BIT OF MLCS2
;          4822  !
;          4823  !       IF SET, AN ERROR MESSAGE IS
;          4824  !       PRINTED AND THE UNIT IS DROPPED
;          4825  !
;          4826  !--
;          4827
;          4828  local
;          4829      DODU_FLG,                                              !DROP UNIT FLAG
;          4830      SAVE;                                                  !TEMP STORAGE LOCATION
;          4831
;          4832  DODU_FLG = ZERO;
;          4833  BGNSUB;
;          4834  CLR_MBUS;
;          4835  SAVE = .MLDA;                                             !READ A DRIVE REGISTER
;          4836  DELAY (ONE_US);                                           !DELAY 1 US
;          4837
;          4838  if .NED IS_SET                                           !TEST THE NED BIT
;          4839  then
;          4840      begin
;          4841      ERRDF (1, ASYNC, DUMPER);                             !IF SET THEN REPORT ERROR AND SET DODU_FLG
;          4842      PRINTB (ONE_FMT, PHR_3);
;          4843      DODU_FLG = ONE;
;          4844      end;
;          4845
;          4846  ENDSUB;
;          4847
;          4848  if .DODU_FLG IS_SET                                       !DROP THIS UNIT IF DODU IS_SET
;          4849  then
;          4850      begin
;          4851      DODU (.ML_LUN);
;          4852      DOCLN;
;          4853      end;
;          4854
;          4855  ENDTST;
;
;                                  .SBTTL  $T3 TEST CODE SECTION
```

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS
                                                                                                         29-Mar-1982 16:21:03    PA:<
031474  004167  153072          $T3:    JSR     R1,$SAVE4
031500  005746                          TST     -(SP)           ;                                                               4807
031502  005003                          CLR     R3
031504  104402                  1$:     TRAP    2               ; DODU.FLG                                                      4832
031506  152777  000040  163730          BISB    #40,@ML.REG+40  ;                                                               4833
031514  016702  164312                  MOV     ML.DUT,R2
031520  042702  177770                  BIC     #177770,R2
031524  142777  000007  163712          BICB    #7,@ML.REG+40
031532  150277  163706                  BISB    R2,@ML.REG+40
031536  017704  163672                  MOV     @ML.REG+30,R4   ; *,SAVE                                                        4835
031542  012701  000001                  MOV     #1,R1           ; *,$$TMP2                                                      4836
031546  001410                  2$:     BEQ     5$
031550  016702  150342                  MOV     L$DLY,R2        ; *,$$TMP1
031554  001403                          BEQ     4$
031556  005016                  3$:     CLR     (SP)            ; $$TMP
031560  005302                          DEC     R2              ; $$TMP1
031562  001375                          BNE     3$
031564  005301                  4$:     DEC     R1              ; $$TMP2
031566  000767                          BR      2$
031570  032777  010000  163646  5$:     BIT     #10000,@ML.REG+40 ;                                                             4838
031576  001420                          BEQ     6$
031600  104455                          TRAP    55              ;                                                               4841
031602  000001                          .WORD   1
031604  012706                          .WORD   ASYNC
031606  026302                          .WORD   DUMPER
031610  012746  011644                  MOV     #PHR.3,-(SP)    ;                                                               4842
031614  012746  010240                  MOV     #ONE.FMT,-(SP)
031620  012746  000002                  MOV     #2,-(SP)
031624  010600                          MOV     SP,R0           ; SP,*
031626  104414                          TRAP    14
031630  012703  000001                  MOV     #1,R3           ; *,DODU.FLG                                                    4843
031634  062706  000006                  ADD     #6,SP           ;                                                               4840
031640  104467                  6$:     TRAP    67              ;                                                               4844
031642  006000                          ROR     R0
031644  103717                          BLO     1$
031646  005303                          DEC     R3              ; DODU.FLG                                                      4848
031650  001004                          BNE     7$
031652  016700  164152                  MOV     ML.LUN,R0       ;                                                               4851
031656  104451                          TRAP    51
031660  104444                          TRAP    44
031662  005726                  7$:     TST     (SP)+           ;                                                               4807
031664  000207                          RTS     PC

                                        ; Routine Size:  61 words
                                        ; Maximum stack depth per invocation:  9 words
```

SEQ 0200


```
8554 ;ML4AD
8555 ;                  TEST CODE SECTION
8556
8560
8561                                        .SBTTL  T3 TEST CODE SECTION
8565 031666                          T3::
8566 031666  004767  177602         1$:     JSR     PC,$T3                          ;                4853
8567 031672  104466                         TRAP    66
8568 031674  006000                         ROR     R0
8569 031676  103773                         BLO     1$
8570 031700  000207                         RTS     PC
8571
8572                                 ; Routine Size:  6 words
8573                                 ; Maximum stack depth per invocation:  0 words
8578
8579
8580 ;          4856  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (56)

```
8582  :ML4AD
8583  :                    TEST CODE SECTION
8584
8585  :          4857  !
8586  :          4858  BGNTST;
8587  :          4859
8588  :          4860  !++
8589  :          4861  ! TEST NUMBER:  TST 4
8590  :          4862  !
8591  :          4863  ! TEST NAME:    DRIVE SELECTION TEST
8592  :          4864  !
8593  :          4865  ! TEST DESCRIPTION:
8594  :          4866  !       THIS TEST TESTS FOR UNIQUE DRIVE SELECTION BY WRITING THE DRIVE
8595  :          4867  !       UNDER TEST (DUT) DRIVE NUMBER INTO ITS DSA REG.  THEN WRITING THE DRIVE
8596  :          4868  !       NUMBERS OF OTHER DRIVES INTO THEIR RESPECTIVE DSA REGISTERS
8597  :          4869  !       AND READING THE DUT DSA FOR ITS DRIVE NUMBER.
8598  :          4870  !
8599  :          4871  ! IMPLICIT INPUTS:
8600  :          4872  !       ML_DUT
8601  :          4873  !       LOADED DURING THE INITIALIZATION CODE AND CONTAINS THE DRIVE
8602  :          4874  !       NUMBER OF THE DRIVE PRESENTLY BEING TESTED.
8603  :          4875  !--
8604  :          4876
8605  :          4877  local
8606  :          4878      SAVE;                                   !TEMPORARY SAVE LOCATION
8607  :          4879
8608  :          4880  BGNSUB;
8609  :          4881  CLR_MBUS;
8610  :          4882  MLDA = .ML_DUT;                             !LOAD THIS DRIVES DRIVE NO. INTO ITS DSA REG
8611  :          4883  SAVE = .MLDA;                               !READ THE REGISTER BACK
8612  :          4884
8613  :          4885  if .SAVE neq .ML_DUT                        !SEE IF DSA HAS DRIVE NUMBER
8614  :          4886  then
8615  :          4887      begin
8616  :          4888      ERRDF (2, INTER, DUMPER);               !ERROR AND EXIT TEST IF DSA NEQ DRIVE NUM
8617  :          4889      PRINTB (SIX_FMT, PHR_4, WRD_12, FNC_3, WRD_37, WRD_13, REG_6);
8618  :          4890      PRINTB (FMT_2, .ML_DUT, .SAVE, (.ML_DUT xor .SAVE));
8619  :          4891      EXIT_TST;
8620  :          4892      end;
8621  :          4893
8622  :          4894  incr DRV_SEL from 0 to 7 do                 !WRITE DRV NO OF OTHER DRIVES INTO THEIR RESPECTIVE DSA REG.
8623  :          4895
8624  :          4896      if .DRV_SEL neq .ML_DUT                 !SKIP IF .DRV_SEL EQL TO THE DRIVE UNDER TEST (DUT)
8625  :          4897      then
8626  :          4898          begin
8627  :          4899          DRV_NUM = .DRV_SEL;                 !SELECT DRIVE TO BE WRITTEN TO
8628  :          4900          MLDA = .DRV_SEL;                    !WRITE DRIVE SEL NO. INTO ITS DSA REG
8629  :          4901          DELAY (ONE_US);                     !DELAY 1 US
8630  :          4902          end;
8631  :          4903
8632  :          4904  DRV_NUM = .ML_DUT;                         !SELECT THE DUT
8633  :          4905  SAVE = .MLDA;                               !READ ITS DSA REG
8634  :          4906
8635  :          4907  if .SAVE neq .ML_DUT                        !SEE IF WRITTING TO OTHER DRIVES CHANGE ITS VALUE
8636  :          4908  then
```

```
8638 ;ML4AD
8639 ;                  TEST CODE SECTION
8640
8641 ;          4909       begin
8642 ;          4910       ERRDF (3, ASYNC, DUMPER);                    !ERROR AND SET DODU_FLG IF CHANGED
8643 ;          4911       PRINTB (THR_FMT, FNC_3, WRD_37, WRD_14);
8644 ;          4912       PRINTB (FMT_1, .ML_DUT, .SAVE);
8645 ;          4913       end;
8646 ;          4914
8647 ;          4915   ENDSUB;
8648 ;          4916   ENDTST;
8652
8653                                          .SBTTL  $T4 TEST CODE SECTION
8657 031702  004167  152646          $T4:    JSR     R1,$SAVE3                    ;                                  4855
8658 031706  005746                          TST     -(SP)
8659 031710  104402                  1$:     TRAP    2                            ;                                  4878
8660 031712  152777  000040  163524          BISB    #40,@ML.REG+40               ;                                  4880
8661 031720  016702  164106                  MOV     ML.DUT,R2
8662 031724  042702  177770                  BIC     #177770,R2
8663 031730  142777  000007  163506          BICB    #7,@ML.REG+40
8664 031736  150277  163502                  BISB    R2,@ML.REG+40
8665 031742  016777  164064  163464          MOV     ML.DUT,@ML.REG+30            ;                                  4882
8666 031750  017703  163460                  MOV     @ML.REG+30,R3                ; *,SAVE                           4883
8667 031754  020367  164052                  CMP     R3,ML.DUT                    ; SAVE,*                           4885
8668 031760  001451                          BEQ     2$
8669 031762  104455                          TRAP    55                           ;                                  4888
8670 031764  000002                          .WORD   2
8671 031766  013114                          .WORD   INTER
8672 031770  026302                          .WORD   DUMPER
8673 031772  012746  012540                  MOV     #REG.6,-(SP)                 ;                                  4889
8674 031776  012746  010660                  MOV     #WRD.13,-(SP)
8675 032002  012746  011122                  MOV     #WRD.37,-(SP)
8676 032006  012746  012172                  MOV     #FNC.3,-(SP)
8677 032012  012746  010650                  MOV     #WRD.12,-(SP)
8678 032016  012746  011676                  MOV     #PHR.4,-(SP)
8679 032022  012746  010322                  MOV     #SIX.FMT,-(SP)
8680 032026  012746  000007                  MOV     #7,-(SP)
8681 032032  010600                          MOV     SP,R0                        ; SP,*
8682 032034  104414                          TRAP    14
8683 032036  016716  163770                  MOV     ML.DUT,(SP)                  ;                                  4890
8684 032042  010302                          MOV     R3,R2                        ; SAVE,*
8685 032044  041602                          BIC     (SP),R2
8686 032046  040316                          BIC     R3,(SP)                      ; SAVE,*
8687 032050  050216                          BIS     R2,(SP)
8688 032052  010346                          MOV     R3,-(SP)                     ; SAVE,*
8689 032054  016746  163752                  MOV     ML.DUT,-(SP)
8690 032060  012746  006506                  MOV     #FMT.2,-(SP)
8691 032064  012746  000004                  MOV     #4,-(SP)
8692 032070  010600                          MOV     SP,R0                        ; SP,*
```

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
                                                                                                            29-Mar-1982 16:21:03    PA:<
 8697 032072  104414                           TRAP    14
 8698 032074  104463                           TRAP    63
 8699 032076  062706  000030                   ADD     #30,SP                  ;                                                   4885
 8700 032102  000513                           BR      9$                      ;                                                   4887
 8701 032104  005002                    2$:    CLR     R2                      ; DRV.SEL                                           4894
 8702 032106  020267  163720            3$:    CMP     R2,ML.DUT               ; DRV.SEL,*                                         4896
 8703 032112  001425                           BEQ     7$
 8704 032114  010201                           MOV     R2,R1                   ; DRV.SEL,*                                         4899
 8705 032116  042701  177770                   BIC     #177770,R1
 8706 032122  142777  000007  163314           BICB    #7,@ML.REG+40
 8707 032130  150177  163310                   BISB    R1,@ML.REG+40
 8708 032134  010277  163274                   MOV     R2,@ML.REG+30           ; DRV.SEL,*                                         4900
 8709 032140  012700  000001                   MOV     #1,R0                   ; *,$$TMP2                                          4901
 8710 032144  001410                    4$:    BEQ     7$
 8711 032146  016701  147744                   MOV     L$DLY,R1                ; *,$$TMP1
 8712 032152  001403                           BEQ     6$
 8713 032154  005016                    5$:    CLR     (SP)                    ; $$TMP
 8714 032156  005301                           DEC     R1                      ; $$TMP1
 8715 032160  001375                           BNE     5$
 8716 032162  005300                    6$:    DEC     R0                      ; $$TMP2
 8717 032164  000767                           BR      4$
 8718 032166  005202                    7$:    INC     R2                      ; DRV.SEL                                           4894
 8719 032170  020227  000007                   CMP     R2,#7                   ; DRV.SEL,*
 8720 032174  003744                           BLE     3$
 8721 032176  016702  163630                   MOV     ML.DUT,R2               ;                                                   4904
 8722 032202  042702  177770                   BIC     #177770,R2
 8723 032206  142777  000007  163230           BICB    #7,@ML.REG+40
 8724 032214  150277  163224                   BISB    R2,@ML.REG+40
 8725 032220  017703  163210                   MOV     @ML.REG+30,R3           ; *,SAVE                                            4905
 8726 032224  020367  163602                   CMP     R3,ML.DUT               ; SAVE,*                                            4907
 8727 032230  001433                           BEQ     8$
 8728 032232  104455                           TRAP    55                      ;                                                   4910
 8729 032234  000003                           .WORD   3
 8730 032236  012706                           .WORD   ASYNC
 8731 032240  026302                           .WORD   DUMPER
 8732 032242  012746  010664                   MOV     #WRD.14,-(SP)           ;                                                   4911
 8733 032246  012746  011122                   MOV     #WRD.37,-(SP)
 8734 032252  012746  012172                   MOV     #FNC.3,-(SP)
 8735 032256  012746  010256                   MOV     #THR.FMT,-(SP)
 8736 032262  012746  000004                   MOV     #4,-(SP)
 8737 032266  010600                           MOV     SP,R0                   ; SP,*
 8738 032270  104414                           TRAP    14
 8739 032272  010316                           MOV     R3,(SP)                 ; SAVE,*                                            4912
 8740 032274  016746  163532                   MOV     ML.DUT,-(SP)
 8741 032300  012746  006442                   MOV     #FMT.1,-(SP)
 8742 032304  012746  000003                   MOV     #3,-(SP)
 8743 032310  010600                           MOV     SP,R0                   ; SP,*
 8744 032312  104414                           TRAP    14
 8745 032314  062706  000020                   ADD     #20,SP                  ;                                                   4909
 8746 032320  104467                    8$:    TRAP    67                      ;                                                   4913
 8747 032322  006000                           ROR     R0
 8748 032324  103002                           BHIS    9$
```

```
                                ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
                                ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
 8753 032326  000167  177356                   JMP     1$
 8754 032332  005726            9$:     TST     (SP)+
 8755 032334  000207                    RTS     PC                              ;               4855
 8756
 8757                           ; Routine Size:  142 words
 8758                           ; Maximum stack depth per invocation:  17 words
 8763
 8764
 8768
 8769                                   .SBTTL  T4 TEST CODE SECTION
 8773 032336                    T4::
 8774 032336  004767  177340    1$:     JSR     PC,$T4                          ;               4915
 8775 032342  104466                    TRAP    66
 8776 032344  006000                    ROR     R0
 8777 032346  103773                    BLO     1$
 8778 032350  000207                    RTS     PC
 8779
 8780                           ; Routine Size:  6 words
 8781                           ; Maximum stack depth per invocation:  0 words
 8786
 8787
 8788 ;          4917  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16.21:03 PA:<NEALE>ML4AD.BLI.4 (57)

```
8790 ;ML4AD
8791 ;                  TEST CODE SECTION
8792
8793 ;          4918  !
8794 ;          4919  BGNTST;
8795 ;          4920
8796 ;          4921  !++
8797 ;          4922  ! TEST NUMBER:  TST 5
8798 ;          4923  !
8799 ;          4924  ! TEST NAME:    READ WRITE REG ONES/ZEROES TEST
8800 ;          4925  !
8801 ;          4926  ! TEST DESCRIPTION:
8802 ;          4927  !       THIS TEST WRITES AND READS A DATA PATTERN OF ALL ONES AND ZEROES TO ALL
8803 ;          4928  !       OF THE ML11'S READ / WRITE REGISTERS.
8804 ;          4929  !
8805 ;          4930  !       ROUTINES WRT_REG AND RD_REG ACCEPT ARGUMENTS TO FURTHER SELECT ROUTINES
8806 ;          4931  !       WHICH ACTUALLY PERFORMS THE READING AND WRITING OF THE REGISTERS.
8807 ;          4932  !
8808 ;          4933  !       THE UNIT IS DROPPED ON DETECTED ERRORS.
8809 ;          4934  !
8810 ;          4935  ! IMPLICIT INPUTS:
8811 ;          4936  !       WT_DATA
8812 ;          4937  !       LOADED BY READ REGISTER ROUTINES AND CONTAINS THE DATA PATTERN WRITTEN
8813 ;          4938  !       TO THE REGISTERS (REPRESENTS GOOD DATA).
8814 ;          4939  !
8815 ;          4940  !       RD_DATA
8816 ;          4941  !       LOADED BY THE READ REGISTER ROUTINES AND CONTAINS THE DATA PATTERN
8817 ;          4942  !       READ FROM THE REGISTER (REPRESENTS BAD DATA).
8818 ;          4943  !--
8819 ;          4944
8820 ;          4945  local
8821 ;          4946      ERR_FLG,                        !ERROR FLAG PASSED TO ROUTINES
8822 ;          4947      TST_PAT,                        !TEST PATTERN
8823 ;          4948      index,                          !POINTS TO REGISTER PRESENTLY BEING TESTED
8824 ;          4949      DODU_FLG;                       !DROP UNIT FLAG
8825 ;          4950
8826 ;          4951  CLR_THRESHOLD;                      !CLEAR ERROR PRINT THRESHOLD
8827 ;          4952  DODU_FLG = ZERO;
8828 ;          4953  TST_PAT = ONES;                     !LOAD TEST PAT WITH ONES
8829 ;          4954
8830 ;          4955  incr TWICE from 0 to 1 do          !REPEAT LOOP TWICE
8831 ;          4956      begin
8832 ;          4957
8833 ;          4958      incr REG_SEL from 0 to 7 do     !TEST ELEVEN WRITE/READ REGISTERS
8834 ;          4959          begin
8835 ;          4960          BGNSUB;
8836 ;          4961          CLR_MBUS;
8837 ;          4962          WRT_REG (.TST_PAT, .REG_SEL, index);    !WRITE TO THE REGISTER
8838 ;          4963          RD_REG (.TST_PAT, .REG_SEL, ERR_FLG);   !READ THE REGISTER
8839 ;          4964
8840 ;          4965          if .ERR_FLG IS_SET              !SEE IF READ FOUND AN ERROR
8841 ;          4966          then
8842 ;          4967              begin                       !IF ERROR FLAG IS SET THEN ERROR AND SET DODU_FLG
8843 ;          4968              CMP_THRESHOLD;              !COMPARE ERROR PRINT THRESHOLD
8844 ;          4969
```

```
;ML4AD
;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (57)
;
;          4970              selectone .REG_SEL of                  !SELECT WHICH MODULE FAILED
;          4971                  set
;          4972
;          4973                  [0 to 3] :
;          4974                      ERRDF (4, ASYNC, DUMPER);      !ASYNC MODULE FAILURE
;          4975
;          4976                  [4 to 5] :
;          4977                      ERRDF (4, SYNC, DUMPER);       !SYNC MODULE FAILURE
;          4978
;          4979                  [6 to 7] :
;          4980                      ERRDF (4, ARR_DAT, DUMPER);    !ARRAY DATA MODULE FAILURE
;          4981                  tes;
;          4982
;          4983              PRINTB (SIX_FMT, PHR_4, WRD_12, FNC_5, FNC_6, WRD_52, WRD_56);
;          4984              PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .WT_DATA, .RD_DATA);
;          4985              DODU_FLG = ONE;
;          4986              end;
;          4987
;          4988          ENDSUB;
;          4989          end;
;          4990
;          4991      TST_PAT = not .TST_PAT;                        !REPEAT AGAIN WITH COMPLIMENT DATA
;          4992      end;
;          4993
;          4994  if .DODU_FLG IS_SET                               !DROP THIS UNIT IF THE DODU_FLG IS_SET
;          4995  then
;          4996      begin
;          4997      DODU (.ML_LUN);
;          4998      DOCLN;
;          4999      end;
;          5000
;          5001  ENDTST;
;
;                                        .SBTTL  $T5 TEST CODE SECTION
032352 004167 152214          $T5:      JSR     R1,$SAVE4                ;                                  4916
032356 024646                           CMP     -(SP),-(SP)
032360 005067 163012                    CLR     P.CNT                    ;                                  4949
032364 005004                           CLR     R4                       ; DODU.FLG                         4952
032366 012702 177777                    MOV     #-1,R2                   ; *,TST.PAT                        4953
032372 005001                           CLR     R1                       ; TWICE                            4955
032374 005003                 1$:       CLR     R3                       ; REG.SEL                          4958
032376 104402                 2$:       TRAP    2                        ;                                  4959
032400 152777 000040 163036             BISB    #40,@ML.REG+40           ;                                  4960
032406 016700 163420                    MOV     ML.DUT,R0
032412 042700 177770                    BIC     #177770,R0
032416 142777 000007 163020             BICB    #7,@ML.REG+40
```

```
                                                                                    29-Mar-1982 16:23:04    TOPS
                                                                                    29-Mar-1982 16:21:03    PA:<
 8902                                    ;ML4AD
 8903                                    ;                  TEST CODE SECTION
 8904
 8905 032424  150077  163014                      BISB    R0,@ML.REG+40
 8906 032430  010246                              MOV     R2,-(SP)
 8907 032432  010346                              MOV     R3,-(SP)                ; TST.PAT,*                     4962
 8908 032434  012746  000010                      MOV     #10,-(SP)               ; REG.SEL,*
 8909 032440  060616                              ADD     SP,(SP)
 8910 032442  004767  172656                      JSR     PC,WRT.REG              ; INDEX,*
 8911 032446  010216                              MOV     R2,(SP)
 8912 032450  010346                              MOV     R3,-(SP)                ; TST.PAT,*                     4963
 8913 032452  012746  000012                      MOV     #12,-(SP)               ; REG.SEL,*
 8914 032456  060616                              ADD     SP,(SP)
 8915 032460  004767  173226                      JSR     PC,RD.REG               ; ERR.FLG,*
 8916 032464  026627  000012  000001              CMP     12(SP),#1               ; ERR.FLG,*                     4965
 8917 032472  001117                              BNE     7$
 8918 032474  005267  162676                      INC     P.CNT                   ;                               4967
 8919 032500  026767  162672  162672              CMP     P.CNT,LIMIT
 8920 032506  003403                              BLE     3$
 8921 032510  062706  000012                      ADD     #12,SP
 8922 032514  000513                              BR      8$
 8923 032516  005703                      3$:     TST     R3                      ; REG.SEL                       4970
 8924 032520  002410                              BLT     4$
 8925 032522  020327  000003                      CMP     R3,#3                   ; REG.SEL,*
 8926 032526  003005                              BGT     4$
 8927 032530  104455                              TRAP    55                      ;                               4974
 8928 032532  000004                              .WORD   4
 8929 032534  012706                              .WORD   ASYNC
 8930 032536  026302                              .WORD   DUMPER
 8931 032540  000425                              BR      6$                      ;
 8932 032542  020327  000004              4$:     CMP     R3,#4                   ;                               4970
 8933 032546  002410                              BLT     5$                      ; REG.SEL,*
 8934 032550  020327  000005                      CMP     R3,#5                   ; REG.SEL,*
 8935 032554  003005                              BGT     5$
 8936 032556  104455                              TRAP    55                      ;                               4977
 8937 032560  000004                              .WORD   4
 8938 032562  012750                              .WORD   SYNC
 8939 032564  026302                              .WORD   DUMPER
 8940 032566  000412                              BR      6$                      ;
 8941 032570  020327  000006              5$:     CMP     R3,#6                   ;                               4970
 8942 032574  002407                              BLT     6$                      ; REG.SEL,*
 8943 032576  020327  000007                      CMP     R3,#7                   ; REG.SEL,*
 8944 032602  003004                              BGT     6$
 8945 032604  104455                              TRAP    55                      ;                               4980
 8946 032606  000004                              .WORD   4
 8947 032610  013012                              .WORD   ARR.DAT
 8948 032612  026302                              .WORD   DUMPER
 8949 032614  012746  011344              6$:     MOV     #WRD.56,-(SP)           ;                               4983
 8950 032620  012746  011310                      MOV     #WRD.52,-(SP)
 8951 032624  012746  012226                      MOV     #FNC.6,-(SP)
 8952 032630  012746  012216                      MOV     #FNC.5,-(SP)
 8953 032634  012746  010650                      MOV     #WRD.12,-(SP)
 8954 032640  012746  011676                      MOV     #PHR.4,-(SP)
 8955 032644  012746  010322                      MOV     #SIX.FMT,-(SP)
 8956 032650  012746  000007                      MOV     #7,-(SP)
```

SEQ 0208


```
                        ;ML4AD
                        ;               TEST CODE SECTION
                                MOV     SP,R0           ; SP,*
                                TRAP    14
                                MOV     RD.DATA,(SP)    ;                       4984
                                MOV     WT.DATA,-(SP)
                                MOV     36(SP),R0       ; INDEX,*
                                ASL     R0
                                ASL     R0
                                ASL     R0
                                MOV     ML.REG(R0),-(SP)
                                MOV     #FMT.16,-(SP)
                                MOV     #4,-(SP)
                                MOV     SP,R0           ; SP,*
                                TRAP    14
                                MOV     #1,R4           ; *,DODU.FLG            4985
                                ADD     #30,SP          ;                       4967
                        7$:     ADD     #12,SP          ;                       4959
                                TRAP    67              ;                       4986
                                ROR     R0
                                BLO     2$
                        8$:     INC     R3              ; REG.SEL               4958
                                CMP     R3,#7           ; REG.SEL,*
                                BLE     2$
                                COM     R2              ; TST.PAT               4991
                                INC     R1              ; TWICE                 4955
                                CMP     R1,#1           ; TWICE,*
                                BLE     1$
                                DEC     R4              ; DODU.FLG              4994
                                BNE     9$
                                MOV     ML.LUN,R0       ;                       4997
                                TRAP    51
                                TRAP    44
                        9$:     CMP     (SP)+,(SP)+     ;                       4916
                                RTS     PC

                        ; Routine Size:  142 words
                        ; Maximum stack depth per invocation:  24 words


                                .SBTTL  T5 TEST CODE SECTION
                        T5::
                        1$:     JSR     PC,$T5          ;                       4999
```

SEQ 0209


```
                                        ;ML4AD
                                        ;               TEST CODE SECTION

 9017 033012  104466                            TRAP    66
 9018 033014  006000                            ROR     R0
 9019 033016  103773                            BLO     1$
 9020 033020  000207                            RTS     PC
 9021
 9022                                   ; Routine Size:  6 words
 9023                                   ; Maximum stack depth per invocation:  0 words
 9028
 9029
 9030 ;          5002  !<BLF/PAGE>
```

```
;ML4AD
;                    TEST CODE SECTION

;          5003  !
;          5004  BGNTST;
;          5005
;          5006  !++
;          5007  ! TEST NUMBER:  TST 6
;          5008  !
;          5009  ! TEST NAME:    READ WRITE REG SHIFTING ONES/ZEROES TEST
;          5010  !
;          5011  ! TEST DESCRIPTION:
;          5012  !       THIS TEST WRITES AND READS A
;          5013  !       SHIFTING ONE'S AND SHIFTING ZEROE'S
;          5014  !       PATTERN TO ALL THE ML11'S
;          5015  !       READ/WRITE REGISTERS
;          5016  !
;          5017  !       ROUTINES WRT_REG AND RD_REG
;          5018  !       ACCEPT ARGUMENTS TO FURTHER
;          5019  !       SELECT ROUTINES WHICH ACTUALLY
;          5020  !       PERFORMS THE READING AND
;          5021  !       WRITING OF THE REGISTERS.
;          5022  !
;          5023  !       THE DRIVE IS DROPPED ON DETECTED
;          5024  !       ERRORS.
;          5025  !
;          5026  ! IMPLICIT INPUTS:
;          5027  !       WT_DATA
;          5028  !         LOADED BY READ REGISTER ROUTINES AND
;          5029  !         CONTAINS THE DATA PATTERN WRITTEN TO THE
;          5030  !         REGISTERS (REPRESENTS GOOD DATA).
;          5031  !
;          5032  !       RD_DATA
;          5033  !         LOADED BY THE READ REGISTER ROUTINES AND
;          5034  !         CONTAINS THE DATA PATTERN READ FROM THE
;          5035  !         REGISTER (REPRESENTS BAD DATA).
;          5036  !
;          5037  !
;          5038  !--
;          5039
;          5040  local
;          5041      ERR_FLG,                        !ERROR FLAG PASSED TO ROUTINE
;          5042      TST_PAT,                        !TEST PATTERN
;          5043      index,                          !POINTS TO REG PRESENTLY BEING TESTED
;          5044      DODU_FLG;                       !DROP UNIT FLAG
;          5045
;          5046  CLR_THRESHOLD;                     !CLEAR ERROR PRINT THRESHOLD
;          5047  DODU_FLG = ZERO;
;          5048  TST_PAT = ONE;                      !LOAD TST_PAT WITH A 1 IN A FILED OF 0'S
;          5049
;          5050  incr SHIFT from 0 to 15 do         !DO SHIFT 16 TIMES
;          5051      begin
;          5052
;          5053      incr TWICE from 0 to 1 do       !REPEAT LOOP TWICE
;          5054          begin
```

```
9088 ;ML4AD
9089 ;                  TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
9090                                                         29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (58)
9091 ;          5055
9092 ;          5056          incr REG_SEL from 0 to 7 do          !TEST ELEVEN READ/WRITE REGISTERS
9093 ;          5057              begin
9094 ;          5058              BGNSUB;
9095 ;          5059              CLR_MBUS;
9096 ;          5060              WRT_REG (.TST_PAT, .REG_SEL, index);      !WRITE TO THE REGISTER
9097 ;          5061              RD_REG (.TST_PAT, .REG_SEL, ERR_FLG);     !READ THE REGISTER
9098 ;          5062
9099 ;          5063              if .ERR_FLG IS_SET                   !SEE IF THE READ FOUND AN ERROR
9100 ;          5064              then
9101 ;          5065                  begin                            !IF THE ERROR FLAG IS SET THEN ERROR
9102 ;          5066                  CMP_THRESHOLD;                   !COMPARE ERROR PRINT THRESHOLD
9103 ;          5067
9104 ;          5068                  selectone .REG_SEL of            !FIND WHICH MODULE FAILED
9105 ;          5069                      set
9106 ;          5070
9107 ;          5071                      [0 to 3] :
9108 ;          5072                          ERRDF (5, ASYNC, DUMPER);     !ASYNC MODULE FAILURE
9109 ;          5073
9110 ;          5074                      [4 to 5] :
9111 ;          5075                          ERRDF (5, SYNC, DUMPER);      !SYNC MODULE FAILURE
9112 ;          5076
9113 ;          5077                      [6 to 7] :
9114 ;          5078                          ERRDF (5, ARR_DAT, DUMPER);   !ARRAY DATA MODULE FAILURE
9115 ;          5079                      tes;
9116 ;          5080
9117 ;          5081                  PRINTB (SIX_FMT, PHR_4, WRD_12, FNC_5, FNC_6, WRD_52, WRD_56);
9118 ;          5082                  PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .WT_DATA, .RD_DATA);
9119 ;          5083                  DODU_FLG = ONE;
9120 ;          5084                  end;
9121 ;          5085
9122 ;          5086              ENDSUB;
9123 ;          5087              end;
9124 ;          5088
9125 ;          5089          TST_PAT = not .TST_PAT;              !REPEAT WITH A 0 IN A FIELD OF 1'S
9126 ;          5090          end;
9127 ;          5091
9128 ;          5092      TST_PAT = .TST_PAT^ONE;                  !SHIFT THE 1 IN THE FIELD OF 0'S
9129 ;          5093      end;
9130 ;          5094
9131 ;          5095  if .DODU_FLG IS_SET                          !DROP THIS UNIT IF DODU_FLG IS SET
9132 ;          5096  then
9133 ;          5097      begin
9134 ;          5098      DODU (.ML_LUN);
9135 ;          5099      DOCLN;
9136 ;          5100      end;
9137 ;          5101
9138 ;          5102  ENDTST;
9142
```

```
                                                                                29-Mar-1982 16:23:04    TOPS
                                                                                29-Mar-1982 16:21:03    PA:<
                        ;ML4AD
                        ;                       TEST CODE SECTION

                                        .SBTTL  $T6 TEST CODE SECTION
 9151 033022 004167 151564      $T6:    JSR     R1,$SAVE5
 9152 033026 024646                     CMP     -(SP),-(SP)                     ;                       5001
 9153 033030 005067 162342              CLR     P.CNT
 9154 033034 005005                     CLR     R5                              ;                       5044
 9155 033036 012703 000001              MOV     #1,R3                           ; DODU.FLG              5047
 9156 033042 005001                     CLR     R1                              ; *,TST.PAT             5048
 9157 033044 005002             1$:     CLR     R2                              ; SHIFT                 5050
 9158 033046 005004             2$:     CLR     R4                              ; TWICE                 5053
 9159 033050 104402             3$:     TRAP    2                               ; REG.SEL               5056
 9160 033052 152777 000040 162364       BISB    #40,@ML.REG+40                  ;                       5057
 9161 033060 016700 162746              MOV     ML.DUT,R0                       ;                       5058
 9162 033064 042700 177770              BIC     #177770,R0
 9163 033070 142777 000007 162346       BICB    #7,@ML.REG+40
 9164 033076 150077 162342              BISB    R0,@ML.REG+40
 9165 033102 010346                     MOV     R3,-(SP)                        ; TST.PAT,*             5060
 9166 033104 010446                     MOV     R4,-(SP)                        ; REG.SEL,*
 9167 033106 012746 000010              MOV     #10,-(SP)
 9168 033112 060616                     ADD     SP,(SP)                         ; INDEX,*
 9169 033114 004767 172204              JSR     PC,WRT.REG
 9170 033120 010316                     MOV     R3,(SP)                         ; TST.PAT,*             5061
 9171 033122 010446                     MOV     R4,-(SP)                        ; REG.SEL,*
 9172 033124 012746 000012              MOV     #12,-(SP)
 9173 033130 060616                     ADD     SP,(SP)                         ; ERR.FLG,*
 9174 033132 004767 172554              JSR     PC,RD.REG
 9175 033136 026627 000012 000001       CMP     12(SP),#1                       ; ERR.FLG,*             5063
 9176 033144 001117                     BNE     8$
 9177 033146 005267 162224              INC     P.CNT                           ;                       5065
 9178 033152 026767 162220 162220       CMP     P.CNT,LIMIT
 9179 033160 003403                     BLE     4$
 9180 033162 062706 000012              ADD     #12,SP
 9181 033166 000513                     BR      9$
 9182 033170 005704             4$:     TST     R4                              ; REG.SEL               5068
 9183 033172 002410                     BLT     5$
 9184 033174 020427 000003              CMP     R4,#3                           ; REG.SEL,*
 9185 033200 003005                     BGT     5$
 9186 033202 104455                     TRAP    55                              ;                       5072
 9187 033204 000005                     .WORD   5
 9188 033206 012706                     .WORD   ASYNC
 9189 033210 026302                     .WORD   DUMPER
 9190 033212 000425                     BR      7$
 9191 033214 020427 000004      5$:     CMP     R4,#4                           ;                       5068
 9192 033220 002410                     BLT     6$                              ; REG.SEL,*
 9193 033222 020427 000005              CMP     R4,#5
 9194 033226 003005                     BGT     6$                              ; REG.SEL,*
 9195 033230 104455                     TRAP    55
 9196 033232 000005                     .WORD   5                               ;                       5075
 9197 033234 012750                     .WORD   SYNC
 9198 033236 026302                     .WORD   DUMPER
```

```
                                  ;ML4AD                                          29-Mar-1982 16:23:04   TOPS
                                  ;          TEST CODE SECTION                    29-Mar-1982 16:21:03   PA:<
033240  000412                            BR      7$                  ;
033242  020427  000006            6$:     CMP     R4,#6               ; REG SEL,*                            5068
033246  002407                            BLT     7$
033250  020427  000007                    CMP     R4,#7               ; REG.SEL,*
033254  003004                            BGT     7$
033256  104455                            TRAP    55                  ;                                      5078
033260  000005                            .WORD   5
033262  013012                            .WORD   ARR.DAT
033264  026302                            .WORD   DUMPER
033266  012746  011344            7$:     MOV     #WRD.56,-(SP)       ;                                      5081
033272  012746  011310                    MOV     #WRD.52,-(SP)
033276  012746  012226                    MOV     #FNC.6,-(SP)
033302  012746  012216                    MOV     #FNC.5,-(SP)
033306  012746  010650                    MOV     #WRD.12,-(SP)
033312  012746  011676                    MOV     #PHR.4,-(SP)
033316  012746  010322                    MOV     #SIX.FMT,-(SP)
033322  012746  000007                    MOV     #7,-(SP)
033326  010600                            MOV     SP,R0               ; SP,*
033330  104414                            TRAP    14
033332  016716  162014                    MOV     RD.DATA,(SP)        ;                                      5082
033336  016746  162006                    MOV     WT.DATA,-(SP)
033342  016600  000036                    MOV     36(SP),R0           ; INDEX,*
033346  006300                            ASL     R0
033350  006300                            ASL     R0
033352  006300                            ASL     R0
033354  016046  015404                    MOV     ML.REG(R0),-(SP)
033360  012746  007436                    MOV     #FMT.16,-(SP)
033364  012746  000004                    MOV     #4,-(SP)
033370  010600                            MOV     SP,R0               ; SP,*
033372  104414                            TRAP    14
033374  012705  000001                    MOV     #1,R5               ; *,DODU.FLG
033400  062706  000030                    ADD     #30,SP              ;                                      5083
033404  062706  000012            8$:     ADD     #12,SP              ;                                      5065
033410  104467                            TRAP    67                  ;                                      5057
033412  006000                            ROR     R0                  ;                                      5084
033414  103615                            BLO     3$
033416  005204                    9$:     INC     R4                  ; REG.SEL                              5056
033420  020427  000007                    CMP     R4,#7               ; REG.SEL,*
033424  003611                            BLE     3$
033426  005103                            COM     R3                  ; TST.PAT                              5089
033430  005202                            INC     R2                  ; TWICE                                5053
033432  020227  000001                    CMP     R2,#1               ; TWICE,*
033436  003603                            BLE     2$
033440  006303                            ASL     R3                  ; TST.PAT                              5092
033442  005201                            INC     R1                  ; SHIFT                                5050
033444  020127  000017                    CMP     R1,#17              ; SHIFT,*
033450  003002                            BGT     10$
033452  000167  177366                    JMP     1$
033456  005305                    10$:    DEC     R5                  ; DODU.FLG                             5095
033460  001004                            BNE     11$
033462  016700  162342                    MOV     ML.LUN,R0           ;                                      5098
033466  104451                            TRAP    51
```

```
                                ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
                                ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<

033470  104444                          TRAP    44
033472  022626                  11$:    CMP     (SP)+,(SP)+                     ;       5001
033474  000207                          RTS     PC

                                ; Routine Size:  150 words
                                ; Maximum stack depth per invocation:  25 words


                                        .SBTTL  T6 TEST CODE SECTION
033476                          T6::
033476  004767  177320          1$:     JSR     PC,$T6                          ;       5100
033502  104466                          TRAP    66
033504  006000                          ROR     R0
033506  103773                          BLO     1$
033510  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words

;       5103    !
;       5104    !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (59)

```
;ML4AD
;                    TEST CODE SECTION

;                5105  !
;                5106  BGNTST;
;                5107
;                5108  !++
;                5109  ! TEST NUMBER:  TST 7
;                5110  !
;                5111  ! TEST NAME:    REGISTER INITIALIZATION TEST
;                5112  !
;                5113  ! TEST DESCRIPTION:
;                5114  !       THIS TEST TESTS THE ABILITY OF
;                5115  !       ALL ACCESSIBLE ML11 REGISTERS
;                5116  !       TO CLEAR OUT REGISTER DATA OF
;                5117  !       ONE'S AND ZEROES PATTERN.
;                5118  !
;                5119  !       ROUTINE WRT_REG WRITES A
;                5120  !       PATTERN TO THE SELECTED REGISTER.
;                5121  !
;                5122  !       A MASS BUS CLEAR IS DONE.
;                5123  !
;                5124  !       THEN ROUTINE RD_REG READS THE
;                5125  !       SELECTED REGISTER FOR CLEARED DATA
;                5126  !       THE DRIVE IS DROPPED ON DETECTED ERRORS.
;                5127  !
;                5128  !       THIS TEST WILL ALSO READ THE
;                5129  !       DRIVE TYPE REGISTER FOR ITS
;                5130  !       INITIAL REGISTER VALUE.
;                5131  !
;                5132  ! IMPLICIT INPUTS:
;                5133  !       THIS FLAG IS NEEDED DUE TO THE UNIQUE
;                5134  !       MANNER IN WHICH THESE REGISTERS
;                5135  !       MUST BE READ.
;                5136  !
;                5137  !       RD_DATA
;                5138  !       LOADED BY THE READ REGISTER ROUTINE
;                5139  !       AND CONTAINS THE DATA PATTERN READ FROM
;                5140  !       THE REGISTER (REPRESENTS BAD DATA).
;                5141  !
;                5142  !       DRIVE TYPE
;                5143  !       LOADED DURING THE INITIALIZATION CODE AND
;                5144  !       STORES THE EXPECTED CONTENTS OF THE DRIVE
;                5145  !       TYPE REGISTER.
;                5146  !--
;                5147
;                5148  local
;                5149      TST_PAT,                    !TEST PATTERN
;                5150      ERR_FLG,                    !ERROR FLAG PASSED TO ROUTINE
;                5151      index,                      !POINTS TO REG PRESENTLY BEING TESTED
;                5152      CLR_DATA,                   !STORES CALCULATED REGISTER CLEAR DATA
;                5153      SAVE,                       !TEMPORARY STORAGE LOCATION
;                5154      DODU_FLG;                   !DROP UNIT FLAG
;                5155
;                5156  CLR_THRESHOLD;                  !CLEAR ERROR PRINT THRESHOLD
```

```
;ML4AD
;                    TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (59)
;
;          5157  ML_REG [19, FORCE_HI] = %o'177777';                   !CLR DATA FOR MLPD IS ONES
;          5158  DODU_FLG = ZERO;
;          5159  TST_PAT = ONES;                                       !BACKGROUND PATTERN
;          5160
;          5161  incr TWICE from 0 to 1 do                             !REPEAT LOOP TWICE
;          5162      begin
;          5163
;          5164      incr REG_SEL from 0 to 10 do                      !TEST THIRTEEN REGISTERS
;          5165          begin
;          5166          BGNSUB;
;          5167          CLR_MBUS;
;          5168          WRT_REG (.TST_PAT, .REG_SEL, index);          !WRITE REGISTER WITH BACKGROUND
;          5169          CLR_DATA = (.AI) or (.IGNORE);                !CALCULATE THE CLEARED DATA PATTERN
;          5170          CLR_MBUS;                                     !CLEAR THE REGISTER
;          5171          RD_REG (.CLR_DATA, .REG_SEL, ERR_FLG);        !READ THE REGISTER FOR THE CLEARED DATA PAT
;          5172
;          5173          if .ERR_FLG IS_SET                            !SEE IF READ FOUND AN ERROR
;          5174          then
;          5175              begin                                     !IF ERROR FLAG IS SET THEN ERROR AND SET DODU_FLG
;          5176              CMP_THRESHOLD;                            !COMPARE ERROR PRINT THRESHOLD
;          5177
;          5178              selectone .REG_SEL of                     !FIND WHICH MODULE FAILED
;          5179                  set
;          5180
;          5181                  [0, 1, 2, 3, 9, 10] :
;          5182                      ERRDF (6, ASYNC, DUMPER);         !ASYNC MODULE FAILURE
;          5183
;          5184                  [4, 5, 8] :
;          5185                      ERRDF (6, SYNC, DUMPER);          !SYNC MODULE FAILURE
;          5186
;          5187                  [6 to 7] :
;          5188                      ERRDF (6, ARR_DAT, DUMPER); !ARRAY DATA MODULE
;          5189                  tes;
;          5190
;          5191              PRINTB (FIV_FMT, PHR_4, WRD_12, WRD_52, FNC_23, WRD_56);
;          5192              PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .CLR_DATA, .RD_DATA);
;          5193              DODU_FLG = ONE;
;          5194              end;
;          5195
;          5196          ENDSUB;
;          5197          end;
;          5198
;          5199      TST_PAT = not .TST_PAT;                           !REPEAT WITH COMPLIMENT BACKGROUND PAT
;          5200      end;
;          5201
;          5202  ML_REG [19, FORCE_HI] = ZEROES;                      !RESTORE MLPD FORCED_HI
;          5203  !
;          5204  !
;          5205  !  NOW TEST THE DRIVE TYPE REGISTER
;          5206  !
;          5207  !
;          5208
```

```
 9409 ;ML4AD
 9410 ;                  TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 9411                                                         29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (59)
 9412 ;          5209  if .MLDT neq .DRIVE_TYPE               !COMPARE REG CONTENTS TO CALCULATED VALUE
 9413 ;          5210  then
 9414 ;          5211      begin
 9415 ;          5212      ERRDF (113, ASYNC, DUMPER);
 9416 ;          5213      PRINTB (TWO_FMT, REG_7, PHR_4);
 9417 ;          5214      PRINTB (FMT_2, .DRIVE_TYPE, .SAVE);
 9418 ;          5215      end;
 9419 ;          5216
 9420 ;          5217  if .DODU_FLG IS_SET                    !DROP THIS UNIT IF DODU_FLG IS_SET
 9421 ;          5218  then
 9422 ;          5219      begin
 9423 ;          5220      DODU (.ML_LUN);
 9424 ;          5221      DOCLN;
 9425 ;          5222      end;
 9426 ;          5223
 9427 ;          5224  ENDTST;
 9431
 9432                                        .SBTTL  $T7 TEST CODE SECTION
 9436 033512 004167 151074          $T7:    JSR     R1,$SAVE5                ;                          5102
 9437 033516 162706 000006                  SUB     #6,SP
 9438 033522 005067 161650                  CLR     P.CNT                    ;                          5154
 9439 033526 012767 177777 162102           MOV     #-1,ML.REG+232           ;                          5157
 9440 033534 005005                         CLR     R5                       ; DODU.FLG                 5158
 9441 033536 012702 177777                  MOV     #-1,R2                   ; *.TST.PAT                5159
 9442 033542 005001                         CLR     R1                       ; TWICE                    5161
 9443 033544 005004                 1$:     CLR     R4                       ; REG.SEL                  5164
 9444 033546 104402                 2$:     TRAP    2                        ;                          5165
 9445 033550 152777 000040 161666           BISB    #40,@ML.REG+40           ;                          5166
 9446 033556 016703 162250                  MOV     ML.DUT,R3
 9447 033562 042703 177770                  BIC     #177770,R3
 9448 033566 142777 000007 161650           BICB    #7,@ML.REG+40
 9449 033574 150377 161644                  BISB    R3,@ML.REG+40
 9450 033600 010246                         MOV     R2,-(SP)                 ; TST.PAT,*                5168
 9451 033602 010446                         MOV     R4,-(SP)                 ; REG.SEL,*
 9452 033604 012746 000012                  MOV     #12,-(SP)
 9453 033610 060616                         ADD     SP,(SP)                  ; INDEX,*
 9454 033612 004767 171506                  JSR     PC,WRT.REG
 9455 033616 016600 000012                  MOV     12(SP),R0                ; INDEX,*                  5169
 9456 033622 006300                         ASL     R0
 9457 033624 006300                         ASL     R0
 9458 033626 006300                         ASL     R0
 9459 033630 010003                         MOV     R0,R3
 9460 033632 016300 015406                  MOV     ML.REG+2(R3),R0
 9461 033636 056300 015412                  BIS     ML.REG+6(R3),R0
 9462 033642 010066 000006                  MOV     R0,6(SP)
 9463 033646 152777 000040 161570           BISB    #40,@ML.REG+40           ; *,CLR.DATA
```

```
                                        ;ML4AD
                                        ;                TEST CODE SECTION
033654  016700  162152                          MOV     ML.DUT,R0
033660  042700  177770                          BIC     #177770,R0
033664  142777  000007  161552                  BICB    #7,@ML.REG+40
033672  150077  161546                          BISB    R0,@ML.REG+40
033676  016616  000006                          MOV     6(SP),(SP)
033702  010446                                  MOV     R4,-(SP)          ; CLR.DATA,*                    5171
033704  012746  000014                          MOV     #14,-(SP)         ; REG.SEL,*
033710  060616                                  ADD     SP,(SP)
033712  004767  171774                          JSR     PC,RD.REG         ; ERR.FLG,*
033716  026627  000014  000001                  CMP     14(SP),#1         ; ERR.FLG,*                     5173
033724  001121                                  BNE     11$
033726  005267  161444                          INC     P.CNT             ;                               5175
033732  026767  161440  161440                  CMP     P.CNT,LIMIT
033740  003403                                  BLE     3$
033742  062706  000012                          ADD     #12,SP
033746  000517                                  BR      13$
033750  005704                          3$:     TST     R4                ; REG.SEL                       5178
033752  002403                                  BLT     4$
033754  020427  000003                          CMP     R4,#3             ; REG.SEL,*
033760  003406                                  BLE     5$
033762  020427  000011                  4$:     CMP     R4,#11            ; REG.SEL,*
033766  002410                                  BLT     6$
033770  020427  000012                          CMP     R4,#12            ; REG.SEL,*
033774  003005                                  BGT     6$
033776  104455                          5$:     TRAP    55                ;                               5182
034000  000006                                  .WORD   6
034002  012706                                  .WORD   ASYNC
034004  026302                                  .WORD   DUMPER
034006  000430                                  BR      10$               ;
034010  020427  000004                  6$:     CMP     R4,#4             ; REG.SEL,*                     5178
034014  002403                                  BLT     7$
034016  020427  000005                          CMP     R4,#5             ; REG.SEL,*
034022  003403                                  BLE     8$
034024  020427  000010                  7$:     CMP     R4,#10            ; REG.SEL,*
034030  001005                                  BNE     9$
034032  104455                          8$:     TRAP    55                ;                               5185
034034  000006                                  .WORD   6
034036  012750                                  .WORD   SYNC
034040  026302                                  .WORD   DUMPER
034042  000412                                  BR      10$               ;
034044  020427  000006                  9$:     CMP     R4,#6             ; REG.SEL,*                     5178
034050  002407                                  BLT     10$
034052  020427  000007                          CMP     R4,#7             ; REG.SEL,*
034056  003004                                  BGT     10$
034060  104455                                  TRAP    55                ;                               5188
034062  000006                                  .WORD   6
034064  013012                                  .WORD   ARR.DAT
034066  026302                                  .WORD   DUMPER
034070  012746  011344                  10$:    MOV     #WRD.56,-(SP)     ;                               5191
034074  012746  012464                          MOV     #FNC.23,-(SP)
034100  012746  011310                          MOV     #WRD.52,-(SP)
034104  012746  010650                          MOV     #WRD.12,-(SP)
```

```
 9521                                   ;ML4AD
 9522                                   ;               TEST CODE SECTION
 9523
 9524 034110  012746  011676                    MOV     #PHR.4,-(SP)
 9525 034114  012746  010304                    MOV     #FIV.FMT,-(SP)
 9526 034120  012746  000006                    MOV     #6,-(SP)
 9527 034124  010600                            MOV     SP,R0                   ; SP,*
 9528 034126  104414                            TRAP    14
 9529 034130  016716  161216                    MOV     RD.DATA,(SP)            ;                               5192
 9530 034134  016646  000030                    MOV     30(SP),-(SP)            ; CLR.DATA,*
 9531 034140  016346  015404                    MOV     ML.REG(R3),-(SP)
 9532 034144  012746  007436                    MOV     #FMT.16,-(SP)
 9533 034150  012746  000004                    MOV     #4,-(SP)
 9534 034154  010600                            MOV     SP,R0                   ; SP,*
 9535 034156  104414                            TRAP    14
 9536 034160  012705  000001                    MOV     #1,R5                   ; *,DODU.FLG                    5193
 9537 034164  062706  000026                    ADD     #26,SP                  ;                               5175
 9538 034170  062706  000012            11$:    ADD     #12,SP                  ;                               5165
 9539 034174  104467                            TRAP    67                      ;                               5194
 9540 034176  006000                            ROR     R0
 9541 034200  103002                            BHIS    13$
 9542 034202  000167  177340            12$:    JMP     2$
 9543 034206  005204            13$:    INC     R4                              ; REG.SEL                       5164
 9544 034210  020427  000012                    CMP     R4,#12                  ; REG.SEL,*
 9545 034214  003772                            BLE     12$
 9546 034216  005102                            COM     R2                      ; TST.PAT                       5199
 9547 034220  005201                            INC     R1                      ; TWICE                         5161
 9548 034222  020127  000001                    CMP     R1,#1                   ; TWICE,*
 9549 034226  003002                            BGT     14$
 9550 034230  000167  177310                    JMP     1$
 9551 034234  005067  161376            14$:    CLR     ML.REG+232              ;                               5202
 9552 034240  027767  161270  161106            CMP     @ML.REG+130,DRIVE.TYPE  ;                               5209
 9553 034246  001431                            BEQ     15$
 9554 034250  104455                            TRAP    55                      ;                               5212
 9555 034252  000161                            .WORD   161
 9556 034254  012706                            .WORD   ASYNC
 9557 034256  026302                            .WORD   DUMPER
 9558 034260  012746  011676                    MOV     #PHR.4,-(SP)            ;                               5213
 9559 034264  012746  012546                    MOV     #REG.7,-(SP)
 9560 034270  012746  010246                    MOV     #TWO.FMT,-(SP)
 9561 034274  012746  000003                    MOV     #3,-(SP)
 9562 034300  010600                            MOV     SP,R0                   ; SP,*
 9563 034302  104414                            TRAP    14
 9564 034304  010416                            MOV     R4,(SP)                 ; SAVE,*                        5214
 9565 034306  016746  161042                    MOV     DRIVE.TYPE,-(SP)
 9566 034312  012746  006506                    MOV     #FMT.2,-(SP)
 9567 034316  012746  000003                    MOV     #3,-(SP)
 9568 034322  010600                            MOV     SP,R0                   ; SP,*
 9569 034324  104414                            TRAP    14
 9570 034326  062706  000016                    ADD     #16,SP                  ;                               5211
 9571 034332  005305            15$:    DEC     R5                              ; DODU.FLG                      5217
 9572 034334  001004                            BNE     16$
 9573 034336  016700  161466                    MOV     ML.LUN,R0               ;                               5220
 9574 034342  104451                            TRAP    51
 9575 034344  104444                            TRAP    44
```

SEQ 0220

```
 9577                                   ;ML4AD
 9578                                   ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS
 9579                                   ;                                                                   29-Mar-1982 16:21:03    PA:<
 9580 034346  062706  000006            16$:    ADD      #6,SP                                 ;                                 5102
 9581 034352  000207                             RTS      PC
 9582
 9583                                   ; Routine Size:  209 words
 9584                                   ; Maximum stack depth per invocation:  25 words
 9589
 9590
 9594
 9595                                           .SBTTL   T7 TEST CODE SECTION
 9599 034354                            T7::
 9600 034354  004767  177132            1$:     JSR      PC,$T7                                ;                                 5222
 9601 034360  104466                             TRAP     66
 9602 034362  006000                             ROR      R0
 9603 034364  103773                             BLO      1$
 9604 034366  000207                             RTS      PC
 9605
 9606
 9607                                   ; Routine Size:  6 words
 9612                                   ; Maximum stack depth per invocation:  0 words
 9613
 9614 ;          5225  !<BLF/PAGE>
```

```
:ML4AD                                                  29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
:               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (60)

:               !
:               !
:               !
:               BGNTST;
:
:               !++
:               ! TEST NUMBER:  TST 8
:               !
:               ! TEST NAME:    REGISTER SELECTION TEST
:               !
:               ! TEST DESCRIPTION:
:               !       THIS TEST TESTS FOR UNIQUE REGISTER
:               !       SELECTION BY FIRST WRITING A BACKGROUND
:               !       PATTERN INTO ALL READ/WRITE REGISTERS
:               !
:               !       IT THEN WRITES A COMPLIMENT
:               !       BACKGROUND PATTERN INTO ONE REGISTER
:               !       AND READS THE REMAINING UNWRITTEN
:               !       REGISTERS FOR AN UNCHANGED BACKGROUND
:               !       PAT
:               !
:               !       THIS PROCEDURE IS REPEATED UNTIL ALL
:               !       REGISTERS HAVE BEEN WRITTEN WITH A
:               !       COMPLIMENT BACKGROUND PATTERN.
:               !
:               !--
:
:               local
:                   DODU_FLG,                           !DROP UNIT FLAG
:                   TST_PAT,                            !TEST PATTERN
:                   ERR_FLG,                            !ERROR FLAG PASSED TO ROUTINE;
:                   index;                              !POINTS TO REGISTER PRESENTLY BEING TESTED.
:
:               CLR_THRESHOLD;                          !CLEAR ERROR PRINT THRESHOLD
:               BGNSUB;
:               CLR_MBUS;
:               DODU_FLG = ZERO;
:               TST_PAT = %o'125252';                   !LOAD TST_PAT WITH ALTERNATE 1'S & 0'S
:               WRT_CS1 (.TST_PAT, 0);                  !WRITE A BACKGROUND INTO ALL THE DIRECTLY
:               WRT_ER (.TST_PAT, 6);                   !ACCESSABLE READ WRITE REGISTERS
:               WRT_DA (.TST_PAT, 3);
:               WRT_PA (.TST_PAT, 8);
:               WRT_E1 (.TST_PAT, 13);
:               WRT_E2 (.TST_PAT, 14);
:
:               incr CNT_1 from 0 to 4 do               !WRITE A COMPLIMENT PATTERN INTO ONE REGISTER
:                   begin
:                   TST_PAT = not .TST_PAT;             !GENERATE THE COMPLIMENT PAT
:
:                   case .CNT_1 from 0 to 4 of          !SELECT THE REGISTER TO WRITE INTO
:                       set
:
```

```
;ML4AD
;                    TEST CODE SECTION
;
;          5278          [0] :
;          5279              WRT_CS1 (.TST_PAT, 0);          !FIRST PASS WRITE COMP PAT TO MLCS1
;          5280
;          5281          [1] :
;          5282              WRT_ER (.TST_PAT, 6);           !SECOND PASS WRITE COMP PAT TO MLER
;          5283
;          5284          [2] :
;          5285              WRT_DA (.TST_PAT, 3);           !THIRD PASS WRITE COMP PAT TO MLDA
;          5286
;          5287          [3] :
;          5288              WRT_PA (.TST_PAT, 8);           !FORTH PASS WRITE COMP PAT TO MLPA
;          5289
;          5290          [4] :
;          5291              WRT_E1 (.TST_PAT, 13);          !FIFTH PASS WRITE COMP PAT TO MLE1
;          5292          tes;
;          5293
;          5294      TST_PAT = not .TST_PAT;                 !COMPLIMENT TST_PAT BACK TO BACKGROUND
;          5295
;          5296      incr CNT_2 from .CNT_1 + 1 to 5 do      !NOW READ THE REMAINING UNWRITTEN REGISTERS FOR AN
;          5297          begin                               !UNCHANGED BACKGROUND
;          5298
;          5299          case .CNT_2 from 0 to 5 of          !SELECT THE REGISTER TO READ
;          5300              set
;          5301
;          5302              [0] :
;          5303                  RD_CS1 (.TST_PAT, index = 0, ERR_FLG);  !READ MLCS1
;          5304
;          5305              [1] :
;          5306                  RD_ER (.TST_PAT, index = 6, ERR_FLG);   !READ MLER
;          5307
;          5308              [2] :
;          5309                  RD_DA (.TST_PAT, index = 3, ERR_FLG);   !READ MLDA
;          5310
;          5311              [3] :
;          5312                  RD_PA (.TST_PAT, index = 8, ERR_FLG);   !READ MLPA
;          5313
;          5314              [4] :
;          5315                  RD_E1 (.TST_PAT, index = 13, ERR_FLG);  !READ MLE1
;          5316
;          5317              [5] :
;          5318                  RD_E2 (.TST_PAT, index = 14, ERR_FLG);  !READ MLE2
;          5319              tes;
;          5320
;          5321      if .ERR_FLG IS_SET                      !SEE IF READ FOUND AN ERROR
;          5322      then
;          5323          begin
;          5324          CMP_THRESHOLD;                      !COMPARE ERROR PRINT THRESHOLD
;          5325
;          5326          selectone .CNT_2 of
;          5327              set
;          5328
;          5329              [0 to 2] :
```

```
9728 ;ML4AD
9729 ;                  TEST CODE SECTION
9730
9731 ;          5330                       ERRDF (110, ASYNC, DUMPER); !ASYNC MODULE FAILURE
9732 ;          5331
9733 ;          5332                   [3 to 5] .
9734 ;          5333                       ERRDF (110, ARR_DAT, DUMPER);       !ARRAY DATA MODULE FAILURE
9735 ;          5334                   tes;
9736 ;          5335
9737 ;          5336                PRINTB (THR_FMT, WRD_38, WRD_37, WRD_10);
9738 ;          5337                PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .WT_DATA, .RD_DATA);
9739 ;          5338                DODU_FLG = ONE;
9740 ;          5339                end;
9741 ;          5340
9742 ;          5341            end;
9743 ;          5342
9744 ;          5343        end;
9745 ;          5344
9746 ;          5345    ENDSUB;
9747 ;          534,
9748 ;          5347    if .DODU_FLG IS_SET                                 !DROP THIS UNIT IF DODU_FLG IS_SET
9749 ;          5348    then
9750 ;          5349        begin
9751 ;          5350        DODU (.ML_LUN);
9752 ;          5351        DOCLN;
9753 ;          5352        end;
9754 ;          5353
9755 ;          5354    ENDTST;
9759
9760
                                .SBTTL  $T8 TEST CODE SECTION
9764 034370 004167 150216       $T8:    JSR     R1,$SAVE5                   ;                                   5224
9765 034374 005746              TST     -(SP)
9766 034376 005067 160774               CLR     P.CNT                       ;                                   5257
9767 034402 104402              1$:     TRAP    2                           ;                                   5259
9768 034404 152777 000040 161032        BISB    #40,@ML.REG+40              ;                                   5260
9769 034412 016703 161414               MOV     ML.DUT,R3
9770 034416 042703 177770               BIC     #177770,R3
9771 034422 142777 000007 161014        BICB    #7,@ML.REG+40
9772 034430 150377 161010               BISB    R3,@ML.REG+40
9773 034434 005005              1$:     CLR     R5                          ; DODU.FLG                          5262
9774 034436 012704 125252               MOV     #-52526,R4                  ; *,TST.PAT                         5263
9775 034442 010446              2$:     MOV     R4,-(SP)                    ; TST.PAT,*                         5264
9776 034444 005046              3$:     CLR     -(SP)
9777 034446 004767 165240               JSR     PC,WRT.CS1
9778 034452 010416              4$:     MOV     R4,(SP)                     ; TST.PAT,*                         5265
9779 034454 012746 000006               MOV     #6,-(SP)
9780 034460 004767 165404               JSR     PC,WRT.ER
9781 034464 010416              5$:     MOV     R4,(SP)                     ; TST.PAT,*                         5266
9782 034466 012746 000003               MOV     #3,-(SP)
```

SEQ 0224


```
 9784
 9785                                   ;ML4AD
 9786                                   ;              TEST CODE SECTION
 9787 034472  004767  165550                    JSR     PC,WRT.DA
 9788 034476  010416                            MOV     R4,(SP)          ; TST.PAT,*                    5267
 9789 034500  012746  000010                    MOV     #10,-(SP)
 9790 034504  004767  166072                    JSR     PC,WRT.PA
 9791 034510  010416                            MOV     R4,(SP)          ;   T.PAT,*                    5268
 9792 034512  012746  000015                    MOV     #15,-(SP)
 9793 034516  004767  166266                    JSR     PC,WRT.E1
 9794 034522  010416                            MOV     R4,(SP)          ; TST.PAT,*                    5269
 9795 034524  012746  000016                    MOV     #16,-(SP)
 9796 034530  004767  166462                    JSR     PC,WRT.E2
 9797 034534  005001                            CLR     R1               ; CNT.1                        5271
 9798 034536  005104            2$:             COM     R4               ; TST.PAT                      5273
 9799 034540  010103                            MOV     R1,R3            ; CNT.1,*                      5275
 9800 034542  006303                            ASL     R3
 9801 034544  066307  034550                    ADD     3$(R3),PC
 9802 034550  000012            3$:             .WORD   4$-3$
 9803 034552  000024                            .WORD   5$-3$
 9804 034554  000040                            .WORD   6$-3$
 9805 034556  000054                            .WORD   7$-3$
 9806 034560  000070                            .WORD   8$-3$
 9807 034562  010446            4$:             MOV     R4,-(SP)         ; TST.PAT,*                    5279
 9808 034564  005046                            CLR     -(SP)
 9809 034566  004767  165120                    JSR     PC,WRT.CS1
 9810 034572  000427                            BR      9$               ;                              5275
 9811 034574  010446            5$:             MOV     R4,-(SP)         ; TST.PAT,*                    5282
 9812 034576  012746  000006                    MOV     #6,-(SP)
 9813 034602  004767  165262                    JSR     PC,WRT.ER
 9814 034606  000421                            BR      9$               ;                              5275
 9815 034610  010446            6$:             MOV     R4,-(SP)         ; TST.PAT,*                    5285
 9816 034612  012746  000003                    MOV     #3,-(SP)
 9817 034616  004767  165424                    JSR     PC,WRT.DA
 9818 034622  000413                            BR      9$               ;                              5275
 9819 034624  010446            7$:             MOV     R4,-(SP)         ; TST.PAT,*                    5288
 9820 034626  012746  000010                    MOV     #10,-(SP)
 9821 034632  004767  165744                    JSR     PC,WRT.PA
 9822 034636  000405                            BR      9$               ;                              5275
 9823 034640  010446            8$:             MOV     R4,-(SP)         ; TST.PAT,*                    5291
 9824 034642  012746  000015                    MOV     #15,-(SP)
 9825 034646  004767  166136                    JSR     PC,WRT.E1
 9826 034652  005104            9$:             COM     R4               ; TST.PAT                      5294
 9827 034654  010103                            MOV     R1,R3            ; CNT.1,CNT.2                  5296
 9828 034656  000167  000416                    JMP     23$
 9829 034662  010300            10$:            MOV     R3,R0            ; CNT.2,*                      5299
 9830 034664  006300                            ASL     R0
 9831 034666  066007  034672                    ADD     11$(R0),PC
 9832 034672  000014            11$:            .WORD   12$-11$
 9833 034674  000036                            .WORD   13$-11$
 9834 034676  000062                            .WORD   14$-11$
 9835 034700  000106                            .WORD   15$-11$
 9836 034702  000132                            .WORD   16$-11$
 9837 034704  000156                            .WORD   17$-11$
 9838 034706  010446            12$:            MOV     R4,-(SP)         ; TST.PAT,*                    5303
```

```
                                                                                29-Mar-1982 16:23:04    TOPS
                                                                                29-Mar-1982 16:21:03    PA:<
                        ;ML4AD
                        ;                TEST CODE SECTION
 9840
 9841
 9842
 9843 034710 005002                      CLR     R2                      ; INDEX
 9844 034712 005046                      CLR     -(SP)
 9845 034714 012746 000030               MOV     #30,-(SP)
 9846 034720 060616                      ADD     SP,(SP)                 ; ERR.FLG,*
 9847 034722 004767 165034               JSR     PC,RD.CS1
 9848 034726 000461                      BR      18$                     ;                                5299
 9849 034730 010446              13$:    MOV     R4,-(SP)                ; TST.PAT,*                      5306
 9850 034732 012702 000006               MOV     #6,R2                   ; *,INDEX
 9851 034736 010246                      MOV     R2,-(SP)                ; INDEX,*
 9852 034740 012746 000030               MOV     #30,-(SP)
 9853 034744 060616                      ADD     SP,(SP)                 ; ERR.FLG,*
 9854 034746 004767 165166               JSR     PC,RD.ER
 9855 034752 000447                      BR      18$                     ;                                5299
 9856 034754 010446              14$:    MOV     R4,-(SP)                ; TST.PAT,*                      5309
 9857 034756 012702 000003               MOV     #3,R2                   ; *,INDEX
 9858 034762 010246                      MOV     R2,-(SP)                ; INDEX,*
 9859 034764 012746 000030               MOV     #30,-(SP)
 9860 034770 060616                      ADD     SP,(SP)                 ; ERR.FLG,*
 9861 034772 004767 165320               JSR     PC,RD.DA
 9862 034776 000435                      BR      18$                     ;                                5299
 9863 035000 010446              15$:    MOV     R4,-(SP)                ; TST.PAT,*                      5312
 9864 035002 012702 000010               MOV     #10,R2                  ; *,INDEX
 9865 035006 010246                      MOV     R2,-(SP)                ; INDEX,*
 9866 035010 012746 000030               MOV     #30,-(SP)
 9867 035014 060616                      ADD     SP,(SP)                 ; ERR.FLG,*
 9868 035016 004767 165644               JSR     PC,RD.PA
 9869 035022 000423                      BR      18$                     ;                                5299
 9870 035024 010446              16$:    MOV     R4,-(SP)                ; TST.PAT,*                      5315
 9871 035026 012702 000015               MOV     #15,R2                  ; *,INDEX
 9872 035032 010246                      MOV     R2,-(SP)                ; INDEX,*
 9873 035034 012746 000030               MOV     #30,-(SP)
 9874 035040 060616                      ADD     SP,(SP)                 ; ERR.FLG,*
 9875 035042 004767 166026               JSR     PC,RD.E1
 9876 035046 000411                      BR      18$                     ;                                5299
 9877 035050 010446              17$:    MOV     R4,-(SP)                ; TST.PAT,*                      5318
 9878 035052 012702 000016               MOV     #16,R2                  ; *,INDEX
 9879 035056 010246                      MOV     R2,-(SP)                ; INDEX,*
 9880 035060 012746 000030               MOV     #30,-(SP)
 9881 035064 060616                      ADD     SP,(SP)                 ; ERR.FLG,*
 9882 035066 004767 166222               JSR     PC,RD.E2
 9883 035072 026627 000030 000001 18$:   CMP     30(SP),#1               ; ERR.FLG,*                      5321
 9884 035100 001075                      BNE     22$
 9885 035102 005267 160270               INC     P.CNT                   ;                                5323
 9886 035106 026767 160264 160264        CMP     P.CNT,LIMIT
 9887 035114 003403                      BLE     19$
 9888 035116 062706 000006               ADD     #6,SP
 9889 035122 000474                      BR      24$
 9890 035124 005703              19$:    TST     R3                      ; CNT.2                          5326
 9891 035126 002410                      BLT     20$
 9892 035130 020327 000002               CMP     R3,#2                   ; CNT.2,*
 9893 035134 003005                      BGT     20$
 9894 035136 104455                      TRAP    55                      ;                                5330
```

SEQ 0226

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION
 9899 035140  000156                            .WORD   156
 9900 035142  012706                            .WORD   ASYNC
 9901 035144  026302                            .WORD   DUMPER
 9902 035146  000412                            BR      21$
 9903 035150  020327  000003            20$:    CMP     R3,#3                   ;                               5326
 9904 035154  002407                            BLT     21$                     ; CNT.2,*
 9905 035156  020327  000005                    CMP     R3,#5                   ; CNT.2,*
 9906 035162  003004                            BGT     21$
 9907 035164  104455                            TRAP    55                      ;                               5333
 9908 035166  000156                            .WORD   156
 9909 035170  013012                            .WORD   ARP.DAT
 9910 035172  026302                            .WORD   DUMPER
 9911 035174  012746  010630            21$:    MOV     #WRD.10,-(SP)           ;                               5336
 9912 035200  012746  011122                    MOV     #WRD.37,-(SP)
 9913 035204  012746  011132                    MOV     #WRD.38,-(SP)
 9914 035210  012746  010256                    MOV     #THR.FMT,-(SP)
 9915 035214  012746  000004                    MOV     #4,-(SP)
 9916 035220  010600                            MOV     SP,R0                   ; SP,*
 9917 035222  104414                            TRAP    14
 9918 035224  016716  160122                    MOV     RD.DATA,(SP)            ;                               5337
 9919 035230  016746  160114                    MOV     WT.DATA,-(SP)
 9920 035234  010200                            MOV     R2,R0                   ; INDEX,*
 9921 035236  006300                            ASL     R0
 9922 035240  006300                            ASL     R0
 9923 035242  006300                            ASL     R0
 9924 035244  016046  015404                    MOV     ML.REG(R0),-(SP)
 9925 035250  012746  007436                    MOV     #FMT.16,-(SP)
 9926 035254  012746  000004                    MOV     #4,-(SP)
 9927 035260  010600                            MOV     SP,R0                   ; SP,*
 9928 035262  104414                            TRAP    14
 9929 035264  012705  000001                    MOV     #1,R5                   ; *,DODU.FLG                    5338
 9930 035270  062706  000022                    ADD     #22,SP                  ;                               5323
 9931 035274  062706  000006            22$:    ADD     #6,SP                   ;                               5297
 9932 035300  005203                    23$:    INC     R3                      ; CNT.2                         5296
 9933 035302  020327  000005                    CMP     R3,#5                   ; CNT.2,*
 9934 035306  003002                            BGT     24$
 9935 035310  000167  177346                    JMP     10$
 9936 035314  022626                    24$:    CMP     (SP)+,(SP)+             ;                               5272
 9937 035316  005201                            INC     R1                      ; CNT.1                         5271
 9938 035320  020127  000004                    CMP     R1,#4                   ; CNT.1,*
 9939 035324  003002                            BGT     25$
 9940 035326  000167  177204                    JMP     2$
 9941 035332  062706  000016            25$:    ADD     #16,SP                  ;                               5259
 9942 035336  104467                            TRAP    67                      ;                               5343
 9943 035340  006000                            ROR     R0
 9944 035342  103002                            BHIS    26$
 9945 035344  000167  177032                    JMP     1$
 9946 035350  005305                    26$:    DEC     R5                      ; DODU.FLG                      5347
 9947 035352  001004                            BNE     27$
 9948 035354  016700  160450                    MOV     ML.LUN,R0               ;                               5350
 9949 035360  104451                            TRAP    51
 9950 035362  104444                            TRAP    44
```

```
                                ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
                                ;                TEST CODE SECTION                      29-Mar-1982 16:21:03    PA:<
 9955 C35364  005726            27$:    TST      (SP)+                          ;                               5224
 9956 035366  000207                    RTS      PC
 9957
 9958                           ; Routine Size:  256 words
 9959                           ; Maximum stack depth per invocation:  28 words
 9964
 9965
 9969
 9970                                    .SBTTL   T8 TEST CODE SECTION
 9974 035370                    T8::
 9975 035370  004767  176774    1$:     JSR      PC,$T8                         ;                               5352
 9976 035374  104466                    TRAP     66
 9977 035376  006000                    ROR      R0
 9978 035400  103773                    BLO      1$
 9979 035402  000207                    RTS      PC
 9980
 9981                           ; Routine Size:  6 words
 9982                           ; Maximum stack depth per invocation:  0 words
 9987
 9988
 9989 ;          5355  !<BLF/PAGE>
```

```
                                                                                29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
                                                                                29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (61)
;ML4AD
;                       TEST CODE SECTION
;
;                5356   !
;                5357   !
;                5358   BGNTST;
;                5359
;                5360   !++
;                5361   ! TEST NUMBER:  TST 9
;                5362   !
;                5363   ! TEST NAME:    PRINT SERIAL NUMBER
;                5364   !
;                5365   ! TEST DESCRIPTION:
;                5366   !       UPON A YES RESPONSE TO THE
;                5367   !       SOFTWARE QUESTION 'PRINT SERIAL NO?
;                5368   !       PRINT OUT THE DRIVE UNDER TEST
;                5369   !       SERIAL NUMBER.
;                5370   !
;                5371   !--
;                5372
;                5373   local
;                5374       D3,                                          !STORES DIGIT 3 OF SERIAL NUMBER
;                5375       D2,                                          !STORES DIGIT 2 OF SERIAL NUMBER
;                5376       D1,                                          !STORES DIGIT 1 OF SERIAL NUMBER
;                5377       D0;                                          !STORES DIGIT 0 OF SERIAL NUMBER
;                5378
;                5379   if .PRSN IS_SET                                  !SEE IF WE PRINT THE SERIAL NUMBER
;                5380   then
;                5381       begin                                        !PRINT THE SERIAL NUMBER IF THE REPLY WAS YES
;                5382       D3 = .SN3;                                   !LOAD DIGIT 3 OF SN INTO D3
;                5383       D2 = .SN2;                                   !LOAD DIGIT 2 OF SN INTO D2
;                5384       D1 = .SN1;                                   !LOAD DIGIT 1 OF SN INTO D1
;                5385       D0 = .SN0;                                   !LOAD DIGIT 0 OF SN INTO D0
;                5386
;                5387       if ((.D3 gtr 9) or (.D2 gtr 9) or (.D1 gtr 9) or (.D0 gtr 9))
;                5388                                                    !SEE IF DIGITS ARE TO BIG FOR COVERSION
;                5389       then
;                5390           PRINTB (FMT_25, PHR_15, .MLSN)           !PRINT OCTAL SN IF TO BIG
;                5391       else
;                5392           PRINTB (FMT_26, PHR_15, .D3, .D2, .D1, .D0);    !ELSE PRINT DECIMAL SERIAL NUMBER
;                5393
;                5394       end;
;                5395
;                5396   ENDTST;
;
;                                       .SBTTL  ST9 TEST CODE SECTION
;035404  004167  147144          $T9:    JSR     R1,$SAVE3                ;                                      5354
;035410  026727  144764  000001          CMP     PRSN,#1                  ;                                      5379
;035416  001101                          BNE     4$
```

```
10047
10048                                   ;ML4AD
10049                                   ;               TEST CODE SECTION
10050 035420  017703  160120                   MOV     @ML.REG+140,R3          ; *,D3                          5382
10051 035424  006203                           ASR     R3                      ; D3
10052 035426  006203                           ASR     R3                      ; D3
10053 035430  006203                           ASR     R3                      ; D3
10054 035432  006203                           ASR     R3                      ; D3
10055 035434  000303                           SWAB    R3                      ; D3
10056 035436  042703  177760                   BIC     #177760,R3              ; *,D3
10057 035442  017700  160076                   MOV     @ML.REG+140,R0          ; *,D2                          5383
10058 035446  000300                           SWAB    R0                      ; D2
10059 035450  042700  177760                   BIC     #177760,R0              ; *,D2
10060 035454  117701  160064                   MOVB    @ML.REG+140,R1          ; *,D1                          5384
10061 035460  006201                           ASR     R1                      ; D1
10062 035462  006201                           ASR     R1                      ; D1
10063 035464  006201                           ASR     R1                      ; D1
10064 035466  006201                           ASR     R1                      ; D1
10065 035470  042701  177760                   BIC     #177760,R1              ; *,D1
10066 035474  117702  160044                   MOVB    @ML.REG+140,R2          ; *,D0                          5385
10067 035500  042702  177760                   BIC     #177760,R2              ; *,D0
10068 035504  020327  000011                   CMP     R3,#11                  ; D3,*                          5387
10069 035510  003011                           BGT     1$
10070 035512  020027  000011                   CMP     R0,#11                  ; D2,*
10071 035516  003006                           BGT     1$
10072 035520  020127  000011                   CMP     R1,#11                  ; D1,*
10073 035524  003003                           BGT     1$
10074 035526  020227  000011                   CMP     R2,#11                  ; D0,*
10075 035532  003413                           BLE     2$
10076 035534  017746  160004           1$:     MOV     @ML.REG+140,-(SP)       ;                               5390
10077 035540  012746  012136                   MOV     #PHR.15,-(SP)
10078 035544  012746  010166                   MOV     #FMT.25,-(SP)
10079 035550  012746  000003                   MOV     #3,-(SP)
10080 035554  010600                           MOV     SP,R0                   ; SP,*
10081 035556  104414                           TRAP    14
10082 035560  000416                           BR      3$                      ;
10083 035562  010246                   2$:     MOV     R2,-(SP)                ; D0,*                          5387
10084 035564  010146                           MOV     R1,-(SP)                ; D1,*                          5392
10085 035566  010046                           MOV     R0,-(SP)                ; D2,*
10086 035570  010346                           MOV     R3,-(SP)                ; D3,*
10087 035572  012746  012136                   MOV     #PHR.15,-(SP)
10088 035576  012746  010206                   MOV     #FMT.26,-(SP)
10089 035602  012746  000006                   MOV     #6,-(SP)
10090 035606  010600                           MOV     SP,R0                   ; SP,*
10091 035610  104414                           TRAP    14
10092 035612  062706  000006                   ADD     #6,SP
10093 035616  062706  000010           3$:     ADD     #10,SP
10094 035622  000207                   4$:     RTS     PC                      ;                               5381
10095                                                                          ;                               5354
10096
10097                                   ; Routine Size:  72 words
                                        ; Maximum stack depth per invocation:  11 words
```

```
10103 ;ML4AD
10104 ;                 TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
10105                                                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (61)
10109
10110                                       .SBTTL  T9 TEST CODE SECTION
10114 035624                          T9::
10115 035624  004767  177554         1$:   JSR     PC,$T9                          ;                        5394
10116 035630  104466                        TRAP    66
10117 035632  006000                        ROR     R0
10118 035634  103773                        BLO     1$
10119 035636  000207                        RTS     PC
10120
10121                                 ; Routine Size:  6 words
10122                                 ; Maximum stack depth per invocation:  0 words
10127
10128
10129 ;          5397  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (62)

```
10131 ;ML4AD
10132 ;                 TEST CODE SECTION
10133 ;
10134 ;          5398   !
10135 ;          5399   !
10136 ;          5400   BGNTST;
10137 ;          5401
10138 ;          5402   !++
10139 ;          5403   ! TEST NUMBER:  TST 10
10140 ;          5404   !
10141 ;          5405   ! TEST NAME:    C-BUS PARITY TEST
10142 ;          5406   !
10143 ;          5407   ! TEST DESCRIPTION:
10144 ;          5408   !       TEST THE CONTROL BUS PARITY
10145 ;          5409   !       DETECTION AND GENERATING BY:
10146 ;          5410   !
10147 ;          5411   !       1.  WRITING BAD PARITY TO DEVICE
10148 ;          5412   !           AND TEST CPAR SET.
10149 ;          5413   !
10150 ;          5414   !       2.  WRITING GOOD PARITY TO DEVICE
10151 ;          5415   !           AND TEST CPAR CLR.
10152 ;          5416   !
10153 ;          5417   !       3.  READING DEVICE AND TEST GOOD
10154 ;          5418   !           PARITY GENERATION BY READING
10155 ;          5419   !           MCPE CLR.
10156 ;          5420   !
10157 ;          5421   !--
10158 ;          5422
10159 ;          5423   local
10160 ;          5424       SAVE,                          !TEMPORARY SAVE LOCATION
10161 ;          5425       TST_PAT;                       !TEST PATTERN
10162 ;          5426
10163 ;          5427   if .PAR_DIS IS_NOT_SET             !SEE IF PARITY IS DISABLED
10164 ;          5428   then
10165 ;          5429       begin                          !BEGIN IF PARITY IS ENABLE
10166 ;          5430       TST_PAT = %o'125252';          !ALTERNATING 1, 0 PATTERN
10167 ;          5431
10168 ;          5432       incr TWICE from 1 to 2 do      !REPEAT LOOP TWICE
10169 ;          5433           begin
10170 ;          5434           CLR_MBUS;
10171 ;          5435           PAT = ONE;                 !GENERATE EVEN PARITY BY SETTING THE PA; BIT
10172 ;          5436           MLDA = .TST_PAT;           !WRITE BAD PARITY ON CONTROL BUS
10173 ;          5437
10174 ;          5438           if .CPAR IS_NOT_SET        !READ CPAR BIT SET
10175 ;          5439           then
10176 ;          5440               begin
10177 ;          5441               ERRDF (7, ASYNC, DUMPER);      !ERROR IF NOT SET
10178 ;          5442               PRINTB (THR_FMT, WRD_5, WRD_7, WRD_9);
10179 ;          5443               end;
10180 ;          5444
10181 ;          5445           CLR_MBUS;                  !CLEAR OUT PAT BIT
10182 ;          5446           MLDA = .TST_PAT;           !WRITE ODD PARITY CONTROL BUS
10183 ;          5447
10184 ;          5448           if .CPAR IS_SET            !READ CPAR BIT CLEARED
10185 ;          5449           then
```

```
 10187 ;ML4AD
 10188 ;                 TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 10189                                                                    29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (62)
 10190 ;          5450              begin
 10191 ;          5451              ERRDF (8, ASYNC, DUMPER);              !ERROR IF SET
 10192 ;          5452              PRINTB (THR_FMT, WRD_6, WRD_7, WRD_9);
 10193 ;          5453              end;
 10194 ;          5454
 10195 ;          5455          CLR_MBUS;
 10196 ;          5456          SAVE = .MLDA;                              !READ A REGISTER FROM DUT
 10197 ;          5457
 10198 ;          5458          if .MCPE IS_SET                            !SEE IF GENERATED GOOD PARITY
 10199 ;          5459          then
 10200 ;          5460              begin
 10201 ;          5461              ERRDF (9, ASYNC, DUMPER);              !ERROR IF MCPE IS_SET
 10202 ;          5462              PRINTB (THR_FMT, WRD_6, WRD_7, WRD_8);
 10203 ;          5463              end;
 10204 ;          5464
 10205 ;          5465          TST_PAT = .TST_PAT^ONE;                    !REPEAT WITH SHIFTED DATA
 10206 ;          5466          end;
 10207 ;          5467
 10208 ;          5468      end
 10209 ;          5469  else
 10210 ;          5470      PRINTB (TWO_FMT, WRD_7, WRD_53);               !JUST PRINT MESSAGE IF PARITY IS DISABLED
 10211 ;          5471
 10212 ;          5472  ENDTST;
 10216
 10217                                      .SBTTL  $T10 TEST CODE SECTION
 10221 035640  004167  146710        $T10: JSR     R1,$SAVE3                ;                                    5396
 10222 035644  005767  156140              TST     PAR.DIS                  ;                                    5427
 10223 035650  001171                      BNE     5$
 10224 035652  012701  125252              MOV     #-52526,R1               ; *.TST.PAT                          5430
 10225 035656  012702  000001              MOV     #1,R2                    ; *.TWICE                            5432
 10226 035662  152777  000040  157554  1$: BISB    #40,@ML.REG+40           ;                                    5433
 10227 035670  016700  160136              MOV     ML.DUT,R0
 10228 035674  042700  177770              BIC     #177770,R0
 10229 035700  142777  000007  157536      BICB    #7,@ML.REG+40
 10230 035706  150077  157532              BISB    R0,@ML.REG+40
 10231 035712  152777  000020  157524      BISB    #20,@ML.REG+40           ;
 10232 035720  010177  157510              MOV     R1,@ML.REG+30            ; TST.PAT,*                          5435
 10233 035724  132777  000010  157532      BITB    #10,@ML.REG+60           ;                                    5436
 10234 035732  001022                      BNE     2$                       ;                                    5438
 10235 035734  104455                      TRAP    55
 10236 035736  000007                      .WORD   7                        ;                                    5441
 10237 035740  012706                      .WORD   ASYNC
 10238 035742  026302                      .WORD   DUMPER
 10239 035744  012746  010616              MOV     #WRD.9,-(SP)             ;                                    5442
 10240 035750  012746  010566              MOV     #WRD.7,-(SP)
 10241 035754  012746  010552              MOV     #WRD.5,-(SP)
```

SEQ 0233

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
10243                                          ;ML4AD
10244                                          ;
10245                                          ;               TEST CODE SECTION
10246 035760  012746  010256                             MOV     #THR.FMT,-(SP)
10247 035764  012746  000004                             MOV     #4,-(SP)
10248 035770  010600                                     MOV     SP,R0          ; SP,*
10249 035772  104414                                     TRAP    14
10250 035774  062706  000012                             ADD     #12,SP         ;
10251 036000  152777  000040  157436  2$:                BISB    #40,@ML.REG+40 ;                                5440
10252 036006  016700  160020                             MOV     ML.DUT,R0      ;                                5443
10253 036012  042700  177770                             BIC     #177770,R0
10254 036016  142777  000007  157420                     BICB    #7,@ML.REG+40
10255 036024  150077  157414                             BISB    R0,@ML.REG+40
10256 036030  010177  157400                             MOV     R1,@ML.REG+30  ; TST.PAT,*
10257 036034  132777  000010  157422                     BITB    #10,@ML.REG+60 ;                                5446
10258 036042  001422                                     BEQ     3$             ;                                5448
10259 036044  104455                                     TRAP    55
10260 036046  000010                                     .WORD   10             ;                                5451
10261 036050  012706                                     .WORD   ASYNC
10262 036052  026302                                     .WORD   DUMPER
10263 036054  012746  010616                             MOV     #WRD.9,-(SP)
10264 036060  012746  010566                             MOV     #WRD.7,-(SP)   ;                                5452
10265 036064  012746  010560                             MOV     #WRD.6,-(SP)
10266 036070  012746  010256                             MOV     #THR.FMT,-(SP)
10267 036074  012746  000004                             MOV     #4,-(SP)
10268 036100  010600                                     MOV     SP,R0          ; SP,*
10269 036102  104414                                     TRAP    14
10270 036104  062706  000012                             ADD     #12,SP         ;
10271 036110  152777  000040  157326  3$:                BISB    #40,@ML.REG+40 ;                                5450
10272 036116  016700  157710                             MOV     ML.DUT,R0      ;                                5453
10273 036122  042700  177770                             BIC     #177770,R0
10274 036126  142777  000007  157310                     BICB    #7,@ML.REG+40
10275 036134  150077  157304                             BISB    R0,@ML.REG+40
10276 036140  017703  157270                             MOV     @ML.REG+30,R3
10277 036144  032777  020000  157232                     BIT     #20000,@ML.REG ; *.SAVE                         5456
10278 036152  001422                                     BEQ     4$             ;                                5458
10279 036154  104455                                     TRAP    55
10280 036156  000011                                     .WORD   11             ;                                5461
10281 036160  012706                                     .WORD   ASYNC
10282 036162  026302                                     .WORD   DUMPER
10283 036164  012746  010602                             MOV     #WRD.8,-(SP)
10284 036170  012746  010566                             MOV     #WRD.7,-(SP)   ;                                5462
10285 036174  012746  010560                             MOV     #WRD.6,-(SP)
10286 036200  012746  010256                             MOV     #THR.FMT,-(SP)
10287 036204  012746  000004                             MOV     #4,-(SP)
10288 036210  010600                                     MOV     SP,R0          ; SP,*
10289 036212  104414                                     TRAP    14
10290 036214  062706  000012                             ADD     #12,SP         ;
10291 036220  006301                            4$:      ASL     R1             ; TST.PAT                       5460
10292 036222  005202                                     INC     R2             ; TWICE                         5465
10293 036224  020227  000002                             CMP     R2,#2          ; TWICE,*                       5432
10294 036230  003614                                     BLE     1$
10295 036232  000207                                     RTS     PC
10296 036234  012746  011316                    5$:      MOV     #WRD.53,-(SP)  ;                                5427
10297 036240  012746  010566                             MOV     #WRD.7,-(SP)   ;                                5470
```

```
10299                                   ;ML4AD
10300                                   ;               TEST CODE SECTION
10301
10302 036244  012746  010246                       MOV     #TWO.FMT,-(SP)
10303 036250  012746  000003                       MOV     #3,-(SP)
10304 036254  010600                               MOV     SP,R0                           ; SP,*
10305 036256  104414                               TRAP    14
10306 036260  062706  000010                       ADD     #10,SP
10307 036264  000207                               RTS     PC                              ;                       5396
10308
10309                                   ; Routine Size:  139 words
10310                                   ; Maximum stack depth per invocation:  9 words
10315
10316
10320
10321                                              .SBTTL  T10 TEST CODE SECTION
10325 036266                            T10::
10326 036266  004767  177346            1$:     JSR     PC,$T10                         ;                       5470
10327 036272  104466                            TRAP    66
10328 036274  006000                            ROR     R0
10329 036276  103773                            BLO     1$
10330 036300  000207                            RTS     PC
10331
10332                                   ; Routine Size:  6 words
10333                                   ; Maximum stack depth per invocation:  0 words
10338
10339
10340 ;          5473  !<BLF/PAGE>
```

```
10342 ;ML4AD
10343 ;                 TEST CODE SECTION
10344
10345 ;          5474  !
10346 ;          5475  !
10347 ;          5476  BGNTST;
10348 ;          5477
10349 ;          5478  !++
10350 ;          5479  ! TEST NUMBER:  TST 11
10351 ;          5480  !
10352 ;          5481  ! TEST NAME:    MEMORY SIZING TEST
10353 ;          5482  !
10354 ;          5483  ! TEST DESCRIPTION:
10355 ;          5484  !
10356 ;          5485  !       THIS TESTS THE ML11'S SIZING
10357 ;          5486  !       LOGIC BY COMPARING THE
10358 ;          5487  !       OPERATORS INPUTED NUMBER OF ARRAYS
10359 ;          5488  !       PRESENT TO THE ML11 SIZING
10360 ;          5489  !       NUMBER OF ARRAYS PRESENT
10361 ;          5490  !       THE DRIVE IS DROPPED ON DETECTED ERRORS.
10362 ;          5491  ! IMPLICIT INPUTS:
10363 ;          5492  !
10364 ;          5493  !       OP_NUM_ARR:
10365 ;          5494  !       LOADED DURING INIT CODE AND
10366 ;          5495  !       STORES OPERATORS INPUT TO THE
10367 ;          5496  !       SOFTWARE QUESTION:
10368 ;          5497  !       NUMBER OF ARRAYS PRESENT?
10369 ;          5498  !
10370 ;          5499  !
10371 ;          5500  !--
10372 ;          5501
10373 ;          5502  BGNSUB;
10374 ;          5503  CLR_MBUS;
10375 ;          5504
10376 ;          5505  if (.OP_NUM_ARR + 1) neq .ML_NUM_ARR          !SEE IF DRIVE SIZED SAME NO. OF ARRAYS AS OP INPUTED
10377 ;          5506  then
10378 ;          5507      begin
10379 ;          5508      ERRDF (10, ASYNC, DUMPER);               !IF NOT EQL THEN ERROR AND SET DODU_FLG
10380 ;          5509      PRINTB (TWO_FMT, FNC_1, WRD_14);
10381 ;          5510      PRINTB (FMT_2, (.OP_NUM_ARR + 1), .ML_NUM_ARR);
10382 ;          5511      end;
10383 ;          5512
10384 ;          5513  ENDSUB;
10385 ;          5514  ENDTST;
10389
10390
10394 036302  004167  146232          $T11:   .SBTTL  $T11 TEST CODE SECTION
10395 036306  104402                   1$:     JSR     R1,$SAVE2                ;                    5472
10396 036310  152777  000040  157126           TRAP    2                        ;                    5476
                                               BISB    #40,@ML.REG+40           ;                    5502
```

SEQ 0236

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
10398                                   ;ML4AD
10399                                   ;
10400                                   ;               TEST CODE SECTION
10401 036316  016700  157510                   MOV     ML.DUT,R0
10402 036322  042700  177770                   BIC     #177770,R0
10403 036326  142777  000007  157110           BICB    #7,@ML.REG+40
10404 036334  150077  157104                   BISB    R0,@ML.REG+40
10405 036340  016700  155436                   MOV     OP.NUM.ARR,R0           ;                       5505
10406 036344  005200                           INC     R0
10407 036346  017702  157152                   MOV     @ML.REG+120,R2
10408 036352  006202                           ASR     R2
10409 036354  006202                           ASR     R2
10410 036356  006202                           ASR     R2
10411 036360  000302                           SWAB    R2
10412 036362  042702  177740                   BIC     #177740,R2
10413 036366  010001                           MOV     R0,R1
10414 036370  020102                           CMP     R1,R2
10415 036372  001442                           BEQ     2$
10416 036374  104455                           TRAP    55                      ;                       5508
10417 036376  000012                           .WORD   12
10418 036400  012706                           .WORD   ASYNC
10419 036402  026302                           .WORD   DUMPER
10420 036404  012746  010664                   MOV     #WRD.14,-(SP)           ;                       5509
10421 036410  012746  012150                   MOV     #FNC.1,-(SP)
10422 036414  012746  010246                   MOV     #TWO.FMT,-(SP)
10423 036420  012746  000003                   MOV     #3,-(SP)
10424 036424  010600                           MOV     SP,R0                   ; SP,*
10425 036426  104414                           TRAP    14
10426 036430  017702  157070                   MOV     @ML.REG+120,R2          ;                       5510
10427 036434  006202                           ASR     R2
10428 036436  006202                           ASR     R2
10429 036440  006202                           ASR     R2
10430 036442  000302                           SWAB    R2
10431 036444  042702  177740                   BIC     #177740,R2
10432 036450  010216                           MOV     R2,(SP)
10433 036452  016746  155324                   MOV     OP.NUM.ARR,-(SP)
10434 036456  005216                           INC     (SP)
10435 036460  012746  006506                   MOV     #FMT.2,-(SP)
10436 036464  012746  000003                   MOV     #3,-(SP)
10437 036470  010600                           MOV     SP,R0                   ; SP,*
10438 036472  104414                           TRAP    14
10439 036474  062706  000016                   ADD     #16,SP                  ;                       5507
10440 036500  104467                   2$:     TRAP    67                      ;                       5511
10441 036502  006000                           ROR     R0
10442 036504  103700                           BLO     1$
10443 036506  000207                           RTS     PC                      ;                       5472
10444
10445                                   ; Routine Size:  67 words
10446                                   ; Maximum stack depth per invocation:  10 words
10451
```

SEQ 0237

```
10459
10460                                   .SBTTL  T11 TEST CODE SECTION
10464 036510                    T11::
10465 036510  004767  177566    1$:     JSR     PC,$T11                 ;          5513
10466 036514  104466                    TRAP    66
10467 036516  006000                    ROR     R0
10468 036520  103773                    BLO     1$
10469 036522  000207                    RTS     PC
10470
10471                           ; Routine Size:  6 words
10472                           ; Maximum stack depth per invocation:  0 words
10477
10478
10479 ;          5515  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (64)

```
10481 ;ML4AD
10482 ;                 TEST CODE SECTION
10483
10484 ;          5516 !
10485 ;          5517 !
10486 ;          5518 BGNTST;
10487 ;          5519
10488 ;          5520 !++
10489 ;          5521 ! TEST NUMBER:  TST 12
10490 ;          5522 !
10491 ;          5523 ! TEST NAME:    NO-OP FUNCTION TEST
10492 ;          5524 !
10493 ;          5525 ! TEST DESCRIPTION:
10494 ;          5526 !
10495 ;          5527 !       TEST IF THE DRIVE CAN PERFORM
10496 ;          5528 !       A NO_OP FUNCTION WITH OUT
10497 ;          5529 !       HANGING THE DRIVE.
10498 ;          5530 !
10499 ;          5531 !       A NO_OP FUNCTION IS WRITTEN
10500 ;          5532 !       INTO MLCS1.
10501 ;          5533 !       THEN GO AND ERROR BITS ARE
10502 ;          5534 !       CHECKED FOR CORRECT STATES.
10503 ;          5535 !
10504 ;          5536 !       THIS UNIT IS DROPPED ON DETECTED
10505 ;          5537 !       ERRORS.
10506 ;          5538 !
10507 ;          5539 ! IMPLICIT INPUTS:      NONE
10508 ;          5540 !
10509 ;          5541 !
10510 ;          5542 !--
10511 ;          5543
10512 ;          5544 BGNSUB;
10513 ;          5545 CLR_MBUS;
10514 ;          5546 MLCS1 = NOOP;                                   !DO A NOOP FUNCTION
10515 ;          5547 DELAY (ONE_US);                                 !DELAY 1 US
10516 ;          5548
10517 ;          5549 if .GO IS_SET                                   !SEE IF GO STILL SET
10518 ;          5550 then
10519 ;          5551     begin
10520 ;          5552     ERRDF (11, ASYNC, DUMPER);                 !ERROR AND SET DODU_FLG IF STILL SET
10521 ;          5553     PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_2, WRD_19);
10522 ;          5554     end;
10523 ;          5555
10524 ;          5556 if .ILF IS_SET                                  !SEE ILF SET
10525 ;          5557 then
10526 ;          5558     begin
10527 ;          5559     ERRDF (12, ASYNC, DUMPER);                 !ERROR AND SET DODU_FLG IF SET
10528 ;          5560     PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_11, FNC_2, WRD_19);
10529 ;          5561     end;
10530 ;          5562
10531 ;          5563 if .OPI IS_SET                                  !SEE IF CPI SET
10532 ;          5564 then
10533 ;          5565     begin
10534 ;          5566     ERRDF (13, ASYNC, DUMPER);                 !ERROR AND SET DODU_FLG IF SET
10535 ;          5567     PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_11, FNC_2, WRD_19);
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (64)

```
10537 ;ML4AD
10538 ;                    TEST CODE SECTION
10539
10540 ;          5568      end;
10541 ;          5569
10542 ;          5570  ENDSUB;
10543 ;          5571  ENDTST;
10547
10548                                        .SBTTL  $T12 TEST CODE SECTION
10552 036524  010146                 $T12:  MOV     R1,-(SP)                ;                        5514
10553 036526  005746                        TST     -(SP)
10554 036530  104402                 1$:    TRAP    2                       ;                        5518
10555 036532  152777  000040  156704        BISB    #40,@ML.REG+40          ;                        5544
10556 036540  016701  157266                MOV     ML.DUT,R1
10557 036544  042701  177770                BIC     #177770,R1
10558 036550  142777  000007  156666        BICB    #7,@ML.REG+40
10559 036556  150177  156662                BISB    R1,@ML.REG+40
10560 036562  012777  000001  156614        MOV     #1,@ML.REG              ;                        5546
10561 036570  012700  000001                MOV     #1,R0                   ; *,$$TMP2               5547
10562 036574  001410                 2$:    BEQ     5$
10563 036576  016701  143314                MOV     L$DLY,R1                ; *,$$TMP1
10564 036602  001403                        BEQ     4$
10565 036604  005016                 3$:    CLR     (SP)                    ; $$TMP
10566 036606  005301                        DEC     R1                      ; $$TMP1
10567 036610  001375                        BNE     3$
10568 036612  005300                 4$:    DEC     R0                      ; $$TMP2
10569 036614  000767                        BR      2$
10570 036616  132777  000001  156560  5$:   BITB    #1,@ML.REG              ;                        5549
10571 036624  001426                        BEQ     6$
10572 036626  104455                        TRAP    55                      ;                        5552
10573 036630  000013                        .WORD   13
10574 036632  012706                        .WORD   ASYNC
10575 036634  026302                        .WORD   DUMPER
10576 036636  012746  010730                MOV     #WRD.19,-(SP)           ;                        5553
10577 036642  012746  012164                MOV     #FNC.2,-(SP)
10578 036646  012746  010640                MOV     #WRD.11,-(SP)
10579 036652  012746  011626                MOV     #PHR.2,-(SP)
10580 036656  012746  010520                MOV     #WRD.1,-(SP)
10581 036662  012746  010304                MOV     #FIV.FMT,-(SP)
10582 036666  012746  000006                MOV     #6,-(SP)
10583 036672  010600                        MOV     SP,R0                   ; SP,*
10584 036674  104414                        TRAP    14
10585 036676  062706  000016                ADD     #16,SP
10586 036702  132777  000001  156554  6$:   BITB    #1,@ML.REG+60           ;                        5551
10587 036710  001426                        BEQ     7$                      ;                        5556
10588 036712  104455                        TRAP    55                      ;                        5559
10589 036714  000014                        .WORD   14
10590 036716  012706                        .WORD   ASYNC
10591 036720  026302                        .WORD   DUMPER
```

```
10593                                   ;ML4AD
10594                                   ;                       TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
10595                                                                                                               29-Mar-1982 16:21:03    PA:<
10596 036722  012746  010730                     MOV     #WRD.19,-(SP)           ;                                                       5560
10597 036726  012746  012164                     MOV     #FNC.2,-(SP)
10598 036732  012746  010640                     MOV     #WRD.11,-(SP)
10599 036736  012746  011714                     MOV     #PHR.5,-(SP)
10600 036742  012746  010536                     MOV     #WRD.3,-(SP)
10601 036746  012746  010304                     MOV     #FIV.FMT,-(SP)
10602 036752  012746  000006                     MOV     #6,-(SP)
10603 036756  010600                           MOV     SP,R0                   ; SP,*
10604 036760  104414                           TRAP    14
10605 036762  062706  000016                     ADD     #16,SP
10606 036766  032777  020000  156470  7$:      BIT     #20000,@ML.REG+60       ;                                                       5558
10607 036774  001426                           BEQ     8$                      ;                                                       5563
10608 036776  104455                           TRAP    55
10609 037000  000015                           .WORD   15                      ;                                                       5566
10610 037002  012706                           .WORD   ASYNC
10611 037004  026302                           .WORD   DUMPER
10612 037006  012746  010730                     MOV     #WRD.19,-(SP)           ;                                                       5567
10613 037012  012746  012164                     MOV     #FNC.2,-(SP)
10614 037016  012746  010640                     MOV     #WRD.11,-(SP)
10615 037022  012746  011714                     MOV     #PHR.5,-(SP)
10616 037026  012746  010544                     MOV     #WRD.4,-(SP)
10617 037032  012746  010304                     MOV     #FIV.FMT,-(SP)
10618 037036  012746  000006                     MOV     #6,-(SP)
10619 037042  010600                           MOV     SP,R0                   ; SP,*
10620 037044  104414                           TRAP    14
10621 037046  062706  000016                     ADD     #16,SP
10622 037052  104467                  8$:      TRAP    67                      ;                                                       5565
10623 037054  006000                           ROR     R0                      ;                                                       5568
10624 037056  103624                           BLO     1$
10625 037060  005726                           TST     (SP)+
10626 037062  012601                           MOV     (SP)+,R1                ;                                                       5514
10627 037064  000207                           RTS     PC
10628
10629                                   ; Routine Size:  113 words
10630                                   ; Maximum stack depth per invocation:  9 words
10635
10636
10640
10641                                           .SBTTL  T12 TEST CODE SECTION
10645 037066                            T12::
10646 037066  004767  177432            1$:     JSR     PC,$T12                 ;                                                       5570
10647 037072  104466                           TRAP    66
```

```
10649                                   ;ML4AD
10650                                   ;               TEST CODE SECTION
10651
10652 037074  006000                              ROR     R0
10653 037076  103773                              BLO     1$
10654 037100  000207                              RTS     PC
10655
10656                                   ; Routine Size:  6 words
10657                                   ; Maximum stack depth per invocation:  0 words
10662
10663
10664 ;          5572  !<BLF/PAGE>
```

```
10666 ;ML4AD
10667 .                 TEST CODE SECTION                       29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
10668                                                           29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (65)
10669 ;          5573  !
10670 ;          5574  !
10671 ;          5575  BGNTST;
10672 ;          5576
10673 ;          5577  !++
10674 ;          5578  ! TEST NUMBER:  TST 13
10675 ;          5579  !
10676 ;          5580  ! TEST NAME:    WRITE CHECK FUNCTION TEST
10677 ;          5581  !
10678 ;          5582  ! TEST DESCRIPTION:
10679 ;          5583  !
10680 ;          5584  !       TEST IF THE DRIVE CAN PERFORM
10681 ;          5585  !       A WRITE CHECK FUNCTION WITHOUT
10682 ;          5586  !       HANGING THE DRIVE.
10683 ;          5587  !
10684 ;          5588  !       A WRITE CHECK FUNCTION IS WRITTEN
10685 ;          5589  !       INTO MLCS1.
10686 ;          5590  !       THEN GO AND ERROR BITS ARE
10687 ;          5591  !       CHECKED FOR CORRECT STATES
10688 ;          5592  !
10689 ;          5593  !--
10690 ;          5594
10691 ;          5595  local
10692 ;          5596      DODU_FLG;                             !DROP UNIT FLAG
10693 ;          5597
10694 ;          5598  BGNSUB;
10695 ;          5599  CLR_MBUS;
10696 ;          5600  DODU_FLG = ZERO;
10697 ;          5601  FIRST_BLK_XFER ();                        !SET UP A FIRST BLK XFERR
10698 ;          5602  MLCS1 = WRT_CHK;                          !DO A WRITE CHECK FUNCTION
10699 ;          5603
10700 ;          5604  if .GO IS_NOT_SET                         !SEE IF THE GO BIT GOT SET
10701 ;          5605  then
10702 ;          5606      begin
10703 ;          5607      ERRDF (14, ASYNC, DUMPER);            !ERROR IF NOT SET
10704 ;          5608      PRINTB (FIV_FMT, WRD_1, PHR_1, WRD_12, FNC_4, WRD_19);
10705 ;          5609
10706 ;          5610      if .DRY IS_NOT_SET                    !SEE IF THE DRY IS SET WHILE GO IS CLEARED
10707 ;          5611      then
10708 ;          5612          begin
10709 ;          5613          ERRDF (15, ASYNC, DUMPER);        !ERROR IF CLEARED
10710 ;          5614          PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_43, WRD_1, PHR_6);
10711 ;          5615          end;
10712 ;          5616
10713 ;          5617      DODU_FLG = ONE;                       !SET DODU_FLG
10714 ;          5618      end
10715 ;          5619  else
10716 ;          5620
10717 ;          5621      if .DRY IS_SET                        !THE GO IS SET SO SEE IF DRY IS CLEARED
10718 ;          5622      then
10719 ;          5623          begin
10720 ;          5624          ERRDF (16, ASYNC, DUMPER);        !ERROR IF DRY IS SET
```

```
10722 ;ML4AD
10723 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
10724 ;                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (65)
10725 ;          5625           PRINTB (FIV_FMT, WRD_2, PHR_2, WRD_43, WRD_1, PHR_5);
10726 ;          5626           DODU_FLG = ONE;
10727 ;          5627           end;
10728 ;          5628
10729 ;          5629   if .ILF IS_SET                                       !SEE IF ILF IS SET
10730 ;          5630   then
10731 ;          5631       begin
10732 ;          5632       ERRDF (17, ASYNC, DUMPER);                       !ERROR IF SET
10733 ;          5633       PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_12, FNC_4, WRD_19);
10734 ;          5634       end;
10735 ;          5635
10736 ;          5636   if .OPI IS_SET                                       !SEE IF OPI IS SET
10737 ;          5637   then
10738 ;          5638       begin
10739 ;          5639       ERRDF (18, ASYNC, DUMPER);                       !ERROR IF SET
10740 ;          5640       PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_4, WRD_19);
10741 ;          5641       end;
10742 ;          5642
10743 ;          5643   DELAY (FRTY_US);                                     !WAIT FO XFERR TO COMPLTE
10744 ;          5644
10745 ;          5645   if .DRY IS_NOT_SET                                   !SEE IF DRY IS SETS AFTER XFERR
10746 ;          5646   then
10747 ;          5647       begin
10748 ;          5648
10749 ;          5649       if .GO IS_SET                                    !IF DRY IS NOT SET THEN SEE IF GO IS SET
10750 ;          5650       then
10751 ;          5651           begin
10752 ;          5652           CLR_MBUS;                                    !IF THE GO IS SET THEN TRY TO CLR GO
10753 ;          5653
10754 ;          5654           if .GO IS_SET then ERRDF (19, ASYNC, DUMPER) else ERRDF (20, SYNC, DUMPER);
10755 ;          5655
10756 ;          5656                                                        !IF GO IS STILL SET THEN ASYNC FAILURE
10757 ;          5657           PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_4, WRD_19);  !ELSE SYNC MODLE FAILURE
10758 ;          5658           end
10759 ;          5659       else
10760 ;          5660           begin
10761 ;          5661           ERRDF (21, ASYNC, DUMPER);                   !ERROR GO AND DRY BOTH CLEARED
10762 ;          5662           PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_11, FNC_4, WRD_19);
10763 ;          5663           end;
10764 ;          5664
10765 ;          5665       DODU_FLG = ONE;                                  !SET THE DODU_FLG
10766 ;          5666       end;
10767 ;          5667
10768 ;          5668   if .GO IS_SET                                        !SEE IF THE GO IS STILL SET
10769 ;          5669   then
10770 ;          5670       begin
10771 ;          5671       CLR_MBUS;                                        !TRY TO CLR GO IF STILL SET
10772 ;          5672
10773 ;          5673       if .GO IS_SET then ERRDF (22, ASYNC, DUMPER) else ERRDF (23, SYNC, DUMPER);
10774 ;          5674
10775 ;          5675                                                        !IF GO IS STILL SET THE ASYNC FAILURE
10776 ;          5676       PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_4, WRD_19);      !ELSE SYNC MODULE FAILURE
```

```
;ML4AD
;                       TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
                                                                            29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (65)
;             5677          DODU_FLG = ONE;
;             5678          end;
;             5679
;             5680      ENDSUB;
;             5681
;             5682      if .DODU_FLG IS_SET                                 !DROP THIS UNIT IF DODU_FLG IS_SET
;             5683      then
;             5684          begin
;             5685          DODU (.ML_LUN);
;             5686          DOCLN;
;             5687          end;
;             5688
;             5689      ENDTST;


                                               .SBTTL  $T13 TEST CODE SECTION
037102  004167  145432               $T13:     JSR     R1,$SAVE2                        ;                   5571
037106  005746                                 TST     -(SP)
037110  104402                       1$:       TRAP    2
037112  152777  000040  156324                 BISB    #40,@ML.REG+40                   ;                   5596
037120  016701  156706                         MOV     ML.DUT,R1                        ;                   5598
037124  042701  177770                         BIC     #177770,R1
037130  142777  000007  156306                 BICB    #7,@ML.REG+40
037136  150177  156302                         BISB    R1,@ML.REG+40
037142  005002                                 CLR     R2                               ; DODU.FLG
037144  004767  161112                         JSR     PC,FIRST.BLK.XFER                                    5600
037150  012777  000051  156226                 MOV     #51,@ML.REG                      ;                   5601
037156  132777  000001  156220                 BITB    #1,@ML.REG                       ;                   5602
037164  001057                                 BNE     2$                               ;                   5604
037166  104455                                 TRAP    55
037170  000016                                 .WORD   16                               ;                   5607
037172  012706                                 .WORD   ASYNC
037174  026302                                 .WORD   DUMPER
037176  012746  010730                         MOV     #WRD.19,-(SP)
037202  012746  012200                         MOV     #FNC.4,-(SP)                     ;                   5608
037206  012746  010650                         MOV     #WRD.12,-(SP)
037212  012746  011610                         MOV     #PHR.1,-(SP)
037216  012746  010520                         MOV     #WRD.1,-(SP)
037222  012746  010304                         MOV     #FIV.FMT,-(SP)
037226  012746  000006                         MOV     #6,-(SP)
037232  010600                                 MOV     SP,R0                            ; SP,*
037234  104414                                 TRAP    14
037236  132777  000200  156210                 BITB    #200,@ML.REG+50                  ;                   5610
037244  001056                                 BNE     3$
037246  104455                                 TRAP    55                               ;                   5613
037250  000017                                 .WORD   17
037252  012706                                 .WORD   ASYNC
```

SEQ 0245


```
10834
10835                                   ;ML4AD
10836                                   ;               TEST CODE SECTION
10837 037254  026302                            .WORD   DUMPER
10838 037256  012746  011726                      MOV     #PHR.6,-(SP)          ;                       5614
10839 037262  012746  010520                      MOV     #WRD.1,-(SP)
10840 037266  012746  011176                      MOV     #WRD.43,-(SP)
10841 037272  012746  011610                      MOV     #PHR.1,-(SP)
10842 037276  012746  010524                      MOV     #WRD.2,-(SP)
10843 037302  012746  010304                      MOV     #FIV.FMT,-(SP)
10844 037306  012746  000006                      MOV     #6,-(SP)
10845 037312  010600                            MOV     SP,R0                 ; SP,*
10846 037314  104414                            TRAP    14
10847 037316  062706  000016                      ADD     #16,SP                ;                       5612
10848 037322  000427                            BR      3$                    ;                       5617
10849 037324  105777  156124          2$:       TSTB    @ML.REG+50            ;                       5621
10850 037330  100030                            BPL     4$
10851 037332  104455                            TRAP    55                    ;                       5624
10852 037334  000020                            .WORD   20
10853 037336  012706                            .WORD   ASYNC
10854 037340  026302                            .WORD   DUMPER
10855 037342  012746  011714                      MOV     #PHR.5,-(SP)          ;                       5625
10856 037346  012746  010520                      MOV     #WRD.1,-(SP)
10857 037352  012746  011176                      MOV     #WRD.43,-(SP)
10858 037356  012746  011626                      MOV     #PHR.2,-(SP)
10859 037362  012746  010524                      MOV     #WRD.2,-(SP)
10860 037366  012746  010304                      MOV     #FIV.FMT,-(SP)
10861 037372  012746  000006                      MOV     #6,-(SP)
10862 037376  010600                            MOV     SP,R0                 ; SP,*
10863 037400  104414                            TRAP    14
10864 037402  012702  000001          3$:       MOV     #1,R2                 ; *,DODU.FLG            5626
10865 037406  062706  000016                      ADD     #16,SP                ;                       5623
10866 037412  132777  000C01  156044  4$:       BITB    #1,@ML.REG+60         ;                       5629
10867 037420  001426                            BEQ     5$
10868 037422  104455                            TRAP    55                    ;                       5632
10869 037424  000021                            .WORD   21
10870 037426  012706                            .WORD   ASYNC
10871 037430  026302                            .WORD   DUMPER
10872 037432  012746  010730                      MOV     #WRD.19,-(SP)         ;                       5633
10873 037436  012746  012200                      MOV     #FNC.4,-(SP)
10874 037442  012746  010650                      MOV     #WRD.12,-(SP)
10875 037446  012746  011714                      MOV     #PHR.5,-(SP)
10876 037452  012746  010536                      MOV     #WRD.3,-(SP)
10877 037456  012746  010304                      MOV     #FIV.FMT,-(SP)
10878 037462  012746  000006                      MOV     #6,-(SP)
10879 037466  010600                            MOV     SP,R0                 ; SP,*
10880 037470  104414                            TRAP    14
10881 037472  062706  000016                      ADD     #16,SP                ;                       5631
10882 037476  032777  020000  155760  5$:       BIT     #20000,@ML.REG+60     ;                       5636
10883 037504  001426                            BEQ     6$
10884 037506  104455                            TRAP    55                    ;                       5639
10885 037510  000022                            .WORD   22
10886 037512  012706                            .WORD   ASYNC
10887 037514  026302                            .WORD   DUMPER
10888 037516  012746  010730                    MOV     #WRD.19,-(SP)         ;                       5640
```

```
10890                                           ;ML4AD
10891                                           ;               TEST CODE SECTION
10892
10893 037522  012746  012200                          MOV     #FNC.4,-(SP)
10894 037526  012746  010650                          MOV     #WRD.12,-(SP)
10895 037532  012746  011714                          MOV     #PHR.5,-(SP)
10896 037536  012746  010544                          MOV     #WRD.4,-(SP)
10897 037542  012746  010304                          MOV     #FIV.FMT,-(SP)
10898 037546  012746  000006                          MOV     #6,-(SP)
10899 037552  010600                                 MOV     SP,R0           ; SP,*
10900 037554  104414                                 TRAP    14
10901 037556  062706  000016                          ADD     #16,SP          ;                                       5638
10902 037562  012700  000050                  6$:     MOV     #50,R0          ; *,$$TMP2                              5643
10903 037566  001410                          7$:     BEQ     10$
10904 037570  016701  142322                          MOV     L$DLY,R1        ; *,$$TMP1
10905 037574  001403                                 BEQ     9$
10906 037576  005016                          8$:     CLR     (SP)            ; $$TMP
10907 037600  005301                                 DEC     R1              ; $$TMP1
10908 037602  001375                                 BNE     8$
10909 037604  005300                          9$:     DEC     R0              ; $$TMP2
10910 037606  000767                                 BR      7$
10911 037610  132777  000200  155636    10$:    BITB    #200,@ML.REG+50 ;                                       5645
10912 037616  001106                                 BNE     15$
10913 037620  132777  000001  155556                  BITB    #1,@ML.REG      ;                                       5649
10914 037626  001452                                 BEQ     13$
10915 037630  152777  000040  155606                  BISB    #40,@ML.REG+40  ;                                       5651
10916 037636  016701  156170                          MOV     ML.DUT,R1
10917 037642  042701  177770                          BIC     #177770,R1
10918 037646  142777  000007  155570                  BICB    #7,@ML.REG+40
10919 037654  150177  155564                          BISB    R1,@ML.REG+40
10920 037660  132777  000001  155516                  BITB    #1,@ML.REG      ;                                       5654
10921 037666  001405                                 BEQ     11$
10922 037670  104455                                 TRAP    55
10923 037672  000023                                 .WORD   23
10924 037674  012706                                 .WORD   ASYNC
10925 037676  026302                                 .WORD   DUMPER
10926 037700  000404                                 BR      12$
10927 037702  104455                          11$:    TRAP    55
10928 037704  000024                                 .WORD   24
10929 037706  012750                                 .WORD   SYNC
10930 037710  026302                                 .WORD   DUMPER
10931 037712  012746  010730                  12$:    MOV     #WRD.19,-(SP)   ;                                       5657
10932 037716  012746  012200                          MOV     #FNC.4,-(SP)
10933 037722  012746  010640                          MOV     #WRD.11,-(SP)
10934 037726  012746  011626                          MOV     #PHR.2,-(SP)
10935 037732  012746  010520                          MOV     #WRD.1,-(SP)
10936 037736  012746  010304                          MOV     #FIV.FMT,-(SP)
10937 037742  012746  000006                          MOV     #6,-(SP)
10938 037746  010600                                 MOV     SP,R0           ; SP,*
10939 037750  104414                                 TRAP    14
10940 037752  000424                                 BR      14$
10941 037754  104455                          13$:    TRAP    55              ;                                       5649
10942 037756  000025                                 .WORD   25              ;                                       5661
10943 037760  012706                                 .WORD   ASYNC
10944 037762  026302                                 .WORD   DUMPER
```

SEQ 0247

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION                                  29-Mar-1982 16:23:04   TOPS
                                                                                                           29-Mar-1982 16:21:03   PA:<
10946
10947
10948
10949 037764  012746  010730                    MOV     #WRD.19,-(SP)           ;                                          5662
10950 037770  012746  012200                    MOV     #FNC.4,-(SP)
10951 037774  012746  010640                    MOV     #WRD.11,-(SP)
10952 040000  012746  011610                    MOV     #PHR.1,-(SP)
10953 040004  012746  010524                    MOV     #WRD.2,-(SP)
10954 040010  012746  010304                    MOV     #FIV.FMT,-(SP)
10955 040014  012746  000006                    MOV     #6,-(SP)
10956 040020  010600                            MOV     SP,R0                   ; SP,*
10957 040022  104414                            TRAP    14
10958 040024  012702  000001            14$:    MOV     #1,R2                   ; *,DODU.FLG                               5665
10959 040030  062706  000016                    ADD     #16,SP                  ;                                          5647
10960 040034  132777  000001  155342    15$:    BITB    #1,@ML.REG              ;                                          5668
10961 040042  001455                            BEQ     18$
10962 040044  152777  000040  155372            BISB    #40,@ML.REG+40          ;                                          5670
10963 040052  016701  155754                    MOV     ML.DUT,R1
10964 040056  042701  177770                    BIC     #177770,R1
10965 040062  142777  000007  155354            BICB    #7,@ML.REG+40
10966 040070  150177  155350                    BISB    R1,@ML.REG+40
10967 040074  132777  000001  155302            BITB    #1,@ML.REG              ;                                          5673
10968 040102  001405                            BEQ     16$
10969 040104  104455                            TRAP    55
10970 040106  000026                            .WORD   26
10971 040110  012706                            .WORD   ASYNC
10972 040112  026302                            .WORD   DUMPER
10973 040114  000404                            BR      17$
10974 040116  104455            16$:    TRAP    55
10975 040120  000027                            .WORD   27
10976 040122  012750                            .WORD   SYNC
10977 040124  026302                            .WORD   DUMPER
10978 040126  012746  010730    17$:    MOV     #WRD.19,-(SP)           ;                                          5676
10979 040132  012746  012200                    MOV     #FNC.4,-(SP)
10980 040136  012746  010640                    MOV     #WRD.11,-(SP)
10981 040142  012746  011626                    MOV     #PHR.2,-(SP)
10982 040146  012746  010520                    MOV     #WRD.1,-(SP)
10983 040152  012746  010304                    MOV     #FIV.FMT,-(SP)
10984 040156  012746  000006                    MOV     #6,-(SP)
10985 040162  010600                            MOV     SP,R0                   ; SP,*
10986 040164  104414                            TRAP    14
10987 040166  012702  000001                    MOV     #1,R2                   ; *,DODU.FLG                               5677
10988 040172  062706  000016                    ADD     #16,SP                  ;                                          5670
10989 040176  104467            18$:    TRAP    67                      ;                                          5678
10990 040200  006000                            ROR     R0
10991 040202  103002                            BHIS    19$
10992 040204  000167  176700                    JMP     1$
10993 040210  005302            19$:    DEC     R2                      ; DODU.FLG                                 5682
10994 040212  001004                            BNE     20$
10995 040214  016700  155610                    MOV     ML.LUN,R0               ;                                          5685
10996 040220  104451                            TRAP    51
10997 040222  104444                            TRAP    44
10998 040224  005726            20$:    TST     (SP)+                   ;                                          5571
10999 040226  000207                            RTS     PC
1*000
```

```
11002                                   ;ML4AD
11003                                   ;               TEST CODE SECTION
11004
11005                                   ; Routine Size:  299 words
11006                                   ; Maximum stack depth per invocation:  18 words
11011
11012
11016
11017                                           .SBTTL  T13 TEST CODE SECTION
11021 040230                            T13::
11022 040230  004767  176646            1$:     JSR     PC,$T13                 ;               5687
11023 040234  104466                            TRAP    66
11024 040236  006000                            ROR     R0
11025 040240  103773                            BLO     1$
11026 040242  000207                            RTS     PC
11027
11028                                   ; Routine Size:  6 words
11029                                   ; Maximum stack depth per invocation:  0 words
11034
11035
11036 ;         5690  !<BLF/PAGE>
```

```
11038 ;ML4AD
11039 ;                 TEST CODE SECTION
11040 ;
11041 ;          5691   !
11042 ;          5692   BGNTST;
11043 ;          5693
11044 ;          5694   !++
11045 ;          5695   ! TEST NUMBER:  TST 14
11046 ;          5696   !
11047 ;          5697   ! TEST NAME:    WRITE FUNCTION TEST
11048 ;          5698   !
11049 ;          5699   ! TEST DESCRIPTION:
11050 ;          5700   !       TEST IF THE DRIVE CAN PERFORM A WRITE FUNCTION WITHOUT
11051 ;          5701   !       HANGING THE DRIVE.
11052 ;          5702   !
11053 ;          5703   !       A WRITE FUNCTION IS WRITTEN INTO MLCS1.  THEN GO AND ERROR BITS ARE
11054 ;          5704   !       CHECKED FOR CORRECT STATUS.  THIS UNIT IS DROPPED ON DETECTED ERRORS.
11055 ;          5705   !--
11056 ;          5706
11057 ;          5707   local
11058 ;          5708       DODU_FLG;                                   !DROP UNIT FLAG
11059 ;          5709
11060 ;          5710   BGNSUB;
11061 ;          5711   CLR_MBUS;
11062 ;          5712   DODU_FLG = ZERO;
11063 ;          5713   FIRST_BLK_XFER ();                              !SET UP A FIRST BLOCK XFERR
11064 ;          5714   MLCS1 = write;                                  !DO A WRITE FUNCTION
11065 ;          5715
11066 ;          5716   if .GO IS_NOT_SET                               !SEE IF THE GO IS SET
11067 ;          5717   then
11068 ;          5718       begin                                       !ERROR IF NOT SET
11069 ;          5719       ERRDF (24, ASYNC, DUMPER);
11070 ;          5720       PRINTB (FIV_FMT, WRD_1, PHR_1, WRD_12, FNC_5, WRD_19);
11071 ;          5721
11072 ;          5722       if .DRY IS_NOT_SET                          !SEE IF DRY SET WITH GO CLEAR
11073 ;          5723       then
11074 ;          5724           begin
11075 ;          5725           ERRDF (25, ASYNC, DUMPER);
11076 ;          5726           PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_43, WRD_1, PHR_6);
11077 ;          5727           end;
11078 ;          5728
11079 ;          5729       DODU_FLG = ONE;                             !SET DODU_FLG
11080 ;          5730       end
11081 ;          5731   else                                            !GO IS SET DURING FUNCTION
11082 ;          5732
11083 ;          5733       if .DRY IS_SET                              !SEE IF DRY CLEAR WITH GO SET
11084 ;          5734       then
11085 ;          5735           begin                                   !ERROR IF SET
11086 ;          5736           ERRDF (26, ASYNC, DUMPER);
11087 ;          5737           PRINTB (FIV_FMT, WRD_2, PHR_2, WRD_43, WRD_1, PHR_5);
11088 ;          5738           DODU_FLG = ONE;
11089 ;          5739           end;
11090 ;          5740
11091 ;          5741   if .ILF IS_SET                                  !DID FUNCTION CAUSE ILF
11092 ;          5742   then
```

```
11094 ;ML4AD
11095 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
11096 ;                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (66)
11097 ;          5743       begin                                           !ERROR IF YES
11098 ;          5744       ERRDF (27, ASYNC, DUMPER);
11099 ;          5745       PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_12, FNC_5, WRD_19);
11100 ;          5746       end;
11101 ;          5747
11102 ;          5748  if .OPI IS_SET                                       !DID FUNCTION CAUSE OPI
11103 ;          5749  then
11104 ;          5750       begin                                           !ERROR IF YES
11105 ;          5751       ERRDF (28, ASYNC, DUMPER);
11106 ;          5752       PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_5, WRD_19);
11107 ;          5753       end;
11108 ;          5754
11109 ;          5755  DELAY (FRTY_US);                                     !WAIT FOR XFERR TO COMPLETE
11110 ;          5756
11111 ;          5757  if .DRY IS_NOT_SET                                   !SEE IF DRY CLEARED AFTER XFERR
11112 ;          5758  then
11113 ;          5759       begin
11114 ;          5760
11115 ;          5761       if .GO IS_SET                                   !TST GO CLR IF DRY NOT SET
11116 ;          5762       then
11117 ;          5763            begin
11118 ;          5764            CLR_MBUS;                                  !CLEAR GO IF STILL SET
11119 ;          5765
11120 ;          5766            if .GO IS_SET then ERRDF (29, ASYNC, DUMPER) else ERRDF (30, SYNC, DUMPER);
11121 ;          5767
11122 ;          5768                                                       !TST GO TO DETERMINE FAILING MOD
11123 ;          5769            PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_5, WRD_19);
11124 ;          5770            end
11125 ;          5771       else                                            !DRY NOT SET AND GO CLEARED
11126 ;          5772            begin                                      !REPORT ERROR
11127 ;          5773            ERRDF (31, ASYNC, DUMPER);
11128 ;          5774            PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_11, FNC_5, WRD_19);
11129 ;          5775            end;
11130 ;          5776
11131 ;          5777       DODU_FLG = ONE;                                 !SET DODU_FLG
11132 ;          5778       end;
11133 ;          5779
11134 ;          5780  if .GO IS_SET                                       !SEE IF GO CLEARED AFTER XFERR.
11135 ;          5781  then
11136 ;          5782       begin
11137 ;          5783       CLR_MBUS;                                       !CLEAR GO IF STILL SET
11138 ;          5784
11139 ;          5785       if .GO IS_SET then ERRDF (32, ASYNC, DUMPER) else ERRDF (33, SYNC, DUMPER);
11140 ;          5786
11141 ;          5787                                                       !TST GO TO DETERMINE FAILING MOD
11142 ;          5788       PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_5, WRD_19);
11143 ;          5789       DODU_FLG = ONE;
11144 ;          5790       end;
11145 ;          5791
11146 ;          5792  ENDSUB;
11147 ;          5793
11148 ;          5794  if .TRE IS_SET                                       !SEE IF XFERR CAUSED A TRANSFER ERROR
```

```
 11150 ;ML4AD
 11151 ;                 TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 11152 ;                                                                    29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (66)
 11153 ;          5795  then
 11154 ;          5796      begin                                          !REPORT ERROR IF SET AND CONTINUE TESTING
 11155 ;          5797      ERRDF (115, INTER, DUMPER);
 11156 ;          5798      PRINTB (SIX_FMT, WRD_61, WRD_20, PHR_5, WRD_12, FNC_5, WRD_19);
 11157 ;          5799      end;
 11158 ;          5800
 11159 ;          5801  if .DODU_FLG IS_SET                                !DROP THIS UNIT IF DODU_FLG SET
 11160 ;          5802  then
 11161 ;          5803      begin
 11162 ;          5804      DODU (.ML_LUN);
 11163 ;          5805      DOCLN;
 11164 ;          5806      end;
 11165 ;          5807
 11166 ;          5808  ENDTST;
 11170
 11171                                      .SBTTL  $T14 TEST CODE SECTION
 11175 040244  004167  144270          $T14: JSR     R1,$SAVE2                ;                                        5689
 11176 040250  005746                        TST     -(SP)
 11177 040252  104402                  1$:   TRAP    2
 11178 040254  152777  000040  155162        BISB    #40,@ML.REG+40           ;                                        5708
 11179 040262  016701  155544                MOV     ML.DUT,R1                ;                                        5710
 11180 040266  042701  177770                BIC     #177770,R1
 11181 040272  142777  000007  155144        BICB    #7,@ML.REG+40
 11182 040300  150177  155140                BISB    R1,@ML.REG+40
 11183 040304  005002                        CLR     R2
 11184 040306  004767  157750                JSR     PC,FIRST.BLK.XFER        ; DODU.FLG                               5712
 11185 040312  012777  000061  155064        MOV     #61,@ML.REG              ;                                        5713
 11186 040320  132777  000001  155056        BITB    #1,@ML.REG               ;                                        5714
 11187 040326  001057                        BNE     2$                       ;                                        5716
 11188 040330  104455                        TRAP    55
 11189 040332  000030                        .WORD   30                       ;                                        5719
 11190 040334  012706                        .WORD   ASYNC
 11191 040336  026302                        .WORD   DUMPER
 11192 040340  012746  010730                MOV     #WRD.19,-(SP)
 11193 040344  012746  012216                MOV     #FNC.5,-(SP)             ;                                        5720
 11194 040350  012746  010650                MOV     #WRD.12,-(SP)
 11195 040354  012746  011610                MOV     #PHR.1,-(SP)
 11196 040360  012746  010520                MOV     #WRD.1,-(SP)
 11197 040364  012746  010304                MOV     #FIV.FMT,-(SP)
 11198 040370  012746  000006                MOV     #6,-(SP)
 11199 040374  010600                        MOV     SP,R0
 11200 040376  104414                        TRAP    14                       ; SP,*
 11201 040400  132777  000200  155046        BITB    #200,@ML.REG+50          ;                                        5722
 11202 040406  001056                        BNE     3$
 11203 040410  104455                        TRAP    55
 11204 040412  000031                        .WORD   31                       ;                                        5725
```

```
11206                                   ;ML4AD
11207                                   ;               TEST CODE SECTION
11208
11209 040414  012706                              .WORD   ASYNC
11210 040416  026302                              .WORD   DUMPER
11211 040420  012746  011726                      MOV     #PHR.6,-(SP)            ;                               5726
11212 040424  012746  010520                      MOV     #WRD.1,-(SP)
11213 040430  012746  011176                      MOV     #WRD.43,-(SP)
11214 040434  012746  011610                      MOV     #PHR.1,-(SP)
11215 040440  012746  010524                      MOV     #WRD.2,-(SP)
11216 040444  012746  010304                      MOV     #FIV.FMT,-(SP)
11217 040450  012746  000006                      MOV     #6,-(SP)
11218 040454  010600                             MOV     SP,R0                   ; SP,*
11219 040456  104414                             TRAP    14
11220 040460  062706  000016                      ADD     #16,SP                  ;                               5724
11221 040464  000427                             BR      3$                      ;                               5729
11222 040466  105777  154762            2$:     TSTB    @ML.REG+50              ;                               5733
11223 040472  100030                             BPL     4$
11224 040474  104455                             TRAP    55                      ;                               5736
11225 040476  000032                             .WORD   32
11226 040500  012706                             .WORD   ASYNC
11227 040502  026302                             .WORD   DUMPER
11228 040504  012746  011714                      MOV     #PHR.5,-(SP)            ;                               5737
11229 040510  012746  010520                      MOV     #WRD.1,-(SP)
11230 040514  012746  011176                      MOV     #WRD.43,-(SP)
11231 040520  012746  011626                      MOV     #PHR.2,-(SP)
11232 040524  012746  010524                      MOV     #WRD.2,-(SP)
11233 040530  012746  010304                      MOV     #FIV.FMT,-(SP)
11234 040534  012746  000006                      MOV     #6,-(SP)
11235 040540  010600                             MOV     SP,R0                   ; SP,*
11236 040542  104414                             TRAP    14
11237 040544  012702  000001            3$:     MOV     #1,R2                   ; *,DODU.FLG                    5738
11238 040550  062706  000016                      ADD     #16,SP                  ;                               5735
11239 040554  132777  000001  154702    4$:     BITB    #1,@ML.REG+60           ;                               5741
11240 040562  001426                             BEQ     5$
11241 040564  104455                             TRAP    55                      ;                               5744
11242 040566  000033                             .WORD   33
11243 040570  012706                             .WORD   ASYNC
11244 040572  026302                             .WORD   DUMPER
11245 040574  012746  010730                      MOV     #WRD.19,-(SP)           ;                               5745
11246 040600  012746  012216                      MOV     #FNC.5,-(SP)
11247 040604  012746  010650                      MOV     #WRD.12,-(SP)
11248 040610  012746  011714                      MOV     #PHR.5,-(SP)
11249 040614  012746  010536                      MOV     #WRD.3,-(SP)
11250 040620  012746  010304                      MOV     #FIV.FMT,-(SP)
11251 040624  012746  000006                      MOV     #6,-(SP)
11252 040630  010600                             MOV     SP,R0                   ; SP,*
11253 040632  104414                             TRAP    14
11254 040634  062706  000016                      ADD     #16,SP                  ;                               5743
11255 040640  032777  020000  154616    5$:     BIT     #20000,@ML.REG+60       ;                               5748
11256 040646  001426                             BEQ     6$
11257 040650  104455                             TRAP    55                      ;                               5751
11258 040652  000034                             .WORD   34
11259 040654  012706                             .WORD   ASYNC
11260 040656  026302                             .WORD   DUMPER
```

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION
11265 040660  012746  010730                   MOV     #WRD.19,-(SP)           ;                                       5752
11266 040664  012746  012216                   MOV     #FNC.5,-(SP)
11267 040670  012746  010650                   MOV     #WRD.12,-(SP)
11268 040674  012746  011714                   MOV     #PHR.5,-(SP)
11269 040700  012746  010544                   MOV     #WRD.4,-(SP)
11270 040704  012746  010304                   MOV     #FIV.FMT,-(SP)
11271 040710  012746  000006                   MOV     #6,-(SP)
11272 040714  010600                          MOV     SP,R0                   ; SP,*
11273 040716  104414                          TRAP    14
11274 040720  062706  000016                   ADD     #16,SP                  ;
11275 040724  012700  000050            6$:    MOV     #50,R0                  ; *,$STMP2                              5750
11276 040730  001410                    7$:    BEQ     10$                     ;                                       5755
11277 040732  016701  141160                   MOV     L$DLY,R1                ; *,$STMP1
11278 040736  001403                          BEQ     9$
11279 040740  005016                    8$:    CLR     (SP)                    ; $STMP
11280 040742  005301                          DEC     R1                      ; $STMP1
11281 040744  001375                          BNE     8$
11282 040746  005300                    9$:    DEC     R0                      ; $STMP2
11283 040750  000767                          BR      7$
11284 040752  132777  000200  154474   10$:    BITB    #200,@ML.REG+50         ;                                       5757
11285 040760  001106                          BNE     15$
11286 040762  132777  000001  154414          BITB    #1,@ML.REG              ;                                       5761
11287 040770  001452                          BEQ     13$
11288 040772  152777  000040  154444          BISB    #40,@ML.REG+40          ;                                       5763
11289 041000  016701  155026                   MOV     ML.DUT,R1
11290 041004  042701  177770                   BIC     #177770,R1
11291 041010  142777  000007  154426          BICB    #7,@ML.REG+40
11292 041016  150177  154422                   BISB    R1,@ML.REG+40
11293 041022  132777  000001  154354          BITB    #1,@ML.REG              ;                                       5766
11294 041030  001405                          BEQ     11$
11295 041032  104455                          TRAP    55
11296 041034  000035                          .WORD   35
11297 041036  012706                          .WORD   ASYNC
11298 041040  026302                          .WORD   DUMPER
11299 041042  000404                          BR      12$
11300 041044  104455                    11$:   TRAP    55
11301 041046  000036                          .WORD   36
11302 041050  012750                          .WORD   SYNC
11303 041052  026302                          .WORD   DUMPER
11304 041054  012746  010730            12$:   MOV     #WRD.19,-(SP)           ;                                       5769
11305 041060  012746  012216                   MOV     #FNC.5,-(SP)
11306 041064  012746  010640                   MOV     #WRD.11,-(SP)
11307 041070  012746  011626                   MOV     #PHR.2,-(SP)
11308 041074  012746  010520                   MOV     #WRD.1,-(SP)
11309 041100  012746  010304                   MOV     #FIV.FMT,-(SP)
11310 041104  012746  000006                   MOV     #6,-(SP)
11311 041110  010600                          MOV     SP,R0                   ; SP,*
11312 041112  104414                          TRAP    14
11313 041114  000424                          BR      14$
11314 041116  104455                    13$:   TRAP    55                      ;                                       5761
11315 041120  000037                          .WORD   37                      ;                                       5773
11316 041122  012706                          .WORD   ASYNC
```

```
                                  ;ML4AD                                                    29-Mar-1982 16:23:04   TOPS
                                  ;            TEST CODE SECTION                            29-Mar-1982 16:21:03   PA:<
041124  026302                         .WORD   DUMPER
041126  012746  010730                 MOV     #WRD.19,-(SP)          ;                                    5774
041132  012746  012216                 MOV     #FNC.5,-(SP)
041136  012746  010640                 MOV     #WRD.11,-(SP)
041142  012746  011610                 MOV     #PHR.1,-(SP)
041146  012746  010524                 MOV     #WRD.2,-(SP)
041152  012746  010304                 MOV     #FIV.FMT,-(SP)
041156  012746  000006                 MOV     #6,-(SP)
041162  010600                         MOV     SP,R0
041164  104414                         TRAP    14                     ; SP,*
041166  012702  000001          14$:   MOV     #1,R2                  ; *,DODU.FLG
041172  062706  000016                 ADD     #16,SP                 ;                                    5777
041176  132777  000001  154200  15$:   BITB    #1,@ML.REG             ;                                    5759
041204  001455                         BEQ     18$                    ;                                    5780
041206  152777  000040  154230         BISB    #40,@ML.REG+40         ;                                    5782
041214  016701  154612                 MOV     ML.DUT,R1
041220  042701  177770                 BIC     #177770,R1
041224  142777  000007  154212         BICB    #7,@ML.REG+40
041232  150177  154206                 BISB    R1,@ML.REG+40
041236  132777  000001  154140         BITB    #1,@ML.REG
041244  001405                         BEQ     16$                    ;                                    5785
041246  104455                         TRAP    55
041250  000040                         .WORD   40
041252  012706                         .WORD   ASYNC
041254  026302                         .WORD   DUMPER
041256  000404                         BR      17$
041260  104455                  16$:   TRAP    55
041262  000041                         .WORD   41
041264  012750                         .WORD   SYNC
041266  026302                         .WORD   DUMPER
041270  012746  010730          17$:   MOV     #WRD.19,-(SP)          ;                                    5788
041274  012746  012216                 MOV     #FNC.5,-(SP)
041300  012746  010640                 MOV     #WRD.11,-(SP)
041304  012746  011626                 MOV     #PHR.2,-(SP)
041310  012746  010520                 MOV     #WRD.1,-(SP)
041314  012746  010304                 MOV     #FIV.FMT,-(SP)
041320  012746  000006                 MOV     #6,-(SP)
041324  010600                         MOV     SP,R0
041326  104414                         TRAP    14                     ; SP,*
041330  012702  000001                 MOV     #1,R2                  ; *,DODU.FLG
041334  062706  000016                 ADD     #16,SP                 ;                                    5789
041340  104467                  18$:   TRAP    67                     ;                                    5782
041342  006000                         ROR     R0                     ;                                    5790
041344  103002                         BHIS    19$
041346  000167  176700                 JMP     1$
041352  032777  040000  154024  19$:   BIT     #40000,@ML.REG         ;                                    5794
041360  001430                         BEQ     20$
041362  104455                         TRAP    55                     ;                                    5797
041364  000163                         .WORD   163
041366  013114                         .WORD   INTER
041370  026302                         .WORD   DUMPER
041372  012746  010730                 MOV     #WRD.19,-(SP)          ;                                    5798
```

```
11374                                   ;ML4AD
11375                                   ;               TEST CODE SECTION                                       29-Mar-1982 16:23:04    TOPS
11376                                                                                                           29-Mar-1982 16:21:03    PA.
11377 041376  012746  012216                      MOV     #FNC.5,-(SP)
11378 041402  012746  010650                      MOV     #WRD.12,-(SP)
11379 041406  012746  011714                      MOV     #PHR.5,-(SP)
11380 041412  012746  010736                      MOV     #WRD.20,-(SP)
11381 041416  012746  011416                      MOV     #WRD.61,-(SP)
11382 041422  012746  010322                      MOV     #SIX.FMT,-(SP)
11383 041426  012746  000007                      MOV     #7,-(SP)
11384 041432  010600                            MOV     SP,R0                           ; SP,*
11385 041434  104414                            TRAP    14
11386 041436  062706  000020                    ADD     #20,SP                          ;
11387 041442  005302                    20$:    DEC     R2                              ; DODU.FLG                      5796
11388 041444  001004                            BNE     21$                                                             5801
11389 041446  016700  154356                    MOV     ML.LUN,R0                       ;                               5804
11390 041452  104451                            TRAP    51
11391 041454  104444                            TRAP    44
11392 041456  005726                    21$:    TST     (SP)+                           ;                               5689
11393 041460  000207                            RTS     PC
11394
11395                                   ; Routine Size:  327 words
11396                                   ; Maximum stack depth per invocation:  18 words
11401
11402
11406
11407                                           .SBTTL  T14 TEST CODE SECTION
11411 041462                            T14::
11412 041462  004767  176556            1$:     JSR     PC,$T14                         ;                               5806
11413 041466  104466                            TRAP    66
11414 041470  006000                            ROR     R0
11415 041472  103773                            BLO     1$
11416 041474  000207                            RTS     PC
11417
11418                                   ; Routine Size:  6 words
11419                                   ; Maximum stack depth per invocation:  0 words
11424
11425
11426 ;          5809  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (67)

```
;ML4AD
;                    TEST CODE SECTION

;          5810  !
;          5811  !
;          5812  BGNTST;
;          5813
;          5814  !++
;          5815  ! TEST NUMBER:  TST 15
;          5816  !
;          5817  ! TEST NAME:    READ FUNCTION TEST
;          5818  !
;          5819  ! TEST DESCRIPTION:
;          5820  !
;          5821  !       TEST IF THE DRIVE CAN PERFORM
;          5822  !       A READ FUNCTION WITHOUT
;          5823  !       HANGING THE DRIVE.
;          5824  !
;          5825  !       A READ FUNCTION IS WRITTEN
;          5826  !       INTO MLCS1
;          5827  !
;          5828  !       THEN GO AND ERROR BITS ARE
;          5829  !       CHECKED FOR CORRECT STATES.
;          5830  !
;          5831  !--
;          5832
;          5833  local
;          5834      DODU_FLG;                                   !DROP UNIT FLAG
;          5835
;          5836  BGNSUB;
;          5837  CLR_MBUS;
;          5838  DODU_FLG = ZERO;
;          5839  FIRST_BLK_XFER ();                              !SET UP A FIRST BLK XFERR
;          5840  MLCS1 = read;                                   !DO A READ FUNCTION
;          5841
;          5842  if .GO IS_NOT_SET                              !SEE IF GO GOT SET
;          5843  then
;          5844      begin                                       !ERROR IF CLEAR
;          5845      ERRDF (34, ASYNC, DUMPER);
;          5846      PRINTB (FIV_FMT, WRD_1, PHR_1, WRD_12, FNC_6, WRD_19);
;          5847
;          5848      if .DRY IS_NOT_SET                          !TST DRY SET WITH GO CLEAR
;          5849      then
;          5850          begin                                   !ERROR IF NOT SET
;          5851          ERRDF (35, ASYNC, DUMPER);
;          5852          PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_43, WRD_1, PHR_6);
;          5853          end;
;          5854
;          5855      DODU_FLG = ONE;                             !SET DODU_FLG
;          5856      end
;          5857  else                                            !GO BIT GOT SET
;          5858
;          5859      if .DRY IS_SET                              !SEE IF DRY IS CLEAR
;          5860      then
;          5861          begin                                   !ERROR IF SET
```

```
11484 ;ML4AD
11485 ;                 TEST CODE SECTION
11486 ;
11487 ;          5862           ERRDF (36, ASYNC, DUMPER);
11488 ;          5863           PRINTB (FIV_FMT, WRD_2, PHR_2, WRD_43, WRD_1, PHR_6);
11489 ;          5864           DODU_FLG = ONE;
11490 ;          5865           end;
11491 ;          5866
11492 ;          5867   if .ILF IS_SET                                  !DID FUNCTION CAUSE ILF
11493 ;          5868   then
11494 ;          5869       begin                                       !ERROR IF YES
11495 ;          5870       ERRDF (37, ASYNC, DUMPER);
11496 ;          5871       PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_12, FNC_6, WRD_19);
11497 ;          5872       end;
11498 ;          5873
11499 ;          5874   if .OPI IS_SET                                  !DID FUNCTION CAUSE OPI
11500 ;          5875   then
11501 ;          5876       begin                                       !ERROR IF YES
11502 ;          5877       ERRDF (38, ASYNC, DUMPER);
11503 ;          5878       PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_6, WRD_19);
11504 ;          5879       end;
11505 ;          5880
11506 ;          5881   DELAY (FRTY_US);                                !WAIT FOR XFERR TO COMPLETE
11507 ;          5882
11508 ;          5883   if .DRY IS_NOT_SET                              !IS DRY SET AFTER XFERR
11509 ;          5884   then
11510 ;          5885       begin
11511 ;          5886
11512 ;          5887       if .GO IS_SET                               !TEST GO CLEAR WITH DRY NOT SET
11513 ;          5888       then
11514 ;          5889           begin
11515 ;          5890           CLR_MBUS;                               !CLEAR GO
11516 ;          5891
11517 ;          5892           if .GO IS_SET then ERRDF (39, ASYNC, DUMPER) else ERRDF (40, SYNC, DUMPER);
11518 ;          5893
11519 ;          5894                                                   !TST GO TO DETERMINE FAILING MOD
11520 ;          5895           PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_6, WRD_19);
11521 ;          5896           end
11522 ;          5897       else                                        !GO BIT CLEAR WITH DRY NOT SET
11523 ;          5898           begin                                   !REPORT ERROR
11524 ;          5899           ERRDF (41, ASYNC, DUMPER);
11525 ;          5900           PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_11, FNC_6, WRD_19);
11526 ;          5901           end;
11527 ;          5902
11528 ;          5903       DODU_FLG = ONE;                             !SET DODU_FLG
11529 ;          5904       end;
11530 ;          5905
11531 ;          5906   if .GO IS_SET                                   !SEE IF GO CLEAR AFTER XFERR
11532 ;          5907   then
11533 ;          5908       begin
11534 ;          5909       CLR_MBUS;                                   !CLEAR GO IF STILL SET
11535 ;          5910
11536 ;          5911       if .GO IS_SET then ERRDF (42, ASYNC, DUMPER) else ERRDF (43, SYNC, DUMPER);
11537 ;          5912
11538 ;          5913                                                   !TST GO TO DETERMINE FAILING MOD
```

```
 11540 ;ML4AD
 11541 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 11542 ;                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (67)
 11543 ;          5914       PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_6, WRD_19);
 11544 ;          5915       DODU_FLG = ONE;
 11545 ;          5916       end;
 11546 ;          5917
 11547 ;          5918   ENDSUB;
 11548 ;          5919
 11549 ;          5920   if .TRE IS_SET                                      !SEE IF XFERR CAUSED A TRANSFER ERROR
 11550 ;          5921   then
 11551 ;          5922       begin                                           !REPORT ERROR IF SET AND CONTINUE TESTING
 11552 ;          5923       ERRDF (116, INTER, DUMPER);
 11553 ;          5924       PRINTB (SIX_FMT, WRD_61, WRD_20, PHR_5, WRD_12, FNC_6, WRD_19);
 11554 ;          5925       end;
 11555 ;          5926
 11556 ;          5927   if .DODU_FLG IS_SET                                 !DROP THIS UNIT IF DODU_FLG SET
 11557 ;          5928   then
 11558 ;          5929       begin
 11559 ;          5930       DODU (.ML_LUN);
 11560 ;          5931       DOCLN;
 11561 ;          5932       end;
 11562 ;          5933
 11563 ;          5934   ENDTST;
 11567
 11568                                          .SBTTL  $T15 TEST CODE SECTION
 11572 041476  004167  143036          $T15:   JSR     R1,$SAVE2                       ;                                5808
 11573 041502  005746                          TST     -(SP)
 11574 041504  104402                  1$:     TRAP    2
 11575 041506  152777  000040  153730          BISB    #40,@ML.REG+40                  ;                                5834
 11576 041514  016701  154312                  MOV     ML.DUT,R1                       ;                                5836
 11577 041520  042701  177770                  BIC     #177770,R1
 11578 041524  142777  000007  153712          BICB    #7,@ML.REG+40
 11579 041532  150177  153706                  BISB    R1,@ML.REG+40
 11580 041536  005002                          CLR     R2
 11581 041540  004767  156516                  JSR     PC,FIRST.BLK.XFER               ; DODU.FLG                       5838
 11582 041544  012777  000071  153632          MOV     #71,@ML.REG                     ;                                5839
 11583 041552  132777  000001  153624          BITB    #1,@ML.REG                      ;                                5840
 11584 041560  001057                          BNE     2$                              ;                                5842
 11585 041562  104455                          TRAP    55
 11586 041564  000042                          .WORD   42                              ;                                5845
 11587 041566  012706                          .WORD   ASYNC
 11588 041570  026302                          .WORD   DUMPER
 11589 041572  012746  010730                  MOV     #WRD.19,-(SP)
 11590 041576  012746  012226                  MOV     #FNC.6,-(SP)                    ;                                5846
 11591 041602  012746  010650                  MOV     #WRD.12,-(SP)
 11592 041606  012746  011610                  MOV     #PHR.1,-(SP)
 11593 041612  012746  010520                  MOV     #WRD.1,-(SP)
 11594 041616  012746  010304                  MOV     #FIV.FMT,-(SP)
```

```
11596                                   ;ML4AD
11597                                   ;               TEST CODE SECTION
11598
11599 041622  012746  000006                    MOV     #6,-(SP)
11600 041626  010600                            MOV     SP,R0           ; SP,*
11601 041630  104414                            TRAP    14
11602 041632  132777  000200  153614            BITB    #200,@ML.REG+50 ;               5848
11603 041640  001056                            BNE     3$              ;               5851
11604 041642  104455                            TRAP    55
11605 041644  000043                            .WORD   43
11606 041646  012706                            .WORD   ASYNC
11607 041650  026302                            .WORD   DUMPER
11608 041652  012746  011726                    MOV     #PHR.6,-(SP)    ;               5852
11609 041656  012746  010520                    MOV     #WRD.1,-(SP)
11610 041662  012746  011176                    MOV     #WRD.43,-(SP)
11611 041666  012746  011610                    MOV     #PHR.1,-(SP)
11612 041672  012746  010524                    MOV     #WRD.2,-(SP)
11613 041676  012746  010304                    MOV     #FIV.FMT,-(SP)
11614 041702  012746  000006                    MOV     #6,-(SP)
11615 041706  010600                            MOV     SP,R0           ; SP,*
11616 041710  104414                            TRAP    14
11617 041712  062706  000016                    ADD     #16,SP          ;               5850
11618 041716  000427                            BR      3$              ;               5855
11619 041720  105777  153530            2$:     TSTB    @ML.REG+50      ;               5859
11620 041724  100030                            BPL     4$
11621 041726  104455                            TRAP    55              ;               5862
11622 041730  000044                            .WORD   44
11623 041732  012706                            .WORD   ASYNC
11624 041734  026302                            .WORD   DUMPER
11625 041736  012746  011726                    MOV     #PHR.6,-(SP)    ;               5863
11626 041742  012746  010520                    MOV     #WRD.1,-(SP)
11627 041746  012746  011176                    MOV     #WRD.43,-(SP)
11628 041752  012746  011626                    MOV     #PHR.2,-(SP)
11629 041756  012746  010524                    MOV     #WRD.2,-(SP)
11630 041762  012746  010304                    MOV     #FIV.FMT,-(SP)
11631 041766  012746  000006                    MOV     #6,-(SP)
11632 041772  010600                            MOV     SP,R0           ; SP,*
11633 041774  104414                            TRAP    14
11634 041776  012702  000001            3$:     MOV     #1,R2           ; *,DODU.FLG    5864
11635 042002  062706  000016                    ADD     #16,SP          ;               5861
11636 042006  132777  000001  153450    4$:     BITB    #1,@ML.REG+60   ;               5867
11637 042014  001426                            BEQ     5$
11638 042016  104455                            TRAP    55              ;               5870
11639 042020  000045                            .WORD   45
11640 042022  012706                            .WORD   ASYNC
11641 042024  026302                            .WORD   DUMPER
11642 042026  012746  010730                    MOV     #WRD.19,-(SP)   ;               5871
11643 042032  012746  012226                    MOV     #FNC.6,-(SP)
11644 042036  012746  010650                    MOV     #WRD.12,-(SP)
11645 042042  012746  011714                    MOV     #PHR.5,-(SP)
11646 042046  012746  010536                    MOV     #WRD.3,-(SP)
11647 042052  012746  010304                    MOV     #FIV.FMT,-(SP)
11648 042056  012746  000006                    MOV     #6,-(SP)
11649 042062  010600                            MOV     SP,R0           ; SP,*
11650 042064  104414                            TRAP    14
```

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS
                                                                                                         29-Mar-1982 16:21:03    PA:<
11652
11653
11654
11655 042066  062706  000016                    ADD     #16,SP
11656 042072  032777  020000  153364  5$:       BIT     #20000,@ML.REG+60       ;                                               5869
11657 042100  001426                            BEQ     6$                      ;                                               5874
11658 042102  104455                            TRAP    55
11659 042104  000046                            .WORD   46                      ;                                               5877
11660 042106  012706                            .WORD   ASYNC
11661 042110  026302                            .WORD   DUMPER
11662 042112  012746  010730                    MOV     #WRD.19,-(SP)           ;                                               5878
11663 042116  012746  012226                    MOV     #FNC.6,-(SP)
11664 042122  012746  010650                    MOV     #WRD.12,-(SP)
11665 042126  012746  011714                    MOV     #PHR.5,-(SP)
11666 042132  012746  010544                    MOV     #WRD.4,-(SP)
11667 042136  012746  010304                    MOV     #FIV.FMT,-(SP)
11668 042142  012746  000006                    MOV     #6,-(SP)
11669 042146  010600                            MOV     SP,R0                   ; SP,*
11670 042150  104414                            TRAP    14
11671 042152  062706  000016                    ADD     #16,SP                  ;
11672 042156  012700  000050            6$:     MOV     #50,R0                  ; *,$$TMP2                                      5876
11673 042162  001410            7$:     BEQ     10$                             ; *,$$TMP1                                      5881
11674 042164  016701  137726                    MOV     L$DLY,R1
11675 042170  001403                            BEQ     9$
11676 042172  005016            8$:     CLR     (SP)                    ; $$TMP
11677 042174  005301                            DEC     R1                      ; $$TMP1
11678 042176  001375                            BNE     8$
11679 042200  005300            9$:     DEC     R0                      ; $$TMP2
11680 042202  000767                            BR      7$
11681 042204  132777  000200  153242  10$:      BITB    #200,@ML.REG+50         ;                                               5883
11682 042212  001106                            BNE     15$
11683 042214  132777  000001  153162            BITB    #1,@ML.REG              ;                                               5887
11684 042222  001452                            BEQ     13$
11685 042224  152777  000040  153212            BISB    #40,@ML.REG+40          ;                                               5889
11686 042232  016701  153574                    MOV     ML.DUT,R1
11687 042236  042701  177770                    BIC     #177770,R1
11688 042242  142777  000007  153174            BICB    #7,@ML.REG+40
11689 042250  150177  153170                    BISB    R1,@ML.REG+40
11690 042254  132777  000001  153122            BITB    #1,@ML.REG              ;                                               5892
11691 042262  001405                            BEQ     11$
11692 042264  104455                            TRAP    55
11693 042266  000047                            .WORD   47
11694 042270  012706                            .WORD   ASYNC
11695 042272  026302                            .WORD   DUMPER
11696 042274  000404                            BR      12$
11697 042276  104455            11$:    TRAP    55
11698 042300  000050                            .WORD   50
11699 042302  012750                            .WORD   SYNC
11700 042304  026302                            .WORD   DUMPER
11701 042306  012746  010730            12$:    MOV     #WRD.19,-(SP)           ;                                               5895
11702 042312  012746  012226                    MOV     #FNC.6,-(SP)
11703 042316  012746  010640                    MOV     #WRD.11,-(SP)
11704 042322  012746  011626                    MOV     #PHR.2,-(SP)
11705 042326  012746  010520                    MOV     #WRD.1,-(SP)
11706 042332  012746  010304                    MOV     #FIV.FMT,-(SP)
```

```
11708                                   ;ML4AD
11709                                   ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS
11710                                                                                                      29-Mar-1982 16:21:03    PA:<
11711 042336  012746  000006                    MOV     #6,-(SP)
11712 042342  010600                            MOV     SP,R0                   ; SP,*
11713 042344  104414                            TRAP    14
11714 042346  000424                            BR      14$                     ;
11715 042350  104455                    13$:    TRAP    55                      ;                                                                  5887
11716 042352  000051                            .WORD   51                      ;                                                                  5899
11717 042354  012706                            .WORD   ASYNC
11718 042356  026302                            .WORD   DUMPER
11719 042360  012746  010730                    MOV     #WRD.19,-(SP)           ;                                                                  5900
11720 042364  012746  012226                    MOV     #FNC.6,-(SP)
11721 042370  012746  010640                    MOV     #WRD.11,-(SP)
11722 042374  012746  011610                    MOV     #PHR.1,-(SP)
11723 042400  012746  010524                    MOV     #WRD.2,-(SP)
11724 042404  012746  010304                    MOV     #FIV.FMT,-(SP)
11725 042410  012746  000006                    MOV     #6,-(SP)
11726 042414  010600                            MOV     SP,R0                   ; SP,*
11727 042416  104414                            TRAP    14
11728 042420  012702  000001            14$:    MOV     #1,R2                   ; *,DODU.FLG                                                       5903
11729 042424  062706  000016                    ADD     #16,SP                  ;                                                                  5885
11730 042430  132777  000001  152746    15$:    BITB    #1,@ML.REG              ;                                                                  5906
11731 042436  001455                            BEQ     18$
11732 042440  152777  000040  152776            BISB    #40,@ML.REG+40          ;                                                                  5908
11733 042446  016701  153360                    MOV     ML.DUT,R1
11734 042452  042701  177770                    BIC     #177770,R1
11735 042456  142777  000007  152760            BICB    #7,@ML.REG+40
11736 042464  150177  152754                    BISB    R1,@ML.REG+40
11737 042470  132777  000001  152706            BITB    #1,@ML.REG              ;                                                                  5911
11738 042476  001405                            BEQ     16$
11739 042500  104455                            TRAP    55
11740 042502  000052                            .WORD   52
11741 042504  012706                            .WORD   ASYNC
11742 042506  026302                            .WORD   DUMPER
11743 042510  000404                            BR      17$
11744 042512  104455                    16$:    TRAP    55
11745 042514  000053                            .WORD   53
11746 042516  012750                            .WORD   SYNC
11747 042520  026302                            .WORD   DUMPER
11748 042522  012746  010730            17$:    MOV     #WRD.19,-(SP)           ;                                                                  5914
11749 042526  012746  012226                    MOV     #FNC.6,-(SP)
11750 042532  012746  010640                    MOV     #WRD.11,-(SP)
11751 042536  012746  011626                    MOV     #PHR.2,-(SP)
11752 042542  012746  010520                    MOV     #WRD.1,-(SP)
11753 042546  012746  010304                    MOV     #FIV.FMT,-(SP)
11754 042552  012746  000006                    MOV     #6,-(SP)
11755 042556  010600                            MOV     SP,R0                   ; SP,*
11756 042560  104414                            TRAP    14
11757 042562  012702  000001                    MOV     #1,R2                   ; *,DODU.FLG                                                       5915
11758 042566  062706  000016                    ADD     #16,SP                  ;                                                                  5908
11759 042572  104467                    18$:    TRAP    67                      ;                                                                  5916
11760 042574  006000                            ROR     R0
11761 042576  103002                            BHIS    19$
11762 042600  000167  176700                    JMP     1$
```

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
                                                                                                            29-Mar-1982 16:21:03    PA:<
11767 042604  032777  040000  152572  19$:    BIT     #40000,@ML.REG                  ;                                           5920
11768 042612  001430                          BEQ     20$
11769 042614  104455                          TRAP    55                              ;                                           5923
11770 042616  000164                          .WORD   164
11771 042620  013114                          .WORD   INTER
11772 042622  026302                          .WORD   DUMPER
11773 042624  012746  010730                  MOV     #WRD.19,-(SP)                   ;                                           5924
11774 042630  012746  012226                  MOV     #FNC.6,-(SP)
11775 042634  012746  010650                  MOV     #WRD.12,-(SP)
11776 042640  012746  011714                  MOV     #PHR.5,-(SP)
11777 042644  012746  010736                  MOV     #WRD.20,-(SP)
11778 042650  012746  011416                  MOV     #WRD.61,-(SP)
11779 042654  012746  010322                  MOV     #SIX.FMT,-(SP)
11780 042660  012746  000007                  MOV     #7,-(SP)
11781 042664  010600                          MOV     SP,R0                           ; SP,*
11782 042666  104414                          TRAP    14
11783 042670  062706  000020                  ADD     #20,SP                          ;                                           5922
11784 042674  005302                  20$:    DEC     R2                              ; DODU.FLG                                  5927
11785 042676  001004                          BNE     21$
11786 042700  016700  153124                  MOV     ML.LUN,R0                       ;                                           5930
11787 042704  104451                          TRAP    51
11788 042706  104444                          TRAP    44
11789 042710  005726                  21$:    TST     (SP)+                           ;                                           5808
11790 042712  000207                          RTS     PC
11791
11792                                   ; Routine Size:  327 words
11793                                   ; Maximum stack depth per invocation:  18 words
11798
11799
11803
11804                                           .SBTTL  T15 TEST CODE SECTION
11808 042714                          T15::
11809 042714  004767  176556          1$:     JSR     PC,$T15                         ;                                           5932
11810 042720  104466                          TRAP    66
11811 042722  006000                          ROR     R0
11812 042724  103773                          BLO     1$
11813 042726  000207                          RTS     PC
11814
11815                                   ; Routine Size:  6 words
11816                                   ; Maximum stack depth per invocation:  0 words
```

SEQ 0263

```
11825
11826
11827 ;          5935  !<BLF/PAGE>
```

```
11829 ;ML4AD
11830 ;                 TEST CODE SECTION
11831 ;
11832 ;          5936  !
11833 ;          5937  BGNTS ;
11834 ;          5938
11835 ;          5939  !++
11836 ;          5940  ! TEST NUMBER:  TS1 16
11837 ;          5941  !
11838 ;          5942  ! TEST NAME:    CLEAR FUNCTION TEST
11839 ;          5943  !
11840 ;          5944  ! TEST DESCRIPTION:
11841 ;          5945  !       TEST IF THE DRIVE CAN PERFORM A CLEAR FUNCTION WITHOUT HANGING THE DRIVE.
11842 ;          5946  !
11843 ;          5947  !       A CLEAR FUNCTION IS WRITTEN INTO MLCS1.
11844 ;          5948  !
11845 ;          5949  !       THEN GO AND ERROR BITS ARE CHECKED FOR CORRECT STATUS.
11846 ;          5950  !       THIS DRIVE IS DROPPED ON DETECTED ERRORS.
11847 ;          5951  !
11848 ;          5952  !--
11849 ;          5953
11850 ;          5954  BGNSUB;
11851 ;          5955  CLR_MBUS;
11852 ;          5956  MLER = ONES;                                  !SET BITS IN ERROR REGISTER
11853 ;          5957  MLCS1 = DRV_CLR;                              !DO A CLEAR FUNCTION
11854 ;          5958  DELAY (ONE_OS);                               !DELAY
11855 ;          5959
11856 ;          5960  if .GO IS_SET                                 !SEE IF GO CLEARED AFTER FUNCTION
11857 ;          5961  then
11858 ;          5962      begin                                     !ERROR IF SET
11859 ;          5963      ERRDF (44, ASYNC, DUMPER);
11860 ;          5964      PRINTB (FIV_FMT, WRD_1, PHR_5, WRD_11, FNC_3, FNC_7, WRD_19);
11861 ;          5965
11862 ;          5966      if .DRY IS_SET                            !TST DRY CLEAR WITH GO SET
11863 ;          5967      then
11864 ;          5968          begin                                 !ERROR IF SET
11865 ;          5969          ERRDF (45, ASYNC, DUMPER);
11866 ;          5970          PRINTB (FIV_FMT, WRD_2, PHR_5, WRD_43, WRD_1, PHR_5);
11867 ;          5971          end;
11868 ;          5972
11869 ;          5973      end
11870 ;          5974  else                                          !GO CLEARED AFTER FUNCTION
11871 ;          5975
11872 ;          5976      if .DRY IS_NOT_SET                        !TST DRY SET WITH GO CLEAR
11873 ;          5977      then
11874 ;          5978          begin                                 !ERROR IF NOT SET
11875 ;          5979          ERRDF (46, ASYNC, DUMPER);
11876 ;          5980          PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_43, WRD_1, PHR_6);
11877 ;          5981          end;
11878 ;          5982
11879 ;          5983  if .ILF IS_SET                                !DID FUNCTION CAUSE ILF
11880 ;          5984  then
11881 ;          5985      begin                                     !ERROR IF YES
11882 ;          5986      ERRDF (47, ASYNC, DUMPER);
11883 ;          5987      PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_12, FNC_3, FNC_7, WRD_19);
```

```
11885 ;ML4AD
11886 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
11887                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (68)
11888 ;          5988       end;
11889 ;          5989
11890 ;          5990   if .OPI IS_SET                                      !DID FUNCTION CAUSE OPI
11891 ;          5991   then
11892 ;          5992       begin                                           !ERROR IF YES
11893 ;          5993       ERRDF (48, ASYNC, DUMPER);
11894 ;          5994       PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_3, FNC_7, WRD_19);
11895 ;          5995       end;
11896 ;          5996
11897 ;          5997   if .MLER neq ZERO                                   !TEST ERROR REGISTER FOR CLEAR
11898 ;          5998   then
11899 ;          5999       begin                                           !ERROR IF NOT CLEAR
11900 ;          6000       ERRDF (49, ASYNC, DUMPER);
11901 ;          6001       PRINTB (SIX_FMT, FNC_3, FNC_7, WRD_19, WRD_14, WRD_13, REG_3);
11902 ;          6002       end;
11903 ;          6003
11904 ;          6004   ENDSUB;
11905 ;          6005   ENDTST;
11909
11910                                          .SBTTL  $T16 TEST CODE SECTION
11914 042730  010146                 $T16:    MOV     R1,-(SP)                ;                                        5934
11915 042732  005746                          TST     -(SP)
11916 042734  104402                 1$:      TRAP    2                       ;                                        5937
11917 042736  152777  000040  152500          BISB    #40,@ML.REG+40          ;                                        5954
11918 042744  016701  153062                  MOV     ML.DUT,R1
11919 042750  042701  177770                  BIC     #177770,R1
11920 042754  142777  000007  152462          BICB    #7,@ML.REG+40
11921 042762  150177  152456                  BISB    R1,@ML.REG+40
11922 042766  012777  177777  152470          MOV     #-1,@ML.REG+60          ;                                        5956
11923 042774  012777  000011  152402          MOV     #11,@ML.REG             ;                                        5957
11924 043002  012700  000001                  MOV     #1,R0                   ; *,$$TMP2                               5958
11925 043006  001410                 2$:      BEQ     5$
11926 043010  016701  137102                  MOV     L$DLY,R1                ; *,$$TMP1
11927 043014  001403                          BEQ     4$
11928 043016  005016                 3$:      CLR     (SP)                    ; $$TMP
11929 043020  005301                          DEC     R1                      ; $$TMP1
11930 043022  001375                          BNE     3$
11931 043024  005300                 4$:      DEC     R0                      ; $$TMP2
11932 043026  000767                          BR      2$
11933 043030  132777  000001  152346 5$:      BITB    #1,@ML.REG              ;                                        5960
11934 043036  001462                          BEQ     7$
11935 043040  104455                          TRAP    55                      ;                                        5963
11936 043042  000054                          .WORD   54
11937 043044  012706                          .WORD   ASYNC
11938 043046  026302                          .WORD   DUMPER
11939 043050  012746  010730                 MOV     #WRD.19,-(SP)           ;                                        5964
```

SEQ 0266


```
11941                                   ;ML4AD
11942                                   ;               TEST CODE SECTION
11943
11944 043054  012746  012234                      MOV     #FNC.7,-(SP)
11945 043060  012746  012172                      MOV     #FNC.3,-(SP)
11946 043064  012746  010640                      MOV     #WRD.11,-(SP)
11947 043070  012746  011714                      MOV     #PHR.5,-(SP)
11948 043074  012746  010520                      MOV     #WRD.1,-(SP)
11949 043100  012746  010304                      MOV     #FIV.FMT,-(SP)
11950 043104  012746  000007                      MOV     #7,-(SP)
11951 043110  010600                              MOV     SP,R0                   ; SP,*
11952 043112  104414                              TRAP    14
11953 043114  105777  152334                      TSTB    @ML.REG+50              ;                       5966
11954 043120  100026                              BPL     6$
11955 043122  104455                              TRAP    55                      ;                       5969
11956 043124  000055                              .WORD   55
11957 043126  012706                              .WORD   ASYNC
11958 043130  026302                              .WORD   DUMPER
11959 043132  012746  011714                      MOV     #PHR.5,-(SP)            ;                       5970
11960 043136  012746  010520                      MOV     #WRD.1,-(SP)
11961 043142  012746  011176                      MOV     #WRD.43,-(SP)
11962 043146  012746  011714                      MOV     #PHR.5,-(SP)
11963 043152  012746  010524                      MOV     #WRD.2,-(SP)
11964 043156  012746  010304                      MOV     #FIV.FMT,-(SP)
11965 043162  012746  000006                      MOV     #6,-(SP)
11966 043166  010600                              MOV     SP,R0                   ; SP,*
11967 043170  104414                              TRAP    14
11968 043172  062706  000016                      ADD     #16,SP                  ;                       5968
11969 043176  062706  000020              6$:     ADD     #20,SP                  ;                       5962
11970 043202  000432                              BR      8$                      ;                       5960
11971 043204  132777  000200  152242    7$:     BITB    #200,@ML.REG+50         ;                       5976
11972 043212  001026                              BNE     8$
11973 043214  104455                              TRAP    55                      ;                       5979
11974 043216  000056                              .WORD   56
11975 043220  012706                              .WORD   ASYNC
11976 043222  026302                              .WORD   DUMPER
11977 043224  012746  011725                      MOV     #PHR.6,-(SP)            ;                       5980
11978 043230  012746  010520                      MOV     #WRD.1,-(SP)
11979 043234  012746  011176                      MOV     #WRD.43,-(SP)
11980 043240  012746  011610                      MOV     #PHR.1,-(SP)
11981 043244  012746  010524                      MOV     #WRD.2,-(SP)
11982 043250  012746  010304                      MOV     #FIV.FMT,-(SP)
11983 043254  012746  000006                      MOV     #6,-(SP)
11984 043260  010600                              MOV     SP,R0                   ; SP,*
11985 043262  104414                              TRAP    14
11986 043264  062706  000016                      ADD     #16,SP                  ;                       5978
11987 043270  132777  000001  152166    8$:     BITB    #1,@ML.REG+60           ;                       5983
11988 043276  001430                              BEQ     9$
11989 043300  104455                              TRAP    55                      ;                       5986
11990 043302  000057                              .WORD   57
11991 043304  012706                              .WORD   ASYNC
11992 043306  026302                              .WORD   DUMPER
11993 043310  012746  010730                      MOV     #WRD.19,-(SP)           ;                       5987
11994 043314  012746  012234                      MOV     #FNC.7,-(SP)
11995 043320  012746  012172                      MOV     #FNC.3,-(SP)
```

```
                                      ;ML4AD                                                   29-Mar-1982 16:23:04   TOPS
                                      ;                TEST CODE SECTION                       29-Mar-1982 16:21:03   PA:<
043324 012746 010650                      MOV     #WRD.12,-(SP)
043330 012746 011714                      MOV     #PHR.5,-(SP)
043334 012746 010536                      MOV     #WRD.3,-(SP)
043340 012746 010304                      MOV     #FIV.FMT,-(SP)
043344 012746 000007                      MOV     #7,-(SP)
043350 010600                             MOV     SP,R0                     ; SP,*
043352 104414                             TRAP    14
043354 062706 000020                      ADD     #20,SP                    ;                      5985
043360 032777 020000 152076   9$:         BIT     #20000,@ML.REG+60         ;                      5990
043366 001430                             BEQ     10$
043370 104455                             TRAP    55                        ;                      5993
043372 000060                             .WORD   60
043374 012706                             .WORD   ASYNC
043376 026302                             .WORD   DUMPER
043400 012746 010730                      MOV     #WRD.19,-(SP)             ;                      5994
043404 012746 012234                      MOV     #FNC.7,-(SP)
043410 012746 012172                      MOV     #FNC.3,-(SP)
043414 012746 010650                      MOV     #WRD.12,-(SP)
043420 012746 011714                      MOV     #PHR.5,-(SP)
043424 012746 010544                      MOV     #WRD.4,-(SP)
043430 012746 010304                      MOV     #FIV.FMT,-(SP)
043434 012746 000007                      MOV     #7,-(SP)
043440 010600                             MOV     SP,R0                     ; SP,*
043442 104414                             TRAP    14
043444 062706 000020                      ADD     #20,SP                    ;                      5992
043450 005777 152010          10$:        TST     @ML.REG+60                ;                      5997
043454 001430                             BEQ     11$
043456 104455                             TRAP    55                        ;                      6000
043460 000061                             .WORD   61
043462 012706                             .WORD   ASYNC
043464 026302                             .WORD   DUMPER
043466 012746 012516                      MOV     #REG.3,-(SP)              ;                      6001
043472 012746 010660                      MOV     #WRD.13,-(SP)
043476 012746 010664                      MOV     #WRD.14,-(SP)
043502 012746 010730                      MOV     #WRD.19,-(SP)
043506 012746 012234                      MOV     #FNC.7,-(SP)
043512 012746 012172                      MOV     #FNC.3,-(SP)
043516 012746 010322                      MOV     #SIX.FMT,-(SP)
043522 012746 000007                      MOV     #7,-(SP)
043526 010600                             MOV     SP,R0                     ; SP,*
043530 104414                             TRAP    14
043532 062706 000020                      ADD     #20,SP                    ;                      5999
043536 104467                 11$:        TRAP    67                        ;                      6002
043540 006000                             ROR     R0
043542 103002                             BHIS    12$
043544 000167 177164                      JMP     1$
043550 005726                 12$:        TST     (SP)+                     ;                      5934
043552 012601                             MOV     (SP)+,R1
043554 000207                             RTS     PC

                                      ; Routine Size:  203 words
                                      ; Maximum stack depth per invocation:  17 words
```

SEQ 0268

```
12053                                   ;ML4AD
12054                                   ;               TEST CODE SECTION
12055
12060
12061
12065
12066                                           .SBTTL  T16 TEST CODE SECTION
12070 043556                            T16::
12071 043556  004767  177146            1$:     JSR     PC,$T16                 ;
12072 043562  104466                            TRAP    66
12073 043564  006000                            ROR     R0
12074 043566  103773                            BLO     1$
12075 043570  000207                            RTS     PC
12076
12077                                   ; Routine Size:  6 words
12078                                   ; Maximum stack depth per invocation:  0 words
12083
12084
12085 ;          6006  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

6004

```
12087 ;ML4AD
12088 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
12089 ;                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (69)
12090 ;          6007   !
12091 ;          6008   BGNTST;
12092 ;          6009
12093 ;          6010   !++
12094 ;          6011   ! TEST NUMBER:  TST 17
12095 ;          6012   !
12096 ;          6013   ! TEST NAME: DIAGNOSTIC REGISTER TESTS
12097 ;          6014   !
12098 ;          6015   ! TEST DESCRIPTION:
12099 ;          6016   !       TEST THE DATA DIAG REGISTERS MLD1, MLD2, MLE2
12100 ;          6017   !       FOR 1'S/0'S, SHIFTING 1'S/0'S AND INITIALIZATION
12101 ;          6018   !
12102 ;          6019   !--
12103 ;          6020
12104 ;          6021   local
12105 ;          6022       CLR_DATA,                                       !CLEAR DATA FOR INIT TEST
12106 ;          6023       SAVE,                                           !TEMPORARY SAVE LOCATION
12107 ;          6024       ERR_FLG,                                        !ERROR FLAG
12108 ;          6025       TST_PAT,                                        !TEST PATTERN
12109 ;          6026       index,                                          !POINTS TO REGISTER PRESENTLY BEING TESTED
12110 ;          6027       DODU_FLG;                                       !DROP UNIT FLAG
12111 ;          6028
12112 ;          6029   DODU_FLG = ZERO;
12113 ;          6030   !
12114 ;          6031   !
12115 ;          6032   !       FIRST TEST THE REGISTERS FOR ONES AND ZEROES
12116 ;          6033   !
12117 ;          6034   !
12118 ;          6035   TST_PAT = ONES;                                     !LOAD TEST PAT WITH ONES
12119 ;          6036   CLR_THRESHOLD;                                      !CLEAR ERROR PRINT THRESHOLD
12120 ;          6037
12121 ;          6038   incr TWICE from 0 to 1 do                           !REPEAT LOOP TWICE
12122 ;          6039       begin
12123 ;          6040
12124 ;          6041       incr REG_SEL from 11 to 13 do                   !TEST ELEVEN WRITE/READ REGISTERS
12125 ;          6042           begin
12126 ;          6043           BGNSUB;
12127 ;          6044           CLR_MBUS;
12128 ;          6045           WRT_REG (.TST_PAT, .REG_SEL, index);        !WRITE TO THE REGISTER
12129 ;          6046           RD_REG (.TST_PAT, .REG_SEL, ERR_FLG);       !READ THE REGISTER
12130 ;          6047
12131 ;          6048           if .ERR_FLG IS_SET                          !SEE IF READ FOUND AN ERROR
12132 ;          6049           then
12133 ;          6050               begin                                   !IF ERROR FLAG IS_SET THEN ERROR AND SET DODU_FLG
12134 ;          6051               CMP_THRESHOLD;                          !COMPARE ERROR PRINT THRESHOLD
12135 ;          6052               ERRDF (121, ARR_DAT, DUMPER);           !ARRAY DATA MODULE FAILURE
12136 ;          6053               PRINTB (SIX_FMT, PHR_4, WRD_12, FNC_5, FNC_6, WRD_52, WRD_56);
12137 ;          6054               PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .WT_DATA, .RD_DATA);
12138 ;          6055               DODU_FLG = ONE;
12139 ;          6056               end;
12140 ;          6057
12141 ;          6058           ENDSUB;
```

```
12143 ;ML4AD
12144 ;                 TEST CODE SECTION
12145
12146 ;          6059          end;
12147 ;          6060
12148 ;          6061      TST_PAT = not .TST_PAT;                 !REPEAT AGAIN WITH COMPLIMENT DATA
12149 ;          6062      end;
12150 ;          6063
12151 ;          6064  !
12152 ;          6065  !
12153 ;          6066  !       TEST THE REGISTERS FOR SHIFTING ONES AND ZEROES
12154 ;          6067  !
12155 ;          6068  !
12156 ;          6069  TST_PAT = ONE;                               !LOAD TST_PAT WITH A 1 IN A FILED OF 0'S
12157 ;          6070
12158 ;          6071  incr SHIFT from 0 to 15 do                   !DO SHIFT 16 TIMES
12159 ;          6072      begin
12160 ;          6073
12161 ;          6074      incr TWICE from 0 to 1 do               !REPEAT LOOP TWICE
12162 ;          6075          begin
12163 ;          6076
12164 ;          6077          incr REG_SEL from 11 to 13 do       !TEST ELEVEN READ/WRITE REGISTERS
12165 ;          6078              begin
12166 ;          6079              BGNSUB;
12167 ;          6080              CLR_MBUS;
12168 ;          6081              WRT_REG (.TST_PAT, .REG_SEL, index);        !WRITE TO THE REGISTER
12169 ;          6082              RD_REG (.TST_PAT, .REG_SEL, ERR_FLG);       !READ THE REGISTER
12170 ;          6083
12171 ;          6084              if .ERR_FLG IS_SET                  !SEE IF THE READ FOUND AN ERROR
12172 ;          6085              then
12173 ;          6086                  begin                           !IF THE ERROR FLAG IS SET THEN ERROR
12174 ;          6087                  CMP_THRESHOLD;                  !COMPARE ERROR PRINT THRESHOLD
12175 ;          6088                  ERRDF (122, ARR_DAT, DUMPER);   !ARRAY DATA MODULE FAILURE
12176 ;          6089                  PRINTB (SIX_FMT, PHR_4, WRD_12, FNC_5, FNC_6, WRD_52, WRD_56);
12177 ;          6090                  PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .WT_DATA, .RD_DATA);
12178 ;          6091                  DODU_FLG = ONE;
12179 ;          6092                  end;
12180 ;          6093
12181 ;          6094              ENDSUB;
12182 ;          6095              end;
12183 ;          6096
12184 ;          6097          TST_PAT = not .TST_PAT;             !REPEAT WITH A 0 IN A FIELD OF 1'S
12185 ;          6098          end;
12186 ;          6099
12187 ;          6100      TST_PAT = .TST_PAT^ONE;                 !SHIFT THE 1 IN THE FIELD OF 0'S
12188 ;          6101      end;
12189 ;          6102
12190 ;          6103  !
12191 ;          6104  !
12192 ;          6105  !       NOW TEST THE REGISTERS FOR INITIALIZATION
12193 ;          6106  !
12194 ;          6107  !
12195 ;          6108  REG_INIT_FLG = ONE;
12196 ;          6109  TST_PAT = ONES;                               !BACKGROUND PATTERN
12197 ;          6110
```

```
 12199 ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 12200 ;                 TEST CODE SECTION                             29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (69)
 12201
 12202 ;          6111   incr TWICE from 0 to 1 do                     !REPEAT LOOP TWICE
 12203 ;          6112       begin
 12204 ;          6113
 12205 ;          6114       incr REG_SEL from 11 to 13 do             !TEST THIRTEEN REGISTERS
 12206 ;          6115           begin
 12207 ;          6116           BGNSUB;
 12208 ;          6117           CLR_MBUS;
 12209 ;          6118           WRT_REG (.TST_PAT, .REG_SEL, index);  !WRITE REGISTER WITH BACKGROUND
 12210 ;          6119           CLR_DATA = (.HI) or (.IGNORE);        !CALCULATE THE CLEARED DATA PATTERN
 12211 ;          6120           REG_INIT_FLG = ONE;
 12212 ;          6121           RD_REG (.CLR_DATA, .REG_SEL, ERR_FLG);!READ THE REGISTER FOR THE CLEARED DATA PAT
 12213 ;          6122
 12214 ;          6123           if .ERR_FLG IS_SET                    !SEE IF READ FOUND AN ERROR
 12215 ;          6124           then
 12216 ;          6125               begin                             !IF ERROR FLAG IS SET THEN ERROR AND SET DODU_FLG
 12217 ;          6126               CMP_THRESHOLD;                    !COMPARE ERROR PRINT THRESHOLD
 12218 ;          6127               ERRDF (118, ARR_DAT, DUMPER);     !ARRAY DATA MODULE
 12219 ;          6128               PRINTB (FIV_FMT, PHR_4, WRD_12, WRD_52, FNC_23, WRD_56);
 12220 ;          6129               PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .CLR_DATA, .RD_DATA);
 12221 ;          6130               DODU_FLG = ONE;
 12222 ;          6131               end;
 12223 ;          6132
 12224 ;          6133           ENDSUB;
 12225 ;          6134           end;
 12226 ;          6135
 12227 ;          6136       TST_PAT = not .TST_PAT;                   !REPEAT WITH COMPLIMENT BACKGROUND PAT
 12228 ;          6137       end;
 12229 ;          6138
 12230 ;          6139   REG_INIT_FLG = ZERO;                          !CLEAR THE FLAG
 12231 ;          6140
 12232 ;          6141   if .DODU_FLG IS_SET                           !DROP THIS UNIT IF DODU_FLG SET
 12233 ;          6142   then
 12234 ;          6143       begin
 12235 ;          6144       DODU (.ML_LUN);
 12236 ;          6145       DOCLN;
 12237 ;          6146       end;
 12238 ;          6147
 12239 ;          6148   ENDTST;
 12243
 12244                                      .SBTTL  $T17 TEST CODE SECTION
 12248 043572  004167  141014       $T17:  JSR     R1,$SAVE5                ;                          6005
 12249 043576  024646                      CMP     -(SP),-(SP)
 12250 043600  005046                      CLR     -(SP)                    ; DODU.FLG                 6029
 12251 043602  012702  177777              MOV     #-1,R2                   ; *.TST.PAT                6035
 12252 043606  005067  151564              CLR     P.CNT
 12253 043612  005004                      CLR     R4                       ; TWICE                    6038
```

```
                                  ;ML4AD                                                       29-Mar-1982 16:23:04   TOPS
                                  ;              TEST CODE SECTION                             29-Mar-1982 16:21:03   PA:<
043614  012701  000013            1$:    MOV     #13,R1                   ; *,REG.SEL                                 6041
043620  104402                    2$:    TRAP    2                        ;                                           6042
043622  152777  000040  151614           BISB    #40,@ML.REG+40           ;                                           6043
043630  016705  152176                   MOV     ML.DUT,R5
043634  042705  177770                   BIC     #177770,R5
043640  142777  000007  151576           BICB    #7,@ML.REG+40
043646  150577  151572                   BISB    R5,@ML.REG+40
043652  010246                           MOV     R2,-(SP)                 ; TST.PAT,*                                 6045
043654  010146                           MOV     R1,-(SP)                 ; REG.SEL,*
043656  012746  000012                   MOV     #12,-(SP)
043662  060616                           ADD     SP,(SP)                  ; INDEX,*
043664  004767  161434                   JSR     PC,WRT.REG
043670  010216                           MOV     R2,(SP)                  ; TST.PAT,*                                 6046
043672  010146                           MOV     R1,-(SP)                 ; REG.SEL,*
043674  012746  000014                   MOV     #14,-(SP)
043700  060616                           ADD     SP,(SP)                  ; ERR.FLG,*
043702  004767  162004                   JSR     PC,RD.REG
043706  026627  000014  000001           CMP     14(SP),#1                ; ERR.FLG,*                                 6048
043714  001065                           BNE     4$
043716  005267  151454                   INC     P.CNT                    ;                                           6050
043722  026767  151450  151450           CMP     P.CNT,LIMIT
043730  003403                           BLE     3$
043732  062706  000012                   ADD     #12,SP
043736  000461                           BR      5$
043740  104455                    3$:    TRAP    55                       ;                                           6052
043742  000171                           .WORD   171
043744  013012                           .WORD   ARR.DAT
043746  026302                           .WORD   DUMPER
043750  012746  011344                   MOV     #WRD.56,-(SP)            ;                                           6053
043754  012746  011310                   MOV     #WRD.52,-(SP)
043760  012746  012226                   MOV     #FNC.6,-(SP)
043764  012746  012216                   MOV     #FNC.5,-(SP)
043770  012746  010650                   MOV     #WRD.12,-(SP)
043774  012746  011676                   MOV     #PHR.4,-(SP)
044000  012746  010322                   MOV     #SIX.FMT,-(SP)
044004  012746  000007                   MOV     #7,-(SP)
044010  010600                           MOV     SP,R0                    ; SP,*
044012  104414                           TRAP    14
044014  016716  151332                   MOV     RD.DATA,(SP)             ;                                           6054
044020  016746  151324                   MOV     WT.DATA,-(SP)
044024  016600  000040                   MOV     40(SP),R0                ; INDEX,*
044030  006300                           ASL     R0
044032  006300                           ASL     R0
044034  006300                           ASL     R0
044036  016046  015404                   MOV     ML.REG(R0),-(SP)
044042  012746  007436                   MOV     #FMT.16,-(SP)
044046  012746  000004                   MOV     #4,-(SP)
044052  010600                           MOV     SP,R0                    ; SP,*
044054  104414                           TRAP    14
044056  012766  000001  000042           MOV     #1,42(SP)                ; *,DODU.FLG                                6055
044064  062706  000030                   ADD     #30,SP                   ;                                           6050
044070  062706  000012            4$:    ADD     #12,SP                   ;                                           6042
```

```
12311                                   ;ML4AD
12312                                   ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS
12313                                                                                                      29-Mar-1982 16:21:03    PA:<
12314 044074  104467                            TRAP    67
12315 044076  006000                            ROR     R0                      ;                                                  6056
12316 044100  103647                            BLO     2$
12317 044102  005201                    5$:     INC     R1
12318 044104  020127  000015                    CMP     R1,#15                  ; REG.SEL                                          6041
12319 044110  003643                            BLE     2$                      ; REG.SEL,*
12320 044112  005102                            COM     R2
12321 044114  005204                            INC     R4                      ; TST.PAT                                          6061
12322 044116  020427  000001                    CMP     R4,#1                   ; TWICE                                            6038
12323 044122  003634                            BLE     1$                      ; TWICE,*
12324 044124  012702  000001                    MOV     #1,R2
12325 044130  005003                            CLR     R3                      ; *,TST.PAT                                        6069
12326 044132  005004                    6$:     CLR     R4                      ; SHIFT                                            6071
12327 044134  012701  000013            7$:     MOV     #13,R1                  ; TWICE                                            6074
12328 044140  104402                    8$:     TRAP    2                       ; *,REG.SEL                                        6077
12329 044142  152777  000040  151274            BISB    #40,@ML.REG+40          ;                                                  6078
12330 044150  016705  151656                    MOV     ML.DUT,R5               ;                                                  6079
12331 044154  042705  177770                    BIC     #177770,R5
12332 044160  142777  000007  151256            BICB    #7,@ML.REG+40
12333 044166  150577  151252                    BISB    R5,@ML.REG+40
12334 044172  010246                            MOV     R2,-(SP)
12335 044174  010146                            MOV     R1,-(SP)                ; TST.PAT,*                                        6081
12336 044176  012746  000012                    MOV     #12,-(SP)               ; REG.SEL,*
12337 044202  060616                            ADD     SP,(SP)
12338 044204  004767  161114                    JSR     PC,WRT.REG              ; INDEX,*
12339 044210  010216                            MOV     R2,(SP)
12340 044212  010146                            MOV     R1,-(SP)                ; TST.PAT,*                                        6082
12341 044214  012746  000014                    MOV     #14,-(SP)               ; REG.SEL,*
12342 044220  060616                            ADD     SP,(SP)
12343 044222  004767  161464                    JSR     PC,RD.REG               ; ERR.FLG,*
12344 044226  026627  000014  000001            CMP     14(SP),#1               ; ERR.FLG,*
12345 044234  001065                            BNE     10$                                                                        6084
12346 044236  005267  151134                    INC     P.CNT                   ;                                                  6086
12347 044242  026767  151130  151130            CMP     P.CNT,LIMIT
12348 044250  003403                            BLE     9$
12349 044252  062706  000012                    ADD     #12,SP
12350 044256  000461                            BR      11$
12351 044260  104455                    9$:     TRAP    55                      ;                                                  6088
12352 044262  000172                            .WORD   172
12353 044264  013012                            .WORD   ARR.DAT
12354 044266  026302                            .WORD   DUMPER
12355 044270  012746  011344                    MOV     #WRD.56,-(SP)           ;                                                  6089
12356 044274  012746  011310                    MOV     #WRD.52,-(SP)
12357 044300  012746  012226                    MOV     #FNC.6,-(SP)
12358 044304  012746  012216                    MOV     #FNC.5,-(SP)
12359 044310  012746  010650                    MOV     #WRD.12,-(SP)
12360 044314  012746  011676                    MOV     #PHR.4,-(SP)
12361 044320  012746  010322                    MOV     #SIX.FMT,-(SP)
12362 044324  012746  000007                    MOV     #7,-(SP)
12363 044330  010600                            MOV     SP,R0                   ; SP,*
12364 044332  104414                            TRAP    14
12365 044334  016716  151012                    MOV     RD.DATA,(SP)            ;                                                  6090
```

```
                                        ;ML4AD                                                 29-Mar-1982 16:23:04   TOPS
                                        ;               TEST CODE SECTION                      29-Mar-1982 16:21:03   PA:<
044340  016746  151004                          MOV     WT.DATA,-(SP)
044344  016600  000040                          MOV     40(SP),R0               ; INDEX,*
044350  006300                                  ASL     R0
044352  006300                                  ASL     R0
044354  006300                                  ASL     R0
044356  016046  015404                          MOV     ML.REG(R0),-(SP)
044362  012746  007436                          MOV     #FMT.16,-(SP)
044366  012746  000004                          MOV     #4,-(SP)
044372  010600                                  MOV     SP,R0                   ; SP,*
044374  104414                                  TRAP    14
044376  012766  000001  000042                  MOV     #1,42(SP)               ; *,DODU.FLG
044404  062706  000030                          ADD     #30,SP                  ;                       6091
044410  062706  000012          10$:            ADD     #12,SP                  ;                       6086
044414  104467                                  TRAP    67                      ;                       6078
044416  006000                                  ROR     R0                      ;                       6092
044420  103647                                  BLO     8$
044422  005201                  11$:            INC     R1                      ; REG.SEL
044424  020127  000015                          CMP     R1,#15                  ; REG.SEL,*             6077
044430  003643                                  BLE     8$
044432  005102                                  COM     R2                      ; TST.PAT
044434  005204                                  INC     R4                      ; TWICE                 6097
044436  020427  000001                          CMP     R4,#1                   ; TWICE,*               6074
044442  003634                                  BLE     7$
044444  006302                                  ASL     R2                      ; TST.PAT
044446  005203                                  INC     R3                      ; SHIFT                 6100
044450  020327  000017                          CMP     R3,#17                  ; SHIFT,*               6071
044454  003626                                  BLE     6$
044456  012767  000001  150674                  MOV     #1,REG.INIT.FLG         ;
044464  012702  177777                          MOV     #-1,R2                  ; *,TST.PAT             6108
044470  005004                                  CLR     R4                      ; TWICE                 6109
044472  012703  000013          12$:            MOV     #13,R3                  ; *,REG.SEL             6111
044476  104402                  13$:            TRAP    2                       ;                       6114
044500  152777  000040  150736                  BISB    #40,@ML.REG+40          ;                       6115
044506  016701  151320                          MOV     ML.DUT,R1               ;                       6116
044512  042701  177770                          BIC     #177770,R1
044516  142777  000007  150720                  BICB    #7,@ML.REG+40
044524  150177  150714                          BISB    R1,@ML.REG+40
044530  010246                                  MOV     R2,-(SP)                ; TST.PAT,*             6118
044532  010346                                  MOV     R3,-(SP)                ; REG.SEL,*
044534  012746  000012                          MOV     #12,-(SP)
044540  060616                                  ADD     SP,(SP)                 ; INDEX,*
044542  004767  160556                          JSR     PC,WRT.REG
044546  016600  000012                          MOV     12(SP),R0               ; INDEX,*               6119
044552  006300                                  ASL     R0
044554  006300                                  ASL     R0
044556  006300                                  ASL     R0
044560  010001                                  MOV     R0,R1
044562  016105  015406                          MOV     ML.REG+2(R1),R5         ; *,CLR.DATA
044566  056105  015412                          BIS     ML.REG+6(R1),R5         ; *,CLR.DATA
044572  012767  000001  150560                  MOV     #1,REG.INIT.FLG         ;                       6120
044600  010516                                  MOV     R5,(SP)                 ; CLR.DATA,*            6121
044602  010346                                  MOV     R3,-(SP)                ; REG.SEL,*
```

SEQ 0275

```
12423                                          ;ML4AD
12424                                          ;              TEST CODE SECTION                                    29-Mar-1982 16:23:04    TOPS
12425                                                                                                               29-Mar-1982 16:21:03    PA:<
12426 044604  012746  000014                          MOV     #14,-(SP)
12427 044610  060616                                  ADD     SP,(SP)
12428 044612  004767  161074                          JSR     PC,RD.REG          ; ERR.FLG,*
12429 044616  026627  000014  000001                  CMP     14(SP),#1          ; ERR.FLG,*                      6123
12430 044624  001055                                  BNE     15$
12431 044626  005267  150544                          INC     P.CNT              ;                                6125
12432 044632  026767  150540  150540                  CMP     P.CNT,LIMIT
12433 044640  003403                                  BLE     14$
12434 044642  062706  000012                          ADD     #12,SP
12435 044646  000451                                  BR      16$
12436 044650  104455                          14$:    TRAP    55                 ;                                6127
12437 044652  000166                                  .WORD   166
12438 044654  013012                                  .WORD   ARR.DAT
12439 044656  026302                                  .WORD   DUMPER
12440 044660  012746  011344                          MOV     #WRD.56,-(SP)      ;                                6128
12441 044664  012746  012464                          MOV     #FNC.23,-(SP)
12442 044670  012746  011310                          MOV     #WRD.52,-(SP)
12443 044674  012746  010650                          MOV     #WRD.12,-(SP)
12444 044700  012746  011676                          MOV     #PHR.4,-(SP)
12445 044704  012746  010304                          MOV     #FIV.FMT,-(SP)
12446 044710  012746  000006                          MOV     #6,-(SP)
12447 044714  010600                                  MOV     SP,R0              ; SP,*
12448 044716  104414                                  TRAP    14
12449 044720  016716  150426                          MOV     RD.DATA,(SP)       ;                                6129
12450 044724  010546                                  MOV     R5,-(SP)           ; CLR.DATA,*
12451 044726  016146  015404                          MOV     ML.REG(R1),-(SP)
12452 044732  012746  007436                          MOV     #FMT.16,-(SP)
12453 044736  012746  000004                          MOV     #4,-(SP)
12454 044742  010600                                  MOV     SP,R0              ; SP,*
12455 044744  104414                                  TRAP    14
12456 044746  012766  000001  000040                  MOV     #1,40(SP)          ; *,DODU.FLG                     6130
12457 044754  062706  000026                          ADD     #26,SP             ;                                6125
12458 044760  062706  000012                  15$:    ADD     #12,SP             ;                                6115
12459 044764  104467                                  TRAP    67                 ;                                6131
12460 044766  006000                                  ROR     R0
12461 044770  103642                                  BLO     13$
12462 044772  005203                          16$:    INC     R3                 ; REG.SEL                        6114
12463 044774  020327  000015                          CMP     R3,#15             ; REG.SEL,*
12464 045000  003636                                  BLE     13$
12465 045002  005102                                  COM     R2                 ; TST.PAT                        6136
12466 045004  005204                                  INC     R4                 ; TWICE                          6111
12467 045006  020427  000001                          CMP     R4,#1              ; TWICE,*
12468 045012  003627                                  BLE     12$
12469 045014  005067  150340                          CLR     REG.INIT.FLG       ;                                6139
12470 045020  021627  000001                          CMP     (SP),#1            ; DODU.FLG,*                     6141
12471 045024  001004                                  BNE     17$
12472 045026  016700  150776                          MOV     ML.LUN,R0          ;                                6144
12473 045032  104451                                  TRAP    51
12474 045034  104444                                  TRAP    44
12475 045036  062706  000006                  17$:    ADD     #6,SP              ;                                6005
12476 045042  000207                                  RTS     PC
12477
```

SEQ 0276


```
12479                                   ;ML4AD
12480                                   ;               TEST CODE SECTION
12481
12482                                   ; Routine Size:  341 words
12483                                   ; Maximum stack depth per invocation:  26 words
12488
12489
12493
12494                                           .SBTTL  T17 TEST CODE SECTION
12498 045044                            T17::
12499 045044  004767  176522            1$:     JSR     PC,$T17                 ;               6146
12500 045050  104466                            TRAP    66
12501 045052  006000                            ROR     R0
12502 045054  103773                            BLO     1$
12503 045056  000207                            RTS     PC
12504
12505                                   ; Routine Size:  6 words
12506                                   ; Maximum stack depth per invocation:  0 words
12511
12512
12513 ;         6149  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (70)

```
12515 ;ML4AD
12516 ;
12517 ;                 TEST CODE SECTION
12518 ;          6150 !
12519 ;          6151 !
12520 ;          6152 BGNTST;
12521 ;          6153
12522 ;          6154 !++
12523 ;          6155 ! TEST NUMBER:  TST 18
12524 ;          6156 !
12525 ;          6157 ! TEST NAME:    COMPOSIT ERROR TEST
12526 ;          6158 !
12527 ;          6159 ! TEST DESCRIPTION:
12528 ;          6160 !       TEST TO SEE IF SETTING EACH
12529 ;          6161 !       BIT IN THE ERROR REGISTER
12530 ;          6162 !       CAUSES A COMPOSIT ERROR BY:
12531 ;          6163 !
12532 ;          6164 !       WRITING A SHIFTING ONE THROUGH
12533 ;          6165 !       THE ERROR REGISTER (SKIPPING THE
12534 ;          6166 !       READ ONLY BITS) AND TESTING THE
12535 ;          6167 !       COMPOSIT ERROR BIT IN MLDS
12536 ;          6168 !       FOR BEING SET AFTER EACH
12537 ;          6169 !       WRITE.
12538 ;          6170 !
12539 ;          6171 !--
12540 ;          6172
12541 ;          6173 local
12542 ;          6174     DAT_PAT,                          !DATA PATTERN
12543 ;          6175     SKIP_MASK;                        !POINTS TO MLER READ ONLY BITS
12544 ;          6176
12545 ;          6177 CLR_MBUS;
12546 ;          6178 SKIP_MASK = %o'163157';                !LOAD SKIP MASK
12547 ;          6179 DAT_PAT = ONE;                         !DATA PATTERN SET BIT 0 IN MLER
12548 ;          6180
12549 ;          6181 incr COUNT from 0 to 15 do             !WRITE AND SHIFT DATA PAT TO MLER 16 TIMES
12550 ;          6182     begin
12551 ;          6183
12552 ;          6184     if (.DAT_PAT and .SKIP_MASK) neq ZERO   !SKIP IF DAT_PAT FALLS ON READ ONLY BIT
12553 ;          6185     then
12554 ;          6186         begin
12555 ;          6187         BGNSUB;
12556 ;          6188         MLER = .DAT_PAT;                !WRITE DATA_PAT TO MLER
12557 ;          6189
12558 ;          6190         if .COMP_ERR IS_NOT_SET         !SEE IF DAT_PAT CAUSED A COMP ERROR
12559 ;          6191         then
12560 ;          6192             begin
12561 ;          6193             ERRDF (50, ASYNC, DUMPER);  !ERROR IF NO COMP ERROR
12562 ;          6194             PRINTB (FOR_FMT, FNC_8, PHR_1, WRD_12, FNC_8);
12563 ;          6195             PRINTB (FMT_4, .DAT_PAT);
12564 ;          6196             end;
12565 ;          6197
12566 ;          6198         ENDSUB;
12567 ;          6199         end;
12568 ;          6200
12569 ;          6201     DAT_PAT = .DAT_PAT^ONE;             !SHIFT DAT_PAT TO NEXT BIT AND REPEAT
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (70)

```
12571 ;ML4AD
12572 ;                 TEST CODE SECTION
12573
12574 ;          6202       end;
12575 ;          6203
12576 ;          6204   ENDTST;
12580
12581                                          .SBTTL  $T18 TEST CODE SECTION
12585 045060  004167  137470          $T18:    JSR     R1,$SAVE3                ;                                  6148
12586 045064  152777  000040  150352           BISB    #40,@ML.REG+40           ;                                  6175
12587 045072  016703  150734                   MOV     ML.DUT,R3
12588 045076  042703  177770                   BIC     #177770,R3
12589 045102  142777  000007  150334           BICB    #7,@ML.REG+40
12590 045110  150377  150330                   BISB    R3,@ML.REG+40
12591 045114  012703  163157                   MOV     #-14621,R3
12592 045120  012701  000001                   MOV     #1,R1                    ; *,SKIP.MASK                      6178
12593 045124  005002                           CLR     R2                       ; *,DAT.PAT                        6179
12594 045126  030103                   1$:     BIT     R1,R3                    ; COUNT                            6181
12595 045130  001445                           BEQ     4$                       ; DAT.PAT,SKIP.MASK                6184
12596 045132  104402                   2$:     TRAP    2                        ;                                  6186
12597 045134  010177  150324                   MOV     R1,@ML.REG+60            ; DAT.PAT,*                        6188
12598 045140  032777  040000  150306           BIT     #40000,@ML.REG+50        ;                                  6190
12599 045146  001033                           BNE     3$                       ;                                  6193
12600 045150  104455                           TRAP    55                       ;
12601 045152  000062                           .WORD   62
12602 045154  012706                           .WORD   ASYNC
12603 045156  026302                           .WORD   DUMPER
12604 045160  012746  012244                   MOV     #FNC.8,-(SP)             ;                                  6194
12605 045164  012746  010650                   MOV     #WRD.12,-(SP)
12606 045170  012746  011610                   MOV     #PHR.1,-(SP)
12607 045174  012746  012244                   MOV     #FNC.8,-(SP)
12608 045200  012746  010270                   MOV     #FOR.FMT,-(SP)
12609 045204  012746  000005                   MOV     #5,-(SP)
12610 045210  010600                           MOV     SP,R0                    ; SP,*
12611 045212  104414                           TRAP    14
12612 045214  010116                           MOV     R1,(SP)                  ; DAT.PAT,*                        6195
12613 045216  012746  006620                   MOV     #FMT.4,-(SP)
12614 045222  012746  000002                   MOV     #2,-(SP)
12615 045226  010600                           MOV     SP,R0                    ; SP,*
12616 045230  104414                           TRAP    14
12617 045232  062706  000020                   ADD     #20,SP                   ;
12618 045236  104467                   3$:     TRAP    67                       ;                                  6192
12619 045240  006000                           ROR     R0                       ;                                  6196
12620 045242  103733                           BLO     2$
12621 045244  006301                   4$:     ASL     R1                       ; DAT.PAT                          6201
12622 045246  005202                           INC     R2                       ; COUNT                            6181
12623 045250  020227  000017                   CMP     R2,#17                   ; COUNT,*
12624 045254  003724                           BLE     1$
12625 045256  000207                           RTS     PC                       ;                                  6148
```

```
                                ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                ;                TEST CODE SECTION                              29-Mar-1982 16:21:03    PA:<

                                ; Routine Size:  64 words
                                ; Maximum stack depth per invocation:  12 words


                                        .SBTTL  T18 TEST CODE SECTION
045260                          T18::
045260  004767  177574          1$:     JSR     PC,$T18                         ;               6202
045264  104466                          TRAP    66
045266  006000                          ROR     R0
045270  103773                          BLO     1$
045272  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


                        6205    !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (71)

```
12664 ;ML4AD
12665 ;                 TEST CODE SECTION
12666 ;
12667 ;          6206 !
12668 ;          6207 !
12669 ;          6208 BGNTST;
12670 ;          6209
12671 ;          6210 !++
12672 ;          6211 ! TEST NUMBER:  TST 19
12673 ;          6212 !
12674 ;          6213 ! TEST NAME:    ATA BIT TEST
12675 ;          6214 !
12676 ;          6215 ! TEST DESCRIPTION:
12677 ;          6216 !
12678 ;          6217 !       TEST THE ATA BIT FOR SETTING
12679 ;          6218 !       BY:
12680 ;          6219 !
12681 ;          6220 !       1.  SETTING A BIT IN THE ERROR
12682 ;          6221 !           REGISTER.
12683 ;          6222 !
12684 ;          6223 !           TEST THE ATA BIT FOR CLEARING
12685 ;          6224 !           AFTER BEING SET BY:
12686 ;          6225 !           1.  WRITING A FUNCTION TO MLCS1.
12687 ;          6226 !           2.  WRITING A ONE INTO THIS
12688 ;          6227 !               UNITS ATA BIT
12689 ;          6228 !
12690 ;          6229 !       TEST THE ATA BIT FOR NOT CLEARING
12691 ;          6230 !       AFTER BEING SET BY:
12692 ;          6231 !
12693 ;          6232 !       1.  WRITING A ONE INTO THE
12694 ;          6233 !           ATA BIT OF THE OTHER
12695 ;          6234 !           UNITS.
12696 ;          6235 !
12697 ;          6236 !--
12698 ;          6237
12699 ;          6238 local
12700 ;          6239     ATA_SAVE : bitvector [8],          !STORES ALL 8 ATA BITS ON READS AND WRITES
12701 ;          6240     DAT_PAT;                           !DATA PATTERN
12702 ;          6241
12703 ;          6242 CLR_MBUS;
12704 ;          6243 MLER = ONE;                            !SET THE ATA BIT
12705 ;          6244 MLER = ZERO;
12706 ;          6245 ATA_SAVE = .MLAS;                      !READ THE ATTN REGISTER
12707 ;          6246
12708 ;          6247 if .ATA_SAVE [.ML_DUT] IS_NOT_SET      !SEE IF THIS DRIVES ATA BIT IS SET
12709 ;          6248 then
12710 ;          6249     begin
12711 ;          6250     ERRDF (51, ASYNC, DUMPER);         !ERROR AND EXIT_TST IF NOT SET
12712 ;          6251     PRINTB (FOR_FMT, WRD_15, PHR_1, WRD_11, FNC_8);
12713 ;          6252     EXIT_TST;
12714 ;          6253     end;
12715 ;          6254
12716 ;          6255 if .ATTN IS_NOT_SET                    !SEE IF THE ATTN BIT IS SET
12717 ;          6256 then
12718 ;          6257     begin
```

```
12720 ;ML4AD
12721 ;                 TEST CODE SECTION
12722 ;
12723 ;          6258       ERRDF (52, ASYNC, DUMPER);                  !ERROR AND EXIT_TST IF NOT SET
12724 ;          6259       PRINTB (FIV_FMT, REG_2, WRD_16, PHR_1, WRD_11, FNC_8);
12725 ;          6260       EXIT_TST;
12726 ;          6261       end;
12727 ;          6262
12728 ;          6263   MLCS1 = NOOP;                                   !TRY TO CLEAR THE ATA BIT WITH NOOP FUNC
12729 ;          6264
12730 ;          6265   if .ATTN IS_SET                                 !SEE IF ATA GOT CLEARED
12731 ;          6266   then
12732 ;          6267       begin
12733 ;          6268       ERRDF (53, ASYNC, DUMPER);                  !ERROR AND EXIT_TST IF SET
12734 ;          6269       PRINTB (FOR_FMT, WRD_15, PHR_2, WRD_11, FNC_2, WRD_19);
12735 ;          6270       EXIT_1ST;
12736 ;          6271       end;
12737 ;          6272
12738 ;          6273   ATA_SAVE = .MLAS;                               !READ THE ATTENTION REGISTER
12739 ;          6274
12740 ;          6275   if .ATA_SAVE [.ML_DUT] IS_SET                   !SEE IF THE ATA REG GOT CLEARED BY NO-OP
12741 ;          6276   then
12742 ;          6277       begin
12743 ;          6278       ERRDF (58, ASYNC, DUMPER);
12744 ;          6279       PRINTB (FIV_FMT, WRD_15, PHR_2, WRD_11, FNC_2, WRD_19);
12745 ;          6280       end;
12746 ;          6281
12747 ;          6282   BGNSUB;
12748 ;          6283   MLER = ONE;                                     !SET THE ATA BIT
12749 ;          6284   MLER = ZERO;
12750 ;          6285   ATA_SAVE = ZEROES;                              !CLEAR ATA_SAVE
12751 ;          6286   ATA_SAVE [.ML_DUT] = ONE;                       !SET ATA_SAVE FOR THIS DRIVE
12752 ;          6287   MLAS = .ATA_SAVE;                               !TRY TO CLEAR THE ATA BY WRITING TO IT.
12753 ;          6288
12754 ;          6289   if .ATTN IS_SET                                 !SEE IF THE ATA GOT CLEARED
12755 ;          6290   then
12756 ;          6291       begin
12757 ;          6292       ERRDF (54, ASYNC, DUMPER);                  !ERROR IF NOT CLEARED
12758 ;          6293       PRINTB (FIV_FMT, WRD_15, PHR_2, WRD_11, WRD_17, REG_5);
12759 ;          6294       end;
12760 ;          6295
12761 ;          6296   ENDSUB;
12762 ;          6297   BGNSUB;
12763 ;          6298   MLER = ONE;                                     !SET THE ATA BIT
12764 ;          6299   MLER = ZERO;
12765 ;          6300   DAT_PAT = ONE;                                  !DATA PATTERN OF ONE IN FIELD OF ZEROES
12766 ;          6301
12767 ;          6302   incr ATA_SEL from 0 to 7 do                     !REPEAT LOOP 8 TIMES
12768 ;          6303       begin
12769 ;          6304
12770 ;          6305       if .ATA_SEL neq .ML_DUT                     !SKIP IF ATA_SEL EQLS THIS DRIVE NO.
12771 ;          6306       then
12772 ;          6307           begin
12773 ;          6308           MLAS = .DAT_PAT;                        !WRITE DAT PAT TO ATA REGISTER
12774 ;          6309           ATA_SAVE = .MLAS;                       !READ ATA REG BACK
```

```
 12776 ;ML4AD
 12777 ;                 TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 12778 ;                                                                  29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (71)
 12779 ;          6310
 12780 ;          6311          if .ATA_SAVE [.ML_DUT] IS_NOT_SET          !SEE IF THIS DRIVE ATA IS CLEARED
 12781 ;          6312          then
 12782 ;          6313              begin
 12783 ;          6314              ERRDF (55, ASYNC, DUMPER);             !ERROR AND EXIT LOOP IF CLEARED
 12784 ;          6315              PRINTB (SIX_FMT, WRD_15, PHR_6, WRD_11, WRD_17, REG_5, PHR_7);
 12785 ;          6316              PRINTB (FMT_7, .DAT_PAT);
 12786 ;          6317              exitloop;
 12787 ;          6318              end;
 12788 ;          6319
 12789 ;          6320          end;
 12790 ;          6321
 12791 ;          6322      DAT_PAT = .DAT_PAT^ONE;                        !SHIFT DAT_PAT AND REPEAT
 12792 ;          6323      end;
 12793 ;          6324
 12794 ;          6325  ENDSUB;
 12795 ;          6326  ENDTST;
 12799
 12800                                          .SBTTL  $T19 TEST CODE SECTION
 12804 045274  004167  137254          $T19:   JSR     R1,$SAVE3
 12805 045300  152777  000040  150136          BISB    #40,@ML.REG+40          ;                       6204
 12806 045306  016703  150520                  MOV     ML.DUT,R3               ;                       6240
 12807 045312  042703  177770                  BIC     #177770,R3
 12808 045316  142777  000007  150120          BICB    #7,@ML.REG+40
 12809 045324  150377  150114                  BISB    R3,@ML.REG+40
 12810 045330  012777  000001  150126          MOV     #1,@ML.REG+60           ;                       6243
 12811 045336  005077  150122                  CLR     @ML.REG+60              ;                       6244
 12812 045342  017746  150126                  MOV     @ML.REG+70,-(SP)        ; *.ATA.SAVE            6245
 12813 045346  016701  150460                  MOV     ML.DUT,R1               ;                       6247
 12814 045352  006201                          ASR     R1
 12815 045354  006201                          ASR     R1
 12816 045356  006201                          ASR     R1
 12817 045360  010600                          MOV     SP,R0                   ; ATA.SAVE,*
 12818 045362  060001                          ADD     R0,R1
 12819 045364  010146                          MOV     R1,-(SP)
 12820 045366  016746  150440                  MOV     ML.DUT,-(SP)
 12821 045372  042716  177770                  BIC     #177770,(SP)
 12822 045376  012746  000001                  MOV     #1,-(SP)
 12823 045402  005046                          CLR     -(SP)
 12824 045404  004767  136224                  JSR     PC,BL$GT2
 12825 045410  062706  000010                  ADD     #10,SP
 12826 045414  005700                          TST     R0
 12827 045416  001026                          BNE     1$
 12828 045420  104455                          TRAP    55                      ;                       6250
 12829 045422  000063                          .WORD   63
 12830 045424  012706                          .WORD   ASYNC
```

SEQ 0283

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
12832                                          ;ML4AD
12833                                          ;                    TEST CODE SECTION
12834
12835 045426  026302                                    .WORD   DUMPER
12836 045430  012746  012244                              MOV     #FNC.8,-(SP)          ;                                  6251
12837 045434  012746  010640                              MOV     #WRD.11,-(SP)
12838 045440  012746  011610                              MOV     #PHR.1,-(SP)
12839 045444  012746  010676                              MOV     #WRD.15,-(SP)
12840 045450  012746  010270                              MOV     #FOR.FMT,-(SP)
12841 045454  012746  000005                              MOV     #5,-(SP)
12842 045460  010600                                      MOV     SP,R0                 ; SP,*
12843 045462  104414                                      TRAP    14
12844 045464  104463                                      TRAP    63
12845 045466  062706  000014                              ADD     #14,SP
12846 045472  000467                                      BR      4$                    ;                                  6247
12847 045474  032777  100000  147752  1$:                 BIT     #100000,@ML.REG+50    ;                                  6249
12848 045502  001026                                      BNE     2$                    ;                                  6255
12849 045504  104455                                      TRAP    55
12850 045506  000064                                      .WORD   64                    ;                                  6258
12851 045510  012706                                      .WORD   ASYNC
12852 045512  026302                                      .WORD   DUMPER
12853 045514  012746  012244                              MOV     #FNC.8,-(SP)
12854 045520  012746  010640                              MOV     #WRD.11,-(SP)         ;                                  6259
12855 045524  012746  011610                              MOV     #PHR.1,-(SP)
12856 045530  012746  010704                              MOV     #WRD.16,-(SP)
12857 045534  012746  012510                              MOV     #REG.2,-(SP)
12858 045540  012746  010304                              MOV     #FIV.FMT,-(SP)
12859 045544  012746  000006                              MOV     #6,-(SP)
12860 045550  010600                                      MOV     SP,R0                 ; SP,*
12861 045552  104414                                      TRAP    14
12862 045554  104463                                      TRAP    63
12863 045556  000433                                      BR      3$
12864 045560  012777  000001  147616  2$:                 MOV     #1,@ML.REG            ;                                  6255
12865 045566  005777  147662                              TST     @ML.REG+50            ;                                  6263
12866 045572  100031                                      BPL     5$                    ;                                  6265
12867 045574  104455                                      TRAP    55
12868 045576  000065                                      .WORD   65                    ;                                  6268
12869 045600  012706                                      .WORD   ASYNC
12870 045602  026302                                      .WORD   DUMPER
12871 045604  012746  010730                              MOV     #WRD.19,-(SP)         ;                                  6269
12872 045610  012746  012164                              MOV     #FNC.2,-(SP)
12873 045614  012746  010640                              MOV     #WRD.11,-(SP)
12874 045620  012746  011626                              MOV     #PHR.2,-(SP)
12875 045624  012746  010676                              MOV     #WRD.15,-(SP)
12876 045630  012746  010270                              MOV     #FOR.FMT,-(SP)
12877 045634  012746  000006                              MOV     #6,-(SP)
12878 045640  010600                                      MOV     SP,R0                 ; SP,*
12879 045642  104414                                      TRAP    14
12880 045644  104463                                      TRAP    63
12881 045646  062706  000016              3$:             ADD     #16,SP                ;                                  6265
12882 045652  000167  000546              4$:             JMP     12$                   ;                                  6267
12883 045656  017716  147612              5$:             MOV     @ML.REG+70,(SP)       ; *,ATA.SAVE                       6273
12884 045662  016701  150144                              MOV     ML.DUT,R1             ;                                  6275
12885 045666  006201                                      ASR     R1
12886 045672  006201                                      ASR     R1
```

```
12888
12889                                   ;ML4AD
12890                                   ;               TEST CODE SECTION
12891 045672  006201                              ASR     R1
12892 045674  010600                              MOV     SP,R0                   ; ATA.SAVE,*
12893 045676  060001                              ADD     R0,R1
12894 045700  010146                              MOV     R1,-(SP)
12895 045702  016746  150124                      MOV     ML.DUT,-(SP)
12896 045706  042716  177770                      BIC     #177770,(SP)
12897 045712  012746  000001                      MOV     #1,-(SP)
12898 045716  005046                              CLR     -(SP)
12899 045720  004767  135710                      JSR     PC,BL$GT2
12900 045724  062706  000010                      ADD     #10,SP
12901 045730  005300                              DEC     R0
12902 045732  001026                              BNE     6$
12903 045734  104455                              TRAP    55                      ;                               6278
12904 045736  000072                              .WORD   72
12905 045740  012706                              .WORD   ASYNC
12906 045742  026302                              .WORD   DUMPER
12907 045744  012746  010730                      MOV     #WRD.19,-(SP)           ;                               6279
12908 045750  012746  012164                      MOV     #FNC.2,-(SP)
12909 045754  012746  010640                      MOV     #WRD.11,-(SP)
12910 045760  012746  011626                      MOV     #PHR.2,-(SP)
12911 045764  012746  010676                      MOV     #WRD.15,-(SP)
12912 045770  012746  010304                      MOV     #FIV.FMT,-(SP)
12913 045774  012746  000006                      MOV     #6,-(SP)
12914 046000  010600                              MOV     SP,R0                   ; SP,*
12915 046002  104414                              TRAP    14
12916 046004  062706  000016                      ADD     #16,SP
12917 046010  104402                      6$:     TRAP    2                       ;                               6277
12918 046012  012777  000001  147444              MOV     #1,@ML.REG+60           ;                               6280
12919 046020  005077  147440                      CLR     @ML.REG+60              ;                               6283
12920 046024  005016                              CLR     (SP)                    ;                               6284
12921 046026  016701  150000                      MOV     ML.DUT,R1               ; ATA.SAVE                      6285
12922 046032  006201                              ASR     R1                      ;                               6286
12923 046034  006201                              ASR     R1
12924 046036  006201                              ASR     R1
12925 046040  010600                              MOV     SP,R0                   ; ATA.SAVE,*
12926 046042  060001                              ADD     R0,R1
12927 046044  010146                              MOV     R1,-(SP)
12928 046046  016746  147760                      MOV     ML.DUT,-(SP)
12929 046052  042716  177770                      BIC     #177770,(SP)
12930 046056  012746  000001                      MOV     #1,-(SP)
12931 046062  011646                              MOV     (SP),-(SP)
12932 046064  004767  136002                      JSR     PC,BL$PU2
12933 046070  016677  000010  147376              MOV     10(SP),@ML.REG+70       ; ATA.SAVE,*                    6287
12934 046076  005777  147352                      TST     @ML.REG+50              ;                               6289
12935 046102  100026                              BPL     7$
12936 046104  104455                              TRAP    55                      ;                               6292
12937 046106  000066                              .WORD   66
12938 046110  012706                              .WORD   ASYNC
12939 046112  026302                              .WORD   DUMPER
12940 046114  012746  012532                      MOV     #REG.5,-(SP)            ;                               6293
12941 046120  012746  010712                      MOV     #WRD.17,-(SP)
12942 046124  012746  010640                      MOV     #WRD.11,-(SP)
```

```
12944                                   ;ML4AD
12945                                   ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS
12946                                                                                                       29-Mar-1982 16:21:03    PA:<
12947 046130  012746  011626                     MOV     #PHR.2,-(SP)
12948 046134  012746  010676                     MOV     #WRD.15,-(SP)
12949 046140  012746  010304                     MOV     #FIV.FMT,-(SP)
12950 046144  012746  000006                     MOV     #6,-(SP)
12951 046150  010600                             MOV     SP,R0                   ; SP,*
12952 046152  104414                             TRAP    14
12953 046154  062706  000016                     ADD     #16,SP                  ;                                                   6291
12954 046160  062706  000010             7$:     ADD     #10,SP                  ;                                                   6280
12955 046164  104467                             TRAP    67                      ;                                                   6294
12956 046166  006000                             ROR     R0
12957 046170  103707                             BLO     6$
12958 046172  104402             8$:             TRAP    2
12959 046174  012777  000001  147262             MOV     #1,@ML.REG+60           ;                                                   6296
12960 046202  005077  147256                     CLR     @ML.REG+60              ;                                                   6298
12961 046206  012703  000001                     MOV     #1,R3                   ;                                                   6299
12962 046212  005002                             CLR     R2                      ; *,DAT.PAT                                         6300
12963 046214  020267  147612             9$:     CMP     R2,ML.DUT               ; ATA.SEL                                           6302
12964 046220  001471                             BEQ     10$                     ; ATA.SEL,*                                         6305
12965 046222  010377  147246                     MOV     R3,@ML.REG+70           ; DAT.PAT,*
12966 046226  017716  147242                     MOV     @ML.REG+70,(SP)         ; *,ATA.SAVE                                        6308
12967 046232  016701  147574                     MOV     ML.DUT,R1               ;                                                   6309
12968 046236  006201                             ASR     R1                                                                          6311
12969 046240  006201                             ASR     R1
12970 046242  006201                             ASR     R1
12971 046244  010600                             MOV     SP,R0
12972 046246  060001                             ADD     R0,R1                   ; ATA.SAVE,*
12973 046250  010146                             MOV     R1,-(SP)
12974 046252  016746  147554                     MOV     ML.DUT,-(SP)
12975 046256  042716  177770                     BIC     #177770,(SP)
12976 046262  012746  000001                     MOV     #1,-(SP)
12977 046266  005046                             CLR     -(SP)
12978 046270  004767  135340                     JSR     PC,BL$GT2
12979 046274  062706  000010                     ADD     #10,SP
12980 046300  005700                             TST     R0
12981 046302  001040                             BNE     10$
12982 046304  104455                             TRAP    55
12983 046306  000067                             .WORD   67                      ;                                                   6314
12984 046310  012706                             .WORD   ASYNC
12985 046312  026302                             .WORD   DUMPER
12986 046314  012746  011740                     MOV     #PHR.7,-(SP)            ;                                                   6315
12987 046320  012746  012532                     MOV     #REG.5,-(SP)
12988 046324  012746  010712                     MOV     #WRD.17,-(SP)
12989 046330  012746  010640                     MOV     #WRD.11,-(SP)
12990 046334  012746  011726                     MOV     #PHR.6,-(SP)
12991 046340  012746  010676                     MOV     #WRD.15,-(SP)
12992 046344  012746  010322                     MOV     #SIX.FMT,-(SP)
12993 046350  012746  000007                     MOV     #7,-(SP)
12994 046354  010600                             MOV     SP,R0                   ; SP,*
12995 046356  104414                             TRAP    14
12996 046360  010316                             MOV     R3,(SP)                 ; DAT.PAT,*                                         6316
12997 046362  012746  007002                     MOV     #FMT.7,-(SP)
12998 046366  012746  000002                     MOV     #2,-(SP)
```

```
13000                                   ;ML4AD
13001                                   ;               TEST CODE SECTION
13002
13003 046372  010600                            MOV     SP,R0                   ; SP,*
13004 046374  104414                            TRAP    14
13005 046376  062706  000024                    ADD     #24,SP                  ;                       6317
13006 046402  000405                            BR      11$
13007 046404  006303                    10$:    ASL     R3                      ; DAT.PAT               6322
13008 046406  005202                            INC     R2                      ; ATA.SEL               6302
13009 046410  020227  000007                    CMP     R2,#7                   ; ATA.SEL,*
13010 046414  003677                            BLE     9$
13011 046416  104467                    11$:    TRAP    67                      ;                       6323
13012 046420  006000                            ROR     R0
13013 046422  103663                            BLO     8$
13014 046424  005726                    12$:    TST     (SP)+                   ;                       6204
13015 046426  000207                            RTS     PC
13016
13017                                   ; Routine Size:  302 words
13018                                   ; Maximum stack depth per invocation:  16 words
13023
13024
13028
13029                                           .SBTTL  T19 TEST CODE SECTION
13033 046430                            T19::
13034 046430  004767  176640            1$:     JSR     PC,$T19                 ;                       6325
13035 046434  104466                            TRAP    66
13036 046436  006000                            ROR     R0
13037 046440  103773                            BLO     1$
13038 046442  000207                            RTS     PC
13039
13040                                   ; Routine Size:  6 words
13041                                   ; Maximum stack depth per invocation:  0 words
13046
13047
13048 ;               6327    !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (72)

```
;ML4AD
;                    TEST CODE SECTION

                6328 !
                6329 !
                6330 BGNTST;
                6331
                6332 !++
                6333 ! TEST NUMBER:  TST 20
                6334 !
                6335 ! TEST NAME:    SEARCH FUNCTION TEST
                6336 !
                6337 ! TEST DESCRIPTION:
                6338 !
                6339 !       TEST THE SEARCH FUNCTION BY:
                6340 !
                6341 !       1.  DOING A SEARCH FUNCTION AT
                6342 !           ARRAY ZERO AND TEST GO,
                6343 !           ERROR BITS AND ATTN FOR
                6344 !           SETTING/NOT SETTING.
                6345 !
                6346 !       2.  DOING SEARCH FUNCTIONS AT ALL
                6347 !           PRESENT ARRAYS' AND TEST ATTN
                6348 !           SET
                6349 !
                6350 !       3.  DOING SEARCH FUNCTIONS AT ALL
                6351 !           NOT PRESENT ARRAYS' AND TEST
                6352 !           ATTN CLEARED.
                6353 !
                6354 !--
                6355
                6356 CLR_MBUS;
                6357 MLDA = ZEROES;
                6358 MLCS1 = SEARCH;                              !DO A SEARCH FUNCTION
                6359
                6360 if .GO IS_SET                              !SEE IF GO IS SET
                6361 then
                6362     begin
                6363     ERRDF (56, ASYNC, DUMPER);            !ERROR IF NOT SET
                6364     PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_10, WRD_19);
                6365     end;
                6366
                6367 if .ILF IS_SET                             !SEE IF ILF IS SET
                6368 then
                6369     begin
                6370     ERRDF (57, ASYNC, DUMPER);            !ERROR IF SET
                6371     PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_11, FNC_11, FNC_10, WRD_19);
                6372     end;
                6373
                6374 if .ATTN IS_NOT_SET                        !SEE IF ATTN IS SET
                6375 then
                6376     begin
                6377     ERRDF (59, ASYNC, DUMPER);            !ERROR IF NOT SET
                6378     PRINTB (FIV_FMT, WRD_16, PHR_1, WRD_12, FNC_10, WRD_19);
                6379     end
```

```
13106 ;ML4AD
13107 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
13108                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (72)
13109 ;          6380  else
13110 ;          6381      begin
13111 ;          6382
13112 ;          6383      CLR_THRESHOLD;                                          !CLEAR ERROR PRINT THRESHOLD
13113 ;          6384
13114 ;          6385  !
13115 ;          6386  ! VERSION CZMLAD CHANGED INCR TO INCRU
13116 ;          6387  !
13117 ;          6388      incru ARR_SEL from 0 to .LST_ARR by .ARR_INC do     !DO SEARCH AT ALL PRESENT ARRAYS
13118 ;          6389          begin
13119 ;          6390          BGNSUB;
13120 ;          6391          CLR_MBUS;
13121 ;          6392          MLDA = .ARR_SEL;                                !LOAD DSA REG WITH ARR_SEL
13122 ;          6393          MLCS1 = SEARCH;                                 !DO A SEARCH FUNCTION
13123 ;          6394
13124 ;          6395          if .OPI IS_SET                                  !READ ATTN
13125 ;          6396          then
13126 ;          6397              begin
13127 ;          6398              CMP_THRESHOLD;                              !COMPARE ERROR PRINT THRESHOLD
13128 ;          6399              ERRDF (60, ASYNC, DUMPER);                  !ERROR IF NOT SET
13129 ;          6400              PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_10, WRD_19);
13130 ;          6401              PRINTB (FMT_9, .ARR_SEL);
13131 ;          6402              end;
13132 ;          6403
13133 ;          6404          ENDSUB;
13134 ;          6405          end;
13135 ;          6406
13136 ;          6407      CLR_THRESHOLD;                                          !CLEAR ERROR PRINT THRESHOLD
13137 ;          6408
13138 ;          6409      if .OP_NUM_ARR lss %o'000017'                   !SEE IF LSS 17 ARRAYS ARE PRESENT
13139 ;          6410      then
13140 ;          6411
13141 ;          6412  !
13142 ;          6413  ! VERSION CZMLAD CHANGED INCR TO INCRU
13143 ;          6414  !
13144 ;          6415          incru ARR_SEL from (.LST_ARR + .ARR_INC) to .ARR_16 by .ARR_INC do
13145 ;          6416                                                          !DO A SEARCH AT ALL NOT PRESENT
13146 ;          6417              begin                                       !ARRAYS IF LSS 17
13147 ;          6418              BGNSUB;
13148 ;          6419              CLR_MBUS;
13149 ;          6420              MLDA = .ARR_SEL;                            !LOAD DSA REG WITH ARR_SEL
13150 ;          6421              MLCS1 = SEARCH;                             !DO A SEARCH FUNCTION
13151 ;          6422
13152 ;          6423              if .OPI IS_NOT_SET                          !SEE IF OPI IS SET
13153 ;          6424              then
13154 ;          6425                  begin
13155 ;          6426                  CMP_THRESHOLD;                          !COMPARE ERROR PRINT THRESHOLD
13156 ;          6427                  ERRDF (61, ASYNC, DUMPER);              !ERROR IF NOT SET
13157 ;          6428                  PRINTB (FIV_FMT, WRD_4, PHR_1, WRD_12, FNC_10, WRD_19);
13158 ;          6429                  PRINTB (FMT_9, .ARR_SEL);
13159 ;          6430                  end;
13160 ;          6431
```

```
13162 ;ML4AD
13163 ;                    TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
13164                                                           29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (72)
13165 ;          6432                ENDSUB;
13166 ;          6433                end;
13167 ;          6434
13168 ;          6435      end;
13169 ;          6436
13170 ;          6437  ENDTST;
13174
13175                                          .SBTTL  $T20 TEST CODE SECTION
13179 046444  004167  136104          $T20:   JSR     R1,$SAVE3
13180 046450  152777  000040  146766          BISB    #40,@ML.REG+40          ;                       6326
13181 046456  016703  147350                  MOV     ML.DUT,R3               ;                       6330
13182 046462  042703  177770                  BIC     #177770,R3
13183 046466  142777  000007  146750          BICB    #7,@ML.REG+40
13184 046474  150377  146744                  BISB    R3,@ML.REG+40
13185 046500  005077  146730                  CLR     @ML.REG+30
13186 046504  012777  000031  146672          MOV     #31,@ML.REG             ;                       6357
13187 046512  132777  000001  146664          BITB    #1,@ML.REG              ;                       6358
13188 046520  001426                          BEQ     1$                      ;                       6360
13189 046522  104455                          TRAP    55
13190 046524  000070                          .WORD   70                      ;                       6363
13191 046526  012706                          .WORD   ASYNC
13192 046530  026302                          .WORD   DUMPER
13193 046532  012746  010730                  MOV     #WRD.19,-(SP)           ;                       6364
13194 046536  012746  012272                  MOV     #FNC.10,-(SP)
13195 046542  012746  010640                  MOV     #WRD.11,-(SP)
13196 046546  012746  011626                  MOV     #PHR.2,-(SP)
13197 046552  012746  010520                  MOV     #WRD.1,-(SP)
13198 046556  012746  010304                  MOV     #FIV.FMT,-(SP)
13199 046562  012746  000006                  MOV     #6,-(SP)
13200 046566  010600                          MOV     SP,R0                   ; SP,*
13201 046570  104414                          TRAP    14
13202 046572  062706  000016                  ADD     #16,SP
13203 046576  132777  000001  146660  1$:     BITB    #1,@ML.REG+60           ;                       6362
13204 046604  001430                          BEQ     2$                      ;                       6367
13205 046606  104455                          TRAP    55
13206 046610  000071                          .WORD   71                      ;                       6370
13207 046612  012706                          .WORD   ASYNC
13208 046614  026302                          .WORD   DUMPER
13209 046616  012746  010730                  MOV     #WRD.19,-(SP)           ;                       6371
13210 046622  012746  012272                  MOV     #FNC.10,-(SP)
13211 046626  012746  012302                  MOV     #FNC.11,-(SP)
13212 046632  012746  010640                  MOV     #WRD.11,-(SP)
13213 046636  012746  011714                  MOV     #PHR.5,-(SP)
13214 046642  012746  010536                  MOV     #WRD.3,-(SP)
13215 046646  012746  010304                  MOV     #FIV.FMT,-(SP)
13215 046652  012746  000007                  MOV     #7,-(SP)
```

SEQ 0290

```
13218                                   ;ML4AD
13219                                   ;                       TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
13220                                                                                                               29-Mar-1982 16:21:03    PA:<
13221 046656  010600                                     MOV     SP,R0                   ; SP,*
13222 046660  104414                                     TRAP    14
13223 046662  062706  000020                             ADD     #20,SP
13224 046666  032777  100000  146560  2$:                BIT     #100000,@ML.REG+50      ;                               6369
13225 046674  001027                                     BNE     3$                      ;                               6374
13226 046676  104455                                     TRAP    55
13227 046700  000073                                     .WORD   73                      ;                               6377
13228 046702  012706                                     .WORD   ASYNC
13229 046704  026302                                     .WORD   DUMPER
13230 046706  012746  010730                             MOV     #WRD.19,-(SP)
13231 046712  012746  012272                             MOV     #FNC.10,-(SP)           ;                               6378
13232 046716  012746  010650                             MOV     #WRD.12,-(SP)
13233 046722  012746  011610                             MOV     #PHR.1,-(SP)
13234 046726  012746  010704                             MOV     #WRD.16,-(SP)
13235 046732  012746  010304                             MOV     #FIV.FMT,-(SP)
13236 046736  012746  000006                             MOV     #6,-(SP)
13237 046742  010600                                     MOV     SP,R0
13238 046744  104414                                     TRAP    14                      ; SP,*
13239 046746  062706  000016                             ADD     #16,SP
13240 046752  000207                                     RTS     PC                      ;                               6376
13241 046754  005067  146416          3$:                CLR     P.CNT                   ;                               6374
13242 046760  016702  145034                             MOV     LST.ARR,R2              ;                               6381
13243 046764  016703  145014                             MOV     ARR.INC,R3              ;                               6388
13244 046770  005001                                     CLR     R1
13245 046772  000475                                     BR      7$                      ; ARR.SEL
13246 046774  104402                          4$:        TRAP    2
13247 046776  152777  000040  146440                     BISB    #40,@ML.REG+40          ;                               6389
13248 047004  016700  147022                             MOV     ML.DUT,R0               ;                               6390
13249 047010  042700  177770                             BIC     #177770,R0
13250 047014  142777  000007  146422                     BICB    #7,@ML.REG+40
13251 047022  150077  146416                             BISB    R0,@ML.REG+40
13252 047026  010177  146402                             MOV     R1,@ML.REG+30           ; ARR.SEL,*
13253 047032  012777  000031  146344                     MOV     #31,@ML.REG             ;                               6392
13254 047040  032777  020000  146416                     BIT     #20000,@ML.REG+60       ;                               6393
13255 047046  001443                                     BEQ     5$                      ;                               6395
13256 047050  005267  146322                             INC     P.CNT
13257 047054  026767  146316  146316                     CMP     P.CNT,LIMIT             ;                               6397
13258 047062  003040                                     BGT     6$
13259 047064  104455                                     TRAP    55
13260 047066  000074                                     .WORD   74                      ;                               6399
13261 047070  012706                                     .WORD   ASYNC
13262 047072  026302                                     .WORD   DUMPER
13263 047074  012746  010730                             MOV     #WRD.19,-(SP)
13264 047100  012746  012272                             MOV     #FNC.10,-(SP)           ;                               6400
13265 047104  012746  010650                             MOV     #WRD.12,-(SP)
13266 047110  012746  011714                             MOV     #PHR.5,-(SP)
13267 047114  012746  010544                             MOV     #WRD.4,-(SP)
13268 047120  012746  010304                             MOV     #FIV.FMT,-(SP)
13269 047124  012746  000006                             MOV     #6,-(SP)
13270 047130  010600                                     MOV     SP,R0
13271 047132  104414                                     TRAP    14                      ; SP,*
13272 047134  010116                                     MOV     R1,(SP)                 ; ARR.SEL,*                     6401
```

SEQ 0291


```
13274                                   ;ML4AD
13275                                   ;
13276                                   ;       TEST CODE SECTION
13277 047136  012746  007064                     MOV     #FMT.9,-(SP)
13278 047142  012746  000002                     MOV     #2,-(SP)
13279 047146  010600                             MOV     SP,R0                   ; SP,*
13280 047150  104414                             TRAP    14
13281 047152  062706  000022                     ADD     #22,SP
13282 047156  104467                     5$:     TRAP    67                      ;                       6397
13283 047160  006000                             ROR     R0                      ;                       6402
13284 047162  103704                             BLO     4$
13285 047164  060301                     6$:     ADD     R3,R1                   ; *,ARR.SEL
13286 047166  020102                     7$:     CMP     R1,R2                   ; ARR.SEL,*             6388
13287 047170  101701                             BLOS    4$
13288 047172  005067  146200                     CLR     P.CNT
13289 047176  026727  144600  000017             CMP     OP.NUM.ARR,#17          ;                       6405
13290 047204  002110                             BGE     12$                     ;                       6409
13291 047206  016701  144606                     MOV     LST.ARR,R1              ;                       6415
13292 047212  066701  144566                     ADD     ARR.INC,R1
13293 047216  016703  144574                     MOV     ARR.16,R3
13294 047222  016702  144556                     MOV     ARR.INC,R2
13295 047226  000475                             BR      11$
13296 047230  104402                     8$:     TRAP    2
13297 047232  152777  000040  146204             BISB    #40,@ML.REG+40          ;                       6417
13298 047240  016700  146566                     MOV     ML.DUT,R0               ;                       6418
13299 047244  042700  177770                     BIC     #177770,R0
13300 047250  142777  000007  146166             BICB    #7,@ML.REG+40
13301 047256  150077  146162                     BISB    R0,@ML.REG+40
13302 047262  010177  146146                     MOV     R1,@ML.REG+30           ; ARR.SEL,*
13303 047266  012777  000031  146110             MOV     #31,@ML.REG             ;                       6420
13304 047274  032777  020000  146162             BIT     #20000,@ML.REG+60       ;                       6421
13305 047302  001043                             BNE     9$                      ;                       6423
13306 047304  005267  146066                     INC     P.CNT
13307 047310  026767  146062  146062             CMP     P.CNT,LIMIT             ;                       6425
13308 047316  003040                             BGT     10$
13309 047320  104455                             TRAP    55
13310 047322  000075                             .WORD   75                      ;                       6427
13311 047324  012706                             .WORD   ASYNC
13312 047326  026302                             .WORD   DUMPER
13313 047330  012746  010730                     MOV     #WRD.19,-(SP)           ;                       6428
13314 047334  012746  012272                     MOV     #FNC.10,-(SP)
13315 047340  012746  010650                     MOV     #WRD.12,-(SP)
13316 047344  012746  011610                     MOV     #PHR.1,-(SP)
13317 047350  012746  010544                     MOV     #WRD.4,-(SP)
13318 047354  012746  010304                     MOV     #FIV.FMT,-(SP)
13319 047360  012746  000006                     MOV     #6,-(SP)
13320 047364  010600                             MOV     SP,R0                   ; SP,*
13321 047366  104414                             TRAP    14
13322 047370  010116                             MOV     R1,(SP)                 ; ARR.SEL,*             6429
13323 047372  012746  007064                     MOV     #FMT.9,-(SP)
13324 047376  012746  000002                     MOV     #2,-(SP)
13325 047402  010600                             MOV     SP,R0                   ; SP,*
13326 047404  104414                             TRAP    14
13327 047406  062706  000022                     ADD     #22,SP
13328 047412  104467                     9$:     TRAP    67                      ;                       6425
                                                                                 ;                       6430
```

```
13330                                   ;ML4AD
13331                                   ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
13332                                                                                                       29-Mar-1982 16:21:03    PA:<
13333 047414  006000                            ROR     R0
13334 047416  103704                            BLO     8$
13335 047420  060201                    10$:    ADD     R2,R1                   ; *,ARR.SEL                                     6415
13336 047422  020103                    11$:    CMP     R1,R3                   ; ARR.SEL,*
13337 047424  101701                            BLOS    8$
13338 047426  000207                    12$:    RTS     PC                      ;                                               6326
13339
13340                                   ; Routine Size:  250 words
13341                                   ; Maximum stack depth per invocation:  13 words
13346
13347
13351
13352                                           .SBTTL  T20 TEST CODE SECTION
13356 047430                            T20::
13357 047430  004767  177010            1$:     JSR     PC,$T20                 ;                                               6435
13358 047434  104466                            TRAP    66
13359 047436  006000                            ROR     R0
13360 047440  103773                            BLO     1$
13361 047442  000207                            RTS     PC
13362
13363                                   ; Routine Size:  6 words
13364                                   ; Maximum stack depth per invocation:  0 words
13369
13370
13371 ;         6438  !<BLF/PAGE>
```

SEQ 0293


```
13373 ;ML4AD
13374 ;                 TEST CODE SECTION
13375 ;
13376 ;          6439   !
13377 ;          6440   BGNTST;
13378 ;          6441
13379 ;          6442   !++
13380 ;          6443   ! TEST NUMBER:  TST 21
13381 ;          6444   !
13382 ;          6445   ! TEST NAME:    POWER FAIL TEST
13383 ;          6446   !
13384 ;          6447   ! TEST DESCRIPTION:
13385 ;          6448   !       THIS TEST IS PART OF THE MANUAL INTERVENTION
13386 ;          6449   !       TEST SECTION.  DRIVE CONDITIONS REVOLVING
13387 ;          6450   !       AROUND POWER FAIL SITUATION ARE TEST FOR.
13388 ;          6451   !       FUNCTION TESTED FOR ARE AS FOLLOWS:
13389 ;          6452   !
13390 ;          6453   !               1.  VV BIT SETTING AND CLEARING
13391 ;          6454   !
13392 ;          6455   !               2.  READ-IN-PRESET FUNCTION SETTING
13393 ;          6456   !                   THE VV BIT.
13394 ;          6457   !
13395 ;          6458   !               3.  ECC INITIALIZE CLEARING OUT MEMORY
13396 ;          6459   !                   WITH BAD BATTERY BACK-UP.
13397 ;          6460   !
13398 ;          6461   !               4.  DRIVE REGISTERS BEING CLEARED BY
13399 ;          6462   !                   DC LO via 'MB_DIS LO'.
13400 ;          6463   !
13401 ;          6464   !               5.  UNSAFE BIT 'UNS' SETTING DURING AC
13402 ;          6465   !                   POWER LOSS.
13403 ;          6466   !--
13404 ;          6467
13405 ;          6468   local
13406 ;          6469       SIZE,                           !STORES THE SIZE OF MASS BUS TRANSFERS
13407 ;          6470       DST,                            !STORES THE DESTINATION ADRS OF MASS BUS TRANSFERS
13408 ;          6471       SRC,                            !STORES THE SOURCE ADRS OF MASS BUS TRANSFERS
13409 ;          6472       DSA_CNT,                        !COUNTS THE NUMBER OF BLOCK TRANSFERS DONE
13410 ;          6473       ERR_FLG,                        !ERROR FLAG
13411 ;          6474       CLR_DATA,                       !STORES THE EXPECTED CLEARED DATA FROM THE REGISTERS
13412 ;          6475       index,                  !INDEX VALUE INTO ML_REG STRUCTURE TO ACCESS THE REGISTERS ADRS
13413 ;          6476       WCE_CNT;                        !COUNT THE NUMBER OF WRITE CHECK ERROR DETECTED
13414 ;          6477
13415 ;          6478   !+
13416 ;          6479   ! PERFORM THIS TEST ONLY IF THE OPERATOR HAS
13417 ;          6480   ! ENABLED MANUAL INTERVENTION TEST VIA THE S/W
13418 ;          6481   ! QUESTIONS.
13419 ;          6482   !-
13420 ;          6483
13421 ;          6484   if .INTERVEN                           !DO THIS TEST IF INTERVEN IS SET
13422 ;          6485   then
13423 ;          6486       begin
13424 ;          6487       !
13425 ;          6488       !
13426 ;          6489       ! PRINT WHICH TEST THIS IS
13427 ;          6490       !
```

```
13429 ;ML4AD
13430 ;                    TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
13431 ;                                                         29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (73)
13432 ;          6491  !
13433 ;          6492      PRINTB (ONE_FMT, T_21);                   !PRINT T_21 MESSAGE
13434 ;          6493      BGNSUB;                                   !START SCOOP LOOP
13435 ;          6494  !
13436 ;          6495  !
13437 ;          6496  ! FIRST TEST THE VV BIT FOR SETTING.
13438 ;          6497  ! TO DO THIS FIRST SEE IF THE BIT IS
13439 ;          6498  ! ALREADY SET.  IF ALREADY SET THEN
13440 ;          6499  ! SEE IF THE READ-IN-PRESET FUNCTION
13441 ;          6500  ! CLEARS THE BIT.  IF NOT ALREADY SET
13442 ;          6501  ! THEN SEE IF THE READ-IN-PRESET FUNC
13443 ;          6502  ! SETS THE BIT.
13444 ;          6503  !
13445 ;          6504  !
13446 ;          6505      CLR_MBUS;                                 !CLEAR THE MASS BUS
13447 ;          6506
13448 ;          6507      if .VV                                    !IS THE VV BIT INITIALLY SET
13449 ;          6508      then                                      !THE VV BIT IS SET
13450 ;          6509          begin                                 !SEE IF THE READ-IN-PRESET FUNC CLEARS IT
13451 ;          6510          MLCS1 = RD_IN_PRE;                    !DO A READ IN PRESET FUNCTION
13452 ;          6511
13453 ;          6512          if not (.VV)                          !DID THE VV GET CLEARED
13454 ;          6513          then
13455 ;          6514              begin                             !THE VV GOT CLEARED
13456 ;          6515              ERRDF (149, ASYNC, DUMPER);       !REPORT THE ERROR TO THE OPERATOR
13457 ;          6516              PRINTB (ONE_FMT, VV_CLEAR);       !TELL WHAT THE ERROR IS
13458 ;          6517              end
13459 ;          6518
13460 ;          6519          end
13461 ;          6520      else                                      !THE VV BIT IS INITIALLY CLEARED
13462 ;          6521          begin                                 !SEE IF A READ-IN-PRESET FUNC SETS THE VV BIT
13463 ;          6522          MLCS1 = RD_IN_PRE;                    !DO A READ-IN-PRESET FUNCTION
13464 ;          6523
13465 ;          6524          if not (.VV)                          !DID THE VV BIT GET SET
13466 ;          6525          then
13467 ;          6526              begin                             !THE VV BIT DID NOT SET
13468 ;          6527              ERRDF (150, ASYNC, DUMPER);       !REPORT THE ERROR TO OPERATOR
13469 ;          6528              PRINTB (TWO_FMT, VV_NOT_SET, FNC_11);     !TELL WHAT THE ERROR IS
13470 ;          6529              end;
13471 ;          6530
13472 ;          6531          end;
13473 ;          6532
13474 ;          6533  !
13475 ;          6534  !
13476 ;          6535  ! NOW TEST DRIVE ERROR CONDITIONS TO SEE
13477 ;          6536  ! IF DOING A READ-IN-PRESET FUNCTION CAUSED
13478 ;          6537  ! ANY UNEXPECTED DRIVE ERRORS.
13479 ;          6538  !
13480 ;          6539  !
13481 ;          6540
13482 ;          6541      if .GO IS_SET                             !DID THE RD-IN-PRE FUNC HANG THE GO BIT
13483 ;          6542      then
```

```
13485 ;ML4AD
13486 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
13487                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (73)
13488 ;          6543          begin                                       !THE GO BIT IS HUNG
13489 ;          6544          ERRDF (62, ASYNC, DUMPER);                  !REPORT THE ERROR TO THE OPERATOR
13490 ;          6545          PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_11, WRD_19); !TELL WHAT THE ERROR IS
13491 ;          6546          end;
13492 ;          6547
13493 ;          6548      if .ILF IS_SET                                  !DID THE RD-IN-PRE FUNC CAUSE A ILLEGAL FUNC ERROR
13494 ;          6549      then
13495 ;          6550          begin                                       !THE ILF BIT IS SET
13496 ;          6551          ERRDF (63, ASYNC, DUMPER);                  !REPORT THE ERROR TO THE OPERATOR
13497 ;          6552          PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_11, FNC_11, WRD_19); !TELL WHAT THE ERROR IS
13498 ;          6553          end;
13499 ;          6554
13500 ;          6555      if .OPI IS_SET                                  !DID THE RD-IN-PRE CAUSE A OPERATION INCOMPLETE ERROR
13501 ;          6556      then
13502 ;          6557          begin                                       !THE OPI BIT IS SET
13503 ;          6558          ERRDF (64, ASYNC, DUMPER);                  !REPORT THE ERROR TO THE OPERATOR
13504 ;          6559          PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_11, FNC_11, WRD_19); !TELL WHAT THE ERROR
13505 ;          6560          end;
13506 ;          6561
13507 ;          6562      ENDSUB;                                         !END SCOOP LOOP
13508 ;          6563  !
13509 ;          6564  !
13510 ;          6565  ! NOW WE'LL TEST THE VV BIT FOR CLEARING
13511 ;          6566  ! AFTER POWER UP WITH BAD BATTERY BACK-UP.
13512 ;          6567  ! ECC INITIALIZE CLEARING MEMORY TO ZEROES
13513 ;          6568  ! AND DC LO CREATING 'MB_DIS_LO' AND CLEARING
13514 ;          6569  ! OUT DRIVE REGISTERS.
13515 ;          6570  !
13516 ;          6571  !
13517 ;          6572      BGNSUB;                                         !START SCOOP LOOP
13518 ;          6573      IO_BUF = ONES;                                  !LOAD THE FIRST IO_BUF WORD WITH ONES DATA
13519 ;          6574      DSA_CNT = -1;                                   !RESET THE DSA COUNTER
13520 ;          6575  !
13521 ;          6576  !
13522 ;          6577  ! WRITE A BACK GROUND OF ALL ONES TO
13523 ;          6578  ! ALL OF MEMORY.  THIS WILL PROVE IF
13524 ;          6579  ! ECC INIT TOOK PLACE AFTER POWER UP
13525 ;          6580  ! WITH BAD BATTERY BACK-UP.
13526 ;          6581  !
13527 ;          6582  !
13528 ;          6583
13529 ;          6584      do                                              !LOAD ALL OF MEMORY WITH ONES DATA
13530 ;          6585          begin
13531 ;          6586          BREAK;                                      !LOOK FOR A CONTROL C
13532 ;          6587          DSA_CNT = .DSA_CNT + 1;                     !UP THE DSA COUNTER
13533 ;          6588          WRT_TRANSFER (SIZE = -256, DST = .DSA_CNT, SRC = IO_BUF);       !DO A BLOCK WRITE TRANSFER
13534 ;          6589          end
13535 ;          6590  !
13536 ;          6591  !VER CZMLAD CHANGED TEST TO UNSIGNED TEST
13537 ;          6592  !
13538 ;          6593      until .DSA_CNT eqlU .LST_BLK;                   !REPEAT WRITING UNTIL THE LAST BLOCK IS WRITTEN
13539 ;          6594
```

SEQ 0296


```
13541 ;ML4AD
13542 ;                 TEST CODE SECTION
13543 ;
13544 ;          6595       ERR_FLG = ONE;                          !CLEAR THE ERROR FLAG
13545 ;          6596       CLR_MBUS;                               !CLEAR THE MASS BUS
13546 ;          6597   !
13547 ;          6598   !
13548 ;          6599   ! TELL THE OPERATOR THE SHUT OFF BOX
13549 ;          6600   ! POWER AND AS THE AC POWER IS GOING
13550 ;          6601   ! DOWN TEST 'UNS BIT' FOR SETTING.
13551 ;          6602   !
13552 ;          6603   !
13553 ;          6604       PRINTB (ONE_FMT, PWR_OFF);              !TELL THE OPERATOR TO TURN BOX POWER OFF
13554 ;          6605
13555 ;          6606       do                                      !SAMPLE THE UNS BIT UNTIL THE DRIVE TIMES OUT
13556 ;          6607           begin
13557 ;          6608           BREAK;                              !LOOK FOR A CONTROL C
13558 ;          6609
13559 ;          6610           if .UNS                             !IS THE UNS BIT SET YET
13560 ;          6611           then
13561 ;          6612               begin                           !IT'S SET
13562 ;          6613               ERR_FLG = ZERO;                 !THE BIT IS NOT IN ERROR SO CLEAR THE ERROR FLAG
13563 ;          6614               exitloop;                       !EXIT THE LOOP
13564 ;          6615               end;
13565 ;          6616
13566 ;          6617           end
13567 ;          6618       until .NED;                             !REPEAT SAMPLING UNS UNTIL NED IS SET
13568 ;          6619
13569 ;          6620       if .ERR_FLG                             !IS THE UNS BIT IN ERROR
13570 ;          6621       then
13571 ;          6622           begin                               !THE BIT IS IN ERROR
13572 ;          6623           ERRDF (151, ASYNC, DUMPER);         !REPORT THE ERROR TO THE OPERATOR
13573 ;          6624           PRINTB (ONE_FMT, UNS_ERR);          !TELL WHAT THE ERROR IS
13574 ;          6625           end;
13575 ;          6626
13576 ;          6627   !
13577 ;          6628   !
13578 ;          6629   ! NOW TELL THE OPERATOR TO TURN
13579 ;          6630   ! THE BOX POWER BACK ON.  ONCE THE
13580 ;          6631   ! DRIVE IS READY TEST THE VV BIT AND
13581 ;          6632   ! SEE IF THE POWER UP WITH BAD BATTERY
13582 ;          6633   ! BACK-UP CLEAR THE VV BIT, CLEARED
13583 ;          6634   ! MEMORY AND CLEARED THE DRIVE REGISTERS.
13584 ;          6635   !
13585 ;          6636   !
13586 ;          6637       PRINTB (ONE_FMT, PWR_ON);               !TELL THE OPERATOR TO TURN BOX POWER BACK ON
13587 ;          6638
13588 ;          6639       do                                      !DO NOTHING
13589 ;          6640           begin
13590 ;          6641           BREAK;                              !LOOK FOR CONTROL C
13591 ;          6642           end
13592 ;          6643       until .DRY;                             !UNTIL THE DRIVE IS READY
13593 ;          6644
13594 ;          6645       if .VV                                  !DID THE POWER FAIL WITH BAD BAT BACK-UP CLEAR THE VV BIT
13595 ;          6646       then
```

```
;ML4AD
;                    TEST CODE SECTION                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (73)

;          6647          begin                                  !THE VV BIT DIDN'T GET CLEARED
;          6648          ERRDF (152, ASYNC, DUMPER);            !REPORT THE ERROR TO THE OPERATOR
;          6649          PRINTB (ONE_FMT, VV_SET);              !TELL WHAT THE ERROR IS
;          6650          end;
;          6651
;          6652      incr REG_SEL from 0 to 6 do                !SEE IF DC LO ON POWER UP CLEARED THE REGISTERS
;          6653          begin
;          6654
;          6655          selectone .REG_SEL of                  !SELECT THIS REGISTERS INDEX VALUE
;          6656              set
;          6657
;          6658              [0] :
;          6659                  index = 0;                     !TEST MLCS1
;          6660
;          6661              [1] :
;          6662                  index = 6;                     !TEST MLER
;          6663
;          6664              [2] :
;          6665                  index = 3;                     !TEST MLDA
;          6666
;          6667              [3] :
;          6668                  index = 10;                    !TEST MLMR
;          6669
;          6670              [4] :
;          6671                  index = 13;                    !TEST MLE1
;          6672
;          6673              [5] :
;          6674                  index = 14;                    !TEST MLE2
;          6675
;          6676              [6] :
;          6677                  index = 8;                     !TEST MLPA
;          6678              tes;
;          6679
;          6680          CLR_DATA = (.HI) or (.IGNORE);         !GENERATE THIS REGISTERS CLEARED DATA
;          6681          RD_REG (.CLR_DATA, .REG_SEL, ERR_FLG); !READ THE REG FOR ITS CLEARD DATA
;          6682
;          6683          if .ERR_FLG                            !DID THE REGISTER GET CLEARED ON POWER UP
;          6684          then
;          6685              begin                              !THE REGISTER'S NOT CLEARED
;          6686              ERRDF (153, ASYNC, DUMPER);        !REPORT THE ERROR TO THE OPERATOR
;          6687              PRINTB (ONE_FMT, MB_DIS_ERR);      !TELL WHAT THE ERROR IS
;          6688              PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .CLR_DATA, .RD_DATA);
;          6689              exitloop;                          !EXIT THE LOOP
;          6690              end;
;          6691
;          6692          end;
;          6693
;          6694      WCE_CNT = ZERO;                            !CLEAR THE WRITE CHECK ERROR COUNTER
;          6695      DSA_CNT = -1;                              !RESET THE DSA COUNTER
;          6696      IO_BUF = ZEROES;                           !LOAD THE FIRST IO_BUF WORD WITH ZEROES DATA
;          6697
;          6698      do                                         !SEARCH ALL OF MEMORY FOR INIT'ED DATA
```

```
 13653 ;ML4AD
 13654 ;                 TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 13655 ;                                                                  29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (73)
 13656 ;          6699          begin
 13657 ;          6700          BREAK;                                   !LOOK FOR CONTROL C
 13658 ;          6701          DSA_CNT = .DSA_CNT + 1;                  !UP THE DSA COUNT
 13659 ;          6702          WRT_CHK_TRANSFER (SIZE = -256, DST = .DSA_CNT, SRC = IO_BUF);   !WRITE CHECK THIS BLOCK
 13660 ;          6703
 13661 ;          6704          if .WCE then WCE_CNT = .WCE_CNT + 1;     !IS THIS BLOCK INIT'ED UP COUNT IF NOT
 13662 ;          6705
 13663 ;          6706          end
 13664 ;          6707  !
 13665 ;          6708  !VER CZMLAD CHANGED TEST TO UNSIGNED TEST
 13666 ;          6709  !
 13667 ;          6710      until .DSA_CNT eqlU .LST_BLK;                !CONTINUE WRITE CHECKING UNTIL THE LAST BLOCK IS DONE
 13668 ;          6711
 13669 ;          6712      if .WCE_CNT gtr 127                          !WAS THERE MORE THAN 127 SECTORS IN ERROR
 13670 ;          6713      then
 13671 ;          6714          begin                                    !MORE THAN 127 SECTORS IN ERROR THERE FOR INIT ERROR
 13672 ;          6715          ERRDF (65, ASYNC, DUMPER);               !REPORT THE ERROR TO THE OPERATOR
 13673 ;          6716          PRINTB (ONE_FMT, NO_INIT);               !TELL WHAT THE ERROR IS
 13674 ;          6717          end;
 13675 ;          6718
 13676 ;          6719      ENDSUB;                                      !END SCOOP LOOP
 13677 ;          6720      end;
 13678 ;          6721
 13679 ;          6722  ENDTST;                                          !END THE TEST
 13683
 13684                                          .SBTTL  ST21 TEST CODE SECTION
 13688 047444  004167  135142          ST21:   JSR     R1,$SAVE5                       ;                       6437
 13689 047450  162706  000012                  SUB     #12,SP
 13690 047454  032767  000001  132726        BIT     #1,INTERVEN                     ;                       6484
 13691 047462  001002                        BNE     1$
 13692 047464  000167  001500                JMP     30$
 13693 047470  012746  004716          1$:   MOV     #T.21,-(SP)
 13694 047474  012746  010240                MOV     #ONE.FMT,-(SP)                  ;                       6492
 13695 047500  012746  000002                MOV     #2,-(SP)
 13696 047504  010600                        MOV     SP,R0
 13697 047506  104414                        TRAP    14                              ; SP,*
 13698 047510  104402                  2$:   TRAP    2
 13699 047512  152777  000040  145724        BISB    #40,@ML.REG+40                  ;                       6493
 13700 047520  016705  146306                MOV     ML.DUT,R5
 13701 047524  042705  177770                BIC     #177770,R5
 13702 047530  142777  000007  145706        BICB    #7,@ML.REG+40
 13703 047536  150577  145702                BISB    R5,@ML.REG+40
 13704 047542  132777  000100  145704        BITB    #100,@ML.REG+50                 ;                       6507
 13705 047550  001426                        BEQ     3$
 13706 047552  012777  000021  145624        MOV     #21,@ML.REG                     ;                       6510
 13707 047560  132777  000100  145666        BITB    #100,@ML.REG+50                 ;                       6512
```

SEQ 0299

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS
                                                                                                         29-Mar-1982 16:21:03    PA:<
13712 047566  001046                                  BNE     4$
13713 047570  104455                                  TRAP    55
13714 047572  000225                                  .WORD   225                             ;                                       6515
13715 047574  012706                                  .WORD   ASYNC
13716 047576  026302                                  .WORD   DUMPER
13717 047600  012746  005414                         MOV     #VV.CLEAR,-(SP)
13718 047604  012746  010240                         MOV     #ONE.FMT,-(SP)                  ;                                       6516
13719 047610  012746  000002                         MOV     #2,-(SP)
13720 047614  010600                                  MOV     SP,R0
13721 047616  104414                                  TRAP    14                              ; SP,*
13722 047620  062706  000006                         ADD     #6,SP
13723 047624  000427                                  BR      4$                              ;                                       6514
13724 047626  012777  000021  145550  3$:             MOV     #21,@ML.REG                     ;                                       6507
13725 047634  132777  000100  145612                 BITB    #100,@ML.REG+50                 ;                                       6522
13726 047642  001020                                  BNE     4$                              ;                                       6524
13727 047644  104455                                  TRAP    55
13728 047646  000226                                  .WORD   226                             ;                                       6527
13729 047650  012706                                  .WORD   ASYNC
13730 047652  026302                                  .WORD   DUMPER
13731 047654  012746  012302                         MOV     #FNC.11,-(SP)
13732 047660  012746  005340                         MOV     #VV.NOT.SET,-(SP)               ;                                       6528
13733 047664  012746  010246                         MOV     #TWO.FMT,-(SP)
13734 047670  012746  000003                         MOV     #3,-(SP)
13735 047674  010600                                  MOV     SP,R0
13736 047676  104414                                  TRAP    14                              ; SP,*
13737 047700  062706  000010                         ADD     #10,SP
13738 047704  132777  000001  145472  4$:             BITB    #1,@ML.REG                      ;                                       6526
13739 047712  001426                                  BEQ     5$                              ;                                       6541
13740 047714  104455                                  TRAP    55
13741 047716  000076                                  .WORD   76                              ;                                       6544
13742 047720  012706                                  .WORD   ASYNC
13743 047722  026302                                  .WORD   DUMPER
13744 047724  012746  010730                         MOV     #WRD.19,-(SP)
13745 047730  012746  012302                         MOV     #FNC.11,-(SP)                   ;                                       6545
13746 047734  012746  010640                         MOV     #WRD.11,-(SP)
13747 047740  012746  011626                         MOV     #PHR.2,-(SP)
13748 047744  012746  010520                         MOV     #WRD.1,-(SP)
13749 047750  012746  010304                         MOV     #FIV.FMT,-(SP)
13750 047754  012746  000006                         MOV     #6,-(SP)
13751 047760  010600                                  MOV     SP,R0
13752 047762  104414                                  TRAP    14                              ; SP,*
13753 047764  062706  000016                         ADD     #16,SP
13754 047770  132777  000001  145466  5$:             BITB    #1,@ML.REG+60                   ;                                       6543
13755 047776  001426                                  BEQ     6$                              ;                                       6548
13756 050000  104455                                  TRAP    55
13757 050002  000077                                  .WORD   77                              ;                                       6551
13758 050004  012706                                  .WORD   ASYNC
13759 050006  026302                                  .WORD   DUMPER
13760 050010  012746  010730                         MOV     #WRD.19,-(SP)
13761 050014  012746  012302                         MOV     #FNC.11,-(SP)                   ;                                       6552
13762 050020  012746  010640                         MOV     #WRD.11,-(SP)
13763 050024  012746  011714                         MOV     #PHR.5,-(SP)
```

SEQ 0300

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
13765                                   ;ML4AD
13766                                   ;               TEST CODE SECTION
13767
13768 050030  012746  010536                   MOV     #WRD.3,-(SP)
13769 050034  012746  010304                   MOV     #FIV.FMT,-(SP)
13770 050040  012746  000006                   MOV     #6,-(SP)
13771 050044  010600                           MOV     SP,R0           ; SP,*
13772 050046  104414                           TRAP    14
13773 050050  062706  000016                   ADD     #16,SP          ;
13774 050054  032777  020000  145402  6$:      BIT     #20000,@ML.REG+60 ;                                     6550
13775 050062  001426                           BEQ     7$              ;                                       6555
13776 050064  104455                           TRAP    55              ;                                       6558
13777 050066  000100                           .WORD   100
13778 050070  012706                           .WORD   ASYNC
13779 050072  026302                           .WORD   DUMPER
13780 050074  012746  010730                   MOV     #WRD.19,-(SP)   ;                                       6559
13781 050100  012746  012302                   MOV     #FNC.11,-(SP)
13782 050104  012746  010640                   MOV     #WRD.11,-(SP)
13783 050110  012746  011714                   MOV     #PHR.5,-(SP)
13784 050114  012746  010544                   MOV     #WRD.4,-(SP)
13785 050120  012746  010304                   MOV     #FIV.FMT,-(SP)
13786 050124  012746  000006                   MOV     #6,-(SP)
13787 050130  010600                           MOV     SP,R0           ; SP,*
13788 050132  104414                           TRAP    14
13789 050134  062706  000016                   ADD     #16,SP          ;
13790 050140  104467                   7$:     TRAP    67              ;                                       6557
13791 050142  006000                           ROR     R0              ;                                       6560
13792 050144  103002                           BHIS    8$
13793 050146  000167  177336                   JMP     2$
13794 050152  104402                   8$:     TRAP    2
13795 050154  012767  177777  143640           MOV     #-1,IO.BUF      ;                                       6562
13796 050162  012703  177777                   MOV     #-1,R3          ;                                       6573
13797 050166  104422                   9$:     TRAP    22              ; *,DSA.CNT                             6574
13798 050170  005203                           INC     R3              ;                                       6585
13799 050172  012766  177400  000006           MOV     #-400,6(SP)     ; DSA.CNT                               6587
13800 050200  012746  177400                   MOV     #-400,-(SP)     ; *,SIZE                                6588
13801 050204  010366  000016                   MOV     R3,16(SP)       ; DSA.CNT,DST
13802 050210  010346                           MOV     R3,-(SP)        ; DSA.CNT,*
13803 050212  012701  014022                   MOV     #IO.BUF,R1      ; *,SRC
13804 050216  010146                           MOV     R1,-(SP)        ; SRC,*
13805 050220  004767  157202                   JSR     PC,WRT.TRANSFER
13806 050224  062706  000006                   ADD     #6,SP           ;
13807 050230  020367  143560                   CMP     R3,LST.BLK      ; DSA.CNT,*                             6585
13808 050234  001354                           BNE     9$              ;                                       6593
13809 050236  012766  000001  000016           MOV     #1,16(SP)       ; *,ERR.FLG                             6595
13810 050244  152777  000040  145172           BISB    #40,@ML.REG+40
13811 050252  016705  145554                   MOV     ML.DUT,R5
13812 050256  042705  177770                   BIC     #177770,R5
13813 050262  142777  000007  145154           BICB    #7,@ML.REG+40
13814 050270  150577  145150                   BISB    R5,@ML.REG+40
13815 050274  012746  005112                   MOV     #PWR.OFF,-(SP)  ;                                       6604
13816 050300  012746  010240                   MOV     #ONE.FMT,-(SP)
13817 050304  012746  000002                   MOV     #2,-(SP)
13818 050310  010600                           MOV     SP,R0           ; SP,*
13819 050312  104414                           TRAP    14
```

SEQ 0301

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
13821                                   ;ML4AD
13822                                   ;
13823                                   ;               TEST CODE SECTION
13824 050314  104422                    10$:    TRAP    22
13825 050316  032777  040000  145140              BIT     #40000,@ML.REG+60       ;                       6607
13826 050324  001403                              BEQ     11$                     ;                       6610
13827 050326  005066  000024                      CLR     24(SP)                  ; ERR.FLG               6613
13828 050332  000404                              BR      12$                     ;                       6614
13829 050334  032777  010000  145102  11$:        BIT     #10000,@ML.REG+40       ;                       6618
13830 050342  001764                              BEQ     10$
13831 050344  032766  000001  000024  12$:        BIT     #1,24(SP)               ; *,ERR.FLG             6620
13832 050352  001416                              BEQ     13$
13833 050354  104455                              TRAP    55                      ;                       6623
13834 050356  000227                              .WORD   227
13835 050360  012706                              .WORD   ASYNC
13836 050362  026302                              .WORD   DUMPER
13837 050364  012746  005272                      MOV     #UNS.ERR,-(SP)
13838 050370  012746  010240                      MOV     #ONE.FMT,-(SP)          ;                       6624
13839 050374  012746  000002                      MOV     #2,-(SP)
13840 050400  010600                              MOV     SP,R0
13841 050402  104414                              TRAP    14                      ; SP,*
13842 050404  062706  000006                      ADD     #6,SP
13843 050410  012716  005202            13$:      MOV     #PWR.ON,(SP)            ;                       6622
13844 050414  012746  010240                      MOV     #ONE.FMT,-(SP)          ;                       6637
13845 050420  012746  000002                      MOV     #2,-(SP)
13846 050424  010600                              MOV     SP,R0
13847 050426  104414                              TRAP    14                      ; SP,*
13848 050430  104422                    14$:      TRAP    22
13849 050432  105777  145016                      TSTB    @ML.REG+50              ;                       6640
13850 050436  100374                              BPL     14$                     ;                       6643
13851 050440  132777  000100  145006              BITB    #100,@ML.REG+50         ;                       6645
13852 050446  001416                              BEQ     15$
13853 050450  104455                              TRAP    55                      ;                       6648
13854 050452  000230                              .WORD   230
13855 050454  012706                              .WORD   ASYNC
13856 050456  026302                              .WORD   DUMPER
13857 050460  012746  005764                      MOV     #VV.SET,-(SP)
13858 050464  012746  010240                      MOV     #ONE.FMT,-(SP)          ;                       6649
13859 050470  012746  000002                      MOV     #2,-(SP)
13860 050474  010600                              MOV     SP,R0
13861 050476  104414                              TRAP    14                      ; SP,*
13862 050500  062706  000006                      ADD     #6,SP
13863 050504  005005                    15$:      CLR     R5                      ;                       6647
13864 050506  005705                    16$:      TST     R5                      ; REG.SEL               6652
13865 050510  001002                              BNE     17$                     ; REG.SEL               6655
13866 050512  005002                              CLR     R2
13867 050514  000443                              BR      23$                     ; INDEX                 6659
13868 050516  020527  000001            17$:      CMP     R5,#1                   ;                       6655
13869 050522  001003                              BNE     18$                     ; REG.SEL,*
13870 050524  012702  000006                      MOV     #6,R2
13871 050530  000435                              BR      23$                     ; *,INDEX               6662
13872 050532  020527  000002            18$:      CMP     R5,#2                   ;                       6655
13873 050536  001003                              BNE     19$                     ; REG.SEL,*
13874 050540  012702  000003                      MOV     #3,R2
13875 050544  000427                              BR      23$                     ; *,INDEX               6665
                                                                                  ;                       6655
```

SEQ 0302

```
13877                                   ;ML4AD
13878                                   ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04   TOPS
13879                                                                                                      29-Mar-1982 16:21:03   PA:<
13880 050546 020527 000003      19$:    CMP     R5,#3           ; REG.SEL,*
13881 050552 001003                     BNE     20$
13882 050554 012702 000012              MOV     #12,R2          ; *,INDEX                                                      6668
13883 050560 000421                     BR      23$             ;                                                              6655
13884 050562 020527 000004      20$:    CMP     R5,#4           ; REG.SEL,*
13885 050566 001003                     BNE     21$
13886 050570 012702 000015              MOV     #15,R2          ; *,INDEX                                                      6671
13887 050574 000413                     BR      23$             ;                                                              6655
13888 050576 020527 000005      21$:    CMP     R5,#5           ; REG.SEL,*
13889 050602 001003                     BNE     22$
13890 050604 012702 000016              MOV     #16,R2          ; *,INDEX                                                      6674
13891 050610 000405                     BR      23$             ;                                                              6655
13892 050612 020527 000006      22$:    CMP     R5,#6           ; REG.SEL,*
13893 050616 001002                     BNE     23$
13894 050620 012702 000010              MOV     #10,R2          ; *,INDEX
13895 050624 010200             23$:    MOV     R2,R0           ; INDEX,*                                                      6677
13896 050626 006300                     ASL     R0                                                                             6680
13897 050630 006300                     ASL     R0
13898 050632 006300                     ASL     R0
13899 050634 010004                     MOV     R0,R4
13900 050636 016400 015406              MOV     ML.REG+2(R4),R0
13901 050642 056400 015412              BIS     ML.REG+6(R4),R0
13902 050646 010066 000024              MOV     R0,24(SP)       ; *,CLR.DATA
13903 050652 010046                     MOV     R0,-(SP)        ; CLR.DATA,*
13904 050654 010546                     MOV     R5,-(SP)        ; REG.SEL,*                                                    6681
13905 050656 012746 000036              MOV     #36,-(SP)
13906 050662 060616                     ADD     SP,(SP)         ; ERR.FLG,*
13907 050664 004767 155022              JSR     PC,RD.REG
13908 050670 032766 000001 000036       BIT     #1,36(SP)       ; *,ERR.FLG                                                    6683
13909 050676 001433                     BEQ     24$
13910 050700 104455                     TRAP    55              ;                                                              6686
13911 050702 000231                     .WORD   231
13912 050704 012706                     .WORD   ASYNC
13913 050706 026302                     .WORD   DUMPER
13914 050710 012746 005570              MOV     #MB.DIS.ERR,-(SP)       ;                                                      6687
13915 050714 012746 010240              MOV     #ONE.FMT,-(SP)
13916 050720 012746 000002              MOV     #2,-(SP)
13917 050724 010600                     MOV     SP,R0           ; SP,*
13918 050726 104414                     TRAP    14
13919 050730 016716 144416              MOV     RD.DATA,(SP)    ;                                                              6688
13920 050734 016646 000040              MOV     40(SP),-(SP)    ; CLR.DATA,*
13921 050740 016446 015404              MOV     ML.REG(R4),-(SP)
13922 050744 012746 007436              MOV     #FMT.16,-(SP)
13923 050750 012746 000004              MOV     #4,-(SP)
13924 050754 010600                     MOV     SP,R0           ; SP,*
13925 050756 104414                     TRAP    14
13926 050760 062706 000024              ADD     #24,SP          ;                                                              6689
13927 050764 000406                     BR      25$
13928 050766 062706 000006      24$:    ADD     #6,SP           ;                                                              6653
13929 050772 005205                     INC     R5              ; REG.SEL                                                      6652
13930 050774 020527 000006              CMP     R5,#6           ; REG.SEL,*
13931 051000 003642                     BLE     16$
```

SEQ 0303

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
13933                                           ;ML4AD
13934                                           ;
13935                                           ;               TEST CODE SECTION
13936 051002  005066  000022          25$:      CLR     22(SP)                  ; WCE.CNT                       6694
13937 051006  012703  177777                    MOV     #-1,R3                  ; *,DSA.CNT                     6695
13938 051012  005067  143004                    CLR     IO.BUF                  ;                               6696
13939 051016  104422                  26$:      TRAP    22                      ;                               6699
13940 051020  005203                            INC     R3                      ; DSA.CNT                       6701
13941 051022  012766  177400  000020            MOV     #-400,20(SP)            ; *,SIZE                        6702
13942 051030  012746  177400                    MOV     #-400,-(SP)
13943 051034  010366  000030                    MOV     R3,30(SP)               ; DSA.CNT,DST
13944 051040  010346                            MOV     R3,-(SP)                ; DSA.CNT,*
13945 051042  012701  014022                    MOV     #IO.BUF,R1              ; *,SRC
13946 051046  010146                            MOV     R1,-(SP)                ; SRC,*
13947 051050  004767  156102                    JSR     PC,WRT.CHK.TRANSFE
13948 051054  032777  040000  144362            BIT     #40000,@ML.REG+40       ;                               6704
13949 051062  001402                            BEQ     27$
13950 051064  005266  000030                    INC     30(SP)                  ; WCE.CNT
13951 051070  062706  000006          27$:      ADD     #6,SP                   ;                               6699
13952 051074  020367  142714                    CMP     R3,LST.BLK              ; DSA.CNT,*                     6710
13953 051100  001346                            BNE     26$
13954 051102  026627  000022  000177            CMP     22(SP),#177             ; WCE.CNT,*                     6712
13955 051110  003416                            BLE     28$
13956 051112  104455                            TRAP    55                      ;                               6715
13957 051114  000101                            .WORD   101
13958 051116  012706                            .WORD   ASYNC
13959 051120  026302                            .WORD   DUMPER
13960 051122  012746  005470                    MOV     #NO.INIT,-(SP)          ;                               6716
13961 051126  012746  010240                    MOV     #ONE.FMT,-(SP)
13962 051132  012746  000002                    MOV     #2,-(SP)
13963 051136  010600                            MOV     SP,R0                   ; SP,*
13964 051140  104414                            TRAP    14
13965 051142  062706  000006                    ADD     #6,SP                   ;                               6714
13966 051146  062706  000012          28$:      ADD     #12,SP                  ;                               6562
13967 051152  104467                            TRAP    67                      ;                               6717
13968 051154  006000                            ROR     R0
13969 051156  103002                            BHIS    29$
13970 051160  000167  176766                    JMP     8$
13971 051164  062706  000006          29$:      ADD     #6,SP                   ;                               6486
13972 051170  062706  000012          30$:      ADD     #12,SP                  ;                               6437
13973 051174  000207                            RTS     PC
13974
13975                                           ; Routine Size:  429 words
13976                                           ; Maximum stack depth per invocation:  29 words
13981
13982
13986
13987                                                   .SBTTL  T21 TEST CODE SECTION
```

SEQ 0304

```
13989                                   ;ML4AD
13990                                   ;                       TEST CODE SECTION
13991                                   ;
13995 051176                            T21::
13996 051176  004767  176242            1$:     JSR     PC,$T21                 ;
13997 051202  104466                            TRAP    66
13998 051204  006000                            ROR     R0
13999 051206  103773                            BLO     1$
14000 051210  000207                            RTS     PC
14001
14002                                   ; Routine Size:  6 words
14003                                   ; Maximum stack depth per invocation:  0 words
14008
14009
14010 ;          6723  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

6720

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V4(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (74)

```
14012 ;ML4AD
14013 ;                 TEST CODE SECTION
14014
14015 ;          6724   !
14016 ;          6725   !
14017 ;          6726   BGNTST;
14018 ;          6727
14019 ;          6728   !++
14020 ;          6729   ! TEST NUMBER:  TST 22
14021 ;          6730   !
14022 ;          6731   ! TEST NAME:    ILLEGAL FUNCTION TEST
14023 ;          6732   !
14024 ;          6733   ! TEST DESCRIPTION:
14025 ;          6734   !
14026 ;          6735   !       TEST THE DETECTION OF ILLEGAL
14027 ;          6736   !       FUNCTIONS WRITTEN TO MLCS1
14028 ;          6737   !       BY:
14029 ;          6738   !
14030 ;          6739   !       WRITING ALL POSSIBLE ILLEGAL
14031 ;          6740   !       FUNCTIONS TO MLCS1.  THEN
14032 ;          6741   !       TEST GO AND ERROR BITS CLEARED.
14033 ;          6742   !
14034 ;          6743   !--
14035 ;          6744
14036 ;          6745   local
14037 ;          6746       BAD_BITS,                        !STORES A COUNT TO GENERATE BAD FUNCTIONS
14038 ;          6747       BAD_FUNC;                        !STORES GENERATED BAD FUNCTION
14039 ;          6748
14040 ;          6749   CLR_THRESHOLD;                       !CLEAR ERROR PRINT THRESHOLD
14041 ;          6750   BAD_BITS = ZEROES;                   !CLEAR BAD BITS
14042 ;          6751
14043 ;          6752   incr CNT_1 from 0 to 2 do            !REPEAT LOOP 3 TIMES
14044 ;          6753       begin
14045 ;          6754       BAD_BITS = .BAD_BITS + %o'2';    !ADD 2 TO BAD_BITS
14046 ;          6755
14047 ;          6756       incr CNT_2 from %o'1' to %o'71' by %o'10' do        !REPEAT LOOP GENERATING 'GOOD' FUNCTIONS
14048 ;          6757           begin
14049 ;          6758           BGNSUB;
14050 ;          6759           CLR_MBUS;
14051 ;          6760           BAD_FUNC = .CNT_2 + .BAD_BITS;   !ADD BAD_BITS TO CNT_2 GENERATING BAD FUNCTIONS
14052 ;          6761           FIRST_BLK_XFER ();               !SET UP A FIRST BLOCK XFERR
14053 ;          6762           MLCS1 = .BAD_FUNC;               !LOAD MLCS1 WITH TWO BAD FUNCTIONS
14054 ;          6763
14055 ;          6764           if .ILF IS_SET                   !SEE IF ILF IS SET
14056 ;          6765           then
14057 ;          6766               begin
14058 ;          6767
14059 ;          6768               if .GO IS_SET                !IF ILF IS SET THEN SEE IF GO IS SET
14060 ;          6769               then
14061 ;          6770                   begin
14062 ;          6771                   CMP_THRESHOLD;           !COMPARE ERROR PRINT THRESHOLD
14063 ;          6772                   ERRDF (66, ASYNC, DUMPER);   !ERROR IF GO SET WITH BAD FUNCTION
14064 ;          6773                   PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_12, WRD_19);
14065 ;          6774                   PRINTB (FMT_12, .BAD_FUNC);
14066 ;          6775                   end;
```

```
 14068 ;ML4AD
 14069 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 14070                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (74)
 14071 ;          6776
 14072 ;          6777                end
 14073 ;          6778            else
 14074 ;          6779                begin
 14075 ;          6780                CMP_THRESHOLD;                          !COMPARE ERROR PRINT THRESHOLD
 14076 ;          6781                ERRDF (67, ASYNC, DUMPER);              !ERROR BAD FUNCTION DON'T CAUSE ILF
 14077 ;          6782                PRINTB (FIV_FMT, WRD_3, PHR_1, WRD_11, FNC_12, WRD_19);
 14078 ;          6783                PRINTB (FMT_12, .BAD_FUNC);
 14079 ;          6784                end;
 14080 ;          6785
 14081 ;          6786            if .OPI IS_SET                              !SEE IF OPI IS SET
 14082 ;          6787            then
 14083 ;          6788                begin
 14084 ;          6789                CMP_THRESHOLD;                          !COMPARE ERROR PRINT THRESHOLD
 14085 ;          6790                ERRDF (68, ASYNC, DUMPER);              !ERROR IF SET
 14086 ;          6791                PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_12, WRD_19);
 14087 ;          6792                PRINTB (FMT_12, .BAD_FUNC);
 14088 ;          6793                end;
 14089 ;          6794
 14090 ;          6795            ENDSUB;
 14091 ;          6796            end;
 14092 ;          6797
 14093 ;          6798        end;
 14094 ;          6799
 14095 ;          6800    ENDTST;
 14099
 14100
                                .SBTTL  $T22 TEST CODE SECTION
 14104 051212  004167  133374  $T22:  JSR     R1,$SAVE5
 14105 051216  005067  144154          CLR     P.CNT                       ;                                   6722
 14106 051222  005004                  CLR     R4                          ;                                   6747
 14107 051224  005001                  CLR     R1                          ; BAD.BITS                          6750
 14108 051226  062704  000002  1$:     ADD     #2,R4                       ; CNT.1                             6752
 14109 051232  012702  000001          MOV     #1,R2                       ; *,BAD.BITS                        6754
 14110 051236  010405          2$:     MOV     R4,R5                       ; *,CNT.2                           6756
 14111 051240  060205                  ADD     R2,R5                       ; BAD.BITS,*                        6760
 14112 051242  104402          3$:     TRAP    2                           ; CNT.2,*
 14113 051244  152777  000040  144172  BISB    #40,@ML.REG+40              ;                                   6757
 14114 051252  016700  144554          MOV     ML.DUT,R0                   ;                                   6758
 14115 051256  042700  177770          BIC     #177770,R0
 14116 051262  142777  000007  144154  BICB    #7,@ML.REG+40
 14117 051270  150077  144150          BISB    R0,@ML.REG+40
 14118 051274  010503                  MOV     R5,R3
 14119 051276  004767  146760          JSR     PC,FIRST.BLK.XFER           ; *,BAD.FUNC                        6760
 14120 051302  010377  144076          MOV     R3,@ML.REG                  ;                                   6761
 14121 051306  132777  000001  144150  BITB    #1,@ML.REG+60               ; BAD.FUNC,*                        6762
 14122 051314  001446                  BEQ     4$                          ;                                   6764
```

SEQ 0307

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
14124                                   ;ML4AD
14125                                   ;               TEST CODE SECTION
14126
14127 051316  132777  000001  144060          BITB    #1,@ML.REG              ;                       6768
14128 051324  001505                          BEQ     6$
14129 051326  005267  144044                  INC     P.CNT                   ;                       6770
14130 051332  026767  144040  144040          CMP     P.CNT,LIMIT
14131 051340  003153                          BGT     8$
14132 051342  104455                          TRAP    55                      ;                       6772
14133 051344  000102                          .WORD   102
14134 051346  012706                          .WORD   ASYNC
14135 051350  026302                          .WORD   DUMPER
14136 051352  012746  010730                  MOV     #WRD.19,-(SP)           ;                       6773
14137 051356  012746  012322                  MOV     #FNC.12,-(SP)
14138 051362  012746  010640                  MOV     #WRD.11,-(SP)
14139 051366  012746  011626                  MOV     #PHR.2,-(SP)
14140 051372  012746  010520                  MOV     #WRD.1,-(SP)
14141 051376  012746  010304                  MOV     #FIV.FMT,-(SP)
14142 051402  012746  000006                  MOV     #6,-(SP)
14143 051406  010600                          MOV     SP,R0                   ; SP,*
14144 051410  104414                          TRAP    14
14145 051412  010316                          MOV     R3,(SP)                 ; BAD.FUNC,*            6774
14146 051414  012746  007224                  MOV     #FMT.12,-(SP)
14147 051420  012746  000002                  MOV     #2,-(SP)
14148 051424  010600                          MOV     SP,R0                   ; SP,*
14149 051426  104414                          TRAP    14
14150 051430  001441                          BR      5$                      ;                       6770
14151 051432  005267  143740          4$:     INC     P.CNT                   ;                       6779
14152 051436  026767  143734  143734          CMP     P.CNT,LIMIT
14153 051444  003111                          BGT     8$
14154 051446  104455                          TRAP    55                      ;                       6781
14155 051450  000103                          .WORD   103
14156 051452  012706                          .WORD   ASYNC
14157 051454  026302                          .WORD   DUMPER
14158 051456  012746  010730                  MOV     #WRD.19,-(SP)           ;                       6782
14159 051462  012746  012322                  MOV     #FNC.12,-(SP)
14160 051466  012746  010640                  MOV     #WRD.11,-(SP)
14161 051472  012746  011610                  MOV     #PHR.1,-(SP)
14162 051476  012746  010536                  MOV     #WRD.3,-(SP)
14163 051502  012746  010304                  MOV     #FIV.FMT,-(SP)
14164 051506  012746  000006                  MOV     #6,-(SP)
14165 051512  010600                          MOV     SP,R0                   ; SP,*
14166 051514  104414                          TRAP    14
14167 051516  010316                          MOV     R3,(SP)                 ; BAD.FUNC,*            6783
14168 051520  012746  007224                  MOV     #FMT.12,-(SP)
14169 051524  012746  000002                  MOV     #2,-(SP)
14170 051530  010600                          MOV     SP,R0                   ; SP,*
14171 051532  104414                          TRAP    14
14172 051534  062706  000022          5$:     ADD     #22,SP                  ;                       6779
14173 051540  032777  020000  143716  6$:     BIT     #20000,@ML.REG+60       ;                       6786
14174 051546  001443                          BEQ     7$
14175 051550  005267  143622                  INC     P.CNT                   ;                       6788
14176 051554  026767  143616  143616          CMP     P.CNT,LIMIT
14177 051562  003042                          BGT     8$
14178 051564  104455                          TRAP    55                      ;                       6790
```

```
14180                                   ;ML4AD
14181                                   ;               TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS
14182                                                                                                    29-Mar-1982 16:21:03    PA:<
14183 051566  000104                            .WORD   104
14184 051570  012706                            .WORD   ASYNC
14185 051572  026302                            .WORD   DUMPER
14186 051574  012746  010730                    MOV     #WRD.19,-(SP)           ;                                                6791
14187 051600  012746  012322                    MOV     #FNC.12,-(SP)
14188 051604  012746  010650                    MOV     #WRD.12,-(SP)
14189 051610  012746  011714                    MOV     #PHR.5,-(SP)
14190 051614  012746  010544                    MOV     #WRD.4,-(SP)
14191 051620  012746  010304                    MOV     #FIV.FMT,-(SP)
14192 051624  012746  000006                    MOV     #6,-(SP)
14193 051630  010600                            MOV     SP,R0                   ; SP,*
14194 051632  104414                            TRAP    14
14195 051634  010316                            MOV     R3,(SP)                 ; BAD.FUNC,*                                     6792
14196 051636  012746  007224                    MOV     #FMT.12,-(SP)
14197 051642  012746  000002                    MOV     #2,-(SP)
14198 051646  010600                            MOV     SP,R0                   ; SP,*
14199 051650  104414                            TRAP    14
14200 051652  062706  000022                    ADD     #22,SP                  ;                                                6788
14201 051656  104467                    7$:     TRAP    67                      ;                                                6793
14202 051660  006000                            ROR     R0
14203 051662  103002                            BHIS    8$
14204 051664  000167  177352                    JMP     3$
14205 051670  062702  000010            8$:     ADD     #10,R2                  ; *,CNT.2                                        6756
14206 051674  020227  000071                    CMP     R2,#71                  ; CNT.2,*
14207 051700  003002                            BGT     9$
14208 051702  000167  177330                    JMP     2$
14209 051706  005201                    9$:     INC     R1                      ; CNT.1                                          6752
14210 051710  020127  000002                    CMP     R1,#2                   ; CNT.1,*
14211 051714  003002                            BGT     10$
14212 051716  000167  177304                    JMP     1$
14213 051722  000207                    10$:    RTS     PC                      ;                                                6722
14214
14215                                   ; Routine Size:  165 words
14216                                   ; Maximum stack depth per invocation:  15 words
14221
14222
14226
14227                                           .SBTTL  T22 TEST CODE SECTION
14231 051724                            T22::
14232 051724  004767  177262            1$:     JSR     PC,$T22                 ;                                                6798
14233 051730  104466                            TRAP    66
14234 051732  006000                            ROR     R0
```

SEQ 0309


```
14236                                   ;ML4AD
14237                                   ;               TEST CODE SECTION
14238
14239 051734  103773                              BLO     1$
14240 051736  000207                              RTS     PC
14241
14242                                   ; Routine Size:  6 words
14243                                   ; Maximum stack depth per invocation:  0 words
14248
14249
14250 ;          6801   !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (75)

```
14252 ;ML4AD
14253 ;                 TEST CODE SECTION
14254 ;
14255 ;          6802  !
14256 ;          6803  !
14257 ;          6804  BGNTST;
14258 ;          6805
14259 ;          6806  !++
14260 ;          6807  ! TEST NUMBER:  TST 23
14261 ;          6808  !
14262 ;          6809  ! TEST NAME:    REGISTER MODIFICATION REFUSED TEST
14263 ;          6810  !
14264 ;          6811  ! TEST DESCRIPTION:
14265 ;          6812  !
14266 ;          6813  !       TEST THE DETECTION OF A
14267 ;          6814  !       REGISTER MODIFICATION REFUSED
14268 ;          6815  !       BY:
14269 ;          6816  !
14270 ;          6817  !       1.  WRITTING TO MLCS1, MLDA
14271 ;          6818  !           AND MLER WHILE THE DRIVE
14272 ;          6819  !           IS BUSY AND TEST RMR
14273 ;          6820  !           BIT SET.
14274 ;          6821  !
14275 ;          6822  !       ALSO SEE IF THE DRIVE ASSERTED
14276 ;          6823  !       EXCEPTION BY TESTING THE TRE BIT SET.
14277 ;          6824  !
14278 ;          6825  !--
14279 ;          6826
14280 ;          6827  incr CNT from 0 to 2 do                    !REPEAT LOOP 3 TIMES
14281 ;          6828      begin
14282 ;          6829      BGNSUB;
14283 ;          6830      CLR_MBUS;
14284 ;          6831      MLCS1 = write;                         !DO A WRITE FUNCTION
14285 ;          6832
14286 ;          6833      case .CNT from 0 to 2 of               !WRITE TO SELECTED REGISTERS FORCING RMR
14287 ;          6834          set
14288 ;          6835
14289 ;          6836          [0] :
14290 ;          6837              MLCS1 = %o'000000';
14291 ;          6838
14292 ;          6839          [1] :
14293 ;          6840              MLDA = ONES;
14294 ;          6841
14295 ;          6842          [2] :
14296 ;          6843              MLER = ONES
14297 ;          6844          tes;
14298 ;          6845
14299 ;          6846      DELAY (FRTY_US);
14300 ;          6847
14301 ;          6848      if .RMR IS_NOT_SET                     !SEE IF RMR GOT SET
14302 ;          6849      then
14303 ;          6850          begin
14304 ;          6851          ERRDF (69, ASYNC, DUMPER);         !ERROR IF NOT SET
14305 ;          6852          PRINTB (FOR_FMT, WRD_21, PHR_1, WRD_11, WRD_21);
14306 ;          6853          end;
```

```
 14308 ;ML4AD
 14309 ;                 TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 14310                                                                    29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (75)
 14311 ;          6854
 14312 ;          6855       if .TRE IS_NOT_SET                          !SEE IF DRIVE ASSERTED EXCEPTION BY TESTING TRE
 14313 ;          6856       then
 14314 ;          6857           begin
 14315 ;          6858           ERRDF (117, SYNC, DUMPER);
 14316 ;          6859           PRINTB (FOR_FMT, WRD_20, PHR_1, WRD_11, WRD_21);
 14317 ;          6860           end;
 14318 ;          6861
 14319 ;          6862       ENDSUB;
 14320 ;          6863       end;
 14321 ;          6864
 14322 ;          6865  ENDTST;
 14326
 14327                                          .SBTTL  $T23 TEST CODE SECTION
 14331 051740  004167  132574          $T23:   JSR     R1,$SAVE2                ;                              6800
 14332 051744  005746                          TST     -(SP)
 14333 051746  005002                          CLR     R2                       ; CNT                          6827
 14334 051750  104402                  1$:     TRAP    2                        ;                              6828
 14335 051752  152777  000040  143464          BISB    #40,@ML.REG+40           ;                              6829
 14336 051760  016701  144046                  MOV     ML.DUT,R1
 14337 051764  042701  177770                  BIC     #177770,R1
 14338 051770  142777  000007  143446          BICB    #7,@ML.REG+40
 14339 051776  150177  143442                  BISB    R1,@ML.REG+40
 14340 052002  012777  000061  143374          MOV     #61,@ML.REG              ;                              6831
 14341 052010  010201                          MOV     R2,R1                    ; CNT,*                        6833
 14342 052012  006301                          ASL     R1
 14343 052014  066107  052020                  ADD     2$(R1),PC
 14344 052020  000006                  2$:     .WORD   3$-2$
 14345 052022  000014                          .WORD   4$-2$
 14346 052024  000024                          .WORD   5$-2$
 14347 052026  005077  143352          3$:     CLR     @ML.REG                  ;                              6837
 14348 052032  000407                          BR      6$                       ;                              6833
 14349 052034  012777  177777  143372  4$:     MOV     #-1,@ML.REG+30           ;                              6840
 14350 052042  000403                          BR      6$                       ;                              6833
 14351 052044  012777  177777  143412  5$:     MOV     #-1,@ML.REG+60           ;                              6843
 14352 052052  012700  000050          6$:     MOV     #50,R0                   ; *,$$TMP2                     6846
 14353 052056  001410                  7$:     BEQ     10$
 14354 052060  016701  130032                  MOV     L$DLY,R1                 ; *,$$TMP1
 14355 052064  001403                          BEQ     9$
 14356 052066  005016                  8$:     CLR     (SP)                     ; $$TMP
 14357 052070  005301                          DEC     R1                       ; $$TMP1
 14358 052072  001375                          BNE     8$
 14359 052074  005300                  9$:     DEC     R0                       ; $$TMP2
 14360 052076  000767                          BR      7$
 14361 052100  132777  000004  143356  10$:    BITB    #4,@ML.REG+60            ;                              6848
 14362 052106  001024                          BNE     11$
```

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION                                29-Mar-1982 16:23:04  TOPS
                                                                                                         29-Mar-1982 16:21:03  PA:<
14364
14365
14366
14367 052110  104455                            TRAP    55
14368 052112  000105                            .WORD   105                     ;                                              6851
14369 052114  012706                            .WORD   ASYNC
14370 052116  026302                            .WORD   DUMPER
14371 052120  012746  010744                    MOV     #WRD.21,-(SP)
14372 052124  012746  010640                    MOV     #WRD.11,-(SP)           ;                                              6852
14373 052130  012746  011610                    MOV     #PHR.1,-(SP)
14374 052134  012746  010744                    MOV     #WRD.21,-(SP)
14375 052140  012746  010270                    MOV     #FOR.FMT,-(SP)
14376 052144  012746  000005                    MOV     #5,-(SP)
14377 052150  010600                            MOV     SP,R0                   ; SP,*
14378 052152  104414                            TRAP    14
14379 052154  062706  000014                    ADD     #14,SP
14380 052160  032777  040000  143216  11$:      BIT     #40000,@ML.REG          ;                                              6850
14381 052166  001024                            BNE     12$                     ;                                              6855
14382 052170  104455                            TRAP    55
14383 052172  000165                            .WORD   165                     ;                                              6858
14384 052174  012750                            .WORD   SYNC
14385 052176  026302                            .WORD   DUMPER
14386 052200  012746  010744                    MOV     #WRD.21,-(SP)
14387 052204  012746  010640                    MOV     #WRD.11,-(SP)           ;                                              6859
14388 052210  012746  011610                    MOV     #PHR.1,-(SP)
14389 052214  012746  010736                    MOV     #WRD.20,-(SP)
14390 052220  012746  010270                    MOV     #FOR.FMT,-(SP)
14391 052224  012746  000005                    MOV     #5,-(SP)
14392 052230  010600                            MOV     SP,R0                   ; SP,*
14393 052232  104414                            TRAP    14
14394 052234  062706  000014                    ADD     #14,SP
14395 052240  104467                    12$:    TRAP    67                      ;                                              6857
14396 052242  006000                            ROR     R0                      ;                                              6860
14397 052244  103641                            BLO     1$
14398 052246  005202                            INC     R2
14399 052250  020227  000002                    CMP     R2,#2                   ; CNT                                          6827
14400 052254  003635                            BLE     1$                      ; CNT,*
14401 052256  005726                            TST     (SP)+
14402 052260  000207                            RTS     PC                      ;                                              6800
14403
14404                                   ; Routine Size:  105 words
14405                                   ; Maximum stack depth per invocation:  10 words
14410
14411
14415
14416                                           .SBTTL  T23 TEST CODE SECTION
```

SEQ 0313

```
 14424 052262                           T23::
 14425 052262  004767  177452          1$:     JSR     PC,$T23                         ;
 14426 052266  104466                          TRAP    66                                      6863
 14427 052270  006000                          ROR     R0
 14428 052272  103773                          BLO     1$
 14429 052274  000207                          RTS     PC
 14430
 14431                                  ; Routine Size:  6 words
 14432                                  ; Maximum stack depth per invocation:  0 words
 14437
 14438
 14439 ;          6866  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (76)

```
14441 ;ML4AD
14442 ;                 TEST CODE SECTION
14443
14444 ;          6867  !
14445 ;          6868  !
14446 ;          6869  BGNTST;
14447 ;          6870
14448 ;          6871  !++
14449 ;          6872  ! TEST NUMBER:  TST 24
14450 ;          6873  !
14451 ;          6874  ! TEST NAME : initial PROM TEST
14452 ;          6875  !
14453 ;          6876  ! TEST DESCRIPTION:
14454 ;          6877  !
14455 ;          6878  !       TEST THE MEMORY ARRAYS' PROM
14456 ;          6879  !       TIMING AND CONTROL LOGIC FOR
14457 ;          6880  !       INITIAL PROM READS BY:
14458 ;          6881  !
14459 ;          6882  !       1.  READING 14 PROM LOCATIONS
14460 ;          6883  !           AND TESTING FOR:
14461 ;          6884  !
14462 ;          6885  !           A. CHECK SUM ERRORS AT
14463 ;          6886  !              EACH ROW COLUMN ADRS
14464 ;          6887  !
14465 ;          6888  !           B. SUM OF EACH PROM BIT SET
14466 ;          6889  !              <9,0> GTR 14.
14467 ;          6890  !
14468 ;          6891  !--
14469 ;          6892
14470 ;          6893  local
14471 ;          6894      OFF_SET_CNT : vector [10, byte],      !COUNTS EACH NIBBLE OFFSET
14472 ;          6895      ROW_ORED_COL : bitvector [16],        !SAVES ROW DATA OR'ED WITH COL DATA
14473 ;          6896      R_C_SAV : bitvector [16],             !TEMP LOCATION FOR ROW COL DATA
14474 ;          6897      PROM_ADRS,                            !PROM ADDRESS
14475 ;          6898      CHK_SUM,                              !CHECK SUM DATA
14476 ;          6899      CHK_SUM_ERR,                          !CHECK SUM ERROR
14477 ;          6900      BAD_NIB_CNT,                          !COUNTS BAD NIBBLES
14478 ;          6901      CNT_14_BAD,                           !COUNTS BAD NIBBLE POSITION EQL 14
14479 ;          6902      DODU_FLG,                             !DROP UNIT FLAG
14480 ;          6903      ERR_FLG;                              !ERROR FLAG
14481 ;          6904
14482 ;          6905  CLR_THRESHOLD;                           !CLEAR ERROR PRINT THRESHOLD
14483 ;          6906  CLR_MBUS;
14484 ;          6907  DODU_FLG = ZERO;
14485 ;          6908  PROM_DIS = ONE;                         !SET PROM DISABLE MODE
14486 ;          6909  CHK_SUM_ERR = ZEROES;
14487 ;          6910
14488 ;          6911  incr CNT from 0 to 9 do                 !CLEAR OFFSET COUNTS
14489 ;          6912      OFF_SET_CNT [.CNT] = ZEROES;
14490 ;          6913
14491 ;          6914  incr ADRS_CNT from 0 to 14 do           !READ PROM DATA FROM 15 ARRAY WORDS
14492 ;          6915      begin
14493 ;          6916      ROW_ORED_COL = ZEROES;              !CLEAR ROW ORED COL SAVE LOCATION
14494 ;          6917      PROM_ADRS = .ADRS_CNT;              !GET COPY OF ADRS_CNT
14495 ;          6918
```

```
14497 ;ML4AD
14498 ;                    TEST CODE SECTION
14499
14500 ;          6919      incr TWICE from 0 to 1 do                    !READ ROW AND COL DATA FOR THIS ARRAY WORD
14501 ;          6920          begin
14502 ;          6921          BAD_NIB_CNT = ZEROES;
14503 ;          6922          ERR_FLG = ZERO;
14504 ;          6923          MLPA = .PROM_ADRS;                       !LOADING MLPA INITIATES PROM READ
14505 ;          6924          DELAY (ONE_US);
14506 ;          6925          R_C_SAV = .MLPD;                         !GET THE ROW OR COL DATA
14507 ;          6926
14508 ;          6927          incr BIT_CNT from 0 to 9 do              !COUNT NUMBER BITS SET IN <9:0>
14509 ;          6928
14510 ;          6929              if .R_C_SAV [.BIT_CNT] IS_SET then BAD_NIB_CNT = .BAD_NIB_CNT + 1;
14511 ;          6930
14512 ;          6931          CHK_SUM = .R_C_SAV<10, 3>;               !GET THE CHECK SUM BITS
14513 ;          6932
14514 ;          6933          if .R_C_SAV [13] IS_SET then CHK_SUM = .CHK_SUM + 1;     !ADD IN BIT 13
14515 ;          6934
14516 ;          6935          if .R_C_SAV [14] IS_SET then CHK_SUM = .CHK_SUM + 1;     !ADD IN BIT 14
14517 ;          6936
14518 ;          6937          if .R_C_SAV [15] IS_SET then CHK_SUM = .CHK_SUM + 1;     !ADD IN BIT 15
14519 ;          6938
14520 ;          6939          if .R_C_SAV [15] IS_SET                  !IS BIT 15 SET
14521 ;          6940          then
14522 ;          6941              begin
14523 ;          6942
14524 ;          6943              if .BAD_...B_CNT lss .CHK_SUM then ERR_FLG = ONE;     !SET ERROR FLG IF CHECK SUM ERROR
14525 ;          6944
14526 ;          6945              end
14527 ;          6946          else
14528 ;          6947              begin
14529 ;          6948
14530 ;          6949              if .BAD_NIB_CNT neq .CHK_SUM then ERR_FLG = ONE;     !SET ERROR FLG IF CHECK SUM ERROR
14531 ;          6950
14532 ;          6951              end;
14533 ;          6952
14534 ;          6953          if .ERR_FLG IS_SET                       !WAS THERE A CHECK SUM ERROR
14535 ;          6954          then
14536 ;          6955              begin                                !REPORT INTERMEDIATE ERROR IF YES
14537 ;          6956              CMP_THRESHOLD;                       !COMPARE ERROR PRINT THRESHOLD
14538 ;          6957              ERRDF (70, INTER, DUMPER);
14539 ;          6958              PRINTB (SIX_FMT, FNC_21, WRD_10, WRD_12, WRD_45, WRD_35, FNC_6);
14540 ;          6959              PRINTB (FMT_10, .CHK_SUM, .R_C_SAV);
14541 ;          6960              DODU_FLG = ONE;
14542 ;          6961              end;
14543 ;          6962
14544 ;          6963          ROW_ORED_COL = (.ROW_ORED_COL) or (.R_C_SAV);    !OR ROW AND COLUMN DATA
14545 ;          6964          PROM_ADRS = (.PROM_ADRS) or (%o'2000');!GET COLUMN DATA
14546 ;          6965          end;
14547 ;          6966
14548 ;          6967      incr index from 0 to 9 do                    !GET TOTAL OFF SET COUNTS FROM THE 15 ARRAY WORDS
14549 ;          6968
14550 ;          6969          if .ROW_ORED_COL [.index] IS_SET then OFF_SET_CNT [.index] = .OFF_SET_CNT [.index] + 1;
14551 ;          6970
```

```
14553 ;ML4AD
14554 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
14555                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (7)
14556 ;          6971       end;
14557 ;          6972
14558 ;          6973   CNT_14_BAD = ZEROES;
14559 ;          6974
14560 ;          6975   incr CNT from 0 to 9 do                             !COUNT HOW MANY OFFSET COUNTS GEQ 14
14561 ;          6976
14562 ;          6977       if .OFF_SET_CNT [.CNT] geq 14 then CNT_14_BAD = .CNT_14_BAD + 1;
14563 ;          6978
14564 ;          6979   if .CNT_14_BAD neq ZERO                             !WAS ANY OFF_SET COUNTS GTR ZERO
14565 ;          6980   then
14566 ;          6981       begin
14567 ;          6982       ERRDF (71, ARR_DAT, DUMPER);                    !ERROR IF YES
14568 ;          6983
14569 ;          6984       if .CNT_14_BAD eql 10                           !WERE ALL 10 NIBBLE OFFSETS GEQ 14
14570 ;          6985       then
14571 ;          6986           PRINTB (SIX_FMT, WRD_55, WRD_54, WRD_46, WRD_51, WRD_42, WRD_40)        !ERROR IF YES
14572 ;          6987       else                                            !IF NO THEN PRINT WHICH ONES WERE
14573 ;          6988           begin
14574 ;          6989           PRINTB (FOR_FMT, WRD_46, WRD_47, WRD_42, WRD_40);       !PRINT MESSAGE
14575 ;          6990
14576 ;          6991           incr CNT from 0 to 9 do                     !FIND OFFSETS GEQ 14
14577 ;          6992
14578 ;          6993               if .OFF_SET_CNT [.CNT] geq 14 then PRINTB (FMT_13, .CNT, (.OFF_SET_CNT [.CNT]));
14579 ;          6994
14580 ;          6995                                                       !PRINT NIBBLE POSITION AND COUNT
14581 ;          6996           end;
14582 ;          6997
14583 ;          6998       DODU_FLG = ONE;
14584 ;          6999       end;
14585 ;          7000
14586 ;          7001   if .DODU_FLG IS_SET                                 !DROP THIS UNIT IF DODU_FLG SET
14587 ;          7002   then
14588 ;          7003       begin
14589 ;          7004       DODU (.ML_LUN);
14590 ;          7005       DOCLN;
14591 ;          7006       end;
14592 ;          7007
14593 ;          7008   ENDTST;
14597
14598
14602 052276  004167  132310          $T24:   .SBTTL  $T24 TEST CODE SECTION
14603 052302  162706  000030                  JSR     R1,$SAVE5                       ;                       6865
14604 052306  005067  143064                  SUB     #30,SP
14605 052312  152777  000040  143124          CLR     P.CNT                           ;                       6903
14606 052320  016705  143506                  BISB    #40,@ML.REG+40                  ;                       6905
14607 052324  042705  177770                  MOV     ML.DUT,R5
                                              BIC     #177770,R5
```

```
14609                                   ;ML4AD
14610                                   ;                       TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
14611                                                                                                               29-Mar-1982 16:21:03    PA:<
14612 052330 142777 000007 143106       BICB    #7,@ML.REG+40
14613 052336 150577 143102              BISB    R5,@ML.REG+40
14614 052342 005066 000002              CLR     2(SP)
14615 052346 152777 000040 143150       BISB    #40,@ML.REG+120         ; DODU.FLG                                  6907
14616 052354 005002                     CLR     R2                      ;                                           6908
14617 052356 012701 000016          1$: MOV     #16,R1                  ; CHK.SUM.ERR                               6909
14618 052362 060601                     ADD     SP,R1                   ;                                           6912
14619 052364 060201                     ADD     R2,R1                   ; OFF.SET.CNT,*
14620 052366 105011                     CLRB    (R1)                    ; CNT,*
14621 052370 005202                     INC     R2
14622 052372 020227 000011              CMP     R2,#11                  ; CNT                                       6911
14623 052376 003767                     BLE     1$                      ; CNT,*
14624 052400 005005                     CLR     R5
14625 052402 005066 000010          2$: CLR     10(SP)                  ; ADRS.CNT                                  6914
14626 052406 010566 000006              MOV     R5,6(SP)                ; ROW.ORED.COL                              6916
14627 052412 005004                     CLR     R4                      ; ADRS.CNT,PROM.ADRS                        6917
14628 052414 005066 000004          3$: CLR     4(SP)                   ; TWICE                                     6919
14629 052420 005016                     CLR     (SP)                    ; BAD.NIB.CNT                               6921
14630 052422 016677 000006 143054       MOV     6(SP),@ML.REG+100       ; ERR.FLG                                   6922
14631 052430 012701 000001              MOV     #1,R1                   ; PROM.ADRS,*                               6923
14632 052434 001411                 4$: BEQ     7$                      ; *,$$TMP2                                  6924
14633 052436 016702 127454              MOV     L$DLY,R2
14634 052442 001404                     BEQ     6$                      ; *,$$TMP1
14635 052444 005066 000014          5$: CLR     14(SP)
14636 052450 005302                     DEC     R2                      ; $$TMP
14637 052452 001374                     BNE     5$                      ; $$TMP1
14638 052454 005301                 6$: DEC     R1
14639 052456 000766                     BR      4$                      ; $$TMP2
14640 052460 017766 143150 000012   7$: MOV     @ML.REG+230,12(SP)      ; *,R.C.SAV                                 6925
14641 052466 005002                     CLR     R2
14642 052470 010201                 8$: MOV     R2,R1                   ; BIT.CNT                                   6927
14643 052472 006201                     ASR     R1                      ; BIT.CNT,*                                 6929
14644 052474 006201                     ASR     R1
14645 052476 006201                     ASR     R1
14646 052500 012700 000012              MOV     #12,R0
14647 052504 060600                     ADD     SP,R0                   ; R.C.SAV,*
14648 052506 060001                     ADD     R0,R1
14649 052510 010146                     MOV     R1,-(SP)
14650 052512 010246                     MOV     R2,-(SP)                ; BIT.CNT,*
14651 052514 042716 177770              BIC     #177770,(SP)
14652 052520 012746 000001              MOV     #1,-(SP)
14653 052524 005046                     CLR     -(SP)
14654 052526 004767 131102              JSR     PC,BL$GT2
14655 052532 062706 000010              ADD     #10,SP
14656 052536 005300                     DEC     R0
14657 052540 001002                     BNE     9$
14658 052542 005266 000004              INC     4(SP)                   ; BAD.NIB.CNT
14659 052546 005202                 9$: INC     R2                      ; BIT.CNT                                   6927
14660 052550 020227 000011              CMP     R2,#11                  ; BIT.CNT,*
14661 052554 003745                     BLE     8$
14662 052556 016603 000012              MOV     12(SP),R3               ; R.C.SAV,CHK.SUM                           6931
14663 052562 006203                     ASR     R3                      ; CHK.SUM
```

SEQ 0318


```
14665                                   ;ML4AD
14666                                   :
14667                                   :               TEST CODE SECTION
14668 052564  006203                            ASR     R3                      ; CHK.SUM
14669 052566  000303                            SWAB    R3                      ; CHK.SUM
14670 052570  042703  177770                    BIC     #177770,R3              ; *,CHK.SUM
14671 052574  132766  000040  000013            BITB    #40,13(SP)              ; *,R.C.SAV+1                   6933
14672 052602  001401                            BEQ     10$
14673 052604  005203                            INC     R3                      ; CHK.SUM
14674 052606  132766  000100  000013  10$:      BITB    #100,13(SP)             ; *,R.C.SAV+1                   6935
14675 052614  001401                            BEQ     11$
14676 052616  005203                            INC     R3                      ; CHK.SUM
14677 052620  005002                    11$:    CLR     R2                      :
14678 052622  105766  000013                    TSTB    13(SP)                  ; R.C.SAV+1                     6937
14679 052626  100002                            BPL     12$
14680 052630  005202                            INC     R2
14681 052632  005203                            INC     R3                      ; CHK.SUM
14682 052634  006002                    12$:    ROR     R2                      :                               6939
14683 052636  103004                            BCC     13$
14684 052640  026603  000004                    CMP     4(SP),R3                ; BAD.NIB.CNT,CHK.SUM           6943
14685 052644  002006                            BGE     15$
14686 052646  000403                            BR      14$
14687 052650  026603  000004            13$:    CMP     4(SP),R3                ; BAD.NIB.CNT,CHK.SUM           6949
14688 052654  001402                            BEQ     15$
14689 052656  012716  000001            14$:    MOV     #1,(SP)                 ; *,ERR.FLG
14690 052662  021627  000001            15$:    CMP     (SP),#1                 ; ERR.FLG,*                     6953
14691 052666  001052                            BNE     16$
14692 052670  005267  142502                    INC     P.CNT                   :                               6955
14693 052674  026767  142476  142476            CMP     P.CNT,LIMIT
14694 052702  003060                            BGT     17$
14695 052704  104455                            TRAP    55                      ;                               6957
14696 052706  000106                            .WORD   106
14697 052710  013114                            .WORD   INTER
14698 052712  026302                            .WORD   DUMPER
14699 052714  012746  012226                    MOV     #FNC.6,-(SP)            ;                               6958
14700 052720  012746  011110                    MOV     #WRD.35,-(SP)
14701 052724  012746  011214                    MOV     #WRD.45,-(SP)
14702 052730  012746  010650                    MOV     #WRD.12,-(SP)
14703 052734  012746  010630                    MOV     #WRD.10,-(SP)
14704 052740  012746  012440                    MOV     #FNC.21,-(SP)
14705 052744  012746  010322                    MOV     #SIX.FMT,-(SP)
14706 052750  012746  000007                    MOV     #7,-(SP)
14707 052754  010600                            MOV     SP,R0                   ; SP,*
14708 052756  104414                            TRAP    14
14709 052760  016616  000032                    MOV     32(SP),(SP)             ; R.C.SAV,*                     6959
14710 052764  010346                            MOV     R3,-(SP)                ; CHK.SUM,*
14711 052766  012746  007116                    MOV     #FMT.10,-(SP)
14712 052772  012746  000003                    MOV     #3,-(SP)
14713 052776  010600                            MOV     SP,R0                   ; SP,*
14714 053000  104414                            TRAP    14
14715 053002  012766  000001  000030            MOV     #1,30(SP)               ; *,DODU.FLG                    6960
14716 053010  062706  000026                    ADD     #26,SP                  :                               6955
14717 053014  056666  000012  000010  16$:      BIS     12(SP),10(SP)           ; R.C.SAV,ROW.ORED.COL          6963
14718 053022  052766  002000  000006            BIS     #2000,6(SP)             ; *,PROM.ADRS                   6964
14719 053030  005204                            INC     R4                      ; TWICE                         6919
```

SEQ 0319

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
14721                                   ;ML4AD
14722                                   ;               TEST CODE SECTION
14723
14724 053032  020427  000001                   CMP     R4,#1                   ; TWICE,*
14725 053036  003002                           BGT     17$
14726 053040  000167  177350                   JMP     3$
14727 053044  005002                   17$:    CLR     R2                      ; INDEX                         6967
14728 053046  010201                   18$:    MOV     R2,R1                   ; INDEX,*                       6969
14729 053050  006201                           ASR     R1
14730 053052  006201                           ASR     R1
14731 053054  006201                           ASR     R1
14732 053056  012700  000010                   MOV     #10,R0
14733 053062  060600                           ADD     SP,R0                   ; ROW.ORED.COL,*
14734 053064  060001                           ADD     R0,R1
14735 053066  010146                           MOV     R1,-(SP)
14736 053070  010246                           MOV     R2,-(SP)                ; INDEX,*
14737 053072  042716  177770                   BIC     #177770,(SP)
14738 053076  012746  000001                   MOV     #1,-(SP)
14739 053102  005046                           CLR     -(SP)
14740 053104  004767  130524                   JSR     PC,BL$GT2
14741 053110  062706  000010                   ADD     #10,SP
14742 053114  005300                           DEC     R0
14743 053116  001005                           BNE     19$
14744 053120  012701  000016                   MOV     #16,R1
14745 053124  060601                           ADD     SP,R1                   ; OFF.SET.CNT,*
14746 053126  060201                           ADD     R2,R1                   ; INDEX,*
14747 053130  105211                           INCB    (R1)
14748 053132  005202                   19$:    INC     R2                      ; INDEX                         6967
14749 053134  020227  000011                   CMP     R2,#11                  ; INDEX,*
14750 053140  003742                           BLE     18$
14751 053142  005205                           INC     R5                      ; ADRS.CNT                      6914
14752 053144  020527  000016                   CMP     R5,#16                  ; ADRS.CNT,*
14753 053150  003002                           BGT     20$
14754 053152  000167  177224                   JMP     2$
14755 053156  005000                   20$:    CLR     R0                      ; CNT.14.BAD                    6973
14756 053160  005001                           CLR     R1                      ; CNT                           6975
14757 053162  012702  000016           21$:    MOV     #16,R2                  ;                               6977
14758 053166  060602                           ADD     SP,R2                   ; OFF.SET.CNT,*
14759 053170  060102                           ADD     R1,R2                   ; CNT,*
14760 053172  121227  000016                   CMPB    (R2),#16
14761 053176  103401                           BLO     22$
14762 053200  005200                           INC     R0                      ; CNT.14.BAD
14763 053202  005201                   22$:    INC     R1                      ; CNT                           6975
14764 053204  020127  000011                   CMP     R1,#11                  ; CNT,*
14765 053210  003764                           BLE     21$
14766 053212  005700                           TST     R0                      ; CNT.14.BAD                    6979
14767 053214  001505                           BEQ     27$
14768 053216  104455                           TRAP    55                      ;                               6982
14769 053220  000107                           .WORD   107
14770 053222  013012                           .WORD   ARR.DAT
14771 053224  026302                           .WORD   DUMPER
14772 053226  020027  000012                   CMP     R0,#12                  ; CNT.14.BAD,*                  6984
14773 053232  001024                           BNE     23$
14774 053234  012746  011150                   MOV     #WRD.40,-(SP)           ;                               6986
14775 053240  012746  011170                   MOV     #WRD.42,-(SP)
```

```
                                      ;ML4AD                                                        29-Mar-1982 16:23:04   TOPS
                                      ;                 TEST CODE SECTION                           29-Mar-1982 16:21:03   PA:<
053244  012746  011276                        MOV     #WRD.51,-(SP)
053250  012746  011226                        MOV     #WRD.46,-(SP)
053254  012746  011326                        MOV     #WRD.54,-(SP)
053260  012746  011336                        MOV     #WRD.55,-(SP)
053264  012746  010322                        MOV     #SIX.FMT,-(SP)
053270  012746  000007                        MOV     #7,-(SP)
053274  010600                                MOV     SP,R0                   ; SP,*
053276  104414                                TRAP    14
053300  022626                                CMP     (SP)+,(SP)+
053302  000445                                BR      26$
053304  012746  011150                23$:    MOV     #WRD.40,-(SP)           ;                       6984
053310  012746  011170                        MOV     #WRD.42,-(SP)           ;                       6989
053314  012746  011240                        MOV     #WRD.47,-(SP)
053320  012746  011226                        MOV     #WRD.46,-(SP)
053324  012746  010270                        MOV     #FOR.FMT,-(SP)
053330  012746  000005                        MOV     #5,-(SP)
053334  010600                                MOV     SP,R0                   ; SP,*
053336  104414                                TRAP    14
053340  005002                                CLR     R2                      ; CNT
053342  012701  000032                24$:    MOV     #32,R1                  ;                       6991
053346  060601                                ADD     SP,R1                   ; OFF.SET.CNT,*         6993
053350  060201                                ADD     R2,R1                   ; CNT,*
053352  121127  000016                        CMPB    (R1),#16
053356  103413                                BLO     25$
053360  005046                                CLR     -(SP)
053362  111116                                MOVB    (R1),(SP)
053364  010246                                MOV     R2,-(SP)                ; CNT,*
053366  012746  007254                        MOV     #FMT.13,-(SP)
053372  012746  000003                        MOV     #3,-(SP)
053376  010600                                MOV     SP,R0                   ; SP,*
053400  104414                                TRAP    14
053402  062706  000010                        ADD     #10,SP
053406  005202                        25$:    INC     R2                      ; CNT
053410  020227  000011                        CMP     R2,#11                  ; CNT,*                 6991
053414  003752                                BLE     24$
053416  012766  000001  000016        26$:    MOV     #1,16(SP)               ; *,DODU.FLG            6998
053424  062706  000014                        ADD     #14,SP                  ;                       6981
053430  026627  000002  000001        27$:    CMP     2(SP),#1                ; DODU.FLG,*            7001
053436  001004                                BNE     28$
053440  016700  142364                        MOV     ML.LUN,R0               ;                       7004
053444  104451                                TRAP    51
053446  104444                                TRAP    44
053450  062706  000030                28$:    ADD     #30,SP                  ;                       6865
053454  000207                                RTS     PC

                                      ; Routine Size:  312 words
                                      ; Maximum stack depth per invocation:  29 words
```

SEQ 0321

```
 14838
 14839                                          .SBTTL  T24 TEST CODE SECTION
 14843 053456                           T24::
 14844 053456  004767  176614           1$:     JSR     PC,$T24                         ;     7006
 14845 053462  104466                           TRAP    66
 14846 053464  006000                           ROR     R0
 14847 053466  103773                           BLO     1$
 14848 053470  000207                           RTS     PC
 14849
 14850                                  ; Routine Size:  6 words
 14851                                  ; Maximum stack depth per invocation:  0 words
 14856
 14857
 14858 ;          7009  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (77)

```
14860 ;ML4AD
14861 ;                 TEST CODE SECTION
14862
14863 ;          7010  !
14864 ;          7011  !
14865 ;          7012  BGNTST;
14866 ;          7013
14867 ;          7014  !++
14868 ;          7015  ! TEST NUMBER:  TST 25
14869 ;          7016  !
14870 ;          7017  ! TEST NAME:    PROM OR FUNCTION TEST
14871 ;          7018  !
14872 ;          7019  ! TEST DESCRIPTION:
14873 ;          7020  !
14874 ;          7021  !       TEST THE HARDWARE ORING OF
14875 ;          7022  !       THE MEMORY ARRAYS' PROM
14876 ;          7023  !       ROW COLUMN DATA BY:
14877 ;          7024  !
14878 ;          7025  !       1.  READING AND STORING 128
14879 ;          7026  !           HARDWARE ORED ROW COLUMN
14880 ;          7027  !           DATA.
14881 ;          7028  !
14882 ;          7029  !       2.  THEN IN PROM DISABLE MODE
14883 ;          7030  !           AND VIA SOFTWARE CONTROL,
14884 ;          7031  !           READ AND OR PROM ROW
14885 ;          7032  !           COLUMN DATA AND COMPARE
14886 ;          7033  !           AGAINST THE RESPECTIVE
14887 ;          7034  !           STORED HARDWARE ORED DATA.
14888 ;          7035  !
14889 ;          7036  !--
14890 ;          7037
14891 ;          7038  local
14892 ;          7039      R_BITS,                          !PROM ROW DATA
14893 ;          7040      C_BITS,                          !PROM COL DATA
14894 ;          7041      SW_ORED,                         !SOFTWARE CALCULATED PROM ORED DATA
14895 ;          7042      HW_SAVE,                         !SOFTWARE  PROM ORED DATA
14896 ;          7043      DODU_FLG;                        !DROP UNIT FLAG
14897 ;          7044
14898 ;          7045  BGNSUB;
14899 ;          7046  CLR_THRESHOLD;                       !CLEAR ERROR PRINT THRESHOLD
14900 ;          7047  CLR_MBUS;
14901 ;          7048  DODU_FLG = ZERO;
14902 ;          7049  DAT_DM = ONE;
14903 ;          7050  FIRST_BLK_XFER ();                   !SET UP A FIRST BLOCK XFER
14904 ;          7051  ML_FUNC = write;                     !DO A WRITE FUNCTION
14905 ;          7052
14906 ;          7053  incr PROM_ADRS from 0 to 127 do      !READ AND STORE 128 HARDWARE PROM ORED DATA
14907 ;          7054      begin
14908 ;          7055      DELAY (ONE_US);
14909 ;          7056      HW_OR_TBL [.PROM_ADRS] = .MLPD;  !READ HARDWARE PROM ORED DATA
14910 ;          7057      DAT_CLK = ONE;                   !CLOCK NEXT ONE OUT
14911 ;          7058      end;
14912 ;          7059
14913 ;          7060  CLR_MBUS;
14914 ;          7061  PROM_DIS = ONE;                      !SET PROM DISABLE MODE
```

```
;ML4AD
;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                   29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (77)
;
;          7062
;          7063  incr PROM_ADRS from 0 to 127 do                   !CALCULATE 128 SW ORED DATA & COMPARE TO HW TABLE
;          7064      begin
;          7065      MLPA = .PROM_ADRS;                            !LOADING MLPA INITIATES A PROM READ
;          7066      DELAY (ONE_US);
;          7067      R_BITS = .MLPD;                               !SAVE ROW DATA
;          7068      MLPA = .PROM_ADRS or %o'2000';                !ENABLE COLUMN DATA ADRS
;          7069      DELAY (ONE_US);
;          7070      C_BITS = .MLPD;                               !SAVE COL DATA
;          7071      SW_ORED = .R_BITS or .C_BITS;                 !CALCULATE SOFTWARE ORED
;          7072      HW_SAVE = .HW_OR_TBL [.PROM_ADRS];            !GET RESPECTIVE HARDWARE ORED
;          7073
;          7074      if (.HW_SAVE<0, 9>) neq (.SW_ORED<0, 9>)      !COMPARE SW & HW ORED
;          7075      then
;          7076          begin                                     !IF NEQ THEN ERROR
;          7077          CMP_THRESHOLD;                            !COMPARE ERROR PRINT THRESHOLD
;          7078          ERRDF (76, ARR_DAT, DUMPER);
;          7079          PRINTB (FOR_FMT, WRD_35, WRD_36, WRD_19, PHR_4);
;          7080          PRINTB (FMT_2, .SW_ORED<0, 9>, .HW_SAVE<0, 9>, (.SW_ORED<0, 9> xor .HW_SAVE<0, 9>));
;          7081          DODU_FLG = ONE;
;          7082          end;
;          7083
;          7084      end;
;          7085
;          7086  ENDSUB;
;          7087
;          7088  if .DODU_FLG IS_SET                               !DROP THIS UNIT IF DODU FLG IS_SET
;          7089  then
;          7090      begin
;          7091      DODU (.ML_LUN);
;          7092      DOCLN;
;          7093      end;
;          7094
;          7095  ENDTST;
;
;
14961 053472  004167  131114          $T25:   .SBTTL  $T25 TEST CODE SECTION
14962 053476  162706  000010                  JSR     R1,$SAVE5                       ;                       7008
14963 053502  104402                  1$:     SUB     #10,SP
14964 053504  005067  141666                  TRAP    2                               ;                       7043
14965 053510  152777  000040  141726          CLR     P.CNT                           ;                       7045
14966 053516  016705  142310                  BISB    #40,@ML.REG+40                  ;                       7046
14967 053522  042705  177770                  MOV     ML.DUT,R5
14968 053526  142777  000007  141710          BIC     #177770,R5
14969 053534  150577  141704                  BICB    #7,@ML.REG+40
14970 053540  005016                          BISB    R5,@ML.REG+40
                                                CLR     (SP)                            ; DODU.FLG              7048
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04   TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03   PA:<
053542  152777  000010  141754                  BISB    #10,@ML.REG+120
053550  004767  144506                          JSR     PC,FIRST.BLK.XFER          ;                                     7049
053554  142777  000077  141622                  BICB    #77,@ML.REG                ;                                     7050
053562  152777  000061  141614                  BISB    #61,@ML.REG                ;                                     7051
053570  005004                                  CLR     R4                         ; PROM.ADRS                           7053
053572  012703  000001                  2$:     MOV     #1,R3                      ; *,$$TMP2                            7055
053576  001411                          3$:     BEQ     6$
053600  016705  126312                          MOV     L$DLY,R5                   ; *,$$TMP1
053604  001404                                  BEQ     5$
053606  005066  000006                  4$:     CLR     6(SP)                      ; $$TMP
053612  005305                                  DEC     R5                         ; $$TMP1
053614  001374                                  BNE     4$
053616  005303                          5$:     DEC     R3                         ; $$TMP2
053620  000766                                  BR      3$
053622  010405                          6$:     MOV     R4,R5                      ; PROM.ADRS,*                         7056
053624  006305                                  ASL     R5
053626  017765  142002  013402                  MOV     @ML.REG+230,HW.OR.TBL(R5)
053634  152777  000020  141662                  BISB    #20,@ML.REG+120            ;                                     7057
053642  005204                                  INC     R4                         ; PROM.ADRS                           7053
053644  020427  000177                          CMP     R4,#177                    ; PROM.ADRS,*
053650  003750                                  BLE     2$
053652  152777  000040  141564                  BISB    #40,@ML.REG+40             ;                                     7058
053660  016705  142146                          MOV     ML.DUT,R5
053664  042705  177770                          BIC     #177770,R5
053670  142777  000007  141546                  BICB    #7,@ML.REG+40
053676  150577  141542                          BISB    R5,@ML.REG+40
053702  152777  000040  141614                  BISB    #40,@ML.REG+120            ;                                     7061
053710  005003                                  CLR     R3                         ; PROM.ADRS                           7063
053712  010377  141566                  7$:     MOV     R3,@ML.REG+100             ; PROM.ADRS,*                         7065
053716  012704  000001                          MOV     #1,R4                      ; *,$$TMP2                            7066
053722  001411                          8$:     BEQ     11$
053724  016705  126166                          MOV     L$DLY,R5                   ; *,$$TMP1
053730  001404                                  BEQ     10$
053732  005066  000006                  9$:     CLR     6(SP)                      ; $$TMP
053736  005305                                  DEC     R5                         ; $$TMP1
053740  001374                                  BNE     9$
053742  005304                          10$:    DEC     R4                         ; $$TMP2
053744  000766                                  BR      8$
053746  017766  141662  000004          11$:    MOV     @ML.REG+230,4(SP)          ; *,R.BITS                            7067
053754  010305                                  MOV     R3,R5                      ; PROM.ADRS,*                         7068
053756  052705  002000                          BIS     #2000,R5
053762  010577  141516                          MOV     R5,@ML.REG+100
053766  012704  000001                          MOV     #1,R4                      ; *,$$TMP2                            7069
053772  001411                          12$:    BEQ     15$
053774  016705  126116                          MOV     L$DLY,R5                   ; *,$$TMP1
054000  001404                                  BEQ     14$
054002  005066  000006                  13$:    CLR     6(SP)                      ; $$TMP
054006  005305                                  DEC     R5                         ; $$TMP1
054010  001374                                  BNE     13$
054012  005304                          14$:    DEC     R4                         ; $$TMP2
054014  000766                                  BR      12$
054016  017766  141612  000002          15$:    MOV     @ML.REG+230,2(SP)          ; *,C.BITS                            7070
```

```
15028                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
15029                                   ;                TEST CODE SECTION                      29-Mar-1982 16:21:03    PA:<
15030
15031 054024  016601  000004                    MOV     4(SP),R1                ; R.BITS,SW.ORED                        7071
15032 054030  056601  000002                    BIS     2(SP),R1                ; C.BITS,SW.ORED
15033 054034  010305                            MOV     R3,R5                   ; PROM.ADRS,*                           7072
15034 054036  006305                            ASL     R5
15035 054040  016502  013402                    MOV     HW.OR.TBL(R5),R2        ; *,HW.SAVE
15036 054044  010104                            MOV     R1,R4                   ; SW.ORED,*                             7074
15037 054046  042704  177000                    BIC     #177000,R4
15038 054052  010205                            MOV     R2,R5                   ; HW.SAVE,*
15039 054054  042705  177000                    BIC     #177000,R5
15040 054060  020504                            CMP     R5,R4
15041 054062  001457                            BEQ     16$
15042 054064  005267  141306                    INC     P.CNT                   ;                                       7076
15043 054070  026767  141302  141302            CMP     P.CNT,LIMIT
15044 054076  003055                            BGT     17$
15045 054100  104455                            TRAP    55                      ;                                       7078
15046 054102  000114                            .WORD   114
15047 054104  013012                            .WORD   ARR.DAT
15048 054106  026302                            .WORD   DUMPER
15049 054110  012746  011676                    MOV     #PHR.4,-(SP)            ;                                       7079
15050 054114  012746  010730                    MOV     #WRD.19,-(SP)
15051 054120  012746  011116                    MOV     #WRD.36,-(SP)
15052 054124  012746  011110                    MOV     #WRD.35,-(SP)
15053 054130  012746  010270                    MOV     #FOR.FMT,-(SP)
15054 054134  012746  000005                    MOV     #5,-(SP)
15055 054140  010600                            MOV     SP,R0                   ; SP,*
15056 054142  104414                            TRAP    14
15057 054144  010205                            MOV     R2,R5                   ; HW.SAVE,*
15058 054146  010116                            MOV     R1,(SP)                 ; SW.ORED,*                             7080
15059 054150  010504                            MOV     R5,R4
15060 054152  040104                            BIC     R1,R4
15061 054154  040516                            BIC     R5,(SP)
15062 054156  050416                            BIS     R4,(SP)
15063 054160  010246                            MOV     R2,-(SP)                ; HW.SAVE,*
15064 054162  042716  177000                    BIC     #177000,(SP)
15065 054166  010146                            MOV     R1,-(SP)                ; SW.ORED,*
15066 054170  042716  177000                    BIC     #177000,(SP)
15067 054174  012746  006506                    MOV     #FMT.2,-(SP)
15068 054200  012746  000004                    MOV     #4,-(SP)
15069 054204  010600                            MOV     SP,R0                   ; SP,*
15070 054206  104414                            TRAP    14
15071 054210  012766  000001  000024            MOV     #1,24(SP)               ; *,DODU.FLG                            7081
15072 054216  062706  000024                    ADD     #24,SP                  ;                                       7076
15073 054222  005203                    16$:    INC     R3                      ; PROM.ADRS                             7063
15074 054224  020327  000177                    CMP     R3,#177                 ; PROM.ADRS,*
15075 054230  003630                            BLE     7$
15076 054232  104467                    17$:    TRAP    67                      ;                                       7084
15077 054234  006000                            ROR     R0
15078 054236  103002                            BHIS    18$
15079 054240  000167  177236                    JMP     1$
15080 054244  021627  000001            18$:    CMP     (SP),#1                 ; DODU.FLG,*                            7088
15081 054250  001004                            BNE     19$
15082 054252  016700  141552                    MOV     ML.LUN,R0               ;                                       7091
```

```
15084                                   ;ML4AD
15085                                   ;               TEST CODE SECTION                                   29-Mar-198? 16:23:04    TOPS
15086                                                                                                       29-Mar-1982 16:21:03    PA:<
15087 054256  104451                             TRAP    51
15088 054260  104444                             TRAP    44
15089 054262  062706  000010            19$:     ADD     #10,SP                          ;                                       7008
15090 054266  000207                             RTS     PC
15091
15092                                   ; Routine Size:  191 words
15093                                   ; Maximum stack depth per invocation:  20 words
15098
15099
15103
15104                                           .SBTTL  T25 TEST CODE SECTION
15108 054270                            T25::
15109 054270  004767  177176            1$:     JSR     PC,$T25                         ;                                       7093
15110 054274  104466                             TRAP    66
15111 054276  006000                             ROR     R0
15112 054300  103773                             BLO     1$
15113 054302  000207                             RTS     PC
15114
15115                                   ; Routine Size:  6 words
15116                                   ; Maximum stack depth per invocation:  0 words
15121
15122
15123 ;         7096  !<BLF/PAGE>
```

SEQ 0327

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (78)

```
15125 ;ML4AD
15126 ;                 TEST CODE SECTION
15127 ;
15128 ;          7097  !
15129 ;          7098  !
15130 ;          7099  BGNTST;
15131 ;          7100  !
15132 ;          7101  !++
15133 ;          7102  ! TEST NUMBER:  TST 26
15134 ;          7103  !
15135 ;          7104  ! TEST NAME:    UV ADRS ERROR TEST
15136 ;          7105  !
15137 ;          7106  ! TEST DESCRIPTION:
15138 ;          7107  !
15139 ;          7108  !       TEST THE DETECTION OF UV ADRS
15140 ;          7109  !       ERRORS BY:
15141 ;          7110  !
15142 ;          7111  !       1.  GENERATING PROM DATA PATTERN
15143 ;          7112  !           FROM 0 TO %O'177777' AND
15144 ;          7113  !           DETERMINE WHETHER RESPECTIVE
15145 ;          7114  !           PATTERN IS GOOD/OR BAD
15146 ;          7115  !           PROM DATA.
15147 ;          7116  !
15148 ;          7117  !       2.  VIA DAT DM AND PROM R/W
15149 ;          7118  !           MODES PRESENT GENERATED
15150 ;          7119  !           PROM DATA TO THE UV ADRS
15151 ;          7120  !           ERR PROM.
15152 ;          7121  !
15153 ;          7122  !       3.  TEST ERROR CONDITIONS FOR
15154 ;          7123  !           CORRECT RESPONCE TO GOOD/
15155 ;          7124  !           OR BAD PROM DATA.
15156 ;          7125  !
15157 ;          7126  !--
15158 ;          7127
15159 ;          7128  local
15160 ;          7129      HIGH_CNT,
15161 ;          7130      PROM_DATA : bitvector [16],     !STORES PROM DATA CHECK SUM BITS
15162 ;          7131      LOW_CNT,                        !STORES PROM DATA
15163 ;          7132      TEMP,                           !STORES SUM OF PROM DATA BITS <9:0>
15164 ;          7133      ERR_FLG,                        !TEMPORARY STORAGE
15165 ;          7134      GTR_FLG;                        !ERROR FLAG
15166 ;          7135                                      !SETS WHEN PROM DATA BIT 15 IS A ONE
15167 ;          7136  CLR_THRESHOLD;                       !CLEAR ERROR PRINT THRESHOLD
15168 ;          7137  PROM_DATA = -1;
15169 ;          7138
15170 ;          7139  do                                  !TEST ALL POSSIBLE PROM DATA COMBINATIONS
15171 ;          7140      begin
15172 ;          7141      PROM_DATA = .PROM_DATA + 1;     !INCREMENT PROM_DATA
15173 ;          7142      BGNSUB;
15174 ;          7143      CLR_MBUS;
15175 ;          7144      ERR_FLG = ZERO;
15176 ;          7145      LOW_CNT = ZEROES;
15177 ;          7146      HIGH_CNT = ZEROES;
15178 ;          7147      GTR_FLG = ZERO;
15179 ;          7148
```

```
15181 ;ML4AD
15182 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
15183 ;                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (78)
15184 ;          7149       incr BIT_CNT from 0 to 9 do                     !COUNT PROM DATA BITS <9:0>
15185 ;          7150
15186 ;          7151           if .PROM_DATA [.BIT_CNT] IS_SET then LOW_CNT = .LOW_CNT + 1;
15187 ;          7152
15188 ;          7153       HIGH_CNT = .PROM_DATA<10, 3>;                   !GET PROM DATA CHECK SUM BITS
15189 ;          7154
15190 ;          7155       if .PROM_DATA [13] IS_SET then HIGH_CNT = .HIGH_CNT + 1;    !ADD IN BIT 13 IF SET
15191 ;          7156
15192 ;          7157       if .PROM_DATA [14] IS_SET then HIGH_CNT = .HIGH_CNT + 1;    !ADD IN BIT 14 IF SET
15193 ;          7158
15194 ;          7159       if .PROM_DATA [15] IS_SET then HIGH_CNT = .HIGH_CNT + 1;    !ADD IN BIT 15 IF SET
15195 ;          7160
15196 ;          7161       if .PROM_DATA [15] IS_SET then GTR_FLG = ONE;       !SET FLAG IF BIT 15 IS SET
15197 ;          7162
15198 ;          7163       DAT_DM_XFER ();                                 !SET UP A DATA DIAG MODE XFERR
15199 ;          7164       PROM_RQ = ONE;                                  !SET PROM READ WRITE
15200 ;          7165       MLPD = .PROM_DATA;                              !LOAD MLPD WITH PROM_DATA
15201 ;          7166       MLCS1 = write;                                  !DO A WRITE FUNCTION
15202 ;          7167       DAT_CLK = ONE;                                  !CLOCK PROM DATA INTO UV PROM
15203 ;          7168
15204 ;          7169       if .UNS IS_SET                                  !SEE IF PROM DATA CAUSED A UV ERROR
15205 ;          7170       then
15206 ;          7171           begin
15207 ;          7172
15208 ;          7173           if .GTR_FLG IS_SET                          !UNS IS_SET. SEE IF GTR FLG IS SET
15209 ;          7174           then
15210 ;          7175               begin
15211 ;          7176               TEMP = .PROM_DATA;                      !LOAD TEMP WITH PROM DATA
15212 ;          7177               TEMP = .TEMP and %o'162000';            !SEE IF THESE BITS ARE SET IN PROM DATA
15213 ;          7178
15214 ;          7179               if .TEMP eql %o'162000'                 !THESE BITS SET AUTOMATICALLY CAUSE A UNS
15215 ;          7180               then
15216 ;          7181                   begin
15217 ;          7182
15218 ;          7183                   if .LOW_CNT geq .HIGH_CNT
15219 ;          7184                                                       !LOW<9:0> SHOULD BE ISS THAN THE HIGH<15:10> IF GTR FLG IS SET
15220 ;          7185                   then
15221 ;          7186                       begin                           !ERROR IF LOW<9:0> IS GEQ HIGH<15:10>
15222 ;          7187                       CMP_THRESHOLD;                  !COMPARE ERROR PRINT THRESHOLD
15223 ;          7188                       ERRDF (72, ARR_DAT, DUMPER);
15224 ;          7189                       PRINTB (SIX_FMT, WRD_34, PHR_5, WRD_32, WRD_6, WRD_33, WRD_24);
15225 ;          7190                       ERR_FLG = ONE;
15226 ;          7191                       end;
15227 ;          7192
15228 ;          7193                   end;
15229 ;          7194
15230 ;          7195               end
15231 ;          7196           else                                        !GTR FLG IS NOT SET
15232 ;          7197               begin
15233 ;          7198
15234 ;          7199               if .LOW_CNT eql .HIGH_CNT               !LOW<9:0> SHOULD BE NEQ HIGH<15:10>
15235 ;          7200               then
```

```
;ML4AD
;                    TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (78)
;          7201                      begin                             !ERROR IF EQL
;          7202                      CMP_THRESHOLD;                    !COMPARE ERROR PRINT THRESHOLD
;          7203                      ERRDF (73, ARR_DAT, DUMPER);
;          7204                      PRINTB (SIX_FMT, WRD_34, PHR_5, WRD_32, WRD_6, WRD_33, WRD_24);
;          7205                      ERR_FLG = ONE;
;          7206                      end;
;          7207
;          7208                  end
;          7209
;          7210              end
;          7211          else                                          !UNS WAS NOT SET
;          7212              begin
;          7213
;          7214              if .GTR_FLG IS_SET                        !SEE IF GTR FLG IS_SET
;          7215              then
;          7216                  begin
;          7217                  TEMP = .PROM_DATA;                    !LOAD TEMP WITH PROM DATA
;          7218                  TEMP = .TEMP and %o'162000';          !SEE IF THESE BITS ARE SET IN PROM DATA
;          7219
;          7220                  if .TEMP neq %o'162000'               !IF THESE BITS ARE SET THEN UNS SHOULD BE SET
;          7221                  then
;          7222                      begin                             !ERROR UNS IS NOT SET
;          7223                      CMP_THRESHOLD;                    !COMPARE ERROR PRINT THRESHOLD
;          7224                      ERRDF (74, ARR_DAT, DUMPER);
;          7225                      PRINTB (SIX_FMT, WRD_34, PHR_1, WRD_32, WRD_5, WRD_33, WRD_24);
;          7226                      ERR_FLG = ONE;
;          7227                      end
;          7228                  else                                  !BITS 162000 ARE NOT SET
;          7229                      begin
;          7230
;          7231                  if .LOW_CNT lss .HIGH_CNT             !LOW<9:0> SHOULD BE GEQ HIGH<15:10>
;          7232                  then
;          7233                      begin                             !ERROR IF LSS
;          7234                      CMP_THRESHOLD;                    !COMPARE ERROR PRINT THRESHOLD
;          7235                      ERRDF (75, ARR_DAT, DUMPER);
;          7236                      PRINTB (SIX_FMT, WRD_34, PHR_1, WRD_32, WRD_5, WRD_33, WRD_24);
;          7237                      ERR_FLG = ONE;
;          7238                      end;
;          7239
;          7240                  end
;          7241
;          7242              end
;          7243          else                                          !GTR_FLG IS NOT SET
;          7244              begin
;          7245
;          7246              if .LOW_CNT neq .HIGH_CNT                 !LOW<9:0> SHOULD EQL HIGH<15:10>
;          7247              then
;          7248                  begin                                 !ERROR IF NEQ
;          7249                  CMP_THRESHOLD;                        !COMPARE ERROR PRINT THRESHOLD
;          7250                  ERRDF (108, ARR_DAT, DUMPER);
;          7251                  PRINTB (SIX_FMT, WRD_34, PHR_1, WRD_32, WRD_5, WRD_33, WRD_24);
;          7252                  ERR_FLG = ONE;
```

```
 15293 ;ML4AD
 15294 ;                    TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 15295                                                           29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (78)
 15296 ;          7253                    end
 15297 ;          7254
 15298 ;          7255                end
 15299 ;          7256
 15300 ;          7257            end;
 15301 ;          7258
 15302 ;          7259        if .ERR_FLG IS_SET                  !SEE IF ERROR FLG GOT SET
 15303 ;          7260        then
 15304 ;          7261            begin
 15305 ;          7262            PRINTB (FMT_7, .PROM_DATA);     !PRINT FAILING PROM_DATA AND SET DODU_FLG
 15306 ;          7263            end;
 15307 ;          7264
 15308 ;          7265        ENDSUB;
 15309 ;          7266        end
 15310 ;          7267    until .PROM_DATA eql %o'177777';        !TRY ALL BIT COMBINATIONS
 15311 ;          7268
 15312 ;          7269    ENDTST;
 15316
 15317                                    .SBTTL  $T26 TEST CODE SECTION
 15321 054304  004167  130302        $T26: JSR     R1,$SAVE5                 ;                          7095
 15322 054310  162706  000010              SUB     #10,SP
 15323 054314  005067  141056              CLR     P.CNT                     ;
 15324 054320  012766  177777  000006      MOV     #-1,6(SP)                 ;                          7134
 15325 054326  005266  000006        1$:   INC     6(SP)                     ; *,PROM.DATA              7137
 15326 054332  104402                2$:   TRAP    2                         ; PROM.DATA                7141
 15327 054334  152777  000040  141102      BISB    #40,@ML.REG+40            ;                          7142
 15328 054342  016705  141464              MOV     ML.DUT,R5
 15329 054346  042705  177770              BIC     #177770,R5
 15330 054352  142777  000007  141064      BICB    #7,@ML.REG+40
 15331 054360  150577  141060              BISB    R5,@ML.REG+40
 15332 054364  005001                      CLR     R1                        ; ERR.FLG                  7144
 15333 054366  005066  000002              CLR     2(SP)                     ; LOW.CNT                  7145
 15334 054372  005002                      CLR     R2                        ; HIGH.CNT                 7146
 15335 054374  005066  000004              CLR     4(SP)                     ; GTR.FLG                  7147
 15336 054400  005005                      CLR     R5                        ; BIT.CNT                  7149
 15337 054402  010504              3$:   MOV     R5,R4                     ; BIT.CNT,*                7151
 15338 054404  006204                      ASR     R4
 15339 054406  006204                      ASR     R4
 15340 054410  006204                      ASR     R4
 15341 054412  012703  000006              MOV     #6,R3
 15342 054416  060603                      ADD     SP,R3                     ; PROM.DATA,*
 15343 054420  060304                      ADD     R3,R4
 15344 054422  010446                      MOV     R4,-(SP)
 15345 054424  010546                      MOV     R5,-(SP)                  ; BIT.CNT,*
 15346 054426  042716  177770              BIC     #177770,(SP)
 15347 054432  012746  000001              MOV     #1,-(SP)
```

```
15349                                   ;ML4AD
15350                                   ;                TEST CODE SECTION
15351
15352 054436 005046                             CLR     -(SP)
15353 054440 004767 127170                      JSR     PC,BL$GT2
15354 054444 062706 000010                      ADD     #10,SP
15355 054450 005300                             DEC     R0
15356 054452 001002                             BNE     4$
15357 054454 005266 000002                      INC     2(SP)                   ; LOW.CNT
15358 054460 005205                     4$:     INC     R5                      ; BIT.CNT                       7149
15359 054462 020527 000011                      CMP     R5,#11                  ; BIT.CNT,*
15360 054466 003745                             BLE     3$
15361 054470 016602 000006                      MOV     6(SP),R2                ; PROM.DATA,HIGH.CNT            7153
15362 054474 006202                             ASR     R2                      ; HIGH.CNT
15363 054476 006202                             ASR     R2                      ; HIGH.CNT
15364 054500 000302                             SWAB    R2                      ; HIGH.CNT
15365 054502 042702 177770                      BIC     #177770,R2              ; *,HIGH.CNT
15366 054506 132766 000040 000007               BITB    #40,7(SP)               ; *,PROM.DATA+1                 7155
15367 054514 001401                             BEQ     5$
15368 054516 005202                             INC     R2                      ; HIGH.CNT
15369 054520 132766 000100 000007       5$:     BITB    #100,7(SP)              ; *,PROM.DATA+1                 7157
15370 054526 001401                             BEQ     6$
15371 054530 005202                             INC     R2                      ; HIGH.CNT
15372 054532 005005                     6$:     CLR     R5                      ;                               7159
15373 054534 105766 000007                      TSTB    7(SP)                   ; PROM.DATA+1
15374 054540 100002                             BPL     7$
15375 054542 005205                             INC     R5
15376 054544 005202                             INC     R2                      ; HIGH.CNT
15377 054546 006005                     7$:     ROR     R5                      ;                               7161
15378 054550 103003                             BCC     8$
15379 054552 012766 000001 000004               MOV     #1,4(SP)                ; *,GTR.FLG
15380 054560 004767 141626              8$:     JSR     PC,DAT.DM.XFER          ;                               7163
15381 054564 152777 000100 140732               BISB    #100,@ML.REG+120        ;                               7164
15382 054572 016605 000006                      MOV     6(SP),R5                ; PROM.DATA,*                   7165
15383 054576 010577 141032                      MOV     R5,@ML.REG+230
15384 054602 012777 000061 140574               MOV     #61,@ML.REG             ;                               7166
15385 054610 152777 000020 140706               BISB    #20,@ML.REG+120         ;                               7167
15386 054616 032777 040000 140640               BIT     #40000,@ML.REG+60       ;                               7169
15387 054624 001514                             BEQ     12$
15388 054626 026627 000004 000001               CMP     4(SP),#1                ; GTR.FLG,*                     7173
15389 054634 001050                             BNE     11$
15390 054636 010516                             MOV     R5,(SP)                 ; *,TEMP                        7176
15391 054640 042716 015777                      BIC     #15777,(SP)             ; *,TEMP                        7177
15392 054644 021627 162000                      CMP     (SP),#-16000            ; TEMP,*                        7179
15393 054650 001003                             BNE     9$
15394 054652 026602 000002                      CMP     2(SP),R2                ; LOW.CNT,HIGH.CNT              7183
15395 054656 002002                             BGE     10$
15396 054660 000167 000516              9$:     JMP     19$
15397 054664 005267 140506              10$:    INC     P.CNT                   ;                               7186
15398 054670 026767 140502 140502               CMP     P.CNT,LIMIT
15399 054676 003146                             BGT     14$
15400 054700 104455                             TRAP    55                      ;                               7188
15401 054702 000110                             .WORD   110
15402 054704 013012                             .WORD   ARR.DAT
15403 054706 026302                             .WORD   DUMPER
```

```
                                                                                29-Mar-1982 16:23:04   TOPS
                                                                                29-Mar-1982 16:21:03   PA:<
15405                                   ;ML4AD
15406                                   ;                TEST CODE SECTION
15407
15408 054710  012746  010774                     MOV     #WRD.24,-(SP)         ;                        7189
15409 054714  012746  011076                     MOV     #WRD.33,-(SP)
15410 054720  012746  010560                     MOV     #WRD.6,-(SP)
15411 054724  012746  011070                     MOV     #WRD.32,-(SP)
15412 054730  012746  011714                     MOV     #PHR.5,-(SP)
15413 054734  012746  011102                     MOV     #WRD.34,-(SP)
15414 054740  012746  010322                     MOV     #SIX.FMT,-(SP)
15415 054744  012746  000007                     MOV     #7,-(SP)
15416 054750  010600                           MOV     SP,R0                 ; SP,*
15417 054752  104414                           TRAP    14
15418 054754  000546                           BR      15$                   ;
15419 054756  026602  000002            11$:    CMP     2(SP),R2              ; LOW.CNT,HIGH.CNT       7190
15420 054762  001336                           BNE     9$                                             7199
15421 054764  005267  140406                     INC     P.CNT
15422 054770  026767  140402  140402           CMP     P.CNT,LIMIT           ;                        7201
15423 054776  003146                           BGT     17$
15424 055000  104455                           TRAP    55
15425 055002  000111                           .WORD   111                   ;                        7203
15426 055004  013012                           .WORD   ARR.DAT
15427 055006  026302                           .WORD   DUMPER
15428 055010  012746  010774                     MOV     #WRD.24,-(SP)
15429 055014  012746  011076                     MOV     #WRD.33,-(SP)         ;                        7204
15430 055020  012746  010560                     MOV     #WRD.6,-(SP)
15431 055024  012746  011070                     MOV     #WRD.32,-(SP)
15432 055030  012746  011714                     MOV     #PHR.5,-(SP)
15433 055034  012746  011102                     MOV     #WRD.34,-(SP)
15434 055040  012746  010322                     MOV     #SIX.FMT,-(SP)
15435 055044  012746  000007                     MOV     #7,-(SP)
15436 055050  010600                           MOV     SP,R0                 ; SP,*
15437 055052  104414                           TRAP    14
15438 055054  000546                           BR      18$                   ;
15439 055056  026627  000004  000001    12$:    CMP     4(SP),#1              ; GTR.FLG,*              7205
15440 055064  001103                           BNE     16$                                            7214
15441 055066  010516                           MOV     R5,(SP)               ; *,TEMP                 7217
15442 055070  042716  015777                     BIC     #15777,(SP)           ; *,TEMP                 7218
15443 055074  021627  162000                     CMP     (SP),#-16000          ; TEMP,*                 7220
15444 055100  001435                           BEQ     13$
15445 055102  005267  140270                     INC     P.CNT                 ;                        7222
15446 055106  026767  140264  140264           CMP     P.CNT,LIMIT
15447 055114  003153                           BGT     21$
15448 055116  104455                           TRAP    55
15449 055120  000112                           .WORD   112                   ;                        7224
15450 055122  013012                           .WORD   ARR.DAT
15451 055124  026302                           .WORD   DUMPER
15452 055126  012746  010774                     MOV     #WRD.24,-(SP)
15453 055132  012746  011076                     MOV     #WRD.33,-(SP)         ;                        7225
15454 055136  012746  010552                     MOV     #WRD.5,-(SP)
15455 055142  012746  011070                     MOV     #WRD.32,-(SP)
15456 055146  012746  011610                     MOV     #PHR.1,-(SP)
15457 055152  012746  011102                     MOV     #WRD.34,-(SP)
15458 055156  012746  010322                     MOV     #SIX.FMT,-(SP)
15459 055162  012746  000007                     MOV     #7,-(SP)
```

```
15461                                   ;ML4AD
15462                                   ;               TEST CODE SECTION
15463
15464 055166 010600                             MOV     SP,R0           ; SP,*
15465 055170 104414                             TRAP    14
15466 055172 000477                             BR      18$             ;                       7226
15467 055174 026602 000002              13$:    CMP     2(SP),R2        ; LOW.CNT,HIGH.CNT      7231
15468 055200 002100                             BGE     19$             ;                       7233
15469 055202 005267 140170                      INC     P.CNT
15470 055206 026767 140164 140164               CMP     P.CNT,LIMIT     ;                       7235
15471 055214 003113                     14$:    BGT     21$
15472 055216 104455                             TRAP    55              ;                       7236
15473 055220 000113                             .WORD   113
15474 055222 013012                             .WORD   ARR.DAT
15475 055224 026302                             .WORD   DUMPER
15476 055226 012746 010774                      MOV     #WRD.24,-(SP)
15477 055232 012746 011076                      MOV     #WRD.33,-(SP)
15478 055236 012746 010552                      MOV     #WRD.5,-(SP)
15479 055242 012746 011070                      MOV     #WRD.32,-(SP)
15480 055246 012746 011610                      MOV     #PHR.1,-(SP)
15481 055252 012746 011102                      MOV     #WRD.34,-(SP)
15482 055256 012746 010322                      MOV     #SIX.FMT,-(SP)
15483 055262 012746 000007                      MOV     #7,-(SP)
15484 055266 010600                             MOV     SP,R0           ; SP,*
15485 055270 104414                             TRAP    14
15486 055272 000437                     15$:    BR      18$             ;                       7237
15487 055274 026602 000002              16$:    CMP     2(SP),R2        ; LOW.CNT,HIGH.CNT      7246
15488 055300 001440                             BEQ     19$             ;                       7248
15489 055302 005267 140070                      INC     P.CNT
15490 055306 026767 140064 140064               CMP     P.CNT,LIMIT     ;                       7250
15491 055314 003053                     17$:    BGT     21$
15492 055316 104455                             TRAP    55              ;                       7251
15493 055320 000154                             .WORD   154
15494 055322 013012                             .WORD   ARR.DAT
15495 055324 026302                             .WORD   DUMPER
15496 055326 012746 010774                      MOV     #WRD.24,-(SP)
15497 055332 012746 011076                      MOV     #WRD.33,-(SP)
15498 055336 012746 010552                      MOV     #WRD.5,-(SP)
15499 055342 012746 011070                      MOV     #WRD.32,-(SP)
15500 055346 012746 011610                      MOV     #PHR.1,-(SP)
15501 055352 012746 011102                      MOV     #WRD.34,-(SP)
15502 055356 012746 010322                      MOV     #SIX.FMT,-(SP)
15503 055362 012746 000007                      MOV     #7,-(SP)
15504 055366 010600                             MOV     SP,R0           ; SP,*
15505 055370 104414                             TRAP    14
15506 055372 012701 000001              18$:    MOV     #1,R1           ; *,ERR.FLG             7252
15507 055376 062706 000020                      ADD     #20,SP          ;                       7248
15508 055402 020127 000001              19$:    CMP     R1,#1           ; ERR.FLG,*             7259
15509 055406 001011                             BNE     20$
15510 055410 010546                             MOV     R5,-(SP)        ;                       7262
15511 055412 012746 007002                      MOV     #FMT.7,-(SP)
15512 055416 012746 000002                      MOV     #2,-(SP)
15513 055422 010600                             MOV     SP,R0           ; SP,*
15514 055424 104414                             TRAP    14
15515 055426 062706 000006                      ADD     #6,SP           ;                       7261
```

```
15517                                   ;ML4AD
15518                                   ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS
15519                                                                                                      29-Mar-1982 16:21:03    PA:<
15520 055432  104467                     20$:    TRAP    67
15521 055434  006000                             ROR     R0                                .                                      7263
15522 055436  103002                             BHIS    21$
15523 055440  000167  176666                     JMP     2$
15524 055444  026627  000006  177777   21$:    CMP     6(SP),#-1                         ; PROM.DATA,*                          7267
15525 055452  001402                             BEQ     22$
15526 055454  000167  176646                     JMP     1$
15527 055460  062706  000010           22$:    ADD     #10,SP                            ;                                      7095
15528 055464  000207                             RTS     PC
15529
15530                                   ; Routine Size:  313 words
15531                                   ; Maximum stack depth per invocation:  18 words
15536
15537
15541
15542                                           .SBTTL  T26 TEST CODE SECTION
15546 055466                            T26::
15547 055466  004767  176612             1$:    JSR     PC,$T26                           ;                                      7267
15548 055472  104466                             TRAP    66
15549 055474  006000                             ROR     R0
15550 055476  103773                             BLO     1$
15551 055500  000207                             RTS     PC
15552
15553                                   ; Routine Size:  6 words
15554                                   ; Maximum stack depth per invocation:  0 words
15559
15560
15561 ;          7270  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (79)

```
15563 ;ML4AD
15564 ;                 TEST CODE SECTION
15565
15566 ;          7271  !
15567 ;          7272  !
15568 ;          7273  BGNTST;
15569 ;          7274
15570 ;          7275  !++
15571 ;          7276  ! TEST NUMBER:  TST 27
15572 ;          7277  !
15573 ;          7278  ! TEST NAME:    INITIAL ARRAY TEST
15574 ;          7279  !
15575 ;          7280  ! TEST DESCRIPTION:
15576 ;          7281  !
15577 ;          7282  !       DUE TO THE NATURE OF THE DEVICE
15578 ;          7283  !       THERE EXISTS KNOWN BAD ARRAY
15579 ;          7284  !       DATA LOCATIONS.
15580 ;          7285  !
15581 ;          7286  !       THEREFORE TO INITIALLY TEST THE ARRAYS'
15582 ;          7287  !       TIMING AND CONTROL LOGIC A BAD
15583 ;          7288  !       NIBBLE THRESHOLD OF 36 BAD NIBBLES
15584 ;          7289  !       OUT OF 100 NIBBLES TESTED WILL BE
15585 ;          7290  !       TOLERATED BEFORE DETERMINING CONTROL
15586 ;          7291  !       LOGIC TO BE IN ERROR.
15587 ;          7292  !
15588 ;          7293  !       THE ARRAYS' ARE INITIALLY TESTED BY:
15589 ;          7294  !
15590 ;          7295  !       1.  VIA DAT_DM MOD WRITE DATA PATTERNS
15591 ;          7296  !           OF 1'S AND 0'S TO 5 ARRAY
15592 ;          7297  !           WORDS.
15593 ;          7298  !
15594 ;          7299  !       2.  TEST EACH NIBBLE (4 BITS) FOR
15595 ;          7300  !           1'S AND 0'S AND COUNT EACH BAD
15596 ;          7301  !           NIBBLE ENCOUNTERED.
15597 ;          7302  !
15598 ;          7303  !       3.  IF ACCUMULATED BAD NIBBLES
15599 ;          7304  !           EXCEED 36 THEN REPORT AN ERROR.
15600 ;          7305  !
15601 ;          7306  !--
15602 ;          7307
15603 ;          7308  local
15604 ;          7309      TST_PAT,                          !TEST PATTERN
15605 ;          7310      BAD_NIB_CNT,                      !NUMBER OF BAD NIBBLES FOUND
15606 ;          7311      ERR_FLG;                          !ERROR FLAG
15607 ;          7312
15608 ;          7313  TST_PAT = ONES;
15609 ;          7314  BAD_NIB_CNT = ZEROES;
15610 ;          7315
15611 ;          7316  incr TWICE from 0 to 1 do             !REPEAT LOOP TWICE
15612 ;          7317      begin
15613 ;          7318      BGNSUB;
15614 ;          7319      CLR_MBUS;
15615 ;          7320      MLD1 = .TST_PAT;                  !LOAD TEST PATTERN INTO DIAG REGISTERS
15616 ;          7321      MLD2 = .TST_PAT;
15617 ;          7322      MLE2 = .TST_PAT;
```

```
15619 ;ML4AD
15620 ;
15621 ;                    TEST CODE SECTION
15622 ;          7323      DAT_DM = ONE;                          !SET DATA DIAG MODE
15623 ;          7324      FIRST_BLK_XFER ();                     !SET UP A FIRST BLK XFERR
15624 ;          7325      MLCS1 = write;                         !DO A MBUS WRITE FUNCTION
15625 ;          7326
15626 ;          7327      incr CNT from 0 to 4 do                !CLOCK 5 WORDS INTO MEMORY
15627 ;          7328          begin
15628 ;          7329          DELAY (ONE_US);
15629 ;          7330          DAT_CLK = ONE;
15630 ;          7331          end;
15631 ;          7332
15632 ;          7333      CLR_MBUS;
15633 ;          7334      DAT_DM = ONE;                          !SET DATA DIAG MODE
15634 ;          7335      FIRST_BLK_XFER ();                     !SET UP A FIRST BLK XFERR
15635 ;          7336      MLCS1 = read;                          !DO A READ FUNCTION
15636 ;          7337      DELAY (ONE_US);
15637 ;          7338
15638 ;          7339      incr ARR_WRD from 0 to 4 do            !READ THE 5 WORD IN MEMORY
15639 ;          7340          begin
15640 ;          7341          DAT_CLK = ONE;                     !CLOCK OUT A WORD INTO THE DIAG REGS
15641 ;          7342          DELAY (ONE_US);
15642 ;          7343          RD_LNG_WRD;                        !READ THE DIAG REGISTERS
15643 ;          7344
15644 ;          7345          incr NIB_PTR from 0 to 9 do        !READ THE 10 NIBBLES IN THE WORD
15645 ;          7346              begin
15646 ;          7347              TST_LNG_WRD (.NIB_PTR, .TST_PAT, ERR_FLG);  !COMPARE TST PAT TO NIBBLE UNDER TEST
15647 ;          7348
15648 ;          7349              if .ERR_FLG IS_SET then BAD_NIB_CNT = .BAD_NIB_CNT + 1;
15649 ;          7350
15650 ;          7351                                             !INCREMENT BAD_NIBBLE COUNT IFERR_FLG SET
15651 ;          7352              end;
15652 ;          7353
15653 ;          7354          end;
15654 ;          7355
15655 ;          7356      TST_PAT = not .TST_PAT;                !REPEAT WITH COMPLIMENT TST PAT
15656 ;          7357      ENDSUB;
15657 ;          7358      end;
15658 ;          7359
15659 ;          7360  if .BAD_NIB_CNT gtr 36                     !SEE IF 36 OUT OF THE 100 XFERRED WHERE BAD
15660 ;          7361  then
15661 ;          7362      begin                                  !ERROR IF GTR 36
15662 ;          7363      ERRDF (77, ASYNC, DUMPER);
15663 ;          7364      PRINTB (FIV_FMT, WRD_22, PHR_4, WRD_12, WRD_45, FNC_14);
15664 ;          7365      DODU (.ML_LUN);
15665 ;          7366      DOCLN;
15666 ;          7367      end;
15667 ;          7368
15668 ;          7369  ENDTST;
15672
15673
                                  .SBTTL  $T27 TEST CODE SECTION
```

```
15675                                   ;ML4AD
15676                                   ;                       TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
15677                                                                                                               29-Mar-1982 16:21:03    PA:<
15681 055502  004167  127104            ST27:   JSR     R1,$SAVE5                                                                           7269
15682 055506  024646                            CMP     -(SP),-(SP)             ;
15683 055510  012701  177777                    MOV     #-1,R1                  ; *,TST.PAT                                                 7313
15684 055514  005046                            CLR     -(SP)                   ; BAD.NIB.CNT                                               7314
15685 055516  005005                            CLR     R5                      ; TWICE                                                     7316
15686 055520  104402                    1$:     TRAP    2                       ;                                                           7317
15687 055522  152777  000040  137714            BISB    #40,@ML.REG+40          ;                                                           7318
15688 055530  016704  140276                    MOV     ML.DUT,R4
15689 055534  042704  177770                    BIC     #177770,R4
15690 055540  142777  000007  137676            BICB    #7,@ML.REG+40
15691 055546  150477  137672                    BISB    R4,@ML.REG+40
15692 055552  010177  140016                    MOV     R1,@ML.REG+170          ; TST.PAT,*                                                 7320
15693 055556  010177  140022                    MOV     R1,@ML.REG+200          ; TST.PAT,*                                                 7321
15694 055562  010177  137776                    MOV     R1,@ML.REG+160          ; TST.PAT,*                                                 7322
15695 055566  152777  000010  137730            BISB    #10,@ML.REG+120         ;                                                           7323
15696 055574  004767  142462                    JSR     PC,FIRST.BLK.XFER       ;                                                           7324
15697 055600  012777  000061  137576            MOV     #61,@ML.REG             ;                                                           7325
15698 055606  005002                            CLR     R2                      ; CNT                                                       7327
15699 055610  012703  000001            2$:     MOV     #1,R3                   ; *,$$TMP2                                                  7329
15700 055614  001411                    3$:     BEQ     6$
15701 055616  016704  124274                    MOV     L$DLY,R4                ; *,$$TMP1
15702 055622  001404                            BEQ     5$
15703 055624  005066  000004            4$:     CLR     4(SP)                   ; $$TMP
15704 055630  005304                            DEC     R4                      ; $$TMP1
15705 055632  001374                            BNE     4$
15706 055634  005303                    5$:     DEC     R3                      ; $$TMP2
15707 055636  000766                            BR      3$
15708 055640  152777  000020  137656    6$:     BISB    #20,@ML.REG+120         ;                                                           7330
15709 055646  005202                            INC     R2                      ; CNT                                                       7327
15710 055650  020227  000004                    CMP     R2,#4                   ; CNT,*
15711 055654  003755                            BLE     2$
15712 055656  152777  000040  137560            BISB    #40,@ML.REG+40          ;                                                           7331
15713 055664  016704  140142                    MOV     ML.DUT,R4
15714 055670  042704  177770                    BIC     #177770,R4
15715 055674  142777  000007  137542            BICB    #7,@ML.REG+40
15716 055702  150477  137536                    BISB    R4,@ML.REG+40
15717 055706  152777  000010  137610            BISB    #10,@ML.REG+120         ;                                                           7334
15718 055714  004767  142342                    JSR     PC,FIRST.BLK.XFER       ;                                                           7335
15719 055720  012777  000071  137456            MOV     #71,@ML.REG             ;                                                           7336
15720 055726  012703  000001                    MOV     #1,R3                   ; *,$$TMP2                                                  7337
15721 055732  001411                    7$:     BEQ     10$
15722 055734  016704  124156                    MOV     L$DLY,R4                ; *,$$TMP1
15723 055740  001404                            BEQ     9$
15724 055742  005066  000004            8$:     CLR     4(SP)                   ; $$TMP
15725 055746  005304                            DEC     R4                      ; $$TMP1
15726 055750  001374                            BNE     8$
15727 055752  005303                    9$:     DEC     R3                      ; $$TMP2
15728 055754  000766                            BR      7$
15729 055756  005002                    10$:    CLR     R2                      ; ARR.WRD                                                   7339
```

```
15731                                   ;ML4AD
15732                                   ;                       TEST CODE SECTION
15733
15734 055760 152777 000020 137536 11$:  BISB    #20,@ML.REG+120
15735 055766 012703 000001              MOV     #1,R3                   ;
15736 055772 001411                     BEQ     15$             12$:    ; *,$$TMP2                      7341
15737 055774 016704 124116              MOV     L$DLY,R4                ; *,$$TMP1                      7342
15738 056000 001404                     BEQ     14$
15739 056002 005066 000004      13$:    CLR     4(SP)                   ; $$TMP
15740 056006 005304                     DEC     R4                      ; $$TMP1
15741 056010 001374                     BNE     13$
15742 056012 005303             14$:    DEC     R3                      ; $$TMP2
15743 056014 000766                     BR      12$
15744 056016 017767 137552 135350 15$:  MOV     @ML.REG+170,D1.TEMP
15745 056024 017767 137554 135344       MOV     @ML.REG+200,D2.TEMP
15746 056032 017767 137526 135340       MOV     @ML.REG+160,E2.TEMP
15747 056040 005004                     CLR     R4
15748 056042 010446             16$:    MOV     R4,-(SP)                ; NIB.PTR                       7345
15749 056044 010146                     MOV     R1,-(SP)                ; NIB.PTR,*                     7347
15750 056046 012746 000010              MOV     #10,-(SP)               ; TST.PAT,*
15751 056052 060616                     ADD     SP,(SP)
15752 056054 004767 142316              JSR     PC,TST.LNG.WRD          ; ERR.FLG,*
15753 056060 026627 000010 000001       CMP     10(SP),#1               ; ERR.FLG,*                     7349
15754 056066 001002                     BNE     17$
15755 056070 005266 000006              INC     6(SP)                   ; BAD.NIB.CNT
15756 056074 062706 000006      17$:    ADD     #6,SP                   ;                               7346
15757 056100 005204                     INC     R4                      ; NIB.PTR                       7345
15758 056102 020427 000011              CMP     R4,#11                  ; NIB.PTR,*
15759 056106 003755                     BLE     16$
15760 056110 005202                     INC     R2                      ; ARR.WRD                       7339
15761 056112 020227 000004              CMP     R2,#4                   ; ARR.WRD,*
15762 056116 003720                     BLE     11$
15763 056120 005101                     COM     R1                      ; TST.PAT                       7356
15764 056122 104467                     TRAP    67
15765 056124 006000                     ROR     R0
15766 056126 103002                     BHIS    19$
15767 056130 000167 177364      18$:    JMP     1$
15768 056134 005205             19$:    INC     R5                      ; TWICE                         7316
15769 056136 020527 000001              CMP     R5,#1                   ; TWICE,*
15770 056142 003772                     BLE     18$
15771 056144 021627 000044              CMP     (SP),#44                ; BAD.NIB.CNT,*                 7360
15772 056150 003432                     BLE     20$
15773 056152 104455                     TRAP    55                      ;                               7363
15774 056154 000115                     .WORD   115
15775 056156 012706                     .WORD   ASYNC
15776 056160 026302                     .WORD   DUMPER
15777 056162 012746 012344              MOV     #FNC.14,-(SP)           ;                               7364
15778 056166 012746 011214              MOV     #WRD.45,-(SP)
15779 056172 012746 010650              MOV     #WRD.12,-(SP)
15780 056176 012746 011676              MOV     #PHR.4,-(SP)
15781 056202 012746 010752              MOV     #WRD.22,-(SP)
15782 056206 012746 010304              MOV     #FIV.FMT,-(SP)
15783 056212 012746 000006              MOV     #6,-(SP)
15784 056216 010600                     MOV     SP,R0                   ; SP,*
15785 056220 104414                     TRAP    14
```

SEQ 0339

```
15787                                   ;ML4AD
15788                                   ;                       TEST CODE SECTION                                       29-Mar-1982 16:23:04    TOPS
15789                                                                                                                   29-Mar-1982 16:21:03    PA:<
15790 056222  016700  137602                      MOV     ML.LUN,R0                       ;                                                       7365
15791 056226  104451                              TRAP    51
15792 056230  104444                              TRAP    44
15793 056232  062706  000016                      ADD     #16,SP                          ;
15794 056236  062706  000006            20$:      ADD     #6,SP                           ;                                                       7362
15795 056242  000207                              RTS     PC                              ;                                                       7269
15796
15797                                   ; Routine Size:  177 words
15798                                   ; Maximum stack depth per invocation:  16 words
15803
15804
15808
15809                                             .SBTTL  T27 TEST CODE SECTION
15813 056244                            T27::
15814 056244  004767  177232            1$:       JSR     PC,$T27                         ;                                                       7367
15815 056250  104466                              TRAP    66
15816 056252  006000                              ROR     R0
15817 056254  103773                              BLO     1$
15818 056256  000207                              RTS     PC
15819
15820                                   ; Routine Size:  6 words
15821                                   ; Maximum stack depth per invocation:  0 words
15826
15827
15828 ;           7370  !<BLF/PAGE>
```

SEQ 0340

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (80)

```
15830 ;ML4AD
15831 ;                 TEST CODE SECTION
15832
15833 ;          7371   !
15834 ;          7372   BGNTST;
15835 ;          7373
15836 ;          7374   !++
15837 ;          7375   ! TEST NUMBER:  TST 28
15838 ;          7376   !
15839 ;          7377   ! TEST NAME:    PROM SELECTION TEST
15840 ;          7378   !
15841 ;          7379   ! TEST DESCRIPTION:
15842 ;          7380   !       DUE TO THE NATURE OF THE DEVICE
15843 ;          7381   !       AND OF THE ARRAY MODULES' UV
15844 ;          7382   !       PROMS, ONLY PROM READS ARE
15845 ;          7383   !       ALLOWED DURING DIAG TESTING.
15846 ;          7384   !
15847 ;          7385   !       THEREFORE THE ARRAY MODULE UV PROMS
15848 ;          7386   !       ARE TESTED FOR UNIQUE SELECTION BY:
15849 ;          7387   !
15850 ;          7388   !       1.  AT EACH PRESENT ARRAY MODULE WRITE 127 ARRAY WORDS WITH 1'S/0'S PATTERN.
15851 ;          7389   !
15852 ;          7390   !       2.  READ THE UV PROMS AT THEIR RESPECTIVE ARRAY WORD LOCATION AND SEE IF
15853 ;          7391   !           THE PROMS MASK BAD NIBBLE LOCATIONS (ENCOUNTERED BAD NIBBLES
15854 ;          7392   !           INDICATES INCORRECT MASKING).  COUNT EACH BAD NIBBLE ENCOUNTERED AT AN
15855 ;          7393   !           ARRAY MODULE.
15856 ;          7394   !
15857 ;          7395   !       3.  ALLOW A THRESHOLD OF 5 BAD NIBBLES AT ANY ARRAY MODULE.
15858 ;          7396   !
15859 ;          7397   !       4.  REPORT PROM SEL ERRORS AT RESPECTIVE ARRAY MODULE IF THE
15860 ;          7398   !           THRESHOLD IS EXCEEDED.
15861 ;          7399   !
15862 ;          7400   ! IMPLICIT INPUTS:
15863 ;          7401   !       PD_TEMP:
15864 ;          7402   !       A BIT VECTOR OF 16 BITS WHERE
15865 ;          7403   !       THE READ PROM DATA IS STORED
15866 ;          7404   !       AND ACCESSED FROM.
15867 ;          7405   !
15868 ;          7406   !       IO_BUF
15869 ;          7407   !       A VECTOR OF 256 WORDS WHERE
15870 ;          7408   !       DATA FOR MBUS READS AND WRITE
15871 ;          7409   !       FUNCTION ARE FOUND.
15872 ;          7410   !
15873 ;          7411   !--
15874 ;          7412
15875 ;          7413   local
15876 ;          7414       DODU_FLG,                       !DROP UNIT FLG
15877 ;          7415       ERR_FLG,                        !ERROR FLG
15878 ;          7416       TST_PAT,                        !TEST PATTERN
15879 ;          7417       ERR_CNT;                        !ERROR COUNT
15880 ;          7418
15881 ;          7419   DODU_FLG = ZERO;
15882 ;          7420   TST_PAT = ONES;
15883 ;          7421
15884 ;          7422   !
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (80)

```
15886 ;ML4AD
15887 ;                 TEST CODE SECTION
15888 ;
15889 ;          7423   !VER CZMLAD CHANGED TEST TO UNSIGNED TEST
15890 ;          7424   !
15891 ;          7425   incrU ARR_SEL from 0 to .LST_ARR by .ARR_INC do        !TEST ALL PRESENT ARRAYS
15892 ;          7426       begin
15893 ;          7427       ERR_CNT = ZERO;
15894 ;          7428       BGNSUB;
15895 ;          7429
15896 ;          7430       incr TWICE from 0 to 1 do                          !REPEAT LOOP TWICE
15897 ;          7431           begin
15898 ;          7432           CLR_MBUS;
15899 ;          7433           MLDT = .TST_PAT;                               !LOAD DATA DIAG REGISTERS WITH TST_PAT
15900 ;          7434           MLD2 = .TST_PAT;
15901 ;          7435           MLE2 = .TST_PAT;
15902 ;          7436           DAT_DM = ONE;                                  !SET DATA DIAG MODE
15903 ;          7437           MLWC = not 255;                                !LOAD WORD COUNT
15904 ;          7438           MLBA = IO_BUF;                                 !LOAD UBUS ADRS
15905 ;          7439           MLDA = .ARR_SEL;                               !LOAD SECTOR
15906 ;          7440           MLCS1 = write;                                 !DO A WRITE FUNCTION
15907 ;          7441
15908 ;          7442           incr CNT from 0 to 127 do                      !CLOCK IN 127 WORDS
15909 ;          7443               begin
15910 ;          7444               DELAY (ONE_US);
15911 ;          7445               DAT_CLK = ONE;
15912 ;          7446               end;
15913 ;          7447
15914 ;          7448           CLR_MBUS;
15915 ;          7449           DAT_DM = ONE;                                  !SET DATA DIAG MODE
15916 ;          7450           MLWC = not 255;                                !LOAD WORD COUNT
15917 ;          7451           MLBA = IO_BUF;                                 !LOAD UBUS ADRS
15918 ;          7452           MLDA = .ARR_SEL;                               !LOAD SECTOR
15919 ;          7453           MLCS1 = read;                                  !DO A READ FUNCTION
15920 ;          7454           DELAY (ONE_US);
15921 ;          7455
15922 ;          7456           incr WD_CNT from 0 to 127 do                   !READ THE 10 WORDS
15923 ;          7457               begin
15924 ;          7458               PD_TEMP = .MLPD;                           !GET PROM DATA FOR THIS WORD
15925 ;          7459               DAT_CLK = ONE;                             !CLOCK THIS WORD INTO DIAG REG
15926 ;          7460               DELAY (ONE_US);
15927 ;          7461               RD_LNG_WRD;                                !READ DIAG REG FOR THIS WORD
15928 ;          7462
15929 ;          7463               incr NIB_PTR from 0 to 9 do                !LOOK AT ALL 10 NIBBLE
15930 ;          7464
15931 ;          7465                   if .PD_TEMP [.NIB_PTR] IS_NOT_SET      !FIND GOOD NIBBLES
15932 ;          7466                   then
15933 ;          7467                       begin
15934 ;          7468                       TST_LNG_WRD (.NIB_PTR, .TST_PAT, ERR_FLG);  !COMPARE NIBBLE TO TST_PAT
15935 ;          7469
15936 ;          7470                       if .ERR_FLG IS_SET then ERR_CNT = .ERR_CNT + 1;
15937 ;          7471
15938 ;          7472                                                          !INCREMENT ERROR COUNT IF ERROR FLG IS SET
15939 ;          7473                       end;
15940 ;          7474
```

```
                                                                                29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
                        TEST CODE SECTION                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (80)

                7475                end;
                7476
                7477            TST_PAT = not .TST_PAT;                     !REPEAT WITH COMPLIMENT DATA
                7478            end;
                7479
                7480        if .ERR_CNT gtr 5                               !ALLOW 5 ERROR BEFORE ERRORING
                7481        then
                7482            begin                                       !ERROR IF GTR 5
                7483            ERRDF (78, ARR_DAT, DUMPER);
                7484            PRINTB (THR_FMT, WRD_35, WRD_37, WRD_10);
                7485            PRINTB (FMT_9, .ARR_SEL);
                7486            DODU_FLG = ONE;
                7487            end;
                7488
                7489        ENDSUB;
                7490
                7491        if .DODU_FLG IS_SET                             !DROP THIS UNIT IF DODU_FLG IS SET
                7492        then
                7493            begin
                7494            DODU (.ML_LUN);
                7495            DOCLN;
                7496            end;
                7497
                7498        end;
                7499
                7500    ENDTST;
```

```
                                .SBTTL  $T28 TEST CODE SECTION
056260 004167 126326        $T28:   JSR     R1,$SAVE5                       ;                               7369
056264 162706 000016                SUB     #16,SP
056270 005066 000010                CLR     10(SP)                          ; DODU.FLG                      7419
056274 012702 177777                MOV     #-1,R2                          ; *,TST.PAT                     7420
056300 016766 135514 000006         MOV     LST.ARR,6(SP)                   ;                               7425
056306 016766 135472 000004         MOV     ARR.INC,4(SP)
056314 005001                       CLR     R1                              ; ARR.SEL
056316 000167 000666                JMP     25$
056322 005066 000002        1$:     CLR     2(SP)                           ; ERR.CNT                       7427
056326 104402               2$:     TRAP    2
056330 005016                       CLR     (SP)                            ; TWICE                         7430
056332 152777 000040 137104 3$:     BISB    #40,@ML.REG+40                  ;                               7431
056340 016705 137466                MOV     ML.DUT,R5
056344 042705 177770                BIC     #177770,R5
056350 142777 000007 137066         BICB    #7,@ML.REG+40
056356 150577 137062                BISB    R5,@ML.REG+40
056362 010277 137206                MOV     R2,@ML.REG+170                  ; TST.PAT,*                     7433
056366 010277 137212                MOV     R2,@ML.REG+200                  ; TST.PAT,*                     7434
```

SEQ 0343

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
15998
15999                                   ;ML4AD
16000                                   :                   TEST CODE SECTION
16001 056372 010277 137166                      MOV     R2,@ML.REG+160          ; TST.PAT,*                                 7435
16002 056376 152777 000010 137120               BISB    #10,@ML.REG+120         ;                                           7436
16003 056404 012777 177400 137002               MOV     #-400,@ML.REG+10        ;                                           7437
16004 056412 012777 014022 137004               MOV     #IO.BUF,@ML.REG+20      ;                                           7438
16005 056420 010177 137010                      MOV     R1,@ML.REG+30           ; ARR.SEL,*                                 7439
16006 056424 012777 000061 136752               MOV     #61,@ML.REG             ;                                           7440
16007 056432 005003                             CLR     R3                      ; CNT                                       7442
16008 056434 012704 000001              4$:     MOV     #1,R4                   ; *,$$TMP2                                  7444
16009 056440 001411                     5$:     BEQ     8$
16010 056442 016705 123450                      MOV     L$DLY,R5                ; *,$$TMP1
16011 056446 001404                             BEQ     7$
16012 056450 005066 000014              6$:     CLR     14(SP)                  ; $$TMP
16013 056454 005305                             DEC     R5                      ; $$TMP1
16014 056456 001374                             BNE     6$
16015 056460 005304                     7$:     DEC     R4                      ; $$TMP2
16016 056462 000766                             BR      5$
16017 056464 152777 000020 137032       8$:     BISB    #20,@ML.REG+120         ;                                           7445
16018 056472 005203                             INC     R3                      ; CNT                                       7442
16019 056474 020327 000177                      CMP     R3,#177                 ; CNT,*
16020 056500 003755                             BLE     4$
16021 056502 152777 000040 136734               BISB    #40,@ML.REG+40          ;                                           7446
16022 056510 016705 137316                      MOV     ML.DUT,R5
16023 056514 042705 177770                      BIC     #177770,R5
16024 056520 142777 000007 136716               BICB    #7,@ML.REG+40
16025 056526 150577 136712                      BISB    R5,@ML.REG+40
16026 056532 152777 000010 136764               BISB    #10,@ML.REG+120         ;                                           7449
16027 056540 012777 177400 136646               MOV     #-400,@ML.REG+10        ;                                           7450
16028 056546 012777 014022 136650               MOV     #IO.BUF,@ML.REG+20      ;                                           7451
16029 056554 010177 136654                      MOV     R1,@ML.REG+30           ; ARR.SEL,*                                 7452
16030 056560 012777 000071 136616               MOV     #71,@ML.REG             ;                                           7453
16031 056566 012704 000001                      MOV     #1,R4                   ; *,$$TMP2                                  7454
16032 056572 001411                     9$:     BEQ     12$
16033 056574 016705 123316                      MOV     L$DLY,R5                ; *,$$TMP1
16034 056600 001404                             BEQ     11$
16035 056602 005066 000014              10$:    CLR     14(SP)                  ; $$TMP
16036 056606 005305                             DEC     R5                      ; $$TMP1
16037 056610 001374                             BNE     10$
16038 056612 005304                     11$:    DEC     R4                      ; $$TMP2
16039 056614 000766                             BR      9$
16040 056616 005003                     12$:    CLR     R3                      ; WD.CNT                                    7456
16041 056620 017767 137010 136514       13$:    MOV     @ML.REG+230,PD.TEMP     ;                                           7458
16042 056626 152777 000020 136670               BISB    #20,@ML.REG+120         ;                                           7459
16043 056634 012704 000001                      MOV     #1,R4                   ; *,$$TMP2                                  7460
16044 056640 001411                     14$:    BEQ     17$
16045 056642 016705 123250                      MOV     L$DLY,R5                ; *,$$TMP1
16046 056646 001404                             BEQ     16$
16047 056650 005066 000014              15$:    CLR     14(SP)                  ; $$TMP
16048 056654 005305                             DEC     R5                      ; $$TMP1
16049 056656 001374                             BNE     15$
16050 056660 005304                     16$:    DEC     R4                      ; $$TMP2
16051 056662 000766                             BR      14$
16052 056664 017767 136704 134502       17$:    MOV     @ML.REG+170,D1.TEMP
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
16057 056672 017767 136706 134476               MOV     @ML.REG+200,D2.TEMP
16058 056700 017767 136660 134472               MOV     @ML.REG+160,E2.TEMP
16059 056706 005004                             CLR     R4
16060 056710 010405                     18$:    MOV     R4,R5                   ; NIB.PTR               7463
16061 056712 006205                             ASR     R5                      ; NIB.PTR,*             7465
16062 056714 006205                             ASR     R5
16063 056716 006205                             ASR     R5
16064 056720 062705 015342                      ADD     #PD.TEMP,R5
16065 056724 010546                             MOV     R5,-(SP)
16066 056726 010446                             MOV     R4,-(SP)                ; NIB.PTR,*
16067 056730 042716 177770                      BIC     #177770,(SP)
16068 056734 012746 000001                      MOV     #1,-(SP)
16069 056740 005046                             CLR     -(SP)
16070 056742 004767 124666                      JSR     PC,BL$GT2
16071 056746 062706 000010                      ADD     #10,SP
16072 056752 005700                             TST     R0
16073 056754 001017                             BNE     20$
16074 056756 010446                             MOV     R4,-(SP)                ; NIB.PTR,*
16075 056760 010246                             MOV     R2,-(SP)                ; TST.PAT,*             7468
16076 056762 012746 000020                      MOV     #20,-(SP)
16077 056766 060616                             ADD     SP,(SP)                 ; ERR.FLG,*
16078 056770 004767 141402                      JSR     PC,TST.LNG.WRD
16079 056774 026627 000020 000001               CMP     20(SP),#1               ; ERR.FLG,*             7470
16080 057002 001002                             BNE     19$
16081 057004 005266 000010                      INC     10(SP)                  ; ERR.CNT
16082 057010 062706 000006              19$:    ADD     #6,SP                   ;                       7467
16083 057014 005204                     20$:    INC     R4                      ; NIB.PTR               7463
16084 057016 020427 000011                      CMP     R4,#11                  ; NIB.PTR,*
16085 057022 003732                             BLE     18$
16086 057024 005203                             INC     R3                      ; WD.CNT
16087 057026 020327 000177                      CMP     R3,#177                 ; WD.CNT,*              7456
16088 057032 003672                             BLE     13$
16089 057034 005102                             COM     R2                      ; TST.PAT               7477
16090 057036 005216                             INC     (SP)                    ; TWICE                 7430
16091 057040 021627 000001                      CMP     (SP),#1                 ; TWICE,*
16092 057044 003002                             BGT     21$
16093 057046 000167 177260                      JMP     3$
16094 057052 026627 000002 000005       21$:    CMP     2(SP),#5                ; ERR.CNT,*             7480
16095 057060 003434                             BLE     22$
16096 057062 104455                             TRAP    55                      ;                       7483
16097 057064 000116                             .WORD   116
16098 057066 013012                             .WORD   ARR.DAT
16099 057070 026302                             .WORD   DUMPER
16100 057072 012746 010630                      MOV     #WRD.10,-(SP)           ;                       7484
16101 057076 012746 011122                      MOV     #WRD.37,-(SP)
16102 057102 012746 011110                      MOV     #WRD.35,-(SP)
16103 057106 012746 010256                      MOV     #THR.FMT,-(SP)
16104 057112 012746 000004                      MOV     #4,-(SP)
16105 057116 010600                             MOV     SP,R0                   ; SP,*
16106 057120 104414                             TRAP    14
16107 057122 010116                             MOV     R1,(SP)                 ; ARR.SEL,*             7485
16108 057124 012746 007064                      MOV     #FMT.9,-(SP)
```

```
16110                                   ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
16111                                   ;                       TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
16112
16113 057130  012746  000002                      MOV     #2,-(SP)
16114 057134  010600                              MOV     SP,R0                   ; SP,*
16115 057136  104414                              TRAP    14
16116 057140  012766  000C01  000026              MOV     #1,26(SP)               ; *,DODU.FLG                                    7486
16117 057146  062706  000016                      ADD     #16,SP                  ;                                               7482
16118 057152  104467                    22$:      TRAP    67                      ;                                               7487
16119 057154  006000                              ROR     R0
16120 057156  103002                              BHIS    23$
16121 057160  000167  177142                      JMP     2$
16122 057164  026627  000010  000001    23$:      CMP     10(SP),#1               ; DODU.FLG,*                                    7491
16123 057172  001004                              BNE     24$
16124 057174  016700  136630                      MOV     ML.LUN,R0               ;                                               7494
16125 057200  104451                              TRAP    51
16126 057202  104444                              TRAP    44
16127 057204  066601  000004              24$:    ADD     4(SP),R1                ; *,ARR.SEL
16128 057210  020166  000006              25$:    CMP     R1,6(SP)                ; ARR.SEL,*                                     7425
16129 057214  101002                              BHI     26$
16130 057216  000167  177100                      JMP     1$
16131 057222  062706  000016              26$:    ADD     #16,SP                  ;                                               7369
16132 057226  000207                              RTS     PC
16133
16134                                   ; Routine Size:  244 words
16135                                   ; Maximum stack depth per invocation:  20 words
16140
16141
16145
16146                                           .SBTTL  T28 TEST CODE SECTION
16150 057230                            T28::
16151 057230  004767  177024            1$:     JSR     PC,$T28                 ;                                               7498
16152 057234  104466                              TRAP    66
16153 057236  006000                              ROR     R0
16154 057240  103773                              BLO     1$
16155 057242  000207                              RTS     PC
16156
16157                                   ; Routine Size:  6 words
16158                                   ; Maximum stack depth per invocation:  0 words
16163
16164 ;         7501  !<BLF/PAGE>
```

SEQ 0346


```
;ML4AD
;                    TEST CODE SECTION

;          7502  !
;          7503  BGNTST;
;          7504
;          7505  !++
;          7506  ! TEST NUMBER:  TST 29
;          7507  !
;          7508  ! TEST NAME:    READ WRITE ARRAYS WITH PROM DATA
;          7509  !
;          7510  ! TEST DESCRIPTION:
;          7511  !       COMBINE THE READING OF ARRAY
;          7512  !       MODULE DATA WITH ARRAY MODULE UV PROM DATA AND
;          7513  !       FIND A GOOD BLOCK WHERE FURTHER
;          7514  !       TESTING WILL BE PERFORMED BY:
;          7515  !
;          7516  !       1.  STARTING AT BLOCK 0 WRITE THE BLOCK WITH SELECTED DATA PATTERNS
;          7517  !           AND READ THE BLOCK AVOIDING ANY BAD NIBBLES POINTED TO BY THE
;          7518  !           PROM DATA.
;          7519  !
;          7520  !           SET ERROR FLAG IF ANY BAD NIBBLES ARE ENCOUNTERED IN BLOCK.
;          7521  !
;          7522  !       2.  REPEAT WRITING/READING THIS BLOCK UNTIL ALL PATTERNS ARE TESTED
;          7523  !           OR THE ERROR FLAG IS SET.
;          7524  !
;          7525  !       3.  IF ALL PATTERN HAVE BEEN TESTED AND THE ERROR FLAG IS NOT SET
;          7526  !           THEN SAVE THIS BLOCK ADDRESS AS THE GOOD BLOCK ADRS AND EXIT TEST.
;          7527  !
;          7528  !       4.  ELSE IF THE ERROR FLG HAS SET THEN REPEAT TEST AT THE NEXT ROW.
;          7529  !           REPEAT UNTIL A GOOD BLOCK IS FOUND OR LAST ROW IS REACHED.
;          7530  !
;          7531  !       5.  IF NO GOOD BLOCK IS FOUND BY LAST ROW THEN REPORT ERROR AND
;          7532  !           EXIT TEST.
;          7533  !
;          7534  ! IMPLICIT INPUTS:
;          7535  !       RAS_INC
;          7536  !       LOADED DURING THE INITIALIZATION CODE AND CONTAINS THE ROW ADDRESS
;          7537  !       INCREMENT VALUE FOR THIS DRIVE.
;          7538  !
;          7539  !
;          7540  !       PD_TEMP:
;          7541  !       A BITVECTOR OF 16 BITS WHERE THE READ PROM DATA IS STORED AND
;          7542  !       ACCESSED FROM.
;          7543  !
;          7544  !       IO_BUF:
;          7545  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE FUNCTION
;          7546  !       ARE FOUND.
;          7547  !
;          7548  !
;          7549  !--
;          7550
;          7551  local
;          7552      WRD_CNT,                                    !WORD COUNT
;          7553      NIB_PTR,                                    !NIBBLE POINTER
```

```
;ML4AD
;                    TEST CODE SECTION                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (81)

;          7554      PASS_CNT,                                  !PASS COUNT
;          7555      NIB_PAT,                                   !NIBBLE PATTERN
;          7556      DONE_FLG,                                  !DONE FLAG
;          7557      ERR_FLG,                                   !ERROR FLAG
;          7558      SECTOR_NO;                                 !SECTOR NUMBER
;          7559
;          7560  BGNSUB;
;          7561  PASS_CNT = -1;
;          7562  SECTOR_NO = ZEROES;
;          7563  DONE_FLG = ZERO;
;          7564
;          7565  do                                             !THIS LOOP DETERMINES WHEN TO STOP
;          7566      begin
;          7567
;          7568      do                                         !THIS LOOP RUNS THE PATTERNS
;          7569          begin
;          7570          PASS_CNT = .PASS_CNT + 1;              !INCREMENT THE PASS COUNT
;          7571          CLR_MBUS;
;          7572          DAT_DM = ONE;                          !SET DATA DIAG MODE
;          7573          MLDA = .SECTOR_NO;                     !LOAD SECTOR NUMBER IN DSA
;          7574          MLWC = not 255;                        !LOAD WORD COUNT
;          7575          MLBA = IO_BUF;                         !LOAD UBUS ADRS
;          7576
;          7577          case .PASS_CNT from 0 to 3 of          !SELECT A NIBBLE PATTERN
;          7578              set
;          7579
;          7580              [0] :
;          7581                  NIB_PAT = %o'000000';          !ZEROES
;          7582
;          7583              [1] :
;          7584                  NIB_PAT = %o'17';              !ONES
;          7585
;          7586              [2] :
;          7587                  NIB_PAT = %o'12';              !ALTERNATING ONE'S, ZEROES
;          7588
;          7589              [3] :
;          7590                  NIB_PAT = %o'15'               !COMPLIMENT ONE'S, ZEROES
;          7591              tes;
;          7592
;          7593      incr LD_CNT from 0 to 9 do                 !LOAD NIBBLE PATTERN INTO NIBBLE SAVE
;          7594          LD_LNG_WRD (.LD_CNT, .NIB_PAT);
;          7595
;          7596      WRT_LNG_WRD;                               !LOAD THE DATA DIAG REGISTERS WITH NIBBLE SAVE
;          7597      MLCS1 = write;                             !DO A WRITE FUNCTION
;          7598
;          7599      incr WRT_CNT from 0 to 127 do              !WRITE PATTERN INTO THIS BLOCK
;          7600          begin
;          7601          DELAY (ONE_US);
;          7602          DAT_CLK = ONE;
;          7603          end;
;          7604
;          7605      CLR_MBUS;
```

SEQ 0348


```
16278 ;ML4AD
16279 ;                 TEST CODE SECTION
16280
16281 ;          7606           DAT_DM = ONE;                          !SET DATA DIAG MODE
16282 ;          7607           MLDA = .SECTOR_NO;                     !LOAD SECTOR NUMBER
16283 ;          7608           MLWC = not 255;                        !LOAD WORD COUNT
16284 ;          7609           MLBA = IO_BUF;                         !LOAD UBUS ADRS
16285 ;          7610           MLCS1 = read;                          !DO A READ FUNCTION
16286 ;          7611           DELAY (ONE_US);
16287 ;          7612           WRD_CNT = -1;                          !RESET THE WORD COUNT
16288 ;          7613
16289 ;          7614           do
16290 ;          7615               begin                              !TEST ONE BLOCK FOR NIBBLE PATTERN
16291 ;          7616               WRD_CNT = .WRD_CNT + 1;
16292 ;          7617               PD_TEMP = .MLPD;                   !GET PROM DATA
16293 ;          7618               DAT_CLK = ONE;                     !CLOCK DATA WORD INTO DIAG REG
16294 ;          7619               DELAY (ONE_US);
16295 ;          7620               RD_LNG_WRD;                        !READ DIAG REGISTERS
16296 ;          7621               NIB_PTR = -1;                      !RESET THE NIBBLE POINTER
16297 ;          7622
16298 ;          7623               do
16299 ;          7624                   begin                          !SEARCH FOR GOOD NIBBLES
16300 ;          7625                   NIB_PTR = .NIB_PTR + 1;        !INCREMENT NIBBLE POINTER
16301 ;          7626
16302 ;          7627                   if .PD_TEMP [.NIB_PTR] IS_NOT_SET      !TEST THIS NIB IF PROM FLAG NOT SET
16303 ;          7628                   then
16304 ;          7629                       TST_LNG_WRD (.NIB_PTR, .NIB_PAT, ERR_FLG);
16305 ;          7630
16306 ;          7631                   end
16307 ;          7632               until (.ERR_FLG) or (.NIB_PTR eql 9);
16308 ;          7633
16309 ;          7634               end
16310 ;          7635           until (.WRD_CNT eql 127);
16311 ;          7636
16312 ;          7637           end
16313 ;          7638       until (.PASS_CNT eql 3) or (.ERR_FLG IS_SET );     !REPEAT UNTIL ALL PAT TESTED OR ERROR FLG GETS SET
16314 ;          7639
16315 ;          7640       if (.PASS_CNT eql 3) and (.ERR_FLG IS_NOT_SET )    !WAS THIS A GOOD BLOCK?
16316 ;          7641       then
16317 ;          7642           begin                                          !YES
16318 ;          7643           DONE_FLG = ONE;                                !SET DONE FLAG
16319 ;          7644           GOOD_BLK = .SECTOR_NO;                         !GOOD BLOCK GETS THIS SECTOR NO
16320 ;          7645           end
16321 ;          7646       else
16322 ;          7647           begin                                          !NO
16323 ;          7648           SECTOR_NO = .SECTOR_NO + .RAS_INC;             !INCREMENT ROW NO
16324 ;          7649           PASS_CNT = -1;                                 !RESET PASS COUNT
16325 ;          7650           end;
16326 ;          7651
16327 ;          7652       end
16328 ;          7653   !
16329 ;          7654   !VER CZMLAD CHANGED TEST TO UNSIGNED TEST
16330 ;          7655   !
16331 ;          7656   until (.DONE_FLG IS_SET ) or (.SECTOR_NO eqlU .LST_ARR + .ARR_INC);
16332 ;          7657
```

SEQ 0349

```
 16334 ;ML4AD
 16335 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 16336                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (81)
 16337 ;          7658
 16338 ;          7659  ENDSUB;                                              !REPEAT UNTIL GOOD BLK FOUND OR AT LST ROW
 16339 ;          7660
 16340 ;          7661  !
 16341 ;          7662  !VER CZMLAD CHANGED TEST TO UNSIGNED TEST
 16342 ;          7663  !
 16343 ;          7664  if .SECTOR_NO eqlU .LST_ARR + .ARR_INC              !SEE IF WE'RE AT THE LAST BLOCK
 16344 ;          7665  then
 16345 ;          7666      begin
 16346 ;          7667      ERRDF (79, INTER, DUMPER);                     !A GOOD BLK NOT FOUND BEFORE LAST BLK
 16347 ;          7668      PRINTB (THR_FMT, WRD_14, PHR_10, FNC_15);      !ERROR AND DROP UNIT
 16348 ;          7669      DODU (.ML_LUN);
 16349 ;          7670      DOCLN;
 16350 ;          7671      end;
 16351 ;          7672
 16352 ;          7673  ENDTST;
 16356
 16357                                          .SBTTL  $T29 TEST CODE SECTION
 16361 057244  004167  125342          $T29:   JSR     R1,$SAVE5                ;                                    7500
 16362 057250  162706  000014                  SUB     #14,SP
 16363 057254  104402                  1$:     TRAP    2                        ;                                    7558
 16364 057256  012766  177777  000002          MOV     #-1,2(SP)                ; *,PASS.CNT                         7561
 16365 057264  005002                          CLR     R2                       ; SECTOR.NO                          7562
 16366 057266  005066  000004                  CLR     4(SP)                    ; DONE.FLG                           7563
 16367 057272  005266  000002          2$:     INC     2(SP)                    ; PASS.CNT                           7570
 16368 057276  152777  000040  136140          BISB    #40,@ML.REG+40
 16369 057304  016705  136522                  MOV     ML.DUT,R5
 16370 057310  042705  177770                  BIC     #177770,R5
 16371 057314  142777  000007  136122          BICB    #7,@ML.REG+40
 16372 057322  150577  136116                  BISB    R5,@ML.REG+40
 16373 057326  152777  000010  136170          BISB    #10,@ML.REG+120          ;                                    7572
 16374 057334  010277  136074                  MOV     R2,@ML.REG+30            ; SECTOR.NO,*                        7573
 16375 057340  012777  177400  136046          MOV     #-400,@ML.REG+10         ;                                    7574
 16376 057346  012777  014022  136050          MOV     #IO.BUF,@ML.REG+20       ;                                    7575
 16377 057354  016605  000002                  MOV     2(SP),R5                 ; PASS.CNT,*                         7577
 16378 057360  006305                          ASL     R5
 16379 057362  066507  057366                  ADD     3$(R5),PC
 16380 057366  000010                  3$:     .WORD   4$-3$
 16381 057370  000014                          .WORD   5$-3$
 16382 057372  000022                          .WORD   6$-3$
 16383 057374  000030                          .WORD   7$-3$
 16384 057376  005001                  4$:     CLR     R1                       ; NIB.PAT                            7581
 16385 057400  000410                          BR      8$                       ;                                    7577
 16386 057402  012701  000017          5$:     MOV     #17,R1                   ; *,NIB.PAT                          7584
 16387 057406  000405                          BR      8$                       ;                                    7577
 16388 057410  012701  000012          6$:     MOV     #12,R1                   ; *,NIB.PAT                          7587
```

```
16390                                   ;ML4AD
16391                                   ;                       TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
16392                                                                                                               29-Mar-1982 16:21:03    PA:<
16393 057414  000402                            BR      8$                                                                                  7577
16394 057416  012701  000015                7$: MOV     #15,R1          ;*,NIB.PAT                                                          7590
16395 057422  005005                        8$: CLR     R5              ;LD.CNT                                                             7593
16396 057424  010546                        9$: MOV     R5,-(SP)        ;LD.CNT,*                                                           7594
16397 057426  010146                            MOV     R1,-(SP)        ;NIB.PAT,*
16398 057430  004767  141720                    JSR     PC,LD.LNG.WRD
16399 057434  022626                            CMP     (SP)+,(SP)+
16400 057436  005205                            INC     R5
16401 057440  020527  000011                    CMP     R5,#11          ;LD.CNT                                                             7593
16402 057444  003767                            BLE     9$              ;LD.CNT,*
16403 057446  016777  133722  136120            MOV     D1.TEMP,@ML.REG+170                                                                 7594
16404 057454  016777  133716  136122            MOV     D2.TEMP,@ML.REG+200
16405 057462  016777  133712  136074            MOV     E2.TEMP,@ML.REG+160
16406 057470  012777  000061  135706            MOV     #61,@ML.REG
16407 057476  005003                            CLR     R3                                                                                  7597
16408 057500  012704  000001                10$: MOV    #1,R4           ;WRT.CNT                                                            7599
16409 057504  001411                        11$: BEQ    14$             ;*,$STMP2                                                           7601
16410 057506  016705  122404                    MOV     L$DLY,R5        ;*,$STMP1
16411 057512  001404                            BEQ     13$
16412 057514  005066  000012                12$: CLR     12(SP)         ;$STMP
16413 057520  005305                            DEC     R5              ;$STMP1
16414 057522  001374                            BNE     12$
16415 057524  005304                        13$: DEC     R4             ;$STMP2
16416 057526  000766                            BR      11$
16417 057530  152777  000020  135766    14$: BISB    #20,@ML.REG+120                                                                     7602
16418 057536  005203                            INC     R3              ;WRT.CNT                                                            7599
16419 057540  020327  000177                    CMP     R3,#177         ;WRT.CNT,*
16420 057544  003755                            BLE     10$
16421 057546  152777  000040  135670            BISB    #40,@ML.REG+40                                                                      7603
16422 057554  016705  136252                    MOV     ML.DUT,R5
16423 057560  042705  177770                    BIC     #177770,R5
16424 057564  142777  000007  135652            BICB    #7,@ML.REG+40
16425 057572  150577  135646                    BISB    R5,@ML.REG+40
16426 057576  152777  000010  135720            BISB    #10,@ML.REG+120
16427 057604  010277  135524                    MOV     R2,@ML.REG+30                                                                       7606
16428 057610  012777  177400  135576            MOV     #-400,@ML.REG+10        ;SECTOR.NO,*                                                7607
16429 057616  012777  014022  135600            MOV     #IO.BUF,@ML.REG+20                                                                  7608
16430 057624  012777  000071  135552            MOV     #71,@ML.REG                                                                         7609
16431 057632  012704  000001                    MOV     #1,R4                                                                               7610
16432 057636  001411                    15$: BEQ    18$             ;*,$STMP2                                                           7611
16433 057640  016705  122252                    MOV     L$DLY,R5
16434 057644  001404                            BEQ     17$             ;*,$STMP1
16435 057646  005066  000012                16$: CLR     12(SP)         ;$STMP
16436 057652  005305                            DEC     R5              ;$STMP1
16437 057654  001374                            BNE     16$
16438 057656  005304                        17$: DEC     R4             ;$STMP2
16439 057660  000766                            BR      15$
16440 057662  012766  177777  000006    18$: MOV     #-1,6(SP)          ;*,WRD.CNT                                                          7612
16441 057670  005266  000006                19$: INC     6(SP)          ;WRD.CNT                                                            7616
16442 057674  017767  135734  135440            MOV     @ML.REG+230,PD.TEMP                                                                 7617
16443 057702  152777  000020  135614            BISB    #20,@ML.REG+120                                                                     7618
16444 057710  012704  000001                    MOV     #1,R4           ;*,$STMP2                                                           7619
```

SEQ 0351


```
16446                                   ;ML4AD
16447                                   ;                TEST CODE SECTION
16448
16449 057714  001411                  20$:    BEQ     23$
16450 057716  016705  122174                  MOV     L$DLY,R5        ; *,$$TMP1
16451 057722  001404                          BEQ     22$
16452 057724  005066  000012          21$:    CLR     12(SP)          ; $$TMP
16453 057730  005305                          DEC     R5              ; $$TMP1
16454 057732  001374                          BNE     21$
16455 057734  005304                  22$:    DEC     R4              ; $$TMP2
16456 057736  000766                          BR      20$
16457 057740  017767  135630  133426  23$:    MOV     @ML.REG+170,D1.TEMP
16458 057746  017767  135632  133422          MOV     @ML.REG+200,D2.TEMP
16459 057754  017767  135604  133416          MOV     @ML.REG+160,E2.TEMP
16460 057762  012716  177777                  MOV     #-1,(SP)
16461 057766  005216                  24$:    INC     (SP)            ; *,NIB.PTR             7621
16462 057770  011605                          MOV     (SP),R5         ; NIB.PTR               7625
16463 057772  006205                          ASR     R5              ; NIB.PTR,*             7627
16464 057774  006205                          ASR     R5
16465 057776  006205                          ASR     R5
16466 060000  062705  015342                  ADD     #PD.TEMP,R5
16467 060004  010546                          MOV     R5,-(SP)
16468 060006  016646  000002                  MOV     2(SP),-(SP)     ; NIB.PTR,*
16469 060012  042716  177770                  BIC     #177770,(SP)
16470 060016  012746  000001                  MOV     #1,-(SP)
16471 060022  005046                          CLR     -(SP)
16472 060024  004767  123604                  JSR     PC,BL$GT2
16473 060030  062706  000010                  ADD     #10,SP
16474 060034  005700                          TST     R0
16475 060036  001011                          BNE     25$
16476 060040  011646                          MOV     (SP),-(SP)      ; NIB.PTR,*
16477 060042  010146                          MOV     R1,-(SP)        ; NIB.PAT,*             7629
16478 060044  012746  000016                  MOV     #16,-(SP)
16479 060050  060616                          ADD     SP,(SP)         ; ERR.FLG,*
16480 060052  004767  140320                  JSR     PC,TST.LNG.WRD
16481 060056  062706  000006                  ADD     #6,SP
16482 060062  016605  000010          25$:    MOV     10(SP),R5       ; ERR.FLG,*             7632
16483 060066  032705  000001                  BIT     #1,R5
16484 060072  001012                          BNE     26$
16485 060074  021627  000011                  CMP     (SP),#11        ; NIB.PTR,*
16486 060100  001332                          BNE     24$
16487 060102  032705  000001                  BIT     #1,R5           ;                       7635
16488 060106  001004                          BNE     26$
16489 060110  026627  000006  000177          CMP     6(SP),#177      ; WRD.CNT,*
16490 060116  001264                          BNE     19$
16491 060120  005004                  26$:    CLR     R4              ;
16492 060122  026627  000002  000003          CMP     2(SP),#3        ; PASS.CNT,*            7638
16493 060130  001002                          BNE     27$
16494 060132  005204                          INC     R4
16495 060134  000403                          BR      28$
16496 060136  020527  000001          27$:    CMP     R5,#1
16497 060142  001030                          BNE     31$
16498 060144  006004                  28$:    ROR     R4              ;                       7640
16499 060146  103010                          BCC     29$
16500 060150  005705                          TST     R5
```

```
16502                                   ;ML4AD
16503                                   ;                       TEST CODE SECTION                       29-Mar-1982 16:23:04    TOPS
16504                                                                                                   29-Mar-1982 16:21:03    PA:<
16505 060152  001006                             BNE     29$                                                                     7643
16506 060154  012766  000001  000004             MOV     #1,4(SP)        ; *,DONE.FLG                                            7644
16507 060162  010267  133620                     MOV     R2,GOOD.BLK     ; SECTOR.NO,*                                           7640
16508 060166  000405                             BR      30$             ;                                                       7648
16509 060170  066702  135152            29$:     ADD     RAS.INC,R2      ; *,SECTOR.NO                                           7649
16510 060174  012766  177777  000002             MOV     #-1,2(SP)       ; *,PASS.CNT                                            7656
16511 060202  026627  000004  000001    30$:     CMP     4(SP),#1        ; DONE.FLG,*
16512 060210  001410                             BEQ     32$
16513 060212  016705  133602                     MOV     LST.ARR,R5
16514 060216  066705  133562                     ADD     ARR.INC,R5
16515 060222  020205                             CMP     R2,R5           ; SECTOR.NO,*
16516 060224  001402                    31$:     BEQ     32$
16517 06022o  000167  177040                     JMP     2$
16518 060232  104467                    32$:     TRAP    67                                                                      7664
16519 060234  006000                             ROR     R0
16520 060236  103002                             BHIS    33$
16521 060240  000167  177010                     JMP     1$
16522 060244  016705  133550            33$:     MOV     LST.ARR,R5      ;
16523 060250  066705  133530                     ADD     ARR.INC,R5
16524 060254  020205                             CMP     R2,R5           ; SECTOR.NO,*
16525 060256  001026                             BNE     34$
16526 060260  104455                             TRAP    55              ;                                                       7667
16527 060262  000117                             .WORD   117
16528 060264  013114                             .WORD   INTER
16529 060266  026302                             .WORD   DUMPER
16530 060270  012746  012360                     MOV     #FNC.15,-(SP)   ;                                                       7668
16531 060274  012746  012006                     MOV     #PHR.10,-(SP)
16532 060300  012746  010664                     MOV     #WRD.14,-(SP)
16533 060304  012746  010256                     MOV     #THR.FMT,-(SP)
16534 060310  012746  000004                     MOV     #4,-(SP)
16535 060314  010600                             MOV     SP,R0           ; SP,*
16536 060316  104414                             TRAP    14
16537 060320  016700  135504                     MOV     ML.LUN,R0       ;                                                       7669
16538 060324  104451                             TRAP    51
16539 060326  104444                             TRAP    44
16540 060330  062706  000012                     ADD     #12,SP
16541 060334  062706  000014            34$:     ADD     #14,SP          ;                                                       7666
16542 060340  000207                             RTS     PC              ;                                                       7500
16543
16544                           ; Routine Size:  287 words
16545                           ; Maximum stack depth per invocation:  17 words
16550
16551
16555
16556                                   .SBTTL  T29 TEST CODE SECTION
```

```
16558                                   ;ML4AD
16559                                   ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
16560                                                                                                       29-Mar-1982 16:21:03    PA:<
16564 060342
16565 060342  004767  176676           1$:     JSR     PC,ST29                         ;                                           7671
16566 060346  104466                           TRAP    66
16567 060350  006000                           ROR     R0
16568 060352  103773                           BLO     1$
16569 060354  000207                           RTS     PC
16570
16571                                   ; Routine Size:  6 words
16572                                   ; Maximum stack depth per invocation:  0 words
16577
16578
16579 ;          7674   !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (82)

```
16581 ;ML4AD
16582 ;                 TEST CODE SECTION
16583 ;
16584 ;          7675 !
16585 ;          7676 !
16586 ;          7677 BGNTST;
16587 ;          7678
16588 ;          7679 !++
16589 ;          7680 ! TEST NUMBER:  TST 30
16590 ;          7681 !
16591 ;          7682 ! TEST NAME:    REFRESH TIMING TEST
16592 ;          7683 !
16593 ;          7684 ! TEST DESCRIPTION:
16594 ;          7685 !
16595 ;          7686 !       TEST THE MEMORY ARRAY MODULES
16596 ;          7687 !       REFRESH TIMING AND CONTROL LOGIC
16597 ;          7688 !       REG BY:
16598 ;          7689 !
16599 ;          7690 !       1.  WRITING ALL ONES INTO THE
16600 ;          7691 !           GOOD BLOCK.
16601 ;          7692 !
16602 ;          7693 !       2.  DELAY FOR 2 MS
16603 ;          7694 !
16604 ;          7695 !       3.  READ THE GOOD BLOCK FOR ONES.
16605 ;          7696 !           KEEP COUNT OF BAD NIBBLES
16606 ;          7697 !           ENCOUNTERED
16607 ;          7698 !
16608 ;          7699 !       4.  ALLOW A BAD NIBBLE THRESHOLD
16609 ;          7700 !           OF 20 BAD NIBBLES OUT OF THE
16610 ;          7701 !           1280 NIBBLES TESTED.
16611 ;          7702 !
16612 ;          7703 !       5.  IF THRESHOLD IS EXCEEDED THEN
16613 ;          7704 !           REPORT ERROR AND DROP THE UNIT.
16614 ;          7705 !
16615 ;          7706 ! IMPLICIT INPUTS:
16616 ;          7707 !
16617 ;          7708 !       PD_TEMP:
16618 ;          7709 !       A BIT VECTOR OF 16 BITS WHERE
16619 ;          7710 !       THE READ PROM DATA IS STORED
16620 ;          7711 !       AND ACCESSED FROM.
16621 ;          7712 !
16622 ;          7713 !--
16623 ;          7714
16624 ;          7715 local
16625 ;          7716     TST_PAT,                          !TEST PATTERN
16626 ;          7717     ERR_CNT,                          !ERROR COUNT
16627 ;          7718     ERR_FLG;                          !ERROR FLAG
16628 ;          7719
16629 ;          7720 BGNSUB;
16630 ;          7721 CLR_MBUS;
16631 ;          7722 ERR_CNT = ZERO;
16632 ;          7723 TST_PAT = ONES;
16633 ;          7724 MLD1 = .TST_PAT;                      !LOAD DATA DIAG REGS WITH TST PAT
16634 ;          7725 MLD2 = .TST_PAT;
16635 ;          7726 MLE2 = .TST_PAT;
```

```
;ML4AD
;                    TEST CODE SECTION                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (82)
;
;          7727  DAT_DM_XFER ();                                !SET UP A DATA DIAG MODE XFERR
;          7728  MLCS1 = write;                                 !DO A WRITE FUNCTION
;          7729
;          7730  incr WRD_CNT from 0 to 127 do                  !WRITE BLOCK WITH TST PAT
;          7731      begin
;          7732      DELAY (ONE_US);
;          7733      DAT_CLK = ONE;
;          7734      end;
;          7735
;          7736  CLR_MBUS;
;          7737  DAT_DM_XFER ();                                !SET UP A DATA DIAG MODE XFERR
;          7738  MLCS1 = read;                                  !DO A READ FUNCTION
;          7739  DELAY (TWO_TH_US);                            !DELAY FOR 2 MS
;          7740
;          7741  incr WRD_CNT from 0 to 127 do                  !READ THE BLOCK
;          7742      begin
;          7743      PD_TEMP = .MLPD;                           !GET THE PROM DATA
;          7744      DAT_CLK = ONE;                             !CLOCK DATA WORD INTO DIAG REG
;          7745      DELAY (ONE_US);
;          7746      RD_LNG_WRD;                                !READ THE DIAG REG
;          7747
;          7748      incr NIB_PTR from 0 to 9 do                !LOOK AT 10 NIBBLES
;          7749          begin
;          7750
;          7751          if .PD_TEMP [.NIB_PTR] IS_NOT_SET then TST_LNG_WRD (.NIB_PTR, .TST_PAT, ERR_FLG);
;          7752
;          7753                                                 !FIND GOOD NIBBLES AND COMPARE THEM
;          7754
;          7755          if .ERR_FLG IS_SET then ERR_CNT = .ERR_CNT + 1; !INCREMENT ERROR COUNT IF ERR_FLG IS SET
;          7756
;          7757          end;
;          7758
;          7759      end;
;          7760
;          7761  ENDSUB;
;          7762
;          7763  if .ERR_CNT gtr 20                             !ALLOW 20 NIBBLES TO FAIL
;          7764  then
;          7765      begin                                      !ERROR IF GTR 20
;          7766      ERRDF (80, ASYNC, DUMPER);
;          7767      PRINTB (FIV_FMT, WRD_22, PHR_4, WRD_12, FNC_16, WRD_48);
;          7768      DODU (.ML_LUN);
;          7769      DOCLN;
;          7770      end;
;          7771
;          7772  ENDTST;
;
;                                        .SBTTL  ST30 TEST CODE SECTION
```

```
060356  004167  124230          $T30:   JSR     R1,$SAVE5
060362  024646                          CMP     -(SP),-(SP)                     ;                       7673
060364  104402                  1$:     TRAP    2
060366  152777  000040  135050          BISB    #40,@ML.REG+40                  ;                       7718
060374  016703  135432                  MOV     ML.DUT,R3                       ;                       7720
060400  042703  177770                  BIC     #177770,R3
060404  142777  000007  135032          BICB    #7,@ML.REG+40
060412  150377  135026                  BISB    R3,@ML.REG+40
060416  005005                          CLR     R5                              ; ERR.CNT
060420  012704  177777                  MOV     #-1,R4                          ; *,TST.PAT             7722
060424  010477  135144                  MOV     R4,@ML.REG+170                  ; TST.PAT,*             7723
060430  010477  135150                  MOV     R4,@ML.REG+200                  ; TST.PAT,*             7724
060434  010477  135124                  MOV     R4,@ML.REG+160                  ; TST.PAT,*             7725
060440  004767  135746                  JSR     PC,DAT.DM.XFER                  ;                       7726
060444  012777  000061  134732          MOV     #61,@ML.REG                     ;                       7727
060452  005001                          CLR     R1                              ; WRD.CNT               7728
060454  012702  000001          2$:     MOV     #1,R2                           ; *,$STMP2              7730
060460  001411                  3$:     BEQ     6$                                                      7732
060462  016703  121430                  MOV     L$DLY,R3                        ; *,$STMP1
060466  001404                          BEQ     5$
060470  005066  000002          4$:     CLR     2(SP)                           ; $STMP
060474  005303                          DEC     R3                              ; $STMP1
060476  001374                          BNE     4$
060500  005302                  5$:     DEC     R2                              ; $STMP2
060502  000766                          BR      3$
060504  152777  000020  135012  6$:     BISB    #20,@ML.REG+120                 ;                       7733
060512  005201                          INC     R1                              ; WRD.CNT               7730
060514  020127  000177                  CMP     R1,#177                         ; WRD.CNT,*
060520  003755                          BLE     2$
060522  152777  000040  134714          BISB    #40,@ML.REG+40                  ;                       7734
060530  016703  135276                  MOV     ML.DUT,R3
060534  042703  177770                  BIC     #177770,R3
060540  142777  000007  134676          BICB    #7,@ML.REG+40
060546  150377  134672                  BISB    R3,@ML.REG+40
060552  004767  135634                  JSR     PC,DAT.DM.XFER                  ;                       7737
060556  012777  000071  134620          MOV     #71,@ML.REG                     ;                       7738
060564  012702  003720                  MOV     #3720,R2                        ; *,$STMP2              7739
060570  001411                  7$:     BEQ     10$
060572  016703  121320                  MOV     L$DLY,R3                        ; *,$STMP1
060576  001404                          BEQ     9$
060600  005066  000002          8$:     CLR     2(SP)                           ; $STMP
060604  005303                          DEC     R3                              ; $STMP1
060606  001374                          BNE     8$
060610  005302                  9$:     DEC     R2                              ; $STMP2
060612  000766                          BR      7$
060614  005001                  10$:    CLR     R1                              ; WRD.CNT               7741
060616  017767  135012  134516  11$:    MOV     @ML.REG+230,PD.TEMP             ;                       7743
060624  152777  000020  134672          BISB    #20,@ML.REG+120                 ;                       7744
060632  012702  000001                  MOV     #1,R2                           ; *,$STMP2              7745
060636  001411                  12$:    BEQ     15$
```

SEQ 0357


```
16749                                   ;ML4AD
16750                                   ;
16751                                   ;       TEST CODE SECTION
16752 060640  016703  121252                     MOV     L$DLY,R3                ; *,$$TMP1
16753 060644  001404                             BEQ     14$
16754 060646  005066  000002             13$:    CLR     2(SP)                   ; $$TMP
16755 060652  005303                             DEC     R3                      ; $$TMP1
16756 060654  001374                             BNE     13$
16757 060656  005302                     14$:    DEC     R2                      ; $$TMP2
16758 060660  000766                             BR      12$
16759 060662  017767  134706  132504     15$:    MOV     @ML.REG+170,D1.TEMP
16760 060670  017767  134710  132500             MOV     @ML.REG+200,D2.TEMP
16761 060676  017767  134662  132474             MOV     @ML.REG+160,E2.TEMP
16762 060704  005002                             CLR     R2                      ; NIB.PTR                       7748
16763 060706  010203                     16$:    MOV     R2,R3                   ; NIB.PTR,*                     7751
16764 060710  006203                             ASR     R3
16765 060712  006203                             ASR     R3
16766 060714  006203                             ASR     R3
16767 060716  062703  015342                     ADD     #PD.TEMP,R3
16768 060722  010346                             MOV     R3,-(SP)
16769 060724  010246                             MOV     R2,-(SP)                ; NIB.PTR,*
16770 060726  042716  177770                     BIC     #177770,(SP)
16771 060732  012746  000001                     MOV     #1,-(SP)
16772 060736  005046                             CLR     -(SP)
16773 060740  004767  122670                     JSR     PC,BL$GT2
16774 060744  062706  000010                     ADD     #10,SP
16775 060750  005700                             TST     R0
16776 060752  001011                             BNE     17$
16777 060754  010246                             MOV     R2,-(SP)                ; NIB.PTR,*
16778 060756  010446                             MOV     R4,-(SP)                ; TST.PAT,*
16779 060760  012746  000006                     MOV     #6,-(SP)
16780 060764  060616                             ADD     SP,(SP)                 ; ERR.FLG,*
16781 060766  004767  137404                     JSR     PC,TST.LNG.WRD
16782 060772  062706  000006                     ADD     #6,SP
16783 060776  021627  000001             17$:    CMP     (SP),#1                 ; ERR.FLG,*                     7755
16784 061002  001001                             BNE     18$
16785 061004  005205                             INC     R5                      ; ERR.CNT
16786 061006  005202                     18$:    INC     R2                      ; NIB.PTR                       7748
16787 061010  020227  000011                     CMP     R2,#11                  ; NIB.PTR,*
16788 061014  003734                             BLE     16$
16789 061016  005201                             INC     R1                      ; WRD.CNT                       7741
16790 061020  020127  000177                     CMP     R1,#177                 ; WRD.CNT,*
16791 061024  003674                             BLE     11$
16792 061026  104467                             TRAP    67                      ;                               7759
16793 061030  006000                             ROR     R0
16794 061032  103002                             BHIS    19$
16795 061034  000167  177324                     JMP     1$
16796 061040  020527  000024             19$:    CMP     R5,#24                  ; ERR.CNT,*                     7763
16797 061044  003432                             BLE     20$
16798 061046  104455                             TRAP    55                      ;                               7766
16799 061050  000120                             .WORD   120
16800 061052  012706                             .WORD   ASYNC
16801 061054  026302                             .WORD   DUMPER
16802 061056  012746  011250                     MOV     #WRD.48,-(SP)           ;                               7767
16803 061062  012746  012372                     MOV     #FNC.16,-(SP)
```

```
16805
16806                                   ;ML4AD
16807                                   ;               TEST CODE SECTION                                       29-Mar-1982 16:23:04    TOPS
16808 061066  012746  010650                                                                                   29-Mar-1982 16:21:03    PA:<
16809 061072  012746  011676                   MOV     #WRD.12,-(SP)
16810 061076  012746  010752                   MOV     #PHR.4,-(SP)
16811 061102  012746  010304                   MOV     #WRD.22,-(SP)
16812 061106  012746  000006                   MOV     #FIV.FMT,-(SP)
16813 061112  010600                           MOV     #6,-(SP)
16814 061114  104414                           MOV     SP,R0                                   ; SP,*
16815 061116  016700  134706                   TRAP    14
16816 061122  104451                           MOV     ML.LUN,R0                               ;                       7768
16817 061124  104444                           TRAP    51
16818 061126  062706  000016                   TRAP    44
16819 061132  022626                           ADD     #16,SP                                  ;                       7765
16820 061134  000207            20$:           CMP     (SP)+,(SP)+                             ;                       7673
16821                                          RTS     PC
16822                                   ; Routine Size:  184 words
16823                                   ; Maximum stack depth per invocation:  15 words
16828
16829
16833
16834                                           .SBTTL  T30 TEST CODE SECTION
16838 061136                            T30::
16839 061136  004767  177214            1$:     JSR     PC,$T30                                 ;                       7770
16840 061142  104466                           TRAP    66
16841 061144  006000                           ROR     R0
16842 061146  103773                           BLO     1$
16843 061150  000207                           RTS     PC
16844
16845                                   ; Routine Size:  6 words
16846                                   ; Maximum stack depth per invocation:  0 words
16851
16852
16853 ;               7773  !<BLF/PAGE>
```

SEQ 0359

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (83)

```
16855 ;ML4AD
16856 ;                 TEST CODE SECTION
16857
16858 ;          7774   !
16859 ;          7775   BGNTST;
16860 ;          7776
16861 ;          7777   !++
16862 ;          7778   ! TEST NUMBER:  TST 31
16863 ;          7779   !
16864 ;          7780   ! TEST NAME:    ADDRESS COUNTER TEST
16865 ;          7781   !
16866 ;          7782   ! TEST DESCRIPTION:
16867 ;          7783   !       TEST THE ABILITY OF THE ADDRESS
16868 ;          7784   !       COUNTER TO SUCCESSFULLY COUNT
16869 ;          7785   !       FROM BLOCK ZERO THROUGH THE
16870 ;          7786   !       DEVICES LAST BLOCK BY:
16871 ;          7787   !
16872 ;          7788   !       1.  WRITING THE LAST BLOCK WITH ONES PATTERN.
16873 ;          7789   !
16874 ;          7790   !       2.  STARTING AT BLOCK ZERO WRITE ZEROES INTO ALL BLOCK UP TO THE
16875 ;          7791   !           LAST BLOCK ADRS MINUS ONE.
16876 ;          7792   !
16877 ;          7793   !           READ THE LAST BLOCK FOR ONES AND ERROR IF ZEROES.
16878 ;          7794   !
16879 ;          7795   !       3.  STARTING AT BLOCK ZERO WRITE ZEROES INTO ALL BLOCK THROUGH THE LAST
16880 ;          7796   !           BLOCK.
16881 ;          7797   !
16882 ;          7798   !           READ THE LAST BLOCK FOR ZEROES AND ERROR IF STILL ONES.
16883 ;          7799   !
16884 ;          7800   ! IMPLICIT INPUTS:
16885 ;          7801   !       PD_TEMP:
16886 ;          7802   !       A BITVECTOR OF 16 BITS WHERE THE READ PROM DATA IS STORED AND ACCESSED FROM.
16887 ;          7803   !--
16888 ;          7804
16889 ;          7805   local
16890 ;          7806       ERR_CNT,                          !ERROR THRESHOLD COUNT
16891 ;          7807       ERR_FLG,                          !ERROR FLG
16892 ;          7808       END_CNT,                          !ENDING SECTOR NUMBER
16893 ;          7809       BG_PAT;                           !BACKGROUND PATTERN
16894 ;          7810
16895 ;          7811   BGNSUB;
16896 ;          7812   CLR_MBUS;
16897 ;          7813   BG_PAT = ONES;                        !BACKGROUND PAT OF ONES
16898 ;          7814   MLD1 = .BG_PAT;                       !LOAD DATA DIAGS WITH BG PAT
16899 ;          7815   MLD2 = .BG_PAT;
16900 ;          7816   MLE2 = .BG_PAT;
16901 ;          7817   DAT_DM = ONE;                         !SET DATA DIAG MODE
16902 ;          7818   LAST_BLK_XFER ();                     !SET UP A LAST BLOCK XFERR
16903 ;          7819   MLCST = write;                        !DO A WRITE FUNCTION
16904 ;          7820
16905 ;          7821   incr WRD_CNT from 0 to 127 do         !WRITE THE LAST BLOCK WITH THE BG PATTERN
16906 ;          7822       begin
16907 ;          7823       DELAY (ONE_US);
16908 ;          7824       DAT_CLK = ONE;
16909 ;          7825       end;
```

```
16911 ;ML4AD
16912 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
16913 ;                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (83)
16914 ;          7826
16915 ;          7827  CLR_MBUS;
16916 ;          7828  DAT_DM = ONE;                                       !SET DATA DIAG MODE
16917 ;          7829  LAST_BLK_XFER ();                                   !SET UP A LAST BLOCK XFERR
16918 ;          7830  MLCST = read;                                       !DO A READ FUNCTION
16919 ;          7831  DELAY (ONE_US);
16920 ;          7832  ERR_CNT = ZERO;                                     !CLEAR ERROR COUNT THRESHOLD
16921 ;          7833
16922 ;          7834  incr WD_CNT from 0 to 127 do                        !READ THE LAST BLOCK FOR BG PATTERN
16923 ;          7835      begin
16924 ;          7836      PD_TEMP = .MLPD;                                !GET PROM DATA
16925 ;          7837      DAT_CLK = ONE;                                  !CLOCK OUT THE DATA WORD
16926 ;          7838      DELAY (ONE_US);
16927 ;          7839      RD_LNG_WRD;                                     !READ THE DATA WORD
16928 ;          7840
16929 ;          7841      incr NIB_PTR from 0 to 9 do                     !LOOK AT 10 NIBBLES
16930 ;          7842          begin
16931 ;          7843
16932 ;          7844          if .PD_TEMP [.NIB_PTR] IS_NOT_SET then TST_LNG_WRD (.NIB_PTR, .BG_PAT, ERR_FLG);
16933 ;          7845
16934 ;          7846                                                          !FIND GOOD NIBBLES AND COMPARE THEM
16935 ;          7847
16936 ;          7848          if .ERR_FLG IS_SET then ERR_CNT = .ERR_CNT + 1; !INC ERROR COUNT IF ERROR DETECTED
16937 ;          7849
16938 ;          7850          end;
16939 ;          7851
16940 ;          7852      end;
16941 ;          7853
16942 ;          7854  if .ERR_CNT gtr 10                                   !IS ERROR THRESHOLD EXCEEDED
16943 ;          7855  then
16944 ;          7856      begin                                           !ERROR IF EXCEEDED
16945 ;          7857      ERRDF (81, INTER, DUMPER);
16946 ;          7858      PRINTB (THR_FMT, PHR_4, WRD_13, FNC_22);
16947 ;          7859      PRINTB (TWO_FMT, FNC_13, WRD_56);
16948 ;          7860      EXIT_TST;
16949 ;          7861      end;
16950 ;          7862
16951 ;          7863  END_CNT = .LST_BLK - 1;                             !END AT LAST BLOCK -1
16952 ;          7864  MLDT = ZEROES;                                      !LOAD DATA DIAG REG WITH COMP BG PAT
16953 ;          7865  MLD2 = ZEROES;
16954 ;          7866  MLE2 = ZEROES;
16955 ;          7867
16956 ;          7868  incr TWICE from 0 to 1 do                          !REPEAT LOOP TWICE
16957 ;          7869      begin
16958 ;          7870      CLR_MBUS;
16959 ;          7871      ERR_CNT = ZERO;                                 !CLEAR THE ERROR COUNT THRESHOLD
16960 ;          7872      DAT_DM = ONE;                                   !SET DATA DIAG MODE
16961 ;          7873      FIRST_BLK_XFER ();                              !SET UP A FIRST BLOCK XFERR
16962 ;          7874      MLCS1 = write;                                  !DO A WRITE FUNCTION
16963 ;          7875
16964 ;          7876  !
16965 ;          7877  !VER CZMLAD CHANGED TEST TO UNSIGNED TEST
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (83)

```
;ML4AD
;                    TEST CODE SECTION
;
;          7878  !
;          7879      incrU BLK_CNT from 0 to .END_CNT do                  !CLOCK THE ADDRESS COUNTER UP TO END_CNT
;          7880
;          7881          incr ADRS_CNT from 0 to 127 do
;          7882              begin
;          7883              DAT_CLK = ONE;
;          7884              end;
;          7885
;          7886      CLR_MBUS;
;          7887      DAT_DM = ONE;                                        !SET DATA DIAG MODE
;          7888      LAST_BLK_XFER ();                                    !SET UP A LAST BLOCK XFERR
;          7889      MLCST = read;                                        !DO A READ FUNCTION
;          7890      DELAY (ONE_US);
;          7891
;          7892      incr WD_CNT from 0 to 127 do                         !READ THE LAST BLOCK FOR BG PATTERN
;          7893          begin
;          7894          PD_TEMP = .MLPD;                                 !GET THE PROM DATA
;          7895          DAT_CLK = ONE;                                   !CLOCK OUT DATA WORD
;          7896          DELAY (ONE_US);
;          7897          RD_LNG_WRD;                                      !READ DATA WORD
;          7898
;          7899          incr NIB_PTR from 0 to 9 do                      !LOOK AT 10 NIBBLES
;          7900              begin
;          7901
;          7902              if .PD_TEMP [.NIB_PTR] IS_NOT_SET then TST_LNG_WRD (.NIB_PTR, .BG_PAT, ERR_FLG);
;          7903
;          7904                                                           !FIND GOOD NIBBLES AND COMPARE THEM
;          7905
;          7906              if .ERR_FLG IS_SET then ERR_CNT = .ERR_CNT + ONE;    !INC ERROR COUNT IF ERROR DECTED
;          7907
;          7908              end;
;          7909
;          7910          end;
;          7911
;          7912      if .ERR_CNT gtr 10                                   !IS ERROR COUNT THRESHOLD EXCEEDED
;          7913      then
;          7914          begin                                            !ERROR IF EXCEEDED
;          7915          ERRDF (82, ASYNC, DUMPER);
;          7916          PRINTB (THR_FMT, WRD_50, WRD_51, WRD_10);
;          7917          end;
;          7918
;          7919      END_CNT = .END_CNT + 1;                              !NOW END AT THE LAST BLOCK
;          7920      BG_PAT = not .BG_PAT;                                !COMPLIMENT THE BG PATTERN AND REPEAT
;          7921      end;
;          7922
;          7923  ENDSUB;
;          7924  ENDTST;
17020
17021                                   .SBTTL  $T31 TEST CODE SECTION
```

SEQ 0362

```
                                        ;ML4AD
                                        ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS
                                                                                                           29-Mar-1982 16:21:03    PA:<
17023
17024
17025
17029 061152  004167  123434            $T31:   JSR     R1,$SAVE5                ;                                                 7772
17030 061156  162706  000010                    SUB     #10,SP
17031 061162  104402                    1$:     TRAP    2                        ;                                                 7809
17032 061164  152777  000040  134252            BISB    #40,@ML.REG+40           ;                                                 7811
17033 061172  016705  134634                    MOV     ML.DUT,R5
17034 061176  042705  177770                    BIC     #177770,R5
17035 061202  142777  000007  134234            BICB    #7,@ML.REG+40
17036 061210  150577  134230                    BISB    R5,@ML.REG+40
17037 061214  012701  177777                    MOV     #-1,R1
17038 061220  010177  134350                    MOV     R1,@ML.REG+170           ; *,BG.PAT                                        7813
17039 061224  010177  134354                    MOV     R1,@ML.REG+200           ; BG.PAT,*                                        7814
17040 061230  010177  134330                    MOV     R1,@ML.REG+160           ; BG.PAT,*                                        7815
17041 061234  152777  000010  134262            BISB    #10,@ML.REG+120          ;                                                 7816
17042 061242  004767  137076                    JSR     PC,LAST.BLK.XFER         ;                                                 7817
17043 061246  012777  000061  134130            MOV     #61,@ML.REG              ;                                                 7818
17044 061254  005002                            CLR     R2                       ;                                                 7819
17045 061256  012703  000001            2$:     MOV     #1,R3                    ; WRD.CNT                                         7821
17046 061262  001411                    3$:     BEQ     6$                       ; *,$$TMP2                                        7823
17047 061264  016704  120626                    MOV     L$DLY,R4
17048 061270  001404                            BEQ     5$                       ; *,$$TMP1
17049 061272  005066  000006            4$:     CLR     6(SP)                    ; $$TMP
17050 061276  005304                            DEC     R4                       ; $$TMP1
17051 061300  001374                            BNE     4$
17052 061302  005303                    5$:     DEC     R3                       ; $$TMP2
17053 061304  000766                            BR      3$
17054 061306  152777  000020  134210    6$:     BISB    #20,@ML.REG+120          ;                                                 7824
17055 061314  005202                            INC     R2                       ; WRD.CNT                                         7821
17056 061316  020227  000177                    CMP     R2,#177                  ; WRD.CNT,*
17057 061322  003755                            BLE     2$
17058 061324  152777  000040  134112            BISB    #40,@ML.REG+40           ;                                                 7825
17059 061332  016705  134474                    MOV     ML.DUT,R5
17060 061336  042705  177770                    BIC     #177770,R5
17061 061342  142777  000007  134074            BICB    #7,@ML.REG+40
17062 061350  150577  134070                    BISB    R5,@ML.REG+40
17063 061354  152777  000010  134142            BISB    #10,@ML.REG+120          ;                                                 7828
17064 061362  004767  136756                    JSR     PC,LAST.BLK.XFER         ;                                                 7829
17065 061366  012777  000071  134010            MOV     #71,@ML.REG              ;                                                 7830
17066 061374  012703  000001                    MOV     #1,R3                    ; *,$$TMP2                                        7831
17067 061400  001411                    7$:     BEQ     10$
17068 061402  016704  120510                    MOV     L$DLY,R4
17069 061406  001404                            BEQ     9$                       ; *,$$TMP1
17070 061410  005066  000006            8$:     CLR     6(SP)                    ; $$TMP
17071 061414  005304                            DEC     R4                       ; $$TMP1
17072 061416  001374                            BNE     8$
17073 061420  005303                    9$:     DEC     R3                       ; $$TMP2
17074 061422  000766                            BR      7$
17075 061424  005016                    10$:    CLR     (SP)                     ; ERR.CNT                                         7832
17076 061426  005002                            CLR     R2                       ; WD.CNT                                          7834
17077 061430  017767  134200  133704    11$:    MOV     @ML.REG+230,PD.TEMP      ;                                                 7836
```

```
17079                                          ;ML4AD
17080                                          ;                TEST CODE SECTION
17081
17082 061436  152777  000020  13406C           BISB    #20,@ML.REG+120
17083 061444  012703  000001                   MOV     #1,R3                   ;                                       7837
17084 061450  001411                   12$:    BEQ     15$                     ; *,$$TMP2                              7838
17085 061452  016704  120440                   MOV     L$DLY,R4
17086 061456  001404                           BEQ     14$                     ; *,$$TMP1
17087 061460  005066  000006           13$:    CLR     6(SP)                   ; $$TMP
17088 061464  005304                           DEC     R4                      ; $$TMP1
17089 061466  001374                           BNE     13$
17090 061470  005303                   14$:    DEC     R3                      ; $$TMP2
17091 061472  000766                           BR      12$
17092 061474  017767  134074  131672   15$:    MOV     @ML.REG+170,D1.TEMP
17093 061502  017767  134076  131666           MOV     @ML.REG+200,D2.TEMP
17094 061510  017767  134050  131662           MOV     @ML.REG+160,E2.TEMP
17095 061516  005004                           CLR     R4
17096 061520  010403                   16$:    MOV     R4,R3                   ; NIB.PTR                               7841
17097 061522  006203                           ASR     R3                      ; NIB.PTR,*                             7844
17098 061524  006203                           ASR     R3
17099 061526  006203                           ASR     R3
17100 061530  062703  015342                   ADD     #PD.TEMP,R3
17101 061534  010346                           MOV     R3,-(SP)
17102 061536  010446                           MOV     R4,-(SP)                ; NIB.PTR,*
17103 061540  042716  177770                   BIC     #177770,(SP)
17104 061544  012746  000001                   MOV     #1,-(SP)
17105 061550  005046                           CLR     -(SP)
17106 061552  004767  122056                   JSR     PC,BL$GT2
17107 061556  062706  000010                   ADD     #10,SP
17108 061562  005700                           TST     R0
17109 061564  001011                           BNE     17$
17110 061566  010446                           MOV     R4,-(SP)                ; NIB.PTR,*
17111 061570  010146                           MOV     R1,-(SP)                ; BG.PAT,*
17112 061572  012746  000012                   MOV     #12,-(SP)
17113 061576  060616                           ADD     SP,(SP)                 ; ERR.FLG,*
17114 061600  004767  136572                   JSR     PC,TST.LNG.WRD
17115 061604  062706  000006                   ADD     #6,SP
17116 061610  026627  000004  000001   17$:    CMP     4(SP),#1                ; ERR.FLG,*                             7848
17117 061616  001001                           BNE     18$
17118 061620  005216                           INC     (SP)                    ; ERR.CNT
17119 061622  005204                   18$:    INC     R4                      ; NIB.PTR                               7841
17120 061624  020427  000011                   CMP     R4,#11                  ; NIB.PTR,*
17121 061630  003733                           BLE     16$
17122 061632  005202                           INC     R2                      ; WD.CNT                                7834
17123 061634  020227  000177                   CMP     R2,#177                 ; WD.CNT,*
17124 061640  003673                           BLE     11$
17125 061642  021627  000012                   CMP     (SP),#12                ; ERR.CNT,*                             7854
17126 061646  003437                           BLE     19$
17127 061650  104455                           TRAP    55                      ;                                       7857
17128 061652  000121                           .WORD   121
17129 061654  013114                           .WORD   INTER
17130 061656  026302                           .WORD   DUMPER
17131 061660  012746  012452                   MOV     #FNC.22,-(SP)           ;                                       7858
17132 061664  012746  010660                   MOV     #WRD.13,-(SP)
17133 061670  012746  011676                   MOV     #PHR.4,-(SP)
```

SEQ 0364

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
17135
17136                                   ;ML4AD
17137                                   ;               TEST CODE SECTION
17138 061674  012746  010256                     MOV     #THR.FMT,-(SP)
17139 061700  012746  000004                     MOV     #4,-(SP)
17140 061704  010600                             MOV     SP,R0           ; SP,*
17141 061706  104414                             TRAP    14
17142 061710  012716  011344                     MOV     #WRD.56,(SP)    ;                       7859
17143 061714  012746  012334                     MOV     #FNC.13,-(SP)
17144 061720  012746  010246                     MOV     #TWO.FMT,-(SP)
17145 061724  012746  000003                     MOV     #3,-(SP)
17146 061730  010600                             MOV     SP,R0           ; SP,*
17147 061732  104414                             TRAP    14
17148 061734  104463                             TRAP    63
17149 061736  062706  000020                     ADD     #20,SP          ;
17150 061742  000167  000560                     JMP     38$             ;                       7854
17151 061746  016766  132042  000002  19$:       MOV     LST.BLK,2(SP)   ; *,END.CNT             7856
17152 061754  005366  000002                     DEC     2(SP)           ; END.CNT               7863
17153 061760  005077  133610                     CLR     @ML.REG+170     ;
17154 061764  005077  133614                     CLR     @ML.REG+200     ;                       7864
17155 061770  005077  133570                     CLR     @ML.REG+160     ;                       7865
17156 061774  005005                             CLR     R5              ;                       7866
17157 061776  152777  000040  133440  20$:       BISB    #40,@ML.REG+40  ; TWICE                 7868
17158 062004  016704  134022                     MOV     ML.DUT,R4       ;                       7869
17159 062010  042704  177770                     BIC     #177770,R4
17160 062014  142777  000007  133422             BICB    #7,@ML.REG+40
17161 062022  150477  133416                     BISB    R4,@ML.REG+40
17162 062026  005016                             CLR     (SP)
17163 062030  152777  000010  133466             BISB    #10,@ML.REG+120 ; ERR.CNT               7871
17164 062036  004767  136220                     JSR     PC,FIRST.BLK.XFER ;                     7872
17165 062042  012777  000061  133334             MOV     #61,@ML.REG     ;                       7873
17166 062050  005003                             CLR     R3              ;                       7874
17167 062052  000411                             BR      23$             ; BLK.CNT               7879
17168 062054  005004                     21$:    CLR     R4
17169 062056  152777  000020  133440  22$:       BISB    #20,@ML.REG+120 ; ADRS.CNT              7881
17170 062064  005204                             INC     R4              ;                       7883
17171 062066  020427  000177                     CMP     R4,#177         ; ADRS.CNT              7881
17172 062072  003771                             BLE     22$             ; ADRS.CNT,*
17173 062074  005203                             INC     R3
17174 062076  020366  000002              23$:   CMP     R3,2(SP)        ; BLK.CNT               7879
17175 062102  101764                             BLOS    21$             ; BLK.CNT,END.CNT
17176 062104  152777  000040  133332             BISB    #40,@ML.REG+40  ;                       7884
17177 062112  016704  133714                     MOV     ML.DUT,R4
17178 062116  042704  177770                     BIC     #177770,R4
17179 062122  142777  000007  133314             BICB    #7,@ML.REG+40
17180 062130  150477  133310                     BISB    R4,@ML.REG+40
17181 062134  152777  000010  133362             BISB    #10,@ML.REG+120 ;
17182 062142  004767  136176                     JSR     PC,LAST.BLK.XFER ;                      7887
17183 062146  012777  000071  133230             MOV     #71,@ML.REG     ;                       7888
17184 062154  012703  000001                     MOV     #1,R3           ;                       7889
17185 062160  001411                     24$:    BEQ     27$             ; *,$$TMP2              7890
17186 062162  016704  117730                     MOV     L$DLY,R4
17187 062166  001404                             BEQ     26$             ; *,$$TMP1
17188 062170  005066  000006              25$:   CLR     6(SP)           ; $$TMP
17189 062174  005304                             DEC     R4              ; $$TMP1
```

SEQ 0365

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
17191
17192                                   ;ML4AD
17193                                   ;                TEST CODE SECTION
17194 062176 001374                             BNE     25$
17195 062200 005303                     26$:    DEC     R3                      ; $$TMP2
17196 062202 000766                             BR      24$
17197 062204 005002                     27$:    CLR     R2                      ; WD.CNT                        7892
17198 062206 017767 133422 133126       28$:    MOV     @ML.REG+230,PD.TEMP     ;                               7894
17199 062214 152777 000020 133302               BISB    #20,@ML.REG+120         ;                               7895
17200 062222 012703 000001                      MOV     #1,R3                   ; *,$$TMP2                      7896
17201 062226 001411                     29$:    BEQ     32$
17202 062230 016704 117662                      MOV     L$DLY,R4                ; *,$$TMP1
17203 062234 001404                             BEQ     31$
17204 062236 005066 000006              30$:    CLR     6(SP)                   ; $$TMP
17205 062242 005304                             DEC     R4                      ; $$TMP1
17206 062244 001374                             BNE     30$
17207 062246 005303                     31$:    DEC     R3                      ; $$TMP2
17208 062250 000766                             BR      29$
17209 062252 017767 133316 131114       32$:    MOV     @ML.REG+170,D1.TEMP
17210 062260 017767 133320 131110               MOV     @ML.REG+200,D2.TEMP
17211 062266 017767 133272 131104               MOV     @ML.REG+160,E2.TEMP
17212 062274 005004                             CLR     R4
17213 062276 010403                     33$:    MOV     R4,R3                   ; NIB.PTR                       7899
17214 062300 006203                             ASR     R3                      ; NIB.PTR,*                     7902
17215 062302 006203                             ASR     R3
17216 062304 006203                             ASR     R3
17217 062306 062703 015342                      ADD     #PD.TEMP,R3
17218 062312 010346                             MOV     R3,-(SP)
17219 062314 010446                             MOV     R4,-(SP)                ; NIB.PTR,*
17220 062316 042716 177770                      BIC     #177770,(SP)
17221 062322 012746 000001                      MOV     #1,-(SP)
17222 062326 005046                             CLR     -(SP)
17223 062330 004767 121300                      JSR     PC,BL$GT2
17224 062334 062706 000010                      ADD     #10,SP
17225 062340 005700                             TST     R0
17226 062342 001011                             BNE     34$
17227 062344 010446                             MOV     R4,-(SP)                ; NIB.PTR,*
17228 062346 010146                             MOV     R1,-(SP)                ; BG.PAT,*
17229 062350 012746 000012                      MOV     #12,-(SP)
17230 062354 060616                             ADD     SP,(SP)                 ; ERR.FLG,*
17231 062356 004767 136014                      JSR     PC,TST.LNG.WRD
17232 062362 062706 000006                      ADD     #6,SP
17233 062366 026627 000004 000001 34$:  CMP     4(SP),#1                ; ERR.FLG,*                     7906
17234 062374 001001                             BNE     35$
17235 062376 005216                             INC     (SP)                    ; ERR.CNT
17236 062400 005204                     35$:    INC     R4                      ; NIB.PTR                       7899
17237 062402 020427 000011                      CMP     R4,#11                  ; NIB.PTR,*
17238 062406 003733                             BLE     33$
17239 062410 005202                             INC     R2                      ; WD.CNT                        7892
17240 062412 020227 000177                      CMP     R2,#177                 ; WD.CNT,*
17241 062416 003673                             BLE     28$
17242 062420 021627 000012                      CMP     (SP),#12                ; ERR.CNT,*                     7912
17243 062424 003422                             BLE     36$
17244 062426 104455                             TRAP    55                      ;                               7915
17245 062430 000122                             .WORD   122
```

SEQ 0366

```
17247                                   ;ML4AD
17248                                   ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
17249                                                                                                       29-Mar-1982 16:21:03    PA:<
17250 062432  012706                             .WORD   ASYNC
17251 062434  026302                             .WORD   DUMPER
17252 062436  012746  010630                    MOV     #WRD.10,-(SP)           ;                                               7916
17253 062442  012746  011276                    MOV     #WRD.51,-(SP)
17254 062446  012746  011270                    MOV     #WRD.50,-(SP)
17255 062452  012746  010256                    MOV     #THR.FMT,-(SP)
17256 062456  012746  000004                    MOV     #4,-(SP)
17257 062462  010600                            MOV     SP,R0                   ; SP,*
17258 062464  104414                            TRAP    14
17259 062466  062706  000012                    ADD     #12,SP                  ;                                               7914
17260 062472  005266  000002            36$:    INC     2(SP)                   ; END.CNT                                       7919
17261 062476  005101                            COM     R1                      ; BG.PAT                                        7920
17262 062500  005205                            INC     R5                      ; TWICE                                         7868
17263 062502  020527  000001                    CMP     R5,#1                   ; TWICE,*
17264 062506  003002                            BGT     37$
17265 062510  000167  177262                    JMP     20$
17266 062514  104467                    37$:    TRAP    67                      ;                                               7921
17267 062516  006000                            ROR     R0
17268 062520  103002                            BHIS    38$
17269 062522  000167  176434                    JMP     1$
17270 062526  062706  000010            38$:    ADD     #10,SP                  ;                                               7772
17271 062532  000207                            RTS     PC
17272
17273                                   ; Routine Size:  377 words
17274                                   ; Maximum stack depth per invocation:  18 words
17279
17280
17284
17285                                           .SBTTL  T31 TEST CODE SECTION
17289 062534                            T31::
17290 062534  004767  176412            1$:     JSR     PC,$T31                 :                                               7923
17291 062540  104466                            TRAP    66
17292 062542  006000                            ROR     R0
17293 062544  103773                            BLO     1$
17294 062546  000207                            RTS     PC
17295
17296                                   ; Routine Size:  6 words
17297                                   ; Maximum stack depth per invocation:  0 words
```

SEQ 0367

```
17303 ;ML4AD
17304 ;                 TEST CODE SECTION                       29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
17305                                                           29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (83)
17306 ;          7925   !<BLF/PAGE>
```

```
17308 ;ML4AD
17309 ;                 TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
17310 ;                                                      29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (84)
17311 ;          7926 !
17312 ;          7927 !
17313 ;          7928 BGNTST;
17314 ;          7929
17315 ;          7930 !++
17316 ;          7931 ! TEST NUMBER:  TST 32
17317 ;          7932 !
17318 ;          7933 ! TEST NAME:    ARRAY MODULE SELECTION TEST
17319 ;          7934 !
17320 ;          7935 ! TEST DESCRIPTION:
17321 ;          7936 !
17322 ;          7937 !       TEST FOR UNIQUE ARRAY MODULE
17323 ;          7938 !       SELECTION BY:
17324 ;          7939 !
17325 ;          7940 !       1.  WRITING THE RESPECTIVE ARRAY
17326 ;          7941 !           MODULE POSITION NUMBER INTO
17327 ;          7942 !           THE FIRST GOOD NIBBLE FOUND
17328 ;          7943 !           IN THE ARRAY.  DO FOR ALL
17329 ;          7944 !           PRESENT ARRAYS.
17330 ;          7945 !
17331 ;          7946 !       2.  READ THE ARRAYS FOR THEIR
17332 ;          7947 !           RESPECTIVE POSTION NUMBERS.
17333 ;          7948 !
17334 ;          7949 ! IMPLICIT INPUTS:
17335 ;          7950 !       PD_TEMP:
17336 ;          7951 !       A BIT VECTOR OF 16 BITS WHERE
17337 ;          7952 !       THE READ PROM DATA IS STORED
17338 ;          7953 !       AND ACCESSED FROM.
17339 ;          7954 !
17340 ;          7955 !       IO_BUF :
17341 ;          7956 !       A VECTOR OF 256 WORDS WHERE
17342 ;          7957 !       DATA FOR MBUS READS AND WRITE
17343 ;          7958 !       FUNCTION ARE FOUND.
17344 ;          7959 !
17345 ;          7960 !--
17346 ;          7961
17347 ;          7962 local
17348 ;          7963     WRDS_TSTED,                          !WORDS TESTED
17349 ;          7964     ARR_SEL,                             !ARRAY SELECT
17350 ;          7965     FND_GD_NIB,                          !FOUND GOOD NIBBLE
17351 ;          7966     ARR_NUM;                             !ARRAY NUMBER
17352 ;          7967
17353 ;          7968 ARR_SEL = ZEROES;
17354 ;          7969 ARR_SEL = .ARR_SEL - .ARR_INC;           !START ARR_SEL BACK ONE ARRAY
17355 ;          7970
17356 ;          7971 incr ARR_CNT from 0 to .OP_NUM_ARR do    !TEST ALL PRESENT ARRAYS
17357 ;          7972     begin
17358 ;          7973     CLR_MBUS;
17359 ;          7974     FND_GD_NIB = ZERO;
17360 ;          7975     WRDS_TSTED = ZERO;
17361 ;          7976     DAT_DM = ONE;                        !SET DATA DIAG MODE
17362 ;          7977     MLWC = not 255;                      !LOAD WORD COUNT
```

```
17364 ;ML4AD
17365 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
17366 ;                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (84)
17367 ;          7978       MLBA = IO_BUF;                                  !LOAD IO_BUF
17368 ;          7979       ARR_SEL = .ARR_SEL + .ARR_INC;                  !LOOK AT NEXT ARRAY
17369 ;          7980       MLDA = .ARR_SEL;                                !LOAD DSA WITH SELECTED ARRAY AT BLOCK ZERO
17370 ;          7981       MLCS1 = write;                                  !DO A WRITE FUNCTION
17371 ;          7982
17372 ;          7983       do                                              !THIS LOOP WRITES ARRAY NUMBERS TO THE ARRAYS
17373 ;          7984           begin
17374 ;          7985           DELAY (ONE_US);
17375 ;          7986           PD_TEMP = .MLPD;                            !GET THE PROM DATA
17376 ;          7987           WRDS_TSTED = .WRDS_TSTED + 1;               !COUNT WORDS TESTED
17377 ;          7988
17378 ;          7989           incr CNT from 0 to 8 do                     !LOOK AT 9 NIBBLES
17379 ;          7990
17380 ;          7991               if .PD_TEMP [.CNT] IS_NOT_SET           !FIND A GOOD NIBBLE
17381 ;          7992               then
17382 ;          7993                   begin
17383 ;          7994
17384 ;          7995                   case .CNT from 0 to 8 of            !SELECT AND LOAD GOOD NIBBLE WITH ARRAY CNT
17385 ;          7996                       set
17386 ;          7997
17387 ;          7998                       [0] :
17388 ;          7999                           (MLD1)<0, 4> = .ARR_CNT;        !NIBBLE 0
17389 ;          8000
17390 ;          8001                       [1] :
17391 ;          8002                           (MLD1)<4, 4> = .ARR_CNT;        !NIBBLE 1
17392 ;          8003
17393 ;          8004                       [2] :
17394 ;          8005                           (MLD1)<8, 4> = .ARR_CNT;        !NIBBLE 2
17395 ;          8006
17396 ;          8007                       [3] :
17397 ;          8008                           (MLD1)<12, 4> = .ARR_CNT;       !NIBBLE 3
17398 ;          8009
17399 ;          8010                       [4] :
17400 ;          8011                           (MLD2)<0, 4> = .ARR_CNT;        !NIBBLE 4
17401 ;          8012
17402 ;          8013                       [5] :
17403 ;          8014                           (MLD2)<4, 4> = .ARR_CNT;        !NIBBLE 5
17404 ;          8015
17405 ;          8016                       [6] :
17406 ;          8017                           (MLD2)<8, 4> = .ARR_CNT;        !NIBBLE 6
17407 ;          8018
17408 ;          8019                       [7] :
17409 ;          8020                           (MLD2)<12, 4> = .ARR_CNT;       !NIBBLE 7
17410 ;          8021
17411 ;          8022                       [8] :
17412 ;          8023                           (MLE2)<8, 4> = .ARR_CNT;        !NIBBLE 8
17413 ;          8024                       tes;
17414 ;          8025
17415 ;          8026               FND_GD_NIB = ONE;                       !SET FOUND GOOD NIBBLE FLG
17416 ;          8027               exitloop;                               !EXIT THE LOOP
17417 ;          8028               end;
17418 ;          8029
```

```
17420 ;ML4AD
17421 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
17422 ;                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (84)
17423 ;          8030           DAT_CLK = ONE;                              !CLOCK GOOD NIBBLE INTO MEMORY & GET NXT PROM WRD
17424 ;          8031           end
17425 ;          8032       until (.FND_GD_NIB IS_SET ) or (.WRDS_TSTED eql 14);        !DO UNTIL FOUND GOOD NIBBLE OR 14 WRDS TSTED
17426 ;          8033
17427 ;          8034       if .WRDS_TSTED eql 14                           !IF 14 WORDS TSTED
17428 ;          8035       then
17429 ;          8036           begin                                       !THEN ERROR AND EXIT TESTED
17430 ;          8037           ERRDF (142, INTER, DUMPER);
17431 ;          8038           PRINTB (SEV_FAT, WRD_14, PHR_10, FNC_15, WRD_12, FNC_17, WRD_37, WRD_56);
17432 ;          8039           EXIT_TST;
17433 ;          8040           end;
17434 ;          8041
17435 ;          8042       end;
17436 ;          8043
17437 ;          8044   ARR_SEL = ZEROES;
17438 ;          8045   ARR_SEL = .ARR_SEL - .ARR_INC;                      !START ARR SEL BACK ONE AGAIN
17439 ;          8046
17440 ;          8047   incr ARR_CNT from 0 to .OP_NUM_ARR do               !TEST ALL PRESENT ARRAYS
17441 ;          8048       begin
17442 ;          8049       BGNSUB;
17443 ;          8050       CLR_MBUS;
17444 ;          8051       FND_GD_NIB = ZERO;
17445 ;          8052       DAT_DM = ONE;                                   !SET DATA DIAG MODE
17446 ;          8053       MLWC = not 255;                                 !LOAD WORD COUNT
17447 ;          8054       MLBA = IO_BUF;                                  !LOAD UBUS ADDRESS
17448 ;          8055       ARR_SEL = .ARR_SEL + .ARR_INC;                  !LOOK AT NEXT ARRAY
17449 ;          8056       MLDA = .ARR_SEL;                                !LOAD DSA WITH ARRAY SELECT
17450 ;          8057       MLCS1 = read;                                   !DO A READ FUNCTION
17451 ;          8058
17452 ;          8059       do                                              !THIS LOOP READS ARRAYS FOR ARRAY NUMBERS
17453 ;          8060           begin
17454 ;          8061           DELAY (ONE_US);
17455 ;          8062           PD_TEMP = .MLPD;                            !GET THE PROM DATA
17456 ;          8063
17457 ;          8064           incr CNT from 0 to 8 do                     !LOOK AT 9 NIBBLES
17458 ;          8065
17459 ;          8066               if .PD_TEMP [.CNT] IS_NOT_SET           !FIND THE GOOD NIBBLES WHERE ARR NUM IS STORED
17460 ;          8067               then
17461 ;          8068                   begin
17462 ;          8069                   DAT_CLK = ONE;                      !CLOCK ARRAY WORD OUT
17463 ;          8070
17464 ;          8071                   case .CNT from 0 to 8 of            !SELECT AND READ GOOD NIBBLE
17465 ;          8072                       set
17466 ;          8073
17467 ;          8074                       [0] :
17468 ;          8075                           ARR_NUM = .(MLD1)<0, 4>;        !NIBBLE 0
17469 ;          8076
17470 ;          8077                       [1] :
17471 ;          8078                           ARR_NUM = .(MLD1)<4, 4>;        !NIBBLE 1
17472 ;          8079
17473 ;          8080                       [2] :
17474 ;          8081                           ARR_NUM = .(MLD1)<8, 4>;        !NIBBLE 2
```

```
17476 ;ML4AD
17477 ;                 TEST CODE SECTION
17478
17479 ;          8082
17480 ;          8083                        [3] :
17481 ;          8084                            ARR_NUM = .(MLD1)<12, 4>;       !NIBBLE 3
17482 ;          8085
17483 ;          8086                        [4] :
17484 ;          8087                            ARR_NUM = .(MLD2)<0, 4>;        !NIBBLE 4
17485 ;          8088
17486 ;          8089                        [5] :
17487 ;          8090                            ARR_NUM = .(MLD2)<4, 4>;        !NIBBLE 5
17488 ;          8091
17489 ;          8092                        [6] :
17490 ;          8093                            ARR_NUM = .(MLD2)<8, 4>;        !NIBBLE 6
17491 ;          8094
17492 ;          8095                        [7] :
17493 ;          8096                            ARR_NUM = .(MLD2)<12, 4>;       !NIBBLE 7
17494 ;          8097
17495 ;          8098                        [8] :
17496 ;          8099                            ARR_NUM = .(MLE2)<8, 4>;        !NIBBLE 8
17497 ;          8100                        tes;
17498 ;          8101
17499 ;          8102                    FND_GD_NIB = ONE;                   !SET FND GD NIB FLG
17500 ;          8103                    exitloop;                           !EXIT LOOP
17501 ;          8104                    end;
17502 ;          8105
17503 ;          8106            DAT_CLK = ONE;                              !CLOCK OUT NEXT PROM LOCATION
17504 ;          8107            end
17505 ;          8108        until .FND_GD_NIB IS_SET;                       !REPEAT UNTIL FOUND THE GOOD NIBBLE
17506 ;          8109
17507 ;          8110        if .ARR_CNT neq .ARR_NUM                        !SEE IF ARRAY CONTAINS IT'S ARR NUM
17508 ;          8111        then
17509 ;          8112            begin                                       !ERROR IF NOT THERE
17510 ;          8113            ERRDF (83, ASYNC, DUMPER);
17511 ;          8114            PRINTB (FOR_FMT, WRD_39, FNC_17, WRD_37, WRD_10);
17512 ;          8115            PRINTB (FMT_14, .ARR_CNT, .ARR_NUM);
17513 ;          8116            end;
17514 ;          8117
17515 ;          8118        ENDSUB;
17516 ;          8119        end;
17517 ;          8120
17518 ;          8121    ENDTST;
17522
17523
17527 062550  004167  122036          $T32:   .SBTTL  $T32 TEST CODE SECTION
                                              JSR     R1,$SAVE5                 ;                       7924
17528 062554  162706  000006                  SUB     #6,SP
17529 062560  005005                          CLR     R5                        ; ARR.SEL               7968
17530 062562  166705  131216                  SUB     ARR.INC,R5                ; *,ARR.SEL             7969
```

```
17532                                   ;ML4AD
17533                                   ;               TEST CODE SECTION
17534
17535 062566 016766 131210 000002       MOV     OP.NUM.ARR,2(SP)
17536 062574 005003                     CLR     R3              ;
17537 062576 000167 000702              JMP     27$             ; ARR.CNT                                       7971
17538 062602 152777 000040 132634 1$:   BISB    #40,@ML.REG+40  ;                                               7972
17539 062610 016702 133216              MOV     ML.DUT,R2
17540 062614 042702 177770              BIC     #177770,R2
17541 062620 142777 000007 132616       BICB    #7,@ML.REG+40
17542 062626 150277 132612              BISB    R2,@ML.REG+40
17543 062632 005016                     CLR     (SP)
17544 062634 005004                     CLR     R4              ; FND.GD.NIB                                    7974
17545 062636 152777 000010 132660       BISB    #10,@ML.REG+120 ; WRDS.TSTED                                    7975
17546 062644 012777 177400 132542       MOV     #-400,@ML.REG+10;                                               7976
17547 062652 012777 014022 132544       MOV     #10.BUF,@ML.REG+20;                                             7977
17548 062660 066705 131120              ADD     ARR.INC,R5      ;                                               7978
17549 062664 010577 132544              MOV     R5,@ML.REG+30   ; *,ARR.SEL                                     7979
17550 062670 012777 000061 132506       MOV     #61,@ML.REG     ; ARR.SEL,*                                     7980
17551 062676 012701 000001          2$: MOV     #1,R1           ;                                               7981
17552 062702 001411                 3$: BEQ     6$              ; *,$$TMP2                                      7985
17553 062704 016702 117206              MOV     L$DLY,R2
17554 062710 001404                     BEQ     5$              ; *,$$TMP1
17555 062712 005066 000004          4$: CLR     4(SP)           ; $$TMP
17556 062716 005302                     DEC     R2              ; $$TMP1
17557 062720 001374                     BNE     4$
17558 062722 005301                 5$: DEC     R1              ; $$TMP2
17559 062724 000766                     BR      3$
17560 062726 017767 132702 132406 6$:   MOV     @ML.REG+230,PD.TEMP;                                            7986
17561 062734 005204                     INC     R4              ; WRDS.TSTED                                    7987
17562 062736 005002                     CLR     R2              ; CNT                                           7989
17563 062740 010201                 7$: MOV     R2,R1           ; CNT,*                                         7991
17564 062742 006201                     ASR     R1
17565 062744 006201                     ASR     R1
17566 062746 006201                     ASR     R1
17567 062750 062701 015342              ADD     #PD.TEMP,R1
17568 062754 010146                     MOV     R1,-(SP)
17569 062756 010246                     MOV     R2,-(SP)        ; CNT,*
17570 062760 042716 177770              BIC     #177770,(SP)
17571 062764 012746 000001              MOV     #1,-(SP)
17572 062770 005046                     CLR     -(SP)
17573 062772 004767 120636              JSR     PC,BL$GT2
17574 062776 062706 000010              ADD     #10,SP
17575 063002 005700                     TST     R0
17576 063004 001155                     BNE     23$
17577 063006 010201                     MOV     R2,R1           ; CNT,*                                         7995
17578 063010 006301                     ASL     R1
17579 063012 066107 063016              ADD     8$(R1),PC
17580 063016 000022                 8$: .WORD   9$-8$
17581 063020 000040                     .WORD   10$-8$
17582 063022 000072                     .WORD   12$-8$
17583 063024 000112                     .WORD   13$-8$
17584 063026 000146                     .WORD   15$-8$
17585 063030 000164                     .WORD   16$-8$
17586 063032 000216                     .WORD   18$-8$
```

```
17588                                   ;ML4AD
17589                                   ;               TEST CODE SECTION
17590
17591 063034  000236                              .WORD   19$-8$
17592 063036  000272                              .WORD   21$-8$
17593 063040  010301                      9$:     MOV     R3,R1
17594 063042  042701  177760                      BIC     #177760,R1      ; ARR.CNT,*     7999
17595 063046  142777  000017  132520              BICB    #17,@ML.REG+170
17596 063054  000412                              BR      11$
17597 063056  010301                      10$:    MOV     R3,R1           ; ARR.CNT,*     8002
17598 063060  006301                              ASL     R1
17599 063062  006301                              ASL     R1
17600 063064  006301                              ASL     R1
17601 063066  006301                              ASL     R1
17602 063070  042701  177417                      BIC     #177417,R1
17603 063074  142777  000360  132472              BICB    #360,@ML.REG+170
17604 063102  150177  132466              11$:    BISB    R1,@ML.REG+170
17605 063106  000511                              BR      22$
17606 063110  010301                      12$:    MOV     R3,R1           ;               7995
17607 063112  000301                              SWAB    R1              ; ARR.CNT,*     8005
17608 063114  042701  170377                      BIC     #170377,R1
17609 063120  042777  007400  132446              BIC     #7400,@ML.REG+170
17610 063126  000413                              BR      14$
17611 063130  010301                      13$:    MOV     R3,R1           ; ARR.CNT,*     8008
17612 063132  000301                              SWAB    R1
17613 063134  006301                              ASL     R1
17614 063136  006301                              ASL     R1
17615 063140  006301                              ASL     R1
17616 063142  006301                              ASL     R1
17617 063144  042701  007777                      BIC     #7777,R1
17618 063150  042777  170000  132416              BIC     #170000,@ML.REG+170
17619 063156  050177  132412              14$:    BIS     R1,@ML.REG+170
17620 063162  000463                              BR      22$
17621 063164  010301                      15$:    MOV     R3,R1           ;               7995
17622 063166  042701  177760                      BIC     #177760,R1      ; ARR.CNT,*     8011
17623 063172  142777  000017  132404              BICB    #17,@ML.REG+200
17624 063200  000412                              BR      17$
17625 063202  010301                      16$:    MOV     R3,R1           ; ARR.CNT,*     8014
17626 063204  006301                              ASL     R1
17627 063206  006301                              ASL     R1
17628 063210  006301                              ASL     R1
17629 063212  006301                              ASL     R1
17630 063214  042701  177417                      BIC     #177417,R1
17631 063220  142777  000360  132356              BICB    #360,@ML.REG+200
17632 063226  150177  132352              17$:    BISB    R1,@ML.REG+200
17633 063232  000437                              BR      22$
17634 063234  010301                      18$:    MOV     R3,R1           ;               7995
17635 063236  000301                              SWAB    R1              ; ARR.CNT,*     8017
17636 063240  042701  170377                      BIC     #170377,R1
17637 063244  042777  007400  132332              BIC     #7400,@ML.REG+200
17638 063252  000413                              BR      20$
17639 063254  010301                      19$:    MOV     R3,R1           ; ARR.CNT,*     8020
17640 063256  000301                              SWAB    R1
17641 063260  006301                              ASL     R1
17642 063262  006301                              ASL     R1
```

```
17644
17645                                   ;ML4AD
17646                                   ;               TEST CODE SECTION
17647 063264  006301                              ASL     R1
17648 063266  006301                              ASL     R1
17649 063270  042701  007777                      BIC     #7777,R1
17650 063274  042777  170000  132302              BIC     #170000,@ML.REG+200
17651 063302  050177  132276              20$:    BIS     R1,@ML.REG+200
17652 063306  000411                              BR      22$
17653 063310  010301                      21$:    MOV     R3,R1                   ;
17654 063312  000301                              SWAB    R1                      ; ARR.CNT,*                             7995
17655 063314  042701  170377                      BIC     #170377,R1                                                      8023
17656 063320  042777  007400  132236              BIC     #7400,@ML.REG+160
17657 063326  050177  132232                      BIS     R1,@ML.REG+160
17658 063332  012716  000001              22$:    MOV     #1,(SP)
17659 063336  000406                              BR      24$                     ; *,FND.GD.NIB                          8026
17660 063340  005202                      23$:    INC     R2                      ;                                       8027
17661 063342  020227  000010                      CMP     R2,#10                  ; CNT                                   7989
17662 063346  003002                              BGT     24$                     ; CNT,*
17663 063350  000167  177364                      JMP     7$
17664 063354  152777  000020  132142      24$:    BISB    #20,@ML.REG+120
17665 063362  021627  000001                      CMP     (SP),#1                 ;                                       8030
17666 063366  001405                              BEQ     25$                     ; FND.GD.NIB,*                          8032
17667 063370  020427  000016                      CMP     R4,#16
17668 063374  001402                              BEQ     25$                     ; WRDS.TSTED,*
17669 063376  000167  177274                      JMP     2$
17670 063402  020427  000016              25$:    CMP     R4,#16                  ; WRDS.TSTED,*                          8034
17671 063406  001035                              BNE     26$
17672 063410  104455                              TRAP    55                      ;                                       8037
17673 063412  000216                              .WORD   216
17674 063414  013114                              .WORD   INTER
17675 063416  026302                              .WORD   DUMPER
17676 063420  012746  011344                      MOV     #WRD.56,-(SP)           ;                                       8038
17677 063424  012746  011122                      MOV     #WRD.37,-(SP)
17678 063430  012746  012404                      MOV     #FNC.17,-(SP)
17679 063434  012746  010650                      MOV     #WRD.12,-(SP)
17680 063440  012746  012360                      MOV     #FNC.15,-(SP)
17681 063444  012746  012006                      MOV     #PHR.10,-(SP)
17682 063450  012746  010664                      MOV     #WRD.14,-(SP)
17683 063454  012746  010342                      MOV     #SEV.FMT,-(SP)
17684 063460  012746  000010                      MOV     #10,-(SP)
17685 063464  010600                              MOV     SP,R0                   ; SP,*
17686 063466  104414                              TRAP    14
17687 063470  104463                              TRAP    63
17688 063472  062706  000022                      ADD     #22,SP                  ;                                       8034
17689 063476  000167  000562                      JMP     56$                     ;                                       8036
17690 063502  005203                      26$:    INC     R3                      ; ARR.CNT                               7971
17691 063504  020366  000002              27$:    CMP     R3,2(SP)                ; ARR.CNT,*
17692 063510  003002                              BGT     28$
17693 063512  000167  177064                      JMP     1$
17694 063516  005005                      28$:    CLR     R5                      ; ARR.SEL                               8044
17695 063520  166705  130260                      SUB     ARR.INC,R5              ; *,ARR.SEL                             8045
17696 063524  016766  130252  000002              MOV     OP.NUM.ARR,2(SP)        ;                                       8047
17697 063532  005004                              CLR     R4                      ; ARR.CNT
17698 063534  000167  000516                      JMP     55$
```

```
                                                                                29-Mar-1982 16:23:04    TOPS
                                                                                29-Mar-1982 16:21:03    PA:<
17700
17701                                   ;ML4AD
17702                                   ;               TEST CODE SECTION
17703 063540  104402                  29$:    TRAP    2
17704 063542  152777  000040  131674          BISB    #40,@ML.REG+40          ;                       8048
17705 063550  016702  132256                  MOV     ML.DUT,R2               ;                       8049
17706 063554  042702  177770                  BIC     #177770,R2
17707 063560  142777  000007  131656          BICB    #7,@ML.REG+40
17708 063566  150277  131652                  BISB    R2,@ML.REG+40
17709 063572  005016                          CLR     (SP)
17710 063574  152777  000010  131722          BISB    #10,@ML.REG+120         ; FND.GD.NIB            8051
17711 063602  012777  177400  131604          MOV     #-400,@ML.REG+10        ;                       8052
17712 063610  012777  014022  131606          MOV     #IO.BUF,@ML.REG+20      ;                       8053
17713 063616  066705  130162                  ADD     ARR.INC,R5              ;                       8054
17714 063622  010577  131606                  MOV     R5,@ML.REG+30           ; *,ARR.SEL             8055
17715 063626  012777  000071  131550          MOV     #71,@ML.REG             ; ARR.SEL,*             8056
17716 063634  012701  000001                  MOV     #1,R1                   ;                       8057
17717 063640  001411                  31$:    BEQ     34$                     ; *,$$TMP2              8061
17718 063642  016702  116250                  MOV     L$DLY,R2
17719 063646  001404                          BEQ     33$                     ; *,$$TMP1
17720 063650  005066  000004          32$:    CLR     4(SP)                   ; $$TMP
17721 063654  005302                          DEC     R2                      ; $$TMP1
17722 063656  001374                          BNE     32$
17723 063660  005301                  33$:    DEC     R1                      ; $$TMP2
17724 063662  000766                          BR      31$
17725 063664  017767  131744  131450  34$:    MOV     @ML.REG+230,PD.TEMP     ;                       8062
17726 063672  005002                          CLR     R2
17727 063674  010201                  35$:    MOV     R2,R1                   ; CNT                   8064
17728 063676  006201                          ASR     R1                      ; CNT,*                 8066
17729 063700  006201                          ASR     R1
17730 063702  006201                          ASR     R1
17731 063704  062701  015342                  ADD     #PD.TEMP,R1
17732 063710  010146                          MOV     R1,-(SP)
17733 063712  010246                          MOV     R2,-(SP)                ; CNT,*
17734 063714  042716  177770                  BIC     #177770,(SP)
17735 063720  012746  000001                  MOV     #1,-(SP)
17736 063724  005046                          CLR     -(SP)
17737 063726  004767  117702                  JSR     PC,BL$GT2
17738 063732  062706  000010                  ADD     #10,SP
17739 063736  005700                          TST     R0
17740 063740  001070                          BNE     50$
17741 063742  152777  000020  131554          BISB    #20,@ML.REG+120         ;                       8069
17742 063750  010201                          MOV     R2,R1                   ; CNT,*                 8071
17743 063752  006301                          ASL     R1
17744 063754  066107  063760                  ADD     36$(R1),PC
17745 063760  000022                  36$:    .WORD   37$-36$
17746 063762  000030                          .WORD   38$-36$
17747 063764  000036                          .WORD   39$-36$
17748 063766  000044                          .WORD   40$-36$
17749 063770  000052                          .WORD   41$-36$
17750 063772  000060                          .WORD   42$-36$
17751 063774  000076                          .WORD   44$-36$
17752 063776  000104                          .WORD   45$-36$
17753 064000  000122                          .WORD   47$-36$
17754 064002  117703  131566          37$:    MOVB    @ML.REG+170,R3          ; *,ARR.NUM             8075
```

SEQ 0376

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
17756                                   ;ML4AD
17757                                   ;               TEST CODE SECTION
17758
17759 064006  000440                            BR      49$
17760 064010  117703  131560              38$:  MOVB    @ML.REG+170,R3          ; *,ARR.NUM                     8078
17761 064014  000413                            BR      43$
17762 064016  017703  131552              39$:  MOV     @ML.REG+170,R3          ; *,ARR.NUM                     8081
17763 064022  000431                            BR      48$
17764 064024  017703  131544              40$:  MOV     @ML.REG+170,R3          ; *,ARR.NUM                     8084
17765 064030  000417                            BR      46$
17766 064032  117703  131546              41$:  MOVB    @ML.REG+200,R3          ; *,ARR.NUM                     8087
17767 064036  000424                            BR      49$
17768 064040  117703  131540              42$:  MOVB    @ML.REG+200,R3          ; *,ARR.NUM                     8090
17769 064044  006203              43$:  ASR     R3                      ; ARR.NUM
17770 064046  006203                    ASR     R3                      ; ARR.NUM
17771 064050  006203                    ASR     R3                      ; ARR.NUM
17772 064052  006203                    ASR     R3                      ; ARR.NUM
17773 064054  000415                            BR      49$
17774 064056  017703  131522              44$:  MOV     @ML.REG+200,R3          ; *,ARR.NUM                     8093
17775 064062  000411                            BR      48$
17776 064064  017703  131514              45$:  MOV     @ML.REG+200,R3          ; *,ARR.NUM                     8096
17777 064070  006203              46$:  ASR     R3                      ; ARR.NUM
17778 064072  006203                    ASR     R3                      ; ARR.NUM
17779 064074  006203                    ASR     R3                      ; ARR.NUM
17780 064076  006203                    ASR     R3                      ; ARR.NUM
17781 064100  000402                            BR      48$
17782 064102  017703  131456              47$:  MOV     @ML.REG+160,R3          ; *,ARR.NUM                     8099
17783 064106  000303              48$:  SWAB    R3                      ; ARR.NUM
17784 064110  042703  177760              49$:  BIC     #177760,R3              ; *,ARR.NUM
17785 064114  012716  000001                    MOV     #1,(SP)                 ; *,FND.GD.NIB                  8102
17786 064120  000404                            BR      51$                     ;                               8103
17787 064122  005202              50$:  INC     R2                      ; CNT                           8064
17788 064124  020227  000010                    CMP     R2,#10                  ; CNT,*
17789 064130  003661                            BLE     35$
17790 064132  152777  000020  131364  51$:  BISB    #20,@ML.REG+120         ;                               8106
17791 064140  021627  000001                    CMP     (SP),#1                 ; FND.GD.NIB,*                  8108
17792 064144  001233                            BNE     30$
17793 064146  020403                            CMP     R4,R3                   ; ARR.CNT,ARR.NUM               8110
17794 064150  001434                            BEQ     52$
17795 064152  104455                            TRAP    55                      ;                               8113
17796 064154  000123                            .WORD   123
17797 064156  012706                            .WORD   ASYNC
17798 064160  026302                            .WORD   DUMPER
17799 064162  012746  010630                    MOV     #WRD.10,-(SP)           ;                               8114
17800 064166  012746  011122                    MOV     #WRD.37,-(SP)
17801 064172  012746  012404                    MOV     #FNC.17,-(SP)
17802 064176  012746  011140                    MOV     #WRD.39,-(SP)
17803 064202  012746  010270                    MOV     #FOR.FMT,-(SP)
17804 064206  012746  000005                    MOV     #5,-(SP)
17805 064212  010600                            MOV     SP,R0                   ; SP,*
17806 064214  104414                            TRAP    14
17807 064216  010316                            MOV     R3,(SP)                 ; ARR.NUM,*                     8115
17808 064220  010446                            MOV     R4,-(SP)                ; ARR.CNT,*
17809 064222  012746  007330                    MOV     #FMT.14,-(SP)
17810 064226  012746  000003                    MOV     #3,-(SP)
```

```
17812                                   ;ML4AD
17813                                   ;                TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS
17814                                                                                        29-Mar-1982 16:21:03    PA:<
17815 064232  010600                            MOV     SP,R0                   ; SP,*
17816 064234  104414                            TRAP    14
17817 064236  062706  000022                    ADD     #22,SP                  ;                                       8112
17818 064242  104467                    52$:    TRAP    67                      ;                                       8116
17819 064244  006000                            ROR     R0
17820 064246  103002                            BHIS    54$
17821 064250  000167  177264            53$:    JMP     29$
17822 064254  005204                    54$:    INC     R4                      ; ARR.CNT
17823 064256  020466  000002            55$:    CMP     R4,2(SP)                ; ARR.CNT,*                             8047
17824 064262  003772                            BLE     53$
17825 064264  062706  000006            56$:    ADD     #6,SP                   ;                                       7924
17826 064270  000207                            RTS     PC
17827
17828                                   ; Routine Size:  425 words
17829                                   ; Maximum stack depth per invocation:  18 words
17834
17835
17839
17840                                           .SBTTL  T32 TEST CODE SECTION
17844 064272                            T32::
17845 064272  004767  176252            1$:     JSR     PC,$T32                 ;                                       8119
17846 064276  104466                            TRAP    66
17847 064300  006000                            ROR     R0
17848 064302  103773                            BLO     1$
17849 064304  000207                            RTS     PC
17850
17851                                   ; Routine Size:  6 words
17852                                   ; Maximum stack depth per invocation:  0 words
17857
17858
17859 ;                 8122  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (85)

```
17861 ;ML4AD
17862 ;                 TEST CODE SECTION
17863 ;
17864 ;          8123 !
17865 ;          8124 BGNTST;
17866 ;          8125
17867 ;          8126 !++
17868 ;          8127 ! TEST NUMBER:  TST 33
17869 ;          8128 !
17870 ;          8129 ! TEST NAME:    SEQUENCER EXISTENCE TEST
17871 ;          8130 !
17872 ;          8131 ! TEST DESCRIPTION:
17873 ;          8132 !
17874 ;          8133 !       TEST SEQUENCER TIMING AND CONTROL
17875 ;          8134 !       LOGIC FOR EXISTENCE BY:
17876 ;          8135 !
17877 ;          8136 !       1.  FIRST WRITING THE GOOD BLOCK VIA DAT_DM MODE WITH A BACKGROUND
17878 ;          8137 !           PATTERN OF ONES.
17879 ;          8138 !
17880 ;          8139 !
17881 ;          8140 !       2.  VIA A MBUS WRITE FUNCTION WRITE ONES INTO THE GOOD BLOCK.
17882 ;          8141 !
17883 ;          8142 !       3.  THEN VIA DAT_DM READ GOOD NIBBLES IN THE GOOD BLOCK AND XOR THEM
17884 ;          8143 !           AGAINST THE BACKGROUND PATTERN.
17885 ;          8144 !
17886 ;          8145 !           RECORD THE NUMBER OF NIBBLES THAT WERE UNCHANGED OR PARTIALLY CHANGED
17887 ;          8146 !           BY THE MBUS WIRTE.
17888 ;          8147 !
17889 ;          8148 !       4.  REPORT A FATAL ERROR AND DROP THIS UNIT IF THE NIBBLES TESTED EQUALS
17890 ;          8149 !           THE COUNT OF UNCHANGED NIBBLES.
17891 ;          8150 !
17892 ;          8151 !           REPORT AN INTERMEDIATE DIAGNOSTIC MESSAGE IF AT LEAST SOME NIBBLES
17893 ;          8152 !           WERE CHANGED BY THE MBUS WRITE.
17894 ;          8153 !
17895 ;          8154 ! IMPLICIT INPUTS:
17896 ;          8155 !       PD_TEMP:
17897 ;          8156 !       A BIT VECTOR OF 16 BITS WHERE THE READ PROM DATA IS STORED AND ACCESSED FROM.
17898 ;          8157 !
17899 ;          8158 !       IO_BUF:
17900 ;          8159 !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE FUNCTIONS
17901 ;          8160 !       ARE FOUND.
17902 ;          8161 !--
17903 ;          8162
17904 ;          8163 local
17905 ;          8164     BG_PAT,                        !BACK GROUND PATTERN
17906 ;          8165     SUM_BAD,                       !SUM NIBBLE ARE BAD
17907 ;          8166     NIB_TSTED,                     !NIBBLES TESTED
17908 ;          8167     ALL_BAD,                       !ALL NIBBLES ARE BAD
17909 ;          8168     RESULT;                        !RESULTS OF XOR
17910 ;          8169
17911 ;          8170 CLR_MBUS:
17912 ;          8171 BG_PAT = ONES;                     !BACKGROUND EQL ONES
17913 ;          8172 MLD1 = .BG_PAT;                    !LOAD DATA DIAG REG WITH BG PAT
17914 ;          8173 MLD2 = .BG_PAT;
17915 ;          8174 MLE2 = .BG_PAT;
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (85)

```
17917 ;ML4AD
17918 ;                 TEST CODE SECTION
17919 ;
17920 ;          8175  DAT_DM_XFER ();                    !SET UP A  DATA DIAG XFERR AT THE GOOD BLOCK
17921 ;          8176  MLCS1 = write;                     !DO A WRITE FUNCTION
17922 ;          8177
17923 ;          8178  incr WD_CNT from 0 to 127 do       !WRITE BLOCK WITH BG PAT
17924 ;          8179      begin
17925 ;          8180      DELAY (ONE_US);
17926 ;          8181      DAT_CLK = ONE;
17927 ;          8182      end;
17928 ;          8183
17929 ;          8184  CLR_MBUS;
17930 ;          8185  IO_BUF = ZEROES;                   !LOAD FIRST WORD OF IO_BUF
17931 ;          8186  BAI = ONE;                         !MAKE XFERR SET ON ONE_ADRS
17932 ;          8187  GD_BLK_XFER ();                    !SET UP A GOOD BLOCK XFERR
17933 ;          8188  MLCS1 = write;                     !DO A WRITE FUNCTION
17934 ;          8189  TIME_OUT_LOOP;
17935 ;          8190  BGNS03;
17936 ;          8191  CLR_MBUS;
17937 ;          8192  ALL_BAD = ZEROES;
17938 ;          8193  SUM_BAD = ZEROES;
17939 ;          8194  NIB_TSTED = ZEROES;
17940 ;          8195  DAT_DM_XFER ();                    !SET UP SAME DATA DIAG XFERR
17941 ;          8196  MLCS1 = read;                      !DO A READ FUNCTION
17942 ;          8197  DELAY (ONE_US);
17943 ;          8198
17944 ;          8199  incr WD_CNT from 0 to 112 do       !READ 113 LONG WORDS
17945 ;          8200      begin
17946 ;          8201      PD_TEMP = .MLPD;               !GET PROM DATA
17947 ;          8202      DAT_CLK = ONE;                 !CLOCK OUT THE DATA WORD
17948 ;          8203      DELAY (ONE_US);
17949 ;          8204      RD_LNG_WRD;                    !READ THE DATA WORD
17950 ;          8205
17951 ;          8206      incr NIB_PTR from 0 to 8 do    !LOOK AT 9 NIBBLES
17952 ;          8207
17953 ;          8208          if .PD_TEMP [.NIB_PTR] eql ZERO    !FIND GOOD NIBBLE
17954 ;          8209          then
17955 ;          8210              begin
17956 ;          8211              NIB_TSTED = .NIB_TSTED + 1;    !INCREMENT COUNT OF NIBBLES TESTED
17957 ;          8212              XOR_LNG_WRD (.NIB_PTR, .BG_PAT, RESULT);    !XOR NIBBLE WITH BG PAT
17958 ;          8213
17959 ;          8214              if .RESULT<0, 4> eql ZERO      !SEE IF ALL BITS IN NIBBLE WERE BAD?
17960 ;          8215              then
17961 ;          8216                  ALL_BAD = .ALL_BAD + 1     !INCREMENT COUNT IF ALL BAD
17962 ;          8217              else
17963 ;          8218
17964 ;          8219                  if .RESULT<0, 4> neq %o'17' then SUM_BAD = .SUM_BAD + 1;
17965 ;          8220
17966 ;          8221                                             !SEE IF SOME BITS IN NIBBLE WERE BAD
17967 ;          8222              end;                           !INCREMENT COUNT IF SOME BAD
17968 ;          8223
17969 ;          8224      end;
17970 ;          8225
17971 ;          8226  ENDSUB;
```

```
 17973 ;ML4AD
 17974 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 17975                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI 4 (85)
 17976 ;          8227
 17977 ;          8228  if .NIB_TSTED eql .ALL_BAD                          !WHERE ALL NIBBLES XFERR'ED ALL BAD?
 17978 ;          8229  then                                                !ERROR IF ALL BAD
 17979 ;          8230      begin
 17980 ;          8231      ERRDF (84, SYNC, DUMPER);
 17981 ;          8232      PRINTB (SEV_FMT, WRD_22, PHR_4, WRD_9, WRD_12, WRD_23, FNC_5, WRD_19);
 17982 ;          8233      DODU (.ML_LON);
 17983 ;          8234      DOCLN;
 17984 ;          8235      end
 17985 ;          8236  else
 17986 ;          8237
 17987 ;          8238      if .SUM_BAD gtr ZERO                            !SEE IF SEE NIBBLE WERE BAD?
 17988 ;          8239      then
 17989 ;          8240          begin                                       !SOME BAD IS OK SO GIVE INTERMEDIATE ERROR
 17990 ;          8241          ERRDF (85, INTER, DUMPER);
 17991 ;          8242          PRINTB (SIX_FMT, PHR_4, WRD_9, WRD_12, WRD_23, FNC_5, WRD_19);
 17992 ;          8243          end;
 17993 ;          8244
 17994 ;          8245  ENDTST;
 17998
 17999                                        .SBTTL  $T33 TEST CODE SECTION
 18003 064306  004167  120300          $T33:  JSR     R1,$SAVE5                ;                                    8121
 18004 064312  162706  000010                 SUB     #10,SP
 18005 064316  152777  000040  131120         BISB    #40,@ML.REG+40           ;                                    8168
 18006 064324  016705  131502                 MOV     ML.DUT,R5
 18007 064330  042705  177770                 BIC     #177770,R5
 18008 064334  142777  000007  131102         BICB    #7,@ML.REG+40
 18009 064342  150577  131076                 BISB    R5,@ML.REG+40
 18010 064346  012766  177777  000002         MOV     #-1,2(SP)                ; *,BG.PAT                           8171
 18011 064354  012777  177777  131212         MOV     #-1,@ML.REG+170          ;                                    8172
 18012 064362  016677  000002  131214         MOV     2(SP),@ML.REG+200        ; BG.PAT,*                           8173
 18013 064370  016677  000002  131166         MOV     2(SP),@ML.REG+160        ; BG.PAT,*                           8174
 18014 064376  004767  132010                 JSR     PC,DAT.DM.XFER           ;                                    8175
 18015 064402  012777  000061  130774         MOV     #61,@ML.REG              ;                                    8176
 18016 064410  005000                         CLR     R0                       ; WD.CNT                             8178
 18017 064412  012701  000001          1$:    MOV     #1,R1                    ; *,$STMP2                           8180
 18018 064416  001411                  2$:    BEQ     5$
 18019 064420  016702  115472                 MOV     L$DLY,R2                 ; *,$STMP1
 18020 064424  001404                         BEQ     4$
 18021 064426  005066  000006          3$:    CLR     6(SP)                    ; $STMP
 18022 064432  005302                         DEC     R2                       ; $STMP1
 18023 064434  001374                         BNE     3$
 18024 064436  005301                  4$:    DEC     R1                       ; $STMP2
 18025 064440  000766                         BR      2$
 18026 064442  152777  000020  131054  5$:    BISB    #20,@ML.REG+120          ;                                    8181
 18027 064450  005200                         INC     R0                       ; WD.CNT                             8178
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;       TEST CODE SECTION                                       29-Mar-1982 16:21:03    PA:<
18029
18030
18031
18032 064452  020027  000177                    CMP     R0,#177                 ; WD.CNT,*
18033 064456  003755                            BLE     1$
18034 064460  152777  000040  130756            BISB    #40,@ML.REG+40          ;                                               8182
18035 064464  016705  131340                    MOV     ML.DUT,R5
18036 064472  042705  177770                    BIC     #177770,R5
18037 064476  142777  000007  130740            BICB    #7,@ML.REG+40
18038 064504  150577  130734                    BISB    R5,@ML.REG+40
18039 064510  005067  127306                    CLR     IO.BUF
18040 064514  152777  000010  130722            BISB    #10,@ML.REG+40          ;                                               8185
18041 064522  004767  133564                    JSR     PC,GD.BLK.XFER          ;                                               8186
18042 064526  012777  000061  130650            MOV     #61,@ML.REG             ;                                               8187
18043 064534  105777  130714            6$:     TSTB    @ML.REG+50              ;                                               8188
18044 064540  100375                            BPL     6$
18045 064542  104402                    7$:     TRAP    2
18046 064544  152777  000040  130672            BISB    #40,@ML.REG+40          ;                                               8189
18047 064552  016703  131254                    MOV     ML.DUT,R3               ;                                               8190
18048 064556  042703  177770                    BIC     #177770,R3
18049 064562  142777  000007  130654            BICB    #7,@ML.REG+40
18050 064570  150377  130650                    BISB    R3,@ML.REG+40
18051 064574  005004                            CLR     R4
18052 064576  005016                            CLR     (SP)                    ; ALL.BAD                                       8192
18053 064600  005005                            CLR     R5                      ; SUM.BAD                                       8193
18054 064602  004767  131604                    JSR     PC,DAT.DM.XFER          ; NIB.TSTED                                     8194
18055 064606  012777  000071  130570            MOV     #71,@ML.REG             ;                                               8195
18056 064614  012701  000001                    MOV     #1,R1                   ;                                               8196
18057 064620  001411                    8$:     BEQ     11$                     ; *,$STMP2                                      8197
18058 064622  016702  115270                    MOV     L$DLY,R2
18059 064626  001404                            BEQ     10$                     ; *,$STMP1
18060 064630  005066  000006            9$:     CLR     6(SP)                   ; $STMP
18061 064634  005302                            DEC     R2                      ; $STMP1
18062 064636  001374                            BNE     9$
18063 064640  005301                    10$:    DEC     R1                      ; $STMP2
18064 064642  000766                            BR      8$
18065 064644  005003                    11$:    CLR     R3                      ; WD.CNT                                        8199
18066 064646  017767  130762  130466    12$:    MOV     @ML.REG+230,PD.TEMP     ;                                               8201
18067 064654  152777  000020  130642            BISB    #20,@ML.REG+120         ;                                               8202
18068 064662  012701  000001                    MOV     #1,R1                   ; *,$STMP2                                      8203
18069 064666  001411                    13$:    BEQ     16$
18070 064670  016702  115222                    MOV     L$DLY,R2                ; *,$STMP1
18071 064674  001404                            BEQ     15$
18072 064676  005066  000006            14$:    CLR     6(SP)                   ; $STMP
18073 064702  005302                            DEC     R2                      ; $STMP1
18074 064704  001374                            BNE     14$
18075 064706  005301                    15$:    DEC     R1                      ; $STMP2
18076 064710  000766                            BR      13$
18077 064712  017767  130656  126454    16$:    MOV     @ML.REG+170,D1.TEMP
18078 064720  017767  130660  126450            MOV     @ML.REG+200,D2.TEMP
18079 064726  017767  130632  126444            MOV     @ML.REG+160,E2.TEMP
18080 064734  005001                            CLR     R1
18081 064736  010102                    17$:    MOV     R1,R2                   ; NIB.PTR                                       8206
18082 064740  006202                            ASR     R2                      ; NIB.PTR,*                                     8208
18083 064742  006202                            ASR     R2
```

SEQ 0382


```
18085
18086                                   ;ML4AD
18087                                   ;                TEST CODE SECTION
18088 064744  006202                            ASR     R2
18089 064746  062702  015342                    ADD     #PD.TEMP,R2
18090 064752  010246                            MOV     R2,-(SP)
18091 064754  010146                            MOV     R1,-(SP)
18092 064756  042716  177770                    BIC     #177770,(SP)          ; NIB.PTR,*
18093 064762  012746  000001                    MOV     #1,-(SP)
18094 064766  005046                            CLR     -(SP)
18095 064770  004767  116640                    JSR     PC,BL$GT2
18096 064774  062706  000010                    ADD     #10,SP
18097 065000  005700                            TST     R0
18098 065002  001027                            BNE     20$
18099 065004  005205                            INC     R5                    ; NIB.TSTED                     8211
18100 065006  010146                            MOV     R1,-(SP)              ; NIB.PTR,*                     8212
18101 065010  016646  000004                    MOV     4(SP),-(SP)           ; BG.PAT,*
18102 065014  012746  000012                    MOV     #12,-(SP)
18103 065020  060616                            ADD     SP,(SP)
18104 065022  004767  134044                    JSR     PC,XOR.LNG.WRD        ; RESULT,*
18105 065026  016602  000012                    MOV     12(SP),R2             ; RESULT,*                      8214
18106 065032  042702  177760                    BIC     #177760,R2
18107 065036  001002                            BNE     18$
18108 065040  005204                            INC     R4                    ; ALL.BAD                       8216
18109 065042  000405                            BR      19$                   ;                               8214
18110 065044  020227  000017            18$:    CMP     R2,#17                ;                               8219
18111 065050  001402                            BEQ     19$
18112 065052  005266  000006                    INC     6(SP)                 ; SUM.BAD
18113 065056  062706  000006            19$:    ADD     #6,SP                 ;                               8210
18114 065062  005201                    20$:    INC     R1                    ; NIB.PTR                       8206
18115 065064  020127  000010                    CMP     R1,#10                ; NIB.PTR,*
18116 065070  003722                            BLE     17$
18117 065072  005203                            INC     R3                    ; WD.CNT                        8199
18118 065074  020327  000160                    CMP     R3,#160               ; WD.CNT,*
18119 065100  003662                            BLE     12$
18120 065102  104467                            TRAP    67                    ;                               8224
18121 065104  006000                            ROR     R0
18122 065106  103615                            BLO     7$
18123 065110  020504                            CMP     R5,R4                 ; NIB.TSTED,ALL.BAD             8228
18124 065112  001037                            BNE     21$
18125 065114  104455                            TRAP    55                    ;                               8231
18126 065116  000124                            .WORD   124
18127 065120  012750                            .WORD   SYNC
18128 065122  026302                            .WORD   DUMPER
18129 065124  012746  010730                    MOV     #WRD.19,-(SP)         ;                               8232
18130 065130  012746  012216                    MOV     #FNC.5,-(SP)
18131 065134  012746  010766                    MOV     #WRD.23,-(SP)
18132 065140  012746  010650                    MOV     #WRD.12,-(SP)
18133 065144  012746  010616                    MOV     #WRD.9,-(SP)
18134 065150  012746  011676                    MOV     #PHR.4,-(SP)
18135 065154  012746  010752                    MOV     #WRD.22,-(SP)
18136 065160  012746  010342                    MOV     #SEV.FMT,-(SP)
18137 065164  012746  000010                    MOV     #10,-(SP)
18138 065170  010600                            MOV     SP,R0                 ; SP,*
18139 065172  104414                            TRAP    14
```

```
18141                                   ;ML4AD
18142                                   ;                       TEST CODE SECTION                                       29-Mar-1982 16:23:04    TOPS
18143                                                                                                                   29-Mar-1982 16:21:03    PA:<
18144 065174  016700  130630                      MOV     ML.LUN,R0                       ;                                                       8233
18145 065200  104451                              TRAP    51
18146 065202  104444                              TRAP    44
18147 065204  062706  000022                      ADD     #22,SP
18148 065210  000432                              BR      22$                             ;                                                       8230
18149 065212  005716                      21$:    TST     (SP)                            ;                                                       8228
18150 065214  003430                              BLE     22$                             ; SUM.BAD                                               8238
18151 065216  104455                              TRAP    55
18152 065220  000125                              .WORD   125                             ;                                                       8241
18153 065222  013114                              .WORD   INTER
18154 065224  026302                              .WORD   DUMPER
18155 065226  012746  010730                      MOV     #WRD.19,-(SP)                   ;                                                       8242
18156 065232  012746  012216                      MOV     #FNC.5,-(SP)
18157 065236  012746  010766                      MOV     #WRD.23,-(SP)
18158 065242  012746  010650                      MOV     #WRD.12,-(SP)
18159 065246  012746  010616                      MOV     #WRD.9,-(SP)
18160 065252  012746  011676                      MOV     #PHR.4,-(SP)
18161 065256  012746  010322                      MOV     #SIX.FMT,-(SP)
18162 065262  012746  000007                      MOV     #7,-(SP)
18163 065266  010600                              MOV     SP,R0
18164 065270  104414                              TRAP    14                              ; SP,*
18165 065272  062706  000020                      ADD     #20,SP
18166 065276  062706  000010              22$:    ADD     #10,SP                          ;                                                       8240
18167 065302  000207                              RTS     PC                              ;                                                       8121
18168
18169                                   ; Routine Size:  255 words
18170                                   ; Maximum stack depth per invocation:  19 words
18175
18176
18180
18181                                           .SBTTL  T33 TEST CODE SECTION
18185 065304                            T33::
18186 065304  004767  176776            1$:     JSR     PC,$T33                         ;                                                       8243
18187 065310  104466                              TRAP    66
18188 065312  006000                              ROR     R0
18189 065314  103773                              BLO     1$
18190 065316  000207                              RTS     PC
18191
18192                                   ; Routine Size:  6 words
18193                                   ; Maximum stack depth per invocation:  0 words
```

```
18202
18203
18204 ;          8246  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (86)

```
18206 ;ML4AD
18207 ;                 TEST CODE SECTION
18208
18209 ;          8247 !
18210 ;          8248 !
18211 ;          8249 BGNTST;
18212 ;          8250
18213 ;          8251 !++
18214 ;          8252 ! TEST NUMBER:  TST 34
18215 ;          8253 !
18216 ;          8254 ! TEST NAME:    SYNC DATA BUS CONTINUITY TEST (WRITE PATH)
18217 ;          8255 !
18218 ;          8256 ! TEST DESCRIPTION:
18219 ;          8257 !
18220 ;          8258 !       TEST THE CONTINUITY OF THE
18221 ;          8259 !       SYNCHRONOUS MODULE WRITE PATH
18222 ;          8260 !       DATA BUS BY:
18223 ;          8261 !
18224 ;          8262 !       1.  VIA DAT_DM MODE WRITE A
18225 ;          8263 !           BACKGROUND PATTERN OF ONES
18226 ;          8264 !           INTO THE GOOD BLOCK.
18227 ;          8265 !
18228 ;          8266 !       2.  VIA MBUS WRITE FUNCTION
18229 ;          8267 !           WRITE A ZEROES PATTERN INTO
18230 ;          8268 !           THE GOOD BLOCK.
18231 ;          8269 !
18232 ;          8270 !       3.  VIA DAT_DM MODE READ GOOD
18233 ;          8271 !           NIBBLES IN THE GOOD BLOCK FOR
18234 ;          8272 !           ZEROES.
18235 ;          8273 !
18236 ;          8274 !       4.  REPEAT WITH COMPLIMENT
18237 ;          8275 !           DATA AND BACKGROUND PATTERNS.
18238 ;          8276 !
18239 ;          8277 ! IMPLICIT INPUTS:
18240 ;          8278 !
18241 ;          8279 !       PD_TEMP
18242 ;          8280 !
18243 ;          8281 !       A BIT VECTOR OF 16 BITS WHERE
18244 ;          8282 !       THE READ PROM DATA IS STORED
18245 ;          8283 !       AND ACCESSED FROM.
18246 ;          8284 !
18247 ;          8285 !       IO_BUF
18248 ;          8286 !
18249 ;          8287 !       A VECTOR OF 256 WORDS WHERE
18250 ;          8288 !       DATA FOR MBUS READS AND WRITE
18251 ;          8289 !       FUNCTION ARE FOUND.
18252 ;          8290 !
18253 ;          8291 !
18254 ;          8292 !--
18255 ;          8293
18256 ;          8294 local
18257 ;          8295     DODU_FLG,                  !DROP UNIT FLAG
18258 ;          8296     BG_PAT,                    !BACKGROUND PATTERN
18259 ;          8297     RESULT,                    !RESULTS FROM XOR
18260 ;          8298     TST_PAT;                   !TEST PATTERN
```

SEQ 0386

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (86)

```
18262 ;ML4AD
18263 ;                 TEST CODE SECTION
18264
18265 ;          8299
18266 ;          8300  CLR_THRESHOLD;                          !CLEAR ERROR PRINT THRESHOLD
18267 ;          8301  DODO_FLG = ZERO;
18268 ;          8302  BG_PAT = ZEROES;                        !BG PAT EQL 0'S
18269 ;          8303  TST_PAT = ONES;                         !TST PAT EQL 1'S
18270 ;          8304
18271 ;          8305  incr TWICE from 0 to 1 do               !REPEAT LOOP TWICE
18272 ;          8306      begin
18273 ;          8307      BGNSUB;
18274 ;          8308      CLR_MBUS;
18275 ;          8309      MLDT = .BG_PAT;                     !LOAD DATA DIAG REG WITH BG PAT
18276 ;          8310      MLD2 = .BG_PAT;
18277 ;          8311      MLE2 = .BG_PAT;
18278 ;          8312      DAT_DM_XFER ();                     !SET UP A DATA DIAG XFERR AT THE GOOD BLK
18279 ;          8313      MLCS1 = write;                      !DO A WRITE FUNCTION
18280 ;          8314
18281 ;          8315      incr WD_CNT from 0 to 127 do        !LOAD BLOCK WITH BG PAT
18282 ;          8316          begin
18283 ;          8317          DELAY (ONE_US);
18284 ;          8318          DAT_CLK = ONE;
18285 ;          8319          end;
18286 ;          8320
18287 ;          8321      CLR_MBUS;
18288 ;          8322      BAI = ONE;                          !SET ON FIRST IO_BUF ADRS
18289 ;          8323      IO_BUF = .TST_PAT;                  !FIRST IO_BUF WORD EQL'S TST_PAT
18290 ;          8324      GD_BLK_XFER ();                     !SET UP A GOOD BLOCK XFERR
18291 ;          8325      MLCS1 = write;                      !DO A WRITE FUNCTION
18292 ;          8326      TIME_OUT_LOOP;
18293 ;          8327      CLR_MBUS;
18294 ;          8328      DAT_DM_XFER ();                     !SET UP A DATA DIAG XFERR AT SAME BLOCK
18295 ;          8329      MLCS1 = read;                       !DO A READ FUNCTION
18296 ;          8330      DELAY (ONE_US);
18297 ;          8331
18298 ;          8332      incr WD_CNT from 0 to 112 do        !READ 113 LONG WORDS
18299 ;          8333          begin
18300 ;          8334          PD_TEMP = .MLPD;                !GET THE PROM DATA
18301 ;          8335          DAT_CLK = ONE;                  !CLOCK OUT THE DATA WORD
18302 ;          8336          DELAY (ONE_US);
18303 ;          8337          RD_LNG_WRD;                     !READ THE DATA WORD
18304 ;          8338
18305 ;          8339          incr NIB_PTR from 0 to 8 do     !LOOK AT 9 NIBBLES
18306 ;          8340
18307 ;          8341              if .PD_TEMP [.NIB_PTR] IS_NOT_SET   !FIND GOOD NIBBLES
18308 ;          8342              then
18309 ;          8343                  begin
18310 ;          8344                  XOR_LNG_WRD (.NIB_PTR, .TST_PAT, RESULT);       !XOR NIBBLE DATA WITH TST_PAT
18311 ;          8345
18312 ;          8346                  if .RESULT<0, 4> neq ZERO       !SEE IF EQUAL
18313 ;          8347                  then
18314 ;          8348                      begin                       !ERROR IF NEQ
18315 ;          8349                      CMP_THRESHOLD;              !COMPARE ERROR PRINT THRESHOLD
18316 ;          8350                      ERRDF (86, SYNC, DUMPER);
```

```
 18318 ;ML4AD
 18319 ;                 TEST CODE SECTION                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 18320                                                                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (86)
 18321 ;          8351                        PRINTB (FIV_FMT, WRD_24, WRD_25, WRD_10, WRD_12, FNC_5);
 18322 ;          8352                        PRINTB (FMT_5, .TST_PAT, .RESULT, .NIB_PTR);
 18323 ;          8353                        DODU_FLG = ONE;
 18324 ;          8354                        end
 18325 ;          8355
 18326 ;          8356                    end
 18327 ;          8357
 18328 ;          8358                end;
 18329 ;          8359
 18330 ;          8360            ENDSUB;
 18331 ;          8361
 18332 ;          8362            if .DODU_FLG IS_SET                          !DROP THIS UNIT IF DODU IS_SET
 18333 ;          8363                                                         !AND EXITS TEST
 18334 ;          8364            then
 18335 ;          8365                begin
 18336 ;          8366                DODU (.ML_LUN);
 18337 ;          8367                DOCLN;
 18338 ;          8368                end;
 18339 ;          8369
 18340 ;          8370            BG_PAT = not .BG_PAT;                        !COMPLIMENT BG_PAT
 18341 ;          8371            TST_PAT = not .TST_PAT;                      !COMPLIMENT TST_PAT AND REPEAT
 18342 ;          8372            end;
 18343 ;          8373
 18344 ;          8374    ENDTST;
 18348
 18349                                    .SBTTL  $T34 TEST CODE SECTION
 18353 065320  004167  117266      $T34: JSR     R1,$SAVE5                    ;                                    8245
 18354 065324  024646                    CMP     -(SP),-(SP)
 18355 065326  005067  130044            CLR     P.CNT                        ;
 18356 065332  005046                    CLR     -(SP)                        ; DODU.FLG                           8298
 18357 065334  005001                    CLR     R1                           ; BG.PAT                             8301
 18358 065336  012702  177777            MOV     #-1,R2                       ; *,TST.PAT                          8302
 18359 065342  005046                    CLR     -(SP)                        ; TWICE                              8303
 18360 065344  104402              1$:   TRAP    2                            ;                                    8305
 18361 065346  152777  000040  130070    BISB    #40,@ML.REG+40               ;                                    8306
 18362 065354  016705  130452            MOV     ML.DUT,R5                    ;                                    8307
 18363 065360  042705  177770            BIC     #177770,R5
 18364 065364  142777  000007  130052    BICB    #7,@ML.REG+40
 18365 065372  150577  130046            BISB    R5,@ML.REG+40
 18366 065376  010177  130172            MOV     R1,@ML.REG+170               ; BG.PAT,*                           8309
 18367 065402  010177  130176            MOV     R1,@ML.REG+200               ; BG.PAT,*                           8310
 18368 065406  010177  130152            MOV     R1,@ML.REG+160               ; BG.PAT,*                           8311
 18369 065412  004767  130774            JSR     PC,DAT.DM.XFER               ;                                    8312
 18370 065416  012777  000061  127760    MOV     #61,@ML.REG                  ;                                    8313
 18371 065424  005003                    CLR     R3                           ; WD.CNT                             8315
 18372 065426  012704  000001      2$:   MOV     #1,R4                        ; *,$STMP2                           8317
```

```
18374                                   ;ML4AD
18375                                   ;                       TEST CODE SECTION
18376
18377 065432 001411                3$:  BEQ     6$
18378 065434 016705 114456              MOV     L$DLY,R5        ; * $$TMP1
18379 065440 001404                     BEQ     5$
18380 065442 005066 000006         4$:  CLR     6(SP)           ; $$TMP
18381 065446 005305                     DEC     R5              ; $$TMP1
18382 065450 001374                     BNE     4$
18383 065452 005304                5$:  DEC     R4              ; $$TMP2
18384 065454 000766                     BR      3$
18385 065456 152777 000020 130040  6$:  BISB    #20,@ML.REG+120 ;                                       8318
18386 065464 005203                     INC     R3              ; WD.CNT                                8315
18387 065466 020327 000177              CMP     R3,#177         ; WD.CNT,*
18388 065472 003755                     BLE     2$
18389 065474 152777 000040 127742       BISB    #40,@ML.REG+40  ;                                       8319
18390 065502 016705 130324              MOV     ML.DUT,R5
18391 065506 042705 177770              BIC     #177770,R5
18392 065512 142777 000007 127724       BICB    #7,@ML.REG+40
18393 065520 150577 127720              BISB    R5,@ML.REG+40
18394 065524 152777 000010 127712       BISB    #10,@ML.REG+40  ;                                       8322
18395 065532 010267 126264              MOV     R2,IO.BUF       ; TST.PAT,*                             8323
18396 065536 004767 132550              JSR     PC,GD.BLK.XFER  ;                                       8324
18397 065542 012777 000061 127634       MOV     #61,@ML.REG     ;                                       8325
18398 065550 105777 127700         7$:  TSTB    @ML.REG+50
18399 065554 100375                     BPL     7$
18400 065556 152777 000040 127660       BISB    #40,@ML.REG+40  ;                                       8326
18401 065564 016705 130242              MOV     ML.DUT,R5
18402 065570 042705 177770              BIC     #177770,R5
18403 065574 142777 000007 127642       BICB    #7,@ML.REG+40
18404 065602 150577 127636              BISB    R5,@ML.REG+40
18405 065606 004767 130600              JSR     PC,DAT.DM.XFER
18406 065612 012777 000071 127564       MOV     #71,@ML.REG     ;                                       8328
18407 065620 012704 000001              MOV     #1,R4           ;                                       8329
18408 065624 001411                8$:  BEQ     11$             ; *,$$TMP2                              8330
18409 065626 016705 114264              MOV     L$DLY,R5        ; *,$$TMP1
18410 065632 001404                     BEQ     10$
18411 065634 005066 000006         9$:  CLR     6(SP)           ; $$TMP
18412 065640 005305                     DEC     R5              ; $$TMP1
18413 065642 001374                     BNE     9$
18414 065644 005304                10$: DEC     R4              ; $$TMP2
18415 065646 000766                     BR      8$
18416 065650 005003                11$: CLR     R3              ; WD.CNT                                8332
18417 065652 017767 127756 127462  12$: MOV     @ML.REG+230,PD.TEMP ;                                   8334
18418 065660 152777 000020 127636       BISB    #20,@ML.REG+120 ;                                       8335
18419 065666 012704 000001              MOV     #1,R4           ; *,$$TMP2                              8336
18420 065672 001411                13$: BEQ     16$
18421 065674 016705 114216              MOV     L$DLY,R5        ; *,$$TMP1
18422 065700 001404                     BEQ     15$
18423 065702 005066 000006         14$: CLR     6(SP)           ; $$TMP
18424 065706 005305                     DEC     R5              ; $$TMP1
18425 065710 001374                     BNE     14$
18426 065712 005304                15$: DEC     R4              ; $$TMP2
18427 065714 000766                     BR      13$
18428 065716 017767 127652 125450  16$: MOV     @ML.REG+170,D1.TEMP
```

```
18430                                          ;ML4AD
18431                                          ;                TEST CODE SECTION
18432
18433 065724  017767  127654  125444           MOV     @ML.REG+200,D2.TEMP
18434 065732  017767  127626  125440           MOV     @ML.REG+160,E2.TEMP
18435 065740  005004                           CLR     R4
18436 065742  010405                    17$:   MOV     R4,R5                   ; NIB.PTR                       8339
18437 065744  006205                           ASR     R5                      ; NIB.PTR,*                     8341
18438 065746  006205                           ASR     R5
18439 065750  006205                           ASR     R5
18440 065752  062705  015342                   ADD     #PD.TEMP,R5
18441 065756  010546                           MOV     R5,-(SP)
18442 065760  010446                           MOV     R4,-(SP)                ; NIB.PTR,*
18443 065762  042716  177770                   BIC     #177770,(SP)
18444 065766  012746  000001                   MOV     #1,-(SP)
18445 065772  005046                           CLR     -(SP)
18446 065774  004767  115634                   JSR     PC,BL$GT2
18447 066000  062706  000010                   ADD     #10,SP
18448 066004  005700                           TST     R0
18449 066006  001071                           BNE     20$
18450 066010  010446                           MOV     R4,-(SP)                ; NIB.PTR,*
18451 066012  010246                           MOV     R2,-(SP)                ; TST.PAT,*                     8344
18452 066014  012746  000012                   MOV     #12,-(SP)
18453 066020  060616                           ADD     SP,(SP)                 ; RESULT,*
18454 066022  004767  133044                   JSR     PC,XOR.LNG.WRD
18455 066026  032766  000017  000012           BIT     #17,12(SP)              ; *,RESULT                      8346
18456 066034  001454                           BEQ     19$
18457 066036  005267  127334                   INC     P.CNT                   ;                               8348
18458 066042  026767  127330  127330           CMP     P.CNT,LIMIT
18459 066050  003403                           BLE     18$
18460 066052  062706  000006                   ADD     #6,SP
18461 066056  000451                           BR      21$
18462 066060  104455                    18$:   TRAP    55                      ;                               8350
18463 066062  000126                           .WORD   126
18464 066064  012750                           .WORD   SYNC
18465 066066  026302                           .WORD   DUMPER
18466 066070  012746  012216                   MOV     #FNC.5,-(SP)            ;                               8351
18467 066074  012746  010650                   MOV     #WRD.12,-(SP)
18468 066100  012746  010630                   MOV     #WRD.10,-(SP)
18469 066104  012746  011002                   MOV     #WRD.25,-(SP)
18470 066110  012746  010774                   MOV     #WRD.24,-(SP)
18471 066114  012746  010304                   MOV     #FIV.FMT,-(SP)
18472 066120  012746  000006                   MOV     #6,-(SP)
18473 066124  010600                           MOV     SP,R0
18474 066126  104414                           TRAP    14                      ; SP,*
18475 066130  010416                           MOV     R4,(SP)                 ; NIB.PTR,*
18476 066132  016646  000030                   MOV     30(SP),-(SP)            ; RESULT,*                      8352
18477 066136  010246                           MOV     R2,-(SP)                ; TST.PAT,*
18478 066140  012746  006650                   MOV     #FMT.5,-(SP)
18479 066144  012746  000004                   MOV     #4,-(SP)
18480 066150  010600                           MOV     SP,R0
18481 066152  104414                           TRAP    14                      ; SP,*
18482 066154  012766  000001  000036           MOV     #1,36(SP)               ; *,DODU.FLG                    8353
18483 066162  062706  000026                   ADD     #26,SP                  ;                               8348
18484 066166  062706  000006            19$:   ADD     #6,SP                   ;                               8343
```

```
                                        ;ML4AD
                                        ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS
                                                                                                            29-Mar-1982 16:21:03    PA:<
066172  005204                  20$:     INC     R4                                ; NIB.PTR                                      8339
066174  020427  000010                   CMP     R4,#10                            ; NIB.PTR,*
066200  003660                           BLE     17$
066202  005203                  21$:     INC     R3                                ; WD.CNT                                       8332
066204  020327  000160                   CMP     R3,#160                           ; WD.CNT,*
066210  003620                           BLE     12$
066212  104467                           TRAP    67                                ;                                              8358
066214  006000                           ROR     R0
066216  103002                           BHIS    23$
066220  000167  177120          22$:     JMP     1$
066224  026627  000002  000001  23$:     CMP     2(SP),#1                          ; DODU.FLG,*                                   8362
066232  001004                           BNE     24$
066234  016700  127570                   MOV     ML.LUN,R0                         ;                                              8366
066240  104451                           TRAP    51
066242  104444                           TRAP    44
066244  005101                  24$:     COM     R1                                ; BG.PAT                                       8370
066246  005102                           COM     R2                                ; TST.PAT                                      8371
066250  005216                           INC     (SP)                              ; TWICE                                        8305
066252  021627  000001                   CMP     (SP),#1                           ; TWICE,*
066256  003760                           BLE     22$
066260  062706  000010                   ADD     #10,SP                            ;                                              8245
066264  000207                           RTS     PC

                                        ; Routine Size:  243 words
                                        ; Maximum stack depth per invocation:  24 words


                                                .SBTTL  T34 TEST CODE SECTION
066266                          T34::
066266  004767  177026          1$:      JSR     PC,$T34                           ;                                              8372
066272  104466                           TRAP    66
066274  006000                           ROR     R0
066276  103773                           BLO     1$
066300  000207                           RTS     PC

                                        ; Routine Size:  6 words
                                        ; Maximum stack depth per invocation:  0 words
```

```
18542 ;ML4AD
18543 ;                 TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
18544                                                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (86)
18545 ;         8375    !<BLF/PAGE>
```

SEQ 0392

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>M_4AD.BLI.4 '87)

```
18547 ;ML4AD
18548 ;                 TEST CODE SECTION
18549 ;
18550 ;          8376  !
18551 ;          8377  BGNTST;
18552 ;          8378
18553 ;          8379  !++
18554 ;          8380  ! TEST NUMBER:  TST 35
18555 ;          8381  !
18556 ;          8382  ! TEST NAME:    SYNC DATA BUS CONIINUITY /READ PATH
18557 ;          8383  !
18558 ;          8384  ! TEST DESCRIPTION:
18559 ;          8385  !       TEST THE CONITNUITY OF THE SYNCHRONOUS MODULE READ
18560 ;          8386  !       DATA BUS BY:
18561 ;          8387  !
18562 ;          8388  !       1.  VIA MBUS WRITE FUNCTION WRITE ONES INTO THE GOOD BLOCK.
18563 ;          8389  !
18564 ;          8390  !       2.  VIA MBUS READ FUNCTION READ THE GOOD BLOCK FOR ONES.
18565 ;          8391  !
18566 ;          8392  !       3.  REPEAT WITH COMPLIMENT DATA PATTERN.
18567 ;          8393  !
18568 ;          8394  ! IMPLICIT INPUTS:
18569 ;          8395  !       IO_BUF
18570 ;          8396  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE
18571 ;          8397  !       FUNCTION ARE FOUND.
18572 ;          8398  !--
18573 ;          8399
18574 ;          8400  local
18575 ;          8401      DODU_FLG,                       !DROP UNIT FLAG
18576 ;          8402      TST_PAT,                        !TEST PATTERN
18577 ;          8403      BG_PAT;                         !BACKGROUND PATTERN
18578 ;          8404
18579 ;          8405  CLR_THRESHOLD;                      !CLEAR ERROR PRINT THRESHOLD
18580 ;          8406  DODU_FLG = ZERO;
18581 ;          8407  TST_PAT = ONES;
18582 ;          8408  BG_PAT = ZEROES;
18583 ;          8409
18584 ;          8410  incr TWICE from 0 to 1 do           !REPEAT LOOP TWICE
18585 ;          8411      begin
18586 ;          8412      CLR_MBUS;
18587 ;          8413      BAI = ONE;                      !SET ON FIRST IO_BUF ADRS
18588 ;          8414      IO_BUF = .TST_PAT;              !FIRST IO_BUF ADRS GET TST PAT
18589 ;          8415      GD_BLK_XFER ();                 !SET UP A GOOD BLOCK XFERR
18590 ;          8416      MLCS1 = write;                  !DO A WRITE FUNCTION (WRITES THE TST_PAT)
18591 ;          8417      TIME_OUT_LOOP;
18592 ;          8418      BGNSUB;
18593 ;          8419      CLR_MBUS;
18594 ;          8420
18595 ;          8421      incr IO_CNT from 0 to 255 do    !LOAD IO_BUF WITH BG PAT
18596 ;          8422          IO_BUF [.IO_CNT] = .BG_PAT;
18597 ;          8423
18598 ;          8424      GD_BLK_XFER ();                 !SET UP A GOOD BLOCK XFERR
18599 ;          8425      MLCS1 = read;                   !DO A READ FUNCTION (READ THE TST PAT)
18600 ;          8426      TIME_OUT_LOOP;
18601 ;          8427
```

```
 18603 ;ML4AD
 18604 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 18605                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (87)
 18606 ;          8428       incr IO_CNT from 0 to 255 do                    !READ THE IO_BUF FOR TEST PATTERN
 18607 ;          8429
 18608 ;          8430           if .IO_BUF [.IO_CNT] neq .TST_PAT           !SEE IF IO_BUF WORD EQLS TST PAT
 18609 ;          8431           then
 18610 ;          8432               begin                                   !ERROR AND SET DODU_FLG IF NEQ
 18611 ;          8433               CMP_THRESHOLD;                          !COMPARE ERROR PRINT THRESHOLD
 18612 ;          8434               ERRDF (87, SYNC, DUMPER);
 18613 ;          8435               PRINTB (SEV_FMT, WRD_24, WRD_25, WRD_10, WRD_12, WRD_23, FNC_6, WRD_19);
 18614 ;          8436               PRINTB (FMT_2, .TST_PAT, .IO_BUF [.IO_CNT], ~(.TST_PAT xor .IO_BUF [.IO_CNT]));
 18615 ;          8437               DODU_FLG = ONE;
 18616 ;          8438               end;
 18617 ;          8439
 18618 ;          8440       ENDSUB;
 18619 ;          8441
 18620 ;          8442       if .DODU_FLG IS_SET                             !DROP THIS UNIT IF DODU_FLG IS SET
 18621 ;          8443       then
 18622 ;          8444           begin
 18623 ;          8445           DODU (.ML_LUN);
 18624 ;          8446           DOCLN;
 18625 ;          8447           end;
 18626 ;          8448
 18627 ;          8449       TST_PAT = not .TST_PAT;                         !COMPLIMENT TST_PAT
 18628 ;          8450       BG_PAT = not .BG_PAT;                           !COMPLIMENT BG_PAT AND REPEAT
 18629 ;          8451       end;
 18630 ;          8452
 18631 ;          8453   ENDTST;
 18635
 18636                                  .SBTTL  $T35 TEST CODE SECTION
 18640 066302 004167 116304             $T35:  JSR     R1,$SAVE5
 18641 066306 005067 127064                    CLR     P.CNT                   ;                                       8374
 18642 066312 005046                           CLR     -(SP)                   ;                                       8403
 18643 066314 012701 177777                    MOV     #-1,R1                  ; DODU.FLG                              8406
 18644 066320 005004                           CLR     R4                      ; *,TST.PAT                             8407
 18645 066322 005005                           CLR     R5                      ; BG.PAT                                8408
 18646 066324 152777 000040 127112      1$:    BISB    #40,@ML.REG+40          ; TWICE                                 8410
 18647 066332 016703 127474                    MOV     ML.DUT,R3               ;                                       8411
 18648 066336 042703 177770                    BIC     #177770,R3
 18649 066342 142777 000007 127074             BICB    #7,@ML.REG+40
 18650 066350 150377 127070                    BISB    R3,@ML.REG+40
 18651 066354 152777 000010 127062             BISB    #10,@ML.REG+40
 18652 066362 010167 125434                    MOV     R1,IO.BUF               ;                                       8413
 18653 066366 004767 131720                    JSR     PC,GD.BLK.XFER          ; TST.PAT,*                             8414
 18654 066372 012777 000061 127004             MOV     #61,@ML.REG             ;                                       8415
 18655 066400 105777 127050             2$:    TSTB    @ML.REG+50              ;                                       8416
 18656 066404 100375                           BPL     2$
 18657 066406 104402                    3$:    TRAP    2                       ;                                       8417
```

```
18659                                   ;ML4AD
18660                                   ;                       TEST CODE SECTION
18661
18662 066410 152777 000040 127026       BISB    #40,@ML.RE+0                    ;                       8418
18663 066416 016703 127410              MOV     ML.DUT,R3
18664 066422 042703 177770              BIC     #177770,R3
18665 066426 142777 000007 127010       BICB    #7,@ML.REG+40
18666 066434 150377 127004              BISB    R3,@ML.REG+40
18667 066440 005002                     CLR     R2                              ; IO.CNT                8421
18668 066442 010203             4$:     MOV     R2,R3                           ; IO.CNT,*              8422
18669 066444 006303                     ASL     R3
18670 066446 010463 014022              MOV     R4,IO.BUF(R3)                   ; BG.PAT,*
18671 066452 005202                     INC     R2                              ; IO.CNT                8421
18672 066454 020227 000377              CMP     R2,#377                         ; IO.CNT,*
18673 066460 003770                     BLE     4$
18674 066462 004767 131624              JSR     PC,GD.BLK.XFER                  ;                       8424
18675 066466 012777 000071 126710       MOV     #71,@ML.REG                     ;                       8425
18676 066474 105777 126754      5$:     TSTB    @ML.REG+50
18677 066500 100375                     BPL     5$
18678 066502 005002                     CLR     R2                              ; IO.CNT                8428
18679 066504 010203             6$:     MOV     R2,R3                           ; IO.CNT,*              8430
18680 066506 006303                     ASL     R3
18681 066510 062703 014022              ADD     #IO.BUF,R3
18682 066514 021301                     CMP     (R3),R1                         ; *,TST.PAT
18683 066516 001462                     BEQ     7$
18684 066520 005267 126652              INC     P.CNT                           ;                       8432
18685 066524 026767 126646 126646       CMP     P.CNT,LIMIT
18686 066532 003060                     BGT     8$
18687 066534 104455                     TRAP    55
18688 066536 000127                     .WORD   127                             ;                       8434
18689 066540 012750                     .WORD   SYNC
18690 066542 026302                     .WORD   DUMPER
18691 066544 012746 010730              MOV     #WRD.19,-(SP)                   ;                       8435
18692 066550 012746 012226              MOV     #FNC.6,-(SP)
18693 066554 012746 010766              MOV     #WRD.23,-(SP)
18694 066560 012746 010650              MOV     #WRD.12,-(SP)
18695 066564 012746 010630              MOV     #WRD.10,-(SP)
18696 066570 012746 011002              MOV     #WRD.25,-(SP)
18697 066574 012746 010774              MOV     #WRD.24,-(SP)
18698 066600 012746 010342              MOV     #SEV.FMT,-(SP)
18699 066604 012746 000010              MOV     #10,-(SP)
18700 066610 010600                     MOV     SP,R0                           ; SP,*
18701 066612 104414                     TRAP    14
18702 066614 011316                     MOV     (R3),(SP)                       ;                       8436
18703 066616 010146                     MOV     R1,-(SP)                        ; TST.PAT,*
18704 066620 046616 000002              BIC     2(SP),(SP)
18705 066624 040166 000002              BIC     R1,2(SP)                        ; TST.PAT,*
18706 066630 052616                     BIS     (SP)+,(SP)
18707 066632 011346                     MOV     (R3),-(SP)
18708 066634 010146                     MOV     R1,-(SP)                        ; TST.PAT,*
18709 066636 012746 006506              MOV     #FMT.2,-(SP)
18710 066642 012746 000004              MOV     #4,-(SP)
18711 066646 010600                     MOV     SP,R0                           ; SP,*
18712 066650 104414                     TRAP    14
18713 066652 012766 000001 000032       MOV     #1,32(SP)                       ; *,DODU.FLG            8437
```

SEQ 0395

```
18715
18716                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
18717                                   ;                TEST CODE SECTION                      29-Mar-1982 16:21:03    PA:<
18718 066660  062706  000032                      ADD     #32,SP                  ;                                       8432
18719 066664  005202              7$:   INC     R2                      ; IO.CNT                                        8428
18720 066666  020227  000377            CMP     R2,#377                 ; IO.CNT,*
18721 066672  003704                    BLE     6$
18722 066674  104467              8$:   TRAP    67                      ;                                               8438
18723 066676  006000                    ROR     R0
18724 066700  103642                    BLO     3$
18725 066702  021627  000001            CMP     (SP),#1                 ; DODU.FLG,*                                    8442
18726 066706  001004                    BNE     9$
18727 066710  016700  127114            MOV     ML.LUN,R0               ;                                               8445
18728 066714  104451                    TRAP    51
18729 066716  104444                    TRAP    44
18730 066720  005101              9$:   COM     R1                      ; TST.PAT                                       8449
18731 066722  005104                    COM     R4                      ; BG.PAT                                        8450
18732 066724  005205                    INC     R5                      ; TWICE                                         8410
18733 066726  020527  000001            CMP     R5,#1                   ; TWICE,*
18734 066732  003002                    BGT     10$
18735 066734  000167  177364            JMP     1$
18736 066740  005726              10$:  TST     (SP)+                   ;                                               8374
18737 066742  000207                    RTS     PC
18738
18739                                   ; Routine Size:  145 words
18740                                   ; Maximum stack depth per invocation:  20 words
18745
18746
18750
18751                                           .SBTTL  T35 TEST CODE SECTION
18755 066744                            T35::
18756 066744  004767  177332            1$:   JSR     PC,$T35                 ;                                       8451
18757 066750  104466                    TRAP    66
18758 066752  006000                    ROR     R0
18759 066754  103773                    BLO     1$
18760 066756  000207                    RTS     PC
18761
18762                                   ; Routine Size:  6 words
18763                                   ; Maximum stack depth per invocation:  0 words
18768
18769 ;          8454  !<BLF/PAGE>
```

SEQ 0396

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (88)

```
18771 ;ML4AD
18772 ;                 TEST CODE SECTION
18773
18774 ;          8455 !
18775 ;          8456 !
18776 ;          8457 BGNTST:
18777 ;          8458
18778 ;          8459 !++
18779 ;          8460 ! TEST NUMBER:  TST 36
18780 ;          8461 !
18781 ;          8462 ! TEST NA'__:    RAM BUS ADRS COUNTER TEST /WRITE PATH
18782 ;          8463 !
18783 ;          8464 ! TEST DESCRIPTION:
18784 ;          8465 !
18785 ;          8466 !       TEST ABILITY OF THE RAM_BUS ADDRESS
18786 ;          8467 !       COUNTERS TO LOAD/UNLOAD THE SKIP
18787 ;          8468 !       RAM DURING WRITE FUNCTIONS BY:
18788 ;          8469 !
18789 ;          8470 !       1.  LOADING A REPEATING COUNT OF 0
18790 ;          8471 !           TO 63 INTO THE NIBBLES OF THE
18791 ;          8472 !           FIRST 64 WORDS OF THE IO_BUF.
18792 ;          8473 !
18793 ;          8474 !       2.  VIA MBUS WRITE FUNCTION WRITE
18794 ;          8475 !           THE CONTENTS OF THE IO_BUF
18795 ;          8476 !           INTO THE GOOD BLOCK.
18796 ;          8477 !
18797 ;          8478 !       3.  VIA DAT_DM READ GOOD NIBBLES IN
18798 ;          8479 !           THE GOOD BLOCK FOR THE UNBROKEN
18799 ;          8480 !           COUNT OF 0 TO 63.
18800 ;          8481 !
18801 ;          8482 !           ONCE A BAD NIBBLE IS ENCOUNTERED
18802 ;          8483 !           MASK THAT NIBBLE FROM FURTHER
18803 ;          8484 !           READS.
18804 ;          8485 !
18805 ;          8486 !       4.  REPEAT READING NIBBLES UNTIL
18806 ;          8487 !           113 WORDS ARE READ OR ALL 10 NIBBLES
18807 ;          8488 !           ARE MASKED.
18808 ;          8489 !
18809 ;          8490 ! IMPLICIT INPUTS:
18810 ;          8491 !
18811 ;          8492 !       PD_TEMP
18812 ;          8493 !
18813 ;          8494 !       A BIT VECTOR OF 16 BITS WHERE
18814 ;          8495 !       THE READ PROM DATA IS STORED
18815 ;          8496 !       AND ACCESSED FROM.
18816 ;          8497 !
18817 ;          8498 !       IO_BUF
18818 ;          8499 !       A VECTOR OF 256 WORDS WHERE
18819 ;          8500 !       DATA FOR MBUS READS AND WRITE
18820 ;          8501 !       FUNCTION ARE FOUND.
18821 ;          8502 !
18822 ;          8503 !--
18823 ;          8504
18824 ;          8505 local
18825 ;          8506     DODU_FLG,                                  !DROP UNIT FLAG
```

```
18827 ;ML4AD
18828 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
18829                                                                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (88)
18830 ;          8507       WRD_CNT,                                        !WORD COUNT
18831 ;          8508       NIB_IGNORE : bitvector [16],                    !NIBBLE IGNORE FLAGS
18832 ;          8509       ERR_FLG,                                        !ERROR FLAG
18833 ;          8510       BAD_NIB_CNT,                                    !BAD NIBBLE COUNT
18834 ;          8511       PASS_CNT,                                       !PASS COUNT
18835 ;          8512       NIB_PAT;                                        !NIBBLE PATTERN
18836 ;          8513
18837 ;          8514   CLR_THRESHOLD;                                      !CLEAR ERROR PRINT THRESHOLD
18838 ;          8515   DODO_FLG = ZERO;
18839 ;          8516   WRD_CNT = ZERO;
18840 ;          8517
18841 ;          8518   incr CNT from 0 to 63 do                            !LOAD 64 WORDS WITH REPEATING COUNTS OF 0-63
18842 ;          8519
18843 ;          8520       incr PAT_CNT from -1 to 11 by 4 do              !LOAD NIBBLES IN WORD WITH REPEATING COUNTS OF 0-63
18844 ;          8521           begin
18845 ;          8522           (IO_BUF [.WRD_CNT])<0, 4> = .PAT_CNT + 1;       !LOAD FIRST NIBBLE IN WORD
18846 ;          8523           (IO_BUF [.WRD_CNT])<4, 4> = .PAT_CNT + 2;       !LOAD SECOND NIBBLE IN WORD
18847 ;          8524           (IO_BUF [.WRD_CNT])<8, 4> = .PAT_CNT + 3;       !LOAD THIRD NIBBLE IN WORD
18848 ;          8525           (IO_BUF [.WRD_CNT])<12, 4> = .PAT_CNT + 4;      !LOAD FORTH NIBBLE IN WORD
18849 ;          8526           WRD_CNT = .WRD_CNT + 1;                         !INCREMENT TO NEXT WORD
18850 ;          8527           end;
18851 ;          8528
18852 ;          8529   BGNSUB;
18853 ;          8530   CLR_MBUS;
18854 ;          8531   GD_BLK_XFER ();                                     !SET UP A GOOD BLOCK XFERR
18855 ;          8532   MLCS1 = write;                                      !DO A WRITE FUNCTION
18856 ;          8533   TIME_OUT_LOOP;
18857 ;          8534   CLR_MBUS;
18858 ;          8535   NIB_IGNORE = ZEROES;
18859 ;          8536   PASS_CNT = ZEROES;
18860 ;          8537   NIB_PAT = ZEROES;
18861 ;          8538   BAD_NIB_CNT = ZEROES;
18862 ;          8539   DAT_DM_XFER ();                                     !SET UP A DATA DIAG MODE AT THE GOOD BLOCK
18863 ;          8540   MLCS1 = read;                                       !DO A READ FUNCTION
18864 ;          8541   DELAY (ONE_US);
18865 ;          8542
18866 ;          8543   do                                                  !LOOP UNTIL THE BLOCK IS READ OR 9 BAD NIBBLES FOUND
18867 ;          8544       begin
18868 ;          8545       PD_TEMP = .MLPD;                                !GET THE PROM DATA
18869 ;          8546       DAT_CLK = ONE;                                  !CLOCK OUT THE DATA WORD
18870 ;          8547       DELAY (ONE_US);
18871 ;          8548       RD_LNG_WRD;                                     !READ THE DATA DIAG REGISTERS
18872 ;          8549
18873 ;          8550       incr NIB_PTR from 0 to 8 do                     !LOOK AT 9 NIBBLES
18874 ;          8551           begin
18875 ;          8552
18876 ;          8553           if .PD_TEMP [.NIB_PTR] IS_NOT_SET           !FIND GOOD NIBBLES
18877 ;          8554           then
18878 ;          8555               begin
18879 ;          8556
18880 ;          8557               if .NIB_IGNORE [.NIB_PTR] IS_NOT_SET        !SEE IF THIS NIBBLE FOUND BAD BEFORE
18881 ;          8558               then
```

```
18883 ;ML4AD
18884 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
18885 ;                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (88)
18886 ;          8559                   begin
18887 ;          8560                   TST_LNG_WRD (.NIB_PTR, .NIB_PAT, ERR_FLG);      !TEST THE NIBBLE IF NEVER FOUND BAD
18888 ;          8561
18889 ;          8562                   if .ERR_FLG IS_SET                  !SEE IF TEST FOUND AN ERROR
18890 ;          8563                   then
18891 ;          8564                       begin                           !ERROR AND SET DODU_FLG IS SET
18892 ;          8565                       CMP_THRESHOLD;                  !COMPARE ERROR PRINT THRESHOLD
18893 ;          8566                       ERRDF (88, ARR_DAT, DUMPER);
18894 ;          8567                       PRINTB (SIX_FMT, FNC_18, WRD_50, WRD_10, WRD_12, FNC_5, WRD_19);
18895 ;          8568                       DODU_FLG = ONE;
18896 ;          8569                       end
18897 ;          8570
18898 ;          8571                   end
18899 ;          8572
18900 ;          8573               end
18901 ;          8574           else                                        !THIS NIBBLE IS BAD
18902 ;          8575               begin
18903 ;          8576               NIB_IGNORE [.NIB_PTR] = ONE;            !SET THIS NIBBLE NIB_IGNORE FLAG
18904 ;          8577               BAD_NIB_CNT = .BAD_NIB_CNT + 1;         !INCREMENT BAD NIB COUNT
18905 ;          8578               end;
18906 ;          8579
18907 ;          8580           NIB_PAT = .NIB_PAT + 1;                     !INCREMENT NIB PAT
18908 ;          8581           end;
18909 ;          8582
18910 ;          8583       PASS_CNT = .PASS_CNT + 1;                       !INCREMENT PASS COUNT
18911 ;          8584       end
18912 ;          8585   until (.PASS_CNT eql 113) or (.BAD_NIB_CNT eql 9);      !REPEAT UNTIL COMPLETE
18913 ;          8586
18914 ;          8587   ENDSUB;
18915 ;          8588
18916 ;          8589   if .DODU_FLG IS_SET                                 !DROP THIS UNIT IF DODU_FLG SET
18917 ;          8590   then
18918 ;          8591       begin
18919 ;          8592       DODU (.ML_LUN);
18920 ;          8593       DOCLN;
18921 ;          8594       end;
18922 ;          8595
18923 ;          8596   ENDTST;
18927
18928                                       .SBTTL  $T36 TEST CODE SECTION
18932 066760  004167  115626        $T36:  JSR     R1,$SAVE5                ;                                  8453
18933 066764  162706  000006               SUB     #6,SP
18934 066770  005067  126402               CLR     P.CNT                    ;                                  8512
18935 066774  005046                       CLR     -(SP)                    ; DODU.FLG                         8515
18936 066776  005003                       CLR     R3                       ; WRD.CNT                          8516
18937 067000  005002                       CLR     R2                       ; CNT                              8518
```

SEQ 0399

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
18939
18940                                   ;ML4AD
18941                                   ;               TEST CODE SECTION
18942 067002  012700  177777      1$:    MOV     #-1,R0
18943 067006  010301              2$:    MOV     R3,R1           ; *,PAT.CNT                     8520
18944 067010  006301                     ASL     R1              ; WRD.CNT,*                     8522
18945 067012  062701  014022             ADD     #IO.BUF,R1
18946 067016  010005                     MOV     R0,R5
18947 067020  005205                     INC     R5              ; PAT.CNT,*
18948 067022  042705  177760             BIC     #177760,R5
18949 067026  142711  000017             BICB    #17,(R1)
18950 067032  150511                     BISB    R5,(R1)
18951 067034  010005                     MOV     R0,R5
18952 067036  062705  000002             ADD     #2,R5           ; PAT.CNT,*                     8523
18953 067042  006305                     ASL     R5
18954 067044  006305                     ASL     R5
18955 067046  006305                     ASL     R5
18956 067050  006305                     ASL     R5
18957 067052  042705  177417             BIC     #177417,R5
18958 067056  142711  000360             BICB    #360,(R1)
18959 067062  150511                     BISB    R5,(R1)
18960 067064  010005                     MOV     R0,R5
18961 067066  062705  000003             ADD     #3,R5           ; PAT.CNT,*                     8524
18962 067072  000305                     SWAB    R5
18963 067074  042705  170377             BIC     #170377,R5
18964 067100  042711  007400             BIC     #7400,(R1)
18965 067104  050511                     BIS     R5,(R1)
18966 067106  010005                     MOV     R0,R5
18967 067110  062705  000004             ADD     #4,R5           ; PAT.CNT,*                     8525
18968 067114  000305                     SWAB    R5
18969 067116  006305                     ASL     R5
18970 067120  006305                     ASL     R5
18971 067122  006305                     ASL     R5
18972 067124  006305                     ASL     R5
18973 067126  042705  007777             BIC     #7777,R5
18974 067132  042711  170000             BIC     #170000,(R1)
18975 067136  050511                     BIS     R5,(R1)
18976 067140  005203                     INC     R3              ; WRD.CNT                       8526
18977 067142  062700  000004             ADD     #4,R0           ; *,PAT.CNT                     8520
18978 067146  020027  000013             CMP     R0,#13          ; PAT.CNT,*
18979 067152  003715                     BLE     2$
18980 067154  005202                     INC     R2              ; CNT
18981 067156  020227  000077             CMP     R2,#77          ; CNT,*                         8518
18982 067162  003707                     BLE     1$
18983 067164  104402              3$:    TRAP    2               ;                               8527
18984 067166  152777  000040  126250     BISB    #40,@ML.REG+40  ;                               8529
18985 067174  016702  126632             MOV     ML.DUT,R2
18986 067200  042702  177770             BIC     #177770,R2
18987 067204  142777  000007  126232     BICB    #7,@ML.REG+40
18988 067212  150277  126226             BISB    R2,@ML.REG+40
18989 067216  004767  131070             JSR     PC,GD.BLK.XFER
18990 067222  012777  000061  126154     MOV     #61,@ML.REG     ;                               8531
18991 067230  105777  126220        4$:  TSTB    @ML.REG+50      ;                               8532
18992 067234  100375                     BPL     4$
18993 067236  152777  000040  126200     BISB    #40,@ML.REG+40  ;                               8533
```

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION
18998 067244  016702  126562                   MOV     ML.DUT,R2
18999 067250  042702  177770                   BIC     #177770,R2
19000 067254  142777  000007  126162           BICB    #7,@ML.REG+40
19001 067262  150277  126156                   BISB    R2,@ML.REG+40
19002 067266  005066  000004                   CLR     4(SP)                   ; NIB.IGNORE                    8535
19003 067272  005004                           CLR     R4                      ; PASS.CNT                      8536
19004 067274  005003                           CLR     R3                      ; NIB.PAT                       8537
19005 067276  005005                           CLR     R5                      ; BAD.NIB.CNT                   8538
19006 067300  004767  127106                   JSR     PC,DAT.DM.XFER          ;                               8539
19007 067304  012777  000071  126072           MOV     #71,@ML.REG             ;                               8540
19008 067312  012701  000001                   MOV     #1,R1                   ; *,$$TMP2                      8541
19009 067316  001411                     5$:   BEQ     8$
19010 067320  016702  112572                   MOV     L$DLY,R2                ; *,$$TMP1
19011 067324  001404                           BEQ     7$
19012 067326  005066  000006               6$:   CLR     6(SP)                   ; $$TMP
19013 067332  005302                           DEC     R2                      ; $$TMP1
19014 067334  001374                           BNE     6$
19015 067336  005301                     7$:   DEC     R1                      ; $$TMP2
19016 067340  000766                           BR      5$
19017 067342  017767  126266  125772   8$:   MOV     @ML.REG+230,PD.TEMP     ;                               8545
19018 067350  152777  000020  126146           BISB    #20,@ML.REG+120         ;                               8546
19019 067356  012701  000001                   MOV     #1,R1                   ; *,$$TMP2                      8547
19020 067362  001411                     9$:   BEQ     12$
19021 067364  016702  112526                   MOV     L$DLY,R2                ; *,$$TMP1
19022 067370  001404                           BEQ     11$
19023 067372  005066  000006              10$:   CLR     6(SP)                   ; $$TMP
19024 067376  005302                           DEC     R2                      ; $$TMP1
19025 067400  001374                           BNE     10$
19026 067402  005301                    11$:   DEC     R1                      ; $$TMP2
19027 067404  000766                           BR      9$
19028 067406  017767  126162  123760   12$:   MOV     @ML.REG+170,D1.TEMP
19029 067414  017767  126164  123754           MOV     @ML.REG+200,D2.TEMP
19030 067422  017767  126136  123750           MOV     @ML.REG+160,E2.TEMP
19031 067430  005001                           CLR     R1                      ; NIB.PTR                       8550
19032 067432  010100                    13$:   MOV     R1,R0                   ; NIB.PTR,*                     8553
19033 067434  006200                           ASR     R0
19034 067436  006200                           ASR     R0
19035 067440  006200                           ASR     R0
19036 067442  012702  000004                   MOV     #4,R2                   ;                               8557
19037 067446  060602                           ADD     SP,R2                   ; NIB.IGNORE,*
19038 067450  060002                           ADD     R0,R2
19039 067452  010046                           MOV     R0,-(SP)                ;                               8553
19040 067454  062716  015342                   ADD     #PD.TEMP,(SP)
19041 067460  010146                           MOV     R1,-(SP)                ; NIB.PTR,*
19042 067462  042716  177770                   BIC     #177770,(SP)
19043 067466  012746  000001                   MOV     #1,-(SP)
19044 067472  005046                           CLR     -(SP)
19045 067474  004767  114134                   JSR     PC,BL$GT2
19046 067500  062706  000010                   ADD     #10,SP
19047 067504  005700                           TST     R0
19048 067506  001077                           BNE     16$
19049 067510  010246                           MOV     R2,-(SP)                ;                               8557
```

```
19051                                   ;ML4AD
19052                                   ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS
19053                                                                                                      29-Mar-1982 16:21:03    PA:<
19054 067512  010146                             MOV     R1,-(SP)                ; NIB.PTR,*
19055 067514  042716  177770                      BIC     #177770,(SP)
19056 067520  012746  000001                      MOV     #1,-(SP)
19057 067524  005046                              CLR     -(SP)
19058 067526  004767  114102                      JSR     PC,BL$GT2
19059 067532  062706  000010                      ADD     #10,SP
19060 067536  005700                              TST     R0
19061 067540  001076                              BNE     17$
19062 067542  010146                             MOV     R1,-(SP)                ; NIB.PTR,*                                        8560
19063 067544  010346                             MOV     R3,-(SP)                ; NIB.PAT,*
19064 067546  012746  000010                      MOV     #10,-(SP)
19065 067552  060616                             ADD     SP,(SP)
19066 067554  004767  130616                      JSR     PC,TST.LNG.WRD          ; ERR.FLG,*
19067 067560  026627  000010  000001              CMP     10(SP),#1               ; ERR.FLG,*                                        8562
19068 067566  001044                              BNE     15$
19069 067570  005267  125602                      INC     P.CNT                   ;                                                  8564
19070 067574  026767  125576  125576              CMP     P.CNT,LIMIT
19071 067602  003403                              BLE     14$
19072 067604  062706  000006                      ADD     #6,SP
19073 067610  000457                              BR      18$
19074 067612  104455                    14$:      TRAP    55                      ;                                                  8566
19075 067614  000130                              .WORD   130
19076 067616  013012                              .WORD   ARR.DAT
19077 067620  026302                              .WORD   DUMPER
19078 067622  012746  010730                      MOV     #WRD.19,-(SP)           ;                                                  8567
19079 067626  012746  012216                      MOV     #FNC.5,-(SP)
19080 067632  012746  010650                      MOV     #WRD.12,-(SP)
19081 067636  012746  010630                      MOV     #WRD.10,-(SP)
19082 067642  012746  011270                      MOV     #WRD.50,-(SP)
19083 067646  012746  012414                      MOV     #FNC.18,-(SP)
19084 067652  012746  010322                      MOV     #SIX.FMT,-(SP)
19085 067656  012746  000007                      MOV     #7,-(SP)
19086 067662  010600                             MOV     SP,R0
19087 067664  104414                              TRAP    14                      ; SP,*
19088 067666  012766  000001  000026              MOV     #1,26(SP)               ; *,DODU.FLG                                       8568
19089 067674  062706  000020                      ADD     #20,SP                  ;                                                  8564
19090 067700  062706  000006            15$:      ADD     #6,SP                   ;                                                  8559
19091 067704  000414                              BR      17$                     ;                                                  8553
19092 067706  010246                    16$:      MOV     R2,-(SP)                ;                                                  8576
19093 067710  010146                             MOV     R1,-(SP)                ; NIB.PTR,*
19094 067712  042716  177770                      BIC     #177770,(SP)
19095 067716  012746  000001                      MOV     #1,-(SP)
19096 067722  011646                             MOV     (SP),-(SP)
19097 067724  004767  114142                      JSR     PC,BL$PU2
19098 067730  005205                              INC     R5
19099 067732  062706  000010                      ADD     #10,SP                  ; BAD.NIB.CNT                                      8577
19100 067736  005203                    17$:      INC     R3                      ;                                                  8575
19101 067740  005201                              INC     R1                      ; NIB.PAT                                          8580
19102 067742  020127  000010                      CMP     R1,#10                  ; NIB.PTR                                          8550
19103 067746  003631                              BLE     13$                     ; NIB.PTR,*
19104 067750  005204                    18$:      INC     R4
19105 067752  020427  000161                      CMP     R4,#161                 ; PASS.CNT                                         8583
                                                                                  ; PASS.CNT,*                                       8585
```

```
19107
19108                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
19109                                   ;                       TEST CODE SECTION               29-Mar-1982 16:21:03    PA:<
19110 067756  001405                            BEQ     19$
19111 067760  020527  000011                    CMP     R5,#11                  ; BAD.NIB.CNT,*
19112 067764  001402                            BEQ     19$
19113 067766  000167  177350                    JMP     8$
19114 067772  104467                  19$:      TRAP    67
19115 067774  006000                            ROR     R0
19116 067776  103002                            BHIS    20$
19117 070000  000167  177160                    JMP     3$
19118 070004  021627  000001          20$:      CMP     (SP),#1                 ; DODU.FLG,*                            8589
19119 070010  001004                            BNE     21$
19120 070012  016700  126012                    MOV     ML.LUN,R0               ;                                       8592
19121 070016  104451                            TRAP    51
19122 070020  104444                            TRAP    44
19123 070022  062706  000010          21$:      ADD     #10,SP                  ;                                       8453
19124 070026  000207                            RTS     PC
19125
19126                                   ; Routine Size:  276 words
19127                                   ; Maximum stack depth per invocation:  21 words
19132
19133
19137
19138                                           .SBTTL  T36 TEST CODE SECTION
19142 070030                            T36::
19143 070030  004767  176724            1$:     JSR     PC,$T36                 ;                                       8594
19144 070034  104466                            TRAP    66
19145 070036  006000                            ROR     R0
19146 070040  103773                            BLO     1$
19147 070042  000207                            RTS     PC
19148
19149                                   ; Routine Size:  6 words
19150                                   ; Maximum stack depth per invocation:  0 words
19155
19156
19157 ;         8597  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (89)

```
19159 ;ML4AD
19160 ;                 TEST CODE SECTION
19161
19162 ;          8598  !
19163 ;          8599  !
19164 ;          8600  BGNTST;
19165 ;          8601
19166 ;          8602  !++
19167 ;          8603  ! TEST NUMBER:  TST 37
19168 ;          8604  !
19169 ;          8605  ! TEST NAME:    RAM BUS ADRS COUNTER TEST /READ PATH
19170 ;          8606  !
19171 ;          8607  ! TEST DESCRIPTION:
19172 ;          8608  !
19173 ;          8609  !       TEST ABILITY OF RAM/BUS ADRS
19174 ;          8610  !       COUNTERS TO LOAD/UNLOAD THE SKIP
19175 ;          8611  !       RAM DURING READ FUNCTIONS BY:
19176 ;          8612  !
19177 ;          8613  !       1.  LOADING A REPEATING COUNT OF 0
19178 ;          8614  !           TO 63 INTO THE NIBBLES OF THE
19179 ;          8615  !           FIRST 64 WORDS OF THE IO_BUF.
19180 ;          8616  !
19181 ;          8617  !       2.  VIA MBUS WRITE FUNCTION WRITE
19182 ;          8618  !           THE CONTENTS OF THE IO_BUF
19183 ;          8619  !           INTO THE GOOD BLOCK.
19184 ;          8620  !
19185 ;          8621  !       3.  CLEAR OUT THE IO_BUF
19186 ;          8622  !
19187 ;          8623  !       4.  VIA MBUS READ FUNCTION READ
19188 ;          8624  !           THE GOOD BLOCK FOR THE REPEATING
19189 ;          8625  !           COUNT OF 0 TO 63.
19190 ;          8626  !
19191 ;          8627  ! IMPLICIT INPUTS:
19192 ;          8628  !
19193 ;          8629  !       IO_BUF
19194 ;          8630  !
19195 ;          8631  !       A VECTOR OF 256 WORDS WHERE
19196 ;          8632  !       DATA FOR MBUS READS AND WRITE
19197 ;          8633  !       FUNCTION ARE FOUND.
19198 ;          8634  !
19199 ;          8635  !--
19200 ;          8636
19201 ;          8637  local
19202 ;          8638      DODU_FLG,                       !DROP UNIT FLAG
19203 ;          8639      WRD_CNT,                        !WORD COUNT
19204 ;          8640      PAT_INC,                        !PATTERN INCREMENT
19205 ;          8641      SIZ_EXP,                        !SIZE EXPRESSION
19206 ;          8642      POS_EXP,                        !POSITIONAL EXPRESSION
19207 ;          8643      TEMP;                           !TEMPORARY STORAGE LOCATION
19208 ;          8644
19209 ;          8645  CLR_THRESHOLD;                      !CLEAR ERROR PRINT THRESHOLD
19210 ;          8646  CLR_MBUS;
19211 ;          8647  DODU_FLG = ZERO;
19212 ;          8648  WRD_CNT = ZERO;
19213 ;          8649
```

```
19215 ;ML4AD
19216 ;                 TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
19217 ;                                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (89)
19218 ;          8650   incr COUNT from 0 to 63 do                          !LOAD 64 WORDS WITH REPEATING  COUNTS OF 0-63
19219 ;          8651
19220 ;          8652       incr PAT_CNT from -1 to 11 by 4 do              !LOAD NIBBLES IN WORD WITH REPEATING COUNT OF 0-63
19221 ;          8653           begin
19222 ;          8654           (IO_BUF [.WRD_CNT])<0, 4> = .PAT_CNT + 1;       !LOAD FIRST NIBBLE IN WORD
19223 ;          8655           (IO_BUF [.WRD_CNT])<4, 4> = .PAT_CNT + 2;       !LOAD SECOND NIBBLE IN WORD
19224 ;          8656           (IO_BUF [.WRD_CNT])<8, 4> = .PAT_CNT + 3;       !LOAD THIRD NIBBLE IN WORD
19225 ;          8657           (IO_BUF [.WRD_CNT])<12, 4> = .PAT_CNT + 4;      !LOAD FORTH NIBBLE IN WORD
19226 ;          8658           WRD_CNT = .WRD_CNT + 1;                         !INCREMENT TO NEXT WORD
19227 ;          8659           end;
19228 ;          8660
19229 ;          8661   GD_BLK_XFER ();                                     !SET UP A GOOD BLOCK XFERR
19230 ;          8662   MLCS1 = write;                                      !DO A WRITE FUNCTION
19231 ;          8663   TIME_OUT_LOOP;
19232 ;          8664   BGNSUB;
19233 ;          8665
19234 ;          8666   incr IO_CNT from 0 to 255 do                        !CLEAR OUT IO_BUF
19235 ;          8667       IO_BUF [.IO_CNT] = ZEROES;
19236 ;          8668
19237 ;          8669   CLR_MBUS;
19238 ;          8670   GD_BLK_XFER ();                                     !SET UP A GOOD BLOCK XFERR
19239 ;          8671   MLCS1 = read;                                       !DO A READ FUNCTION
19240 ;          8672   TIME_OUT_LOOP;
19241 ;          8673   CLR_MBUS;
19242 ;          8674   SIZ_EXP = 4;                                        !FIELD SIZE FOR NIBBLES ALWAYS 4 BITS
19243 ;          8675   WRD_CNT = 0;
19244 ;          8676
19245 ;          8677   incr COUNT from 0 to 63 do                          !READ 64 WORDS IN IO_BUF
19246 ;          8678
19247 ;          8679       incr PAT_CNT from -1 to 11 by 4 do              !READ REPEATING COUNTS OF 0-63
19248 ;          8680           begin
19249 ;          8681           POS_EXP = ZERO;                             !FIELD SELECTOR SELECTS THE FOUR NIBBLES
19250 ;          8682           PAT_INC = ONE;
19251 ;          8683           TEMP = .IO_BUF [.WRD_CNT];                  !GET A WORD OUT OF IO_BUF
19252 ;          8684
19253 ;          8685           incr CNT from 0 to 3 do                     !READ THE FOUR NIBBLES IN WORD
19254 ;          8686               begin
19255 ;          8687
19256 ;          8688               if .TEMP<.POS_EXP, .SIZ_EXP> neq (.PAT_CNT + .PAT_INC)
19257 ;          8689                                                       !COMPARE NIBBLE WITH RESPECTIVE 0 -63 CNT
19258 ;          8690               then
19259 ;          8691                   begin                               !ERROR AND SET DODU_FLG IF NEQ
19260 ;          8692                   CMP_THRESHOLD;                      !COMPARE ERROR PRINT THRESHOLD
19261 ;          8693                   ERRDF (89, ARR_DAT, DUMPER);
19262 ;          8694                   PRINTB (SIX_FMT, FNC_18, WRD_50, WRD_10, WRD_12, FNC_6, WRD_19);
19263 ;          8695                   DODU_FLG = ONE;
19264 ;          8696                   end;
19265 ;          8697
19266 ;          8698               POS_EXP = .POS_EXP + 4;                 !POINT TO THE NEXT NIBBLE IN WORD
19267 ;          8699               PAT_INC = .PAT_INC + 1;                 !INCREMENT THE 0-03 COUNT
19268 ;          8700               end;
19269 ;          8701
```

SEQ 0405

```
 19271 ;ML4AD
 19272 ;                 TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 19273                                                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (89)
 19274 ;          8702          WRD_CNT = .WRD_CNT + 1;       !GET THE NEXT IO_BUF WORD
 19275 ;          8703          end;
 19276 ;          8704
 19277 ;          8705  ENDSUB;
 19278 ;          8706
 19279 ;          8707  if .DODU_FLG IS_SET                   !DROP THIS UNIT IF DODU_FLG SET
 19280 ;          8708  then
 19281 ;          8709      begin
 19282 ;          8710      DODU (.ML_LUN);
 19283 ;          8711      DOCLN;
 19284 ;          8712      end;
 19285 ;          8713
 19286 ;          8714  ENDTST;
 19290
 19291                                    .SBTTL  $T37 TEST CODE SECTION
 19295 070044  004167  114542      $T37:  JSR     R1,$SAVE5                ;                          8596
 19296 070050  162706  000012             SUB     #12,SP
 19297 070054  005067  125316             CLR     P.CNT
 19298 070060  152777  000040  125356     BISB    #40,@ML.REG+40           ;                          8643
 19299 070066  016705  125740             MOV     ML.DUT,R5                ;                          8645
 19300 070072  042705  177770             BIC     #177770,R5
 19301 070076  142777  000007  125340     BICB    #7,@ML.REG+40
 19302 070104  150577  125334             BISB    R5,@ML.REG+40
 19303 070110  005066  000010             CLR     10(SP)                   ; DODU.FLG                 8647
 19304 070114  005066  000002             CLR     2(SP)                    ; WRD.CNT                  8648
 19305 070120  005002                     CLR     R2                       ; COUNT                    8650
 19306 070122  012703  177777      1$:    MOV     #-1,R3                   ; *,PAT.CNT                8652
 19307 070126  016604  000002      2$:    MOV     2(SP),R4                 ; WRD.CNT,*                8654
 19308 070132  006304                     ASL     R4
 19309 070134  062704  014022             ADD     #IO.BUF,R4
 19310 070140  010305                     MOV     R3,R5                    ; PAT.CNT,*
 19311 070142  005205                     INC     R5
 19312 070144  042705  177760             BIC     #177760,R5
 19313 070150  142714  000017             BICB    #17,(R4)
 19314 070154  150514                     BISB    R5,(R4)
 19315 070156  010305                     MOV     R3,R5                    ; PAT.CNT,*                8655
 19316 070160  062705  000002             ADD     #2,R5
 19317 070164  006305                     ASL     R5
 19318 070166  006305                     ASL     R5
 19319 070170  006305                     ASL     R5
 19320 070172  006305                     ASL     R5
 19321 070174  042705  177417             BIC     #177417,R5
 19322 070200  142714  000360             BICB    #360,(R4)
 19323 070204  150514                     BISB    R5,(R4)
 19324 070206  010305                     MOV     R3,R5                    ; PAT.CNT,*                8656
 19325 070210  062705  000003             ADD     #3,R5
```

SEQ 0406


```
                                        ;ML4AD
                                        ;                TEST CODE SECTION
19330 070214  000305                            SWAB    R5
19331 070216  042705  170377                      BIC     #170377,R5
19332 070222  042714  007400                      BIC     #7400,(R4)
19333 070226  050514                              BIS     R5,(R4)
19334 070230  010305                              MOV     R3,R5                   ; PAT.CNT,*                     8657
19335 070232  062705  000004                      ADD     #4,R5
19336 070236  000305                              SWAB    R5
19337 070240  006305                              ASL     R5
19338 070242  006305                              ASL     R5
19339 070244  006305                              ASL     R5
19340 070246  006305                              ASL     R5
19341 070250  042705  007777                      BIC     #7777,R5
19342 070254  042714  170000                      BIC     #170000,(R4)
19343 070260  050514                              BIS     R5,(R4)
19344 070262  005266  000002                      INC     2(SP)                   ; WRD.CNT                       8658
19345 070266  062703  000004                      ADD     #4,R3                   ; *,PAT.CNT                     8652
19346 070272  020327  000013                      CMP     R3,#13                  ; PAT.CNT,*
19347 070276  003713                              BLE     2$
19348 070300  005202                              INC     R2                      ; COUNT                         8650
19349 070302  020227  000077                      CMP     R2,#77                  ; COUNT,*
19350 070306  003705                              BLE     1$
19351 070310  004767  127776                      JSR     PC,GD.BLK.XFER          ;                               8661
19352 070314  012777  000061  125062              MOV     #61,@ML.REG             ;                               8662
19353 070322  105777  125126           3$:        TSTB    @ML.REG+50
19354 070326  100375                              BPL     3$
19355 070330  104402                    4$:       TRAP    2                       ;                               8663
19356 070332  005003                              CLR     R3                      ; IO.CNT                        8666
19357 070334  010304                    5$:       MOV     R3,R4                   ; IO.CNT,*                      8667
19358 070336  006304                              ASL     R4
19359 070340  005064  014022                      CLR     IO.BUF(R4)
19360 070344  005203                              INC     R3                      ; IO.CNT                        8666
19361 070346  020327  000377                      CMP     R3,#377                 ; IO.CNT,*
19362 070352  003770                              BLE     5$
19363 070354  152777  000040  125062              BISB    #40,@ML.REG+40          ;                               8667
19364 070362  016704  125444                      MOV     ML.DUT,R4
19365 070366  042704  177770                      BIC     #177770,R4
19366 070372  142777  000007  125044              BICB    #7,@ML.REG+40
19367 070400  150477  125040                      BISB    R4,@ML.REG+40
19368 070404  004767  127702                      JSR     PC,GD.BLK.XFER
19369 070410  012777  000071  124766              MOV     #71,@ML.REG             ;                               8670
19370 070416  105777  125032           6$:        TSTB    @ML.REG+50              ;                               8671
19371 070422  100375                              BPL     6$
19372 070424  152777  000040  125012              BISB    #40,@ML.REG+40          ;                               8672
19373 070432  016704  125374                      MOV     ML.DUT,R4
19374 070436  042704  177770                      BIC     #177770,R4
19375 070442  142777  000007  124774              BICB    #7,@ML.REG+40
19376 070450  150477  124770                      BISB    R4,@ML.REG+40
19377 070454  012766  000004  000006              MOV     #4,6(SP)                ; *,SIZ.EXP                     8674
19378 070462  005066  000002                      CLR     2(SP)                   ; WRD.CNT                       8675
19379 070466  005001                              CLR     R1                      ; COUNT                         8677
19380 070470  012703  177777           7$:        MOV     #-1,R3                  ; *,PAT.CNT                     8679
19381 070474  005005                    8$:       CLR     R5                      ; POS.EXP                       8681
```

SEQ 0407

```
19383                                          ;ML4AD
19384                                          ;
19385                                          ;       TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS
19386 070476  012716  000001                                MOV     #1,(SP)                                  29-Mar-1982 16:21:03    PA:<
19387 070502  016604  000002                                MOV     2(SP),R4        ; *,PAT.INC                                      8682
19388 070506  006304                                       ASL     R4              ; WRD.CNT,*                                      8683
19389 070510  016466  014022  000004                       MOV     IO.BUF(R4),4(SP)
19390 070516  005002                                       CLR     R2              ; *,TEMP
19391 070520  016646  000004                        9$:    MOV     4(SP),-(SP)     ; CNT                                            8685
19392 070524  010546                                       MOV     R5,-(SP)        ; TEMP,*                                         8688
19393 070526  016646  000012                               MOV     12(SP),-(SP)    ; POS.EXP,*
19394 070532  005046                                       CLR     -(SP)           ; SIZ.EXP,*
19395 070534  004767  112752                               JSR     PC,BL$GT1
19396 070540  062706  000010                               ADD     #10,SP
19397 070544  010304                                       MOV     R3,R4
19398 070546  061604                                       ADD     (SP),R4         ; PAT.CNT,*
19399 070550  020004                                       CMP     R0,R4           ; PAT.INC,*
19400 070552  001441                                       BEQ     10$
19401 070554  005267  124616                               INC     P.CNT
19402 070560  026767  124612  124612                       CMP     P.CNT,LIMIT     ;                                                8691
19403 070566  003042                                       BGT     11$
19404 070570  104455                                       TRAP    55              ;                                                8693
19405 070572  000131                                       .WORD   131
19406 070574  013012                                       .WORD   ARR.DAT
19407 070576  026302                                       .WORD   DUMPER
19408 070600  012746  010730                               MOV     #WRD.19,-(SP)   ;                                                8694
19409 070604  012746  012226                               MOV     #FNC.6,-(SP)
19410 070610  012746  010650                               MOV     #WRD.12,-(SP)
19411 070614  012746  010630                               MOV     #WRD.10,-(SP)
19412 070620  012746  011270                               MOV     #WRD.50,-(SP)
19413 070624  012746  012414                               MOV     #FNC.18,-(SP)
19414 070630  012746  010322                               MOV     #SIX.FMT,-(SP)
19415 070634  012746  000007                               MOV     #7,-(SP)
19416 070640  010600                                       MOV     SP,R0           ; SP,*
19417 070642  104414                                       TRAP    14
19418 070644  012766  000001  000030                       MOV     #1,30(SP)       ; *,DODU.FLG                                     8695
19419 070652  062706  000020                               ADD     #20,SP          ;                                                8691
19420 070656  062705  000004                        10$:   ADD     #4,R5           ; *,POS.EXP                                      8698
19421 070662  005216                                       INC     (SP)            ; PAT.INC                                        8699
19422 070664  005202                                       INC     R2              ; CNT                                            8685
19423 070666  020227  000003                               CMP     R2,#3           ; CNT,*
19424 070672  003712                                       BLE     9$
19425 070674  005266  000002                        11$:   INC     2(SP)           ; WRD.CNT                                        8702
19426 070700  062703  000004                               ADD     #4,R3           ; *,PAT.CNT                                      8679
19427 070704  020327  000013                               CMP     R3,#13          ; PAT.CNT,*
19428 070710  003671                                       BLE     8$
19429 070712  005201                                       INC     R1              ; COUNT                                          8677
19430 070714  020127  000077                               CMP     R1,#77          ; COUNT,*
19431 070720  003663                                       BLE     7$
19432 070722  104467                                       TRAP    67              ;                                                8703
19433 070724  006000                                       ROR     R0
19434 070726  103600                                       BLO     4$
19435 070730  026627  000010  000001                       CMP     10(SP),#1       ; DODU.FLG,*                                     8707
19436 070736  001004                                       BNE     12$
19437 070740  016700  125064                               MOV     ML.LUN,R0       ;                                                8710
```

SEQ 0408

```
19439                                   ;ML4AD
19440                                   ;               TEST CODE SECTION                                       29-Mar-1982 16:23:04    TOPS
19441                                                                                                           29-Mar-1982 16:21:03    PA:<
19442 070744  104455                             TRAP    51
19443 070746  104444                             TRAP    44
19444 070750  062706  000012               12$:  ADD     #12.SP                          ;                                       8596
19445 070754  000207                             RTS     PC
19446
19447                                   ; Routine Size:  229 words
19448                                   ; Maximum stack depth per invocation:  19 words
19453
19454
19458
19459                                           .SBTTL  T37 TEST CODE SECTION
19463 070756                            T37::
19464 070756  004767  177062            1$:   JSR     PC,$T37                         ;                                       8712
19465 070762  104466                            TRAP    66
19466 070764  006000                            ROR     R0
19467 070766  103773                            BLO     1$
19468 070770  000207                            RTS     PC
19469
19470                                   ; Routine Size:  6 words
19471                                   ; Maximum stack depth per invocation:  0 words
19476
19477
19478 ;          8715  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (90)

```
19480 ;ML4AD
19481 ;                 TEST CODE SECTION
19482
19483 ;          8716 !
19484 ;          8717 !
19485 ;          8718 BGNTST;
19486 ;          8719
19487 ;          8720 !++
19488 ;          8721 ! TEST NUMBER:  TST 38
19489 ;          8722 !
19490 ;          8723 ! TEST NAME:    SYNC DATA BUS BIT UNIQUENESS TEST/WRITE PATH
19491 ;          8724 !
19492 ;          8725 ! TEST DESCRIPTION:
19493 ;          8726 !
19494 ;          8727 !       TEST SYNCHRONOUS DATA BUS FOR
19495 ;          8728 !       DATA BIT UNIQUENESS BY:
19496 ;          8729 !
19497 ;          8730 !       1.  LOADING THE FIRST 16 WORDS IN
19498 ;          8731 !           THE IO_BUF WITH A SHIFTING
19499 ;          8732 !           ZERO IN A FIELD OF ONES PATTERN.
19500 ;          8733 !
19501 ;          8734 !       2.  VIA MBUS WRITE FUNCTION WRITE
19502 ;          8735 !           SHIFTING PATTERN THROUGH THE
19503 ;          8736 !           DATA BUS AND INTO THE GOOD
19504 ;          8737 !           BLOCK.
19505 ;          8738 !
19506 ;          8739 !       3.  VIA DAT_DM MODE READ THE
19507 ;          8740 !           GOOD BLOCK AND SAVE ALL GOOD
19508 ;          8741 !           NIBBLE DATA, IN THEIR PROPER
19509 ;          8742 !           SEQUENCE, INTO A STACK
19510 ;          8743 !           STRUCTURE.
19511 ;          8744 !
19512 ;          8745 !       4.  INTERRIGATE STACK STRUCTURE FOR
19513 ;          8746 !           SHIFTED DATA PATTERN.
19514 ;          8747 !
19515 ;          8748 !
19516 ;          8749 ! IMPLICIT INPUTS:
19517 ;          8750 !
19518 ;          8751 !       PD_TEMP
19519 ;          8752 !       A BIT VECTOR OF 16 BITS WHERE
19520 ;          8753 !       THE READ PROM DATA IS STORED
19521 ;          8754 !       AND ACCESSED FROM.
19522 ;          8755 !
19523 ;          8756 !       IO_BUF
19524 ;          8757 !       A VECTOR OF 256 WORDS WHERE
19525 ;          8758 !       DATA FOR MBUS READS AND WRITE
19526 ;          8759 !       FUNCTION ARE FOUND.
19527 ;          8760 !
19528 ;          8761 !       STACK
19529 ;          8762 !       A VECTOR OF 198 BYTE LOCATIONS
19530 ;          8763 !       WHERE GOOD NIBBLE DATA IS STORED
19531 ;          8764 !       WHEN STRIPPING AWAY BAD NIBBLE
19532 ;          8765 !       LOCATIONS OF A BLOCK.
19533 ;          8766 !
19534 ;          8767 !
```

```
;ML4AD
;                    TEST CODE SECTION                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                                                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (90)
;
;          8768  !--
;          8769
;          8770  local
;          8771      SAV_NIB,                                  !STORES THE SHIFTED BIT PATTERN
;          8772      DODD_FLG,                                 !DROP UNIT FLAG
;          8773      NIB_BIT,                                  !NIBBLE PATTERN
;          8774      TST_PAT,                                  !TEST PATTERN
;          8775      ALL_ONES_1,                               !HOW MANY '17' NIBBLE PATTERN POSITION TO READ
;          8776      ALL_ONES_2,                               !HOW MANY '17' NIBBLE PATTERN POSITION TO READ
;          8777      STK_PTR,                                  !STACK POINTER
;          8778      COUNT;                                    !COUNTER
;          8779
;          8780  CLR_THRESHOLD;                               !CLEAR ERROR PRINT THRESHOLD
;          8781  BGNSUB;
;          8782  CLR_MBUS;
;          8783  DODD_FLG = ZERO;
;          8784  TST_PAT = ONE;                                !ONE IN A FIELD OF ZEROES
;          8785
;          8786  incr CNT from 0 to 15 do                     !WRITE 16 WORDS WITH SHIFTING 0 IN FIELD OF 1'S.
;          8787      begin
;          8788      IO_BUF [.CNT] = not .TST_PAT;
;          8789      TST_PAT = .TST_PAT^ONE;
;          8790      end;
;          8791
;          8792  GD_BLK_XFER ();                               !SET UP A GOOD BLOCK XFERR
;          8793  MLCS1 = write;                                !WRITE SHIFTING PATTERN THROUGH SYNC BUS
;          8794  TIME_OUT_LOOP;
;          8795  STRIPPER (21, 8);                             !CALL ROUTINE TO STRIP OUT BAD NIBBLE DATA FROM ARRAY WORDS
;          8796  STK_PTR = -1;                                 !RESET THE STACK POINTER
;          8797  NIB_BIT = ONE;                                !SHIFTING NIBBLE PAT OF 1 IN FIELD OF 0'S
;          8798  ALL_ONES_1 = ZERO;                           !READ NO '17' NIBBLE PATTERN ON FIRST PASS
;          8799  ALL_ONES_2 = 3;                              !READ THREE '17' NIBBLE PATTERN ON FIRST PASS
;          8800
;          8801  incr BY_FOUR_WRDS from 0 to 3 do             !READ 4 GROUPS OF 4 WORDS
;          8802      begin
;          8803
;          8804      incr BY_ONE_WRD from 0 to 3 do            !READ 4 GROUPS OF 1 WORD
;          8805          begin
;          8806          COUNT = ZERO;                         !CLEAR COUNT
;          8807
;          8808          until .COUNT eql .ALL_ONES_1 do       !READ X NUMBER OF '17' NIBBLE PAT
;          8809              begin
;          8810              COUNT = .COUNT + 1;               !INCREMENT COUNT
;          8811              STK_PTR = .STK_PTR + 1;           !INCREMENT STACK POINTER
;          8812
;          8813              if (.stack [.STK_PTR]) neq %o'000017'       !COMPARE STACK WITH '17'
;          8814              then
;          8815                  begin                         !ERROR AND SET DODU_FLG IF NEQ
;          8816                  CMP_THRESHOLD;                !COMPARE ERROR PRINT THRESHOLD
;          8817                  ERRDF (90, SYNC, DUMPER);
;          8818                  PRINTB (SIX_FMT, WRD_23, WRD_39, PHR_4, WRD_12, FNC_5, WRD_19);
;          8819                  PRINTB (FMT_5, ONES, .stack [.STK_PTR], .STK_PTR);
```

SEQ 0411


```
19592 ;ML4AD
19593 ;                 TEST CODE SECTION
19594 ;
19595 ;          8820                   DODU_FLG = ONE;
19596 ;          8821                   end;
19597 ;          8822
19598 ;          8823               end;
19599 ;          8824
19600 ;          8825           STK_PTR = .STK_PTR + 1;                  !INCREMENT THE STACK POINTER
19601 ;          8826           SAV_NIB = ( not .NIB_BIT) and (%o'000017');     !GENERATE THE SHIFTED BIT
19602 ;          8827
19603 ;          8828           if (.stack [.STK_PTR]) neq (.SAV_NIB)    !COMPARE STACK TO SHIFTED BIT
19604 ;          8829           then
19605 ;          8830               begin                                !ERROR AND SET DODU_FLG IF NEQ
19606 ;          8831               CMP_THRESHOLD;                       !COMPARE ERROR PRINT THRESHOLD
19607 ;          8832               ERRDF (91, SYNC, DUMPER);
19608 ;          8833               PRINTB (SIX_FMT, WRD_23, WRD_39, PHR_4, WRD_12, FNC_5, WRD_19);
19609 ;          8834               PRINTB (FMT_15, .STK_PTR);
19610 ;          8835               PRINTB (FMT_5, .SAV_NIB, .stack [.STK_PTR]);
19611 ;          8836               DODU_FLG = ONE;
19612 ;          8837               end;
19613 ;          8838
19614 ;          8839           COUNT = ZEROES;                          !CLEAR COUNT
19615 ;          8840
19616 ;          8841           until .COUNT eql .ALL_ONES_2 do          !READ X NUMBER OF '17' NIBBLE PAT
19617 ;          8842               begin
19618 ;          8843               COUNT = .COUNT + 1;                  !INCREMENT COUNT
19619 ;          8844               STK_PTR = .STK_PTR + 1;              !INCREMENT STACK POINTER
19620 ;          8845
19621 ;          8846               if (.stack [.STK_PTR]) neq %o'000017'        !COMPARE STACK POINTER WITH  '17'
19622 ;          8847               then
19623 ;          8848                   begin                            !ERROR AND SET DODU_FLG IF SET
19624 ;          8849                   CMP_THRESHOLD;                   !COMPARE ERROR PRINT THRESHOLD
19625 ;          8850                   ERRDF (92, SYNC, DUMPER);
19626 ;          8851                   PRINTB (SIX_FMT, WRD_23, WRD_39, PHR_4, WRD_12, FNC_5, WRD_19);
19627 ;          8852                   PRINTB (FMT_5, ONES, .stack [.STK_PTR], .STK_PTR);
19628 ;          8853                   DODU_FLG = ONE;
19629 ;          8854                   end;
19630 ;          8855
19631 ;          8856               end;
19632 ;          8857
19633 ;          8858               NIB_BIT = .NIB_BIT^ONE;              !SHIFT THE SHIFTED NIBBLE BIT
19634 ;          8859               end;
19635 ;          8860
19636 ;          8861           NIB_BIT = ONE;                           !RESET THE SHIFTED NIBBLE BIT
19637 ;          8862           ALL_ONES_1 = .ALL_ONES_1 + 1;            !READ ONE MORE '17' PATTERN
19638 ;          8863           ALL_ONES_2 = .ALL_ONES_2 - 1;            !READ ONE LESS '17' PATTERN
19639 ;          8864           end;
19640 ;          8865
19641 ;          8866   ENDSUB;
19642 ;          8867
19643 ;          8868   if .DODU_FLG IS_SET                             !DROP THIS UNIT IF DODU_FLG SET
19644 ;          8869   then
19645 ;          8870       begin
19646 ;          8871       DODU (.ML_LUN);
```

SEQ 0412


```
19648 ;ML4AD
19649 ;                 TEST CODE SECTION
19650
19651 ;          8872       DOCLN;
19652 ;          8873       end;
19653 ;          8874
19654 ;          8875   ENDTST;
19658
19659                                          .SBTTL  $T38 TEST CODE SECTION
19663 070772  004167  113614          $T38:   JSR     R1,$SAVE5                ;                                8714
19664 070776  162706  000012                  SUB     #12,SP
19665 071002  005067  124370                  CLR     P.CNT
19666 071006  104402                  1$:     TRAP    2                        ;                                8778
19667 071010  152777  000040  124426          BISB    #40,@ML.REG+40           ;                                8780
19668 071016  016705  125010                  MOV     ML.DUT,R5                ;                                8781
19669 071022  042705  177770                  BIC     #177770,R5
19670 071026  142777  000007  124410          BICB    #7,@ML.REG+40
19671 071034  150577  124404                  BISB    R5,@ML.REG+40
19672 071040  005001                          CLR     R1
19673 071042  012766  000001  000006          MOV     #1,6(SP)                 ; DODU.FLG                       8783
19674 071050  005004                          CLR     R4                       ; *,TST.PAT                      8784
19675 071052  010405                  2$:     MOV     R4,R5                    ; CNT                            8786
19676 071054  006305                          ASL     R5                       ; CNT,*                          8788
19677 071056  016665  000006  014022          MOV     6(SP),IO.BUF(R5)         ; TST.PAT,*
19678 071064  005165  014022                  COM     IO.BUF(R5)
19679 071070  006366  000006                  ASL     6(SP)                    ; TST.PAT                        8789
19680 071074  005204                          INC     R4                       ; CNT                            8786
19681 071076  020427  000017                  CMP     R4,#17                   ; CNT,*
19682 071102  003763                          BLE     2$
19683 071104  004767  127202                  JSR     PC,GD.BLK.XFER           ;                                8792
19684 071110  012777  000061  124266          MOV     #61,@ML.REG              ;                                8793
19685 071116  105777  124332          3$:     TSTB    @ML.REG+50
19686 071122  100375                          BPL     3$
19687 071124  012746  000025                  MOV     #25,-(SP)
19688 071130  012746  000010                  MOV     #10,-(SP)                ;                                8795
19689 071134  004767  125304                  JSR     PC,STRIPPER
19690 071140  012702  177777                  MOV     #-1,R2
19691 071144  012766  000001  000004          MOV     #1,4(SP)                 ; *,STK.PTR                      8796
19692 071152  005066  000010                  CLR     10(SP)                   ; *,NIB.BIT                      8797
19693 071156  012766  000003  000006          MOV     #3,6(SP)                 ; ALL.ONES.1                     8798
19694 071164  005004                          CLR     R4                       ; *,ALL.ONES.2                   8799
19695 071166  005005                  4$:     CLR     R5                       ; BY.FOUR.WRDS                   8801
19696 071170  005003                  5$:     CLR     R3                       ; BY.ONE.WRD                     8804
19697 071172  020366  000010          6$:     CMP     R3,10(SP)                ; COUNT                          8806
19698 071176  001463                          BEQ     7$                       ; COUNT,ALL.ONES.1               8808
19699 071200  005203                          INC     R3
19700 071202  005202                          INC     R2                       ; COUNT                          8810
19701 071204  126227  015034  000017          CMPB    STACK(R2),#17            ; STK.PTR                        8811
19702 071212  001767                          BEQ     6$                       ; *(STK.PTR),*                   8813
```

```
19704                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
19705                                   ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
19706
19707 071214  005267  124156                    INC     P.CNT                           ;                               8815
19708 071220  026767  124152  124152            CMP     P.CNT,LIMIT
19709 071226  003047                            BGT     7$
19710 071230  104455                            TRAP    55                              ;                               8817
19711 071232  000132                            .WORD   132
19712 071234  012750                            .WORD   SYNC
19713 071236  026302                            .WORD   DUMPER
19714 071240  012746  010730                    MOV     #WRD.19,-(SP)                   ;                               8818
19715 071244  012746  012216                    MOV     #FNC.5,-(SP)
19716 071250  012746  010650                    MOV     #WRD.12,-(SP)
19717 071254  012746  011676                    MOV     #PHR.4,-(SP)
19718 071260  012746  011140                    MOV     #WRD.39,-(SP)
19719 071264  012746  010766                    MOV     #WRD.23,-(SP)
19720 071270  012746  010322                    MOV     #SIX.FMT,-(SP)
19721 071274  012746  000007                    MOV     #7,-(SP)
19722 071300  010600                            MOV     SP,R0                           ; SP,*
19723 071302  104414                            TRAP    14
19724 071304  010216                            MOV     R2,(SP)                         ; STK.PTR,*                     8819
19725 071306  005046                            CLR     -(SP)
19726 071310  116216  015034                    MOVB    STACK(R2),(SP)                  ; *(STK.PTR),*
19727 071314  012746  177777                    MOV     #-1,-(SP)
19728 071320  012746  006650                    MOV     #FMT.5,-(SP)
19729 071324  012746  000004                    MOV     #4,-(SP)
19730 071330  010600                            MOV     SP,R0                           ; SP,*
19731 071332  104414                            TRAP    14
19732 071334  012701  000001                    MOV     #1,R1                           ; *,DODU.FLG                    8820
19733 071340  062706  000030                    ADD     #30,SP                          ;                               8815
19734 071344  000712                            BR      6$                              ;                               8808
19735 071346  005202                    7$:     INC     R2                              ; STK.PTR                       8825
19736 071350  012766  000017  000014            MOV     #17,14(SP)                      ; *,SAV.NIB                     8826
19737 071356  046666  000004  000014            BIC     4(SP),14(SP)                    ; NIB.BIT,SAV.NIB
19738 071364  005000                            CLR     R0                              ;                               8828
19739 071366  156200  015034                    BISB    STACK(R2),R0                    ; *(STK.PTR),*
19740 071372  020066  000014                    CMP     R0,14(SP)                       ; *,SAV.NIB
19741 071376  001462                            BEQ     8$
19742 071400  005267  123772                    INC     P.CNT                           ;                               8830
19743 071404  026767  123766  123766            CMP     P.CNT,LIMIT
19744 071412  003153                            BGT     11$
19745 071414  104455                            TRAP    55                              ;                               8832
19746 071416  000133                            .WORD   133
19747 071420  012750                            .WORD   SYNC
19748 071422  026302                            .WORD   DUMPER
19749 071424  012746  010730                    MOV     #WRD.19,-(SP)                   ;                               8833
19750 071430  012746  012216                    MOV     #FNC.5,-(SP)
19751 071434  012746  010650                    MOV     #WRD.12,-(SP)
19752 071440  012746  011676                    MOV     #PHR.4,-(SP)
19753 071444  012746  011140                    MOV     #WRD.39,-(SP)
19754 071450  012746  010766                    MOV     #WRD.23,-(SP)
19755 071454  012746  010322                    MOV     #SIX.FMT,-(SP)
19756 071460  012746  000007                    MOV     #7,-(SP)
19757 071464  010600                            MOV     SP,R0                           ; SP,*
19758 071466  104414                            TRAP    14
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;                TEST CODE SECTION                              29-Mar-1982 16:21:03    PA:<
071470  010216                                  MOV     R2,(SP)
071472  012746  007370                          MOV     #FMT.15,-(SP)          ; STK.PTR,*                                      8834
071476  012746  000002                          MOV     #2,-(SP)
071502  010600                                  MOV     SP,R0                  ; SP,*
071504  104414                                  TRAP    14
071506  005016                                  CLR     (SP)                   ;                                                8835
071510  116216  015034                          MOVB    STACK(R2),(SP)         ; *(STK.PTR),*
071514  016646  000040                          MOV     40(SP),-(SP)           ; SAV.NIB,*
071520  012746  006650                          MOV     #FMT.5,-(SP)
071524  012746  000003                          MOV     #3,-(SP)
071530  010600                                  MOV     SP,R0                  ; SP,*
071532  104414                                  TRAP    14
071534  012701  000001                          MOV     #1,R1                  ; *,DODU.FLG                                     8836
071540  062706  000032                          ADD     #32,SP                 ;                                                8830
071544  005003                          8$:     CLR     R3                     ; COUNT                                          8839
071546  020366  000006                  9$:     CMP     R3,6(SP)               ; COUNT,ALL.ONES.2                               8841
071552  001463                                  BEQ     10$
071554  005203                                  INC     R3                     ; COUNT                                          8843
071556  005202                                  INC     R2                     ; STK.PTR                                        8844
071560  126227  015034  000017                  CMPB    STACK(R2),#17          ; *(STK.PTR),*                                   8846
071566  001767                                  BEQ     9$
071570  005267  123602                          INC     P.CNT                  ;                                                8848
071574  026767  123576  123576                  CMP     P.CNT,LIMIT
071602  003047                                  BGT     10$
071604  104455                                  TRAP    55                     ;                                                8850
071606  000134                                  .WORD   134
071610  012750                                  .WORD   SYNC
071612  026302                                  .WORD   DUMPER
071614  012746  010730                          MOV     #WRD.19,-(SP)          ;                                                8851
071620  012746  012216                          MOV     #FNC.5,-(SP)
071624  012746  010650                          MOV     #WRD.12,-(SP)
071630  012746  011676                          MOV     #PHR.4,-(SP)
071634  012746  011140                          MOV     #WRD.39,-(SP)
071640  012746  010766                          MOV     #WRD.23,-(SP)
071644  012746  010322                          MOV     #SIX.FMT,-(SP)
071650  012746  000007                          MOV     #7,-(SP)
071654  010600                                  MOV     SP,R0                  ; SP,*
071656  104414                                  TRAP    14
071660  010216                                  MOV     R2,(SP)                ; STK.PTR,*                                      8852
071662  005046                                  CLR     -(SP)
071664  116216  015034                          MOVB    STACK(R2),(SP)         ; *(STK.PTR),*
071670  012746  177777                          MOV     #-1,-(SP)
071674  012746  006650                          MOV     #FMT.5,-(SP)
071700  012746  000004                          MOV     #4,-(SP)
071704  010600                                  MOV     SP,R0                  ; SP,*
071706  104414                                  TRAP    14
071710  012701  000001                          MOV     #1,R1                  ; *,DODU.FLG                                     8853
071714  062706  000030                          ADD     #30,SP                 ;                                                8848
071720  000712                                  BR      9$                     ;                                                8841
071722  006366  000004          10$:    ASL     4(SP)                  ; NIB.BIT                                        8858
071726  005205                                  INC     R5                     ; BY.ONE.WRD                                     88C4
071730  020527  000003                          CMP     R5,#3                  ; BY.ONE.WRD,*
```

```
                                        ;ML4AD
                                        ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS
                                                                                                            29-Mar-1982 16:21:03    PA:<
                                                 BGT     11$
                                                 JMP     5$
000004                          11$:     MOV     #1,4(SP)                ; *,NIB.BIT                                              8861
                                         INC     10(SP)                  ; ALL.ONES.1                                             8862
                                         DEC     6(SP)                   ; ALL.ONES.2                                             8863
                                         INC     R4                      ; BY.FOUR.WRDS                                           8801
                                         CMP     R4,#3                   ; BY.FOUR.WRDS,*
                                         BGT     12$
                                         JMP     4$
                                 12$:    CMP     (SP)+,(SP)+             ;                                                        8780
                                         TRAP    67                      ;                                                        8864
                                         ROR     R0
                                         BHIS    13$
                                         JMP     1$
                                 13$:    DEC     R1                      ; DODU.FLG                                               8868
                                         BNE     14$
                                         MOV     ML.LUN,R0               ;                                                        8871
                                         TRAP    51
                                         TRAP    44
                                 14$:    ADD     #12,SP                  ;                                                        8714
                                         RTS     PC

                                        ; Routine Size:  272 words
                                        ; Maximum stack depth per invocation:  26 words


                                                 .SBTTL  T38 TEST CODE SECTION
                                        T38::
                                 1$:     JSR     PC,$T38                 ;                                                        8873
                                         TRAP    66
                                         ROR     R0
                                         BLO     1$
                                         RTS     PC

                                        ; Routine Size:  6 words
                                        ; Maximum stack depth per invocation:  0 words
;          8876  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (91)

```
;ML4AD
;                    TEST CODE SECTION

                8877  !
                8878  BGNTST;
                8879
                8880  !++
                8881  ! TEST NUMBER:  TST 39
                8882  !
                8883  ! TEST NAME:    SYNC DATA BUS BIT UNIQUENESS TEST/READ PATH
                8884  !
                8885  ! TEST DESCRIPTION:
                8886  !       TEST SYNCHRONOUS DATA BUS READ
                8887  !       PATH FOR DATA BIT UNIQUENESS BY:
                8888  !
                8889  !       1.  LOADING THE FIRST 16 WORDS IN THE IO_BUF WITH A SHIFTING
                8890  !           ZERO IN A FIELD OF ONES PATTERN.
                8891  !
                8892  !       2.  VIA MBUS WRITE FUNCTION WRITE SHIFTING PATTERN INTO THE GOOD BLOCK.
                8893  !
                8894  !       3.  CLEAR THE IO_BUF.
                8895  !
                8896  !       4.  VIA MBUS READ FUNCTION READ THE SHIFTING PATTERN THROUGH THE
                8897  !           READ PATH.
                8898  !
                8899  !       5.  INTERIGATE THE IO_BUF FOR THE SHIFTING PATTERN.
                8900  !
                8901  ! IMPLICIT INPUTS:
                8902  !       IO_BUF
                8903  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE FUNCTION ARD FOUND.
                8904  !--
                8905
                8906  local
                8907      DODU_FLG,                       !DROP UNIT FLAG
                8908      TST_PAT;                        !TEST PATTERN
                8909
                8910  CLR_MBUS;
                8911  DODU_FLG = ZERO;
                8912  TST_PAT = ONE;                      !ONE IN A FIELD OF ZEROES
                8913
                8914  incr CNT from 0 to 15 do            !WRITE 16 WORDS WITH SHIFTED 0 IN A FIELD OF 1'S
                8915      begin
                8916      IO_BUF [.CNT] = not .TST_PAT;
                8917      TST_PAT = .TST_PAT^ONE;
                8918      end;
                8919
                8920  GD_BLK_XFER ();                     !SET UP A GOOD BLOCK XFERR
                8921  MLCS1 = write;                      !WRITE SHIFTING PATTERN
                8922  TIME_OUT_LOOP;
                8923  BGNSUB;
                8924
                8925  incr CNT from 0 to 15 do            !CLEAR OUT THE IO_BUF
                8926      IO_BUF [.CNT] = ZEROES;
                8927
                8928  CLR_MBUS;
```

```
 19928 ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 19929 ;                 TEST CODE SECTION                             29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (91)
 19930
 19931 ;          8929  GD_BLK_XFER ();                                !SET UP A GOOD BLOCK XFERR
 19932 ;          8930  ML[S1 = read;                                  !READ SHIFTING PATTERN THROUGH SYNC BUS
 19933 ;          8931  TIME_OUT_LOOP;
 19934 ;          8932  TST_PAT = ONE;                                 !SHIFTING PATTERN
 19935 ;          8933
 19936 ;          8934  incr CNT from 0 to 15 do                      !READ IO_BUF FOR SHIFTING 0 IN FIELD OF 1'S
 19937 ;          8935      begin
 19938 ;          8936
 19939 ;          8937      if .IO_BUF [.CNT] neq ( not .TST_PAT)     !COMPARE IO_BUF TO SHIFTED PAT
 19940 ;          8938      then
 19941 ;          8939          begin                                 !ERROR AND SET DODU_FLG IF NEQ
 19942 ;          8940          ERRDF (93, SYNC, DUMPER);
 19943 ;          8941          PRINTB (FIV_FMT, WRD_23, FNC_6, WRD_19, WRD_39, PHR_4);
 19944 ;          8942          PRINTB (FMT_2, ( not .TST_PAT), .IO_BUF [.CNT], ( not .TST_PAT xor .IO_BUF [.CNT]));
 19945 ;          8943          DODU_FLG = ONE;
 19946 ;          8944          end;
 19947 ;          8945
 19948 ;          8946      TST_PAT = .TST_PAT^ONE;                   !SHIFT THE PATTERN AND REPEAT
 19949 ;          8947      end;
 19950 ;          8948
 19951 ;          8949  ENDSUB;
 19952 ;          8950
 19953 ;          8951  if .DODU_FLG IS_SET                           !DROP THIS UNIT IF DODU_FLG IS_SET
 19954 ;          8952  then
 19955 ;          8953      begin
 19956 ;          8954      DODU (.ML_LUN);
 19957 ;          8955      DOCLN;
 19958 ;          8956      end;
 19959 ;          8957
 19960 ;          8958  ENDTST;
 19964
 19965                                          .SBTTL  $T39 TEST CODE SECTION
 19969 072046  004167  112540                $T39:   JSR     R1,$SAVE5                       ;                       8875
 19970 072052  152777  000040  123364               BISB    #40,@ML.REG+40                  ;                       8908
 19971 072060  016705  123746                       MOV     ML.OUT,R5
 19972 072064  042705  177770                       BIC     #177770,R5
 19973 072070  142777  000007  123346               BICB    #7,@ML.REG+40
 19974 072076  150577  123342                       BISB    R5,@ML.REG+40
 19975 072102  005005                               CLR     R5                              ; DODU.FLG              8911
 19976 072104  012704  000001                       MOV     #1,R4                           ; *,TST.PAT             8912
 19977 072110  005000                               CLR     R0                              ; CNT                   8914
 19978 072112  010001                         1$:   MOV     R0,R1                           ; CNT,*                 8916
 19979 072114  006301                               ASL     R1
 19980 072116  010461  014022                       MOV     R4,IO.BUF(R1)                   ; TST.PAT,*
 19981 072122  005161  014022                       COM     IO.BUF(R1)
 19982 072126  006304                               ASL     R4                              ; TST.PAT               8917
```

SEQ 0418

```
19984                                   ;ML4AD
19985                                   ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
19986                                                                                                       29-Mar-1982 16:21:03    PA:<
19987 072130  005200                              INC     R0                      ; CNT                                           8914
19988 072132  020027  000017                      CMP     R0,#17                  ; CNT,*
19989 072136  003765                              BLE     1$
19990 072140  004767  126146                      JSR     PC,GD.BLK.XFER          ;                                       8920
19991 072144  012777  000061  123232              MOV     #61,@ML.REG             ;                                       8921
19992 072152  105777  123276          2$:         TSTB    @ML.REG+50
19993 072156  100375                              BPL     2$
19994 072160  104402                    3$:       TRAP    2                       ;                                       8922
19995 072162  005000                              CLR     R0                      ; CNT                                   8925
19996 072164  010001                    4$:       MOV     R0,R1                   ; CNT,*                                 8926
19997 072166  006301                              ASL     R1
19998 072170  005061  014022                      CLR     IO.BUF(R1)
19999 072174  005200                              INC     R0                      ; CNT                                   8925
20000 072176  020027  000017                      CMP     R0,#17                  ; CNT,*
20001 072202  003770                              BLE     4$
20002 072204  152777  000040  123232              BISB    #40,@ML.REG+40          ;                                       8926
20003 072212  016703  123614                      MOV     ML.DUT,R3
20004 072216  042703  177770                      BIC     #177770,R3
20005 072222  142777  000007  123214              BICB    #7,@ML.REG+40
20006 072230  150377  123210                      BISB    R3,@ML.REG+40
20007 072234  004767  126052                      JSR     PC,GD.BLK.XFER          ;                                       8929
20008 072240  012777  000071  123136              MOV     #71,@ML.REG             ;                                       8930
20009 072246  105777  123202          5$:         TSTB    @ML.REG+50
20010 072252  100375                              BPL     5$
20011 072254  012704  000001                      MOV     #1,R4                   ; *,TST.PAT                             8932
20012 072260  005002                              CLR     R2                      ; CNT                                   8934
20013 072262  010201                    6$:       MOV     R2,R1                   ; CNT,*                                 8937
20014 072264  006301                              ASL     R1
20015 072266  012703  014022                      MOV     #IO.BUF,R3
20016 072272  060103                              ADD     R1,R3
20017 072274  010401                              MOV     R4,R1                   ; TST.PAT,*
20018 072276  005101                              COM     R1
20019 072300  021301                              CMP     (R3),R1
20020 072302  001447                              BEQ     7$
20021 072304  104455                              TRAP    55                      ;                                       8940
20022 072306  000135                              .WORD   135
20023 072310  012750                              .WORD   SYNC
20024 072312  026302                              .WORD   DUMPER
20025 072314  012746  011676                      MOV     #PHR.4,-(SP)            ;                                       8941
20026 072320  012746  011140                      MOV     #WRD.36,-(SP)
20027 072324  012746  010730                      MOV     #WRD.19,-(SP)
20028 072330  012746  012226                      MOV     #FNC.6,-(SP)
20029 072334  012746  010766                      MOV     #WRD.23,-(SP)
20030 072340  012746  010304                      MOV     #FIV.FMT,-(SP)
20031 072344  012746  000006                      MOV     #6,-(SP)
20032 072350  010600                              MOV     SP,R0                   ; SP,*
20033 072352  104414                              TRAP    14
20034 072354  011316                              MOV     (R3),(SP)               ;                                       8942
20035 072356  010146                              MOV     R1,-(SP)
20036 072360  046616  000002                      BIC     2(SP),(SP)
20037 072364  040166  000002                      BIC     R1,2(SP)
20038 072370  052616                              BIS     (SP)+,(SP)
```

```
20040                                   ;ML4AD
20041                                   ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
20042                                                                                                       29-Mar-1982 16:21:03    PA:<
20043 072372  011346                             MOV     (R3),-(SP)
20044 072374  010146                             MOV     R1,-(SP)
20045 072376  012746  006506                     MOV     #FMT.2,-(SP)
20046 072402  012746  000004                     MOV     #4,-(SP)
20047 072406  010600                             MOV     SP,R0                          ; SP,*
20048 072410  104414                             TRAP    14
20049 072412  012705  000001                     MOV     #1,R5                          ; *,DODU.FLG                    8943
20050 072416  062706  000026                     ADD     #26,SP                         ;                               8939
20051 072422  006304                     7$:     ASL     R4                             ; TST.PAT                       8946
20052 072424  005202                             INC     R2                             ; CNT                           8934
20053 072426  020227  000017                     CMP     R2,#17                         ; CNT,*
20054 072432  003713                             BLE     6$
20055 072434  104467                             TRAP    67                             ;                               8947
20056 072436  006000                             ROR     R0
20057 072440  103647                             BLO     3$
20058 072442  005305                             DEC     R5                             ; DODU.FLG                      8951
20059 072444  001004                             BNE     8$
2006C 072446  016700  123356                     MOV     ML.LUN,R0                      ;                               8954
20061 072452  104451                             TRAP    51
20062 072454  104444                             TRAP    44
20063 072456  000207                     8$:     RTS     PC                             ;                               8875
20064
20065                                   ; Routine Size:  133 words
20066                                   ; Maximum stack depth per invocation:  17 words
20071
20072
20076
20077                                           .SBTTL  T39 TEST CODE SECTION
20081 072460                            T39::
20082 072460  004767  177362            1$:     JSR     PC,$T39                        ;                               8956
20083 072464  104466                             TRAP    66
20084 072466  006000                             ROR     R0
20085 072470  103773                             BLO     1$
20086 072472  000207                             RTS     PC
20087
20088                                   ; Routine Size:  6 words
20089                                   ; Maximum stack depth per invocation:  0 words
20094 ;          8959  !<BLF/PAGE>
```

```
20096 ;ML4AD
20097 ;                 TEST CODE SECTION
20098
20099 ;          8960  !
20100 ;          8961  !
20101 ;          8962  BGNTST;
20102 ;          8963
20103 ;          8964  !++
20104 ;          8965  ! TEST NUMBER:  TST 40
20105 ;          8966  !
20106 ;          8967  ! TEST NAME:    ARRAY ADDRESS MUX TEST
20107 ;          8968  !
20108 ;          8969  ! TEST DESCRIPTION:
20109 ;          8970  !
20110 ;          8971  !       TEST FOR UNIQUE MOS RAM ROW
20111 ;          8972  !       AND COLUMN ADDRESSING BY:
20112 ;          8973  !
20113 ;          8974  !       1.  FIRST FINDING A ERROR FREE
20114 ;          8975  !           16K OR 64K CHUNK OF MEMORY.
20115 ;          8976  !           THIS REPRESENTS ONE ROW OF
20116 ;          8977  !           EITHER 16K OR 64K MOS RAMS.
20117 ;          8978  !
20118 ;          8979  !       2.  WRITE A BACKGROUND OF ALL
20119 ;          8980  !           ONES INTO THE GOOD CHUNK
20120 ;          8981  !
20121 ;          8982  !       3.  WRITE ZEROES INTO THE FIRST
20122 ;          8983  !           BLOCK OF THE GOOD CHUNK.
20123 ;          8984  !
20124            8985  !       4.  READ REMAINING BLOCKS IN
20125 ;          8986  !           GOOD CHUNK FOR ONES.
20126 ;          8987  !
20127 ;          8988  !
20128 ;          8989  ! IMPLICIT INPUTS:
20129 ;          8990  !
20130 ;          8991  !       IO_BUF
20131 ;          8992  !
20132 ;          8993  !       A VECTOR OF 256 WORDS
20133 ;          8994  !       WHERE DATA FOR MBUS
20134 ;          8995  !       READ AND WRITE TRANSFERS
20135 ;          8996  !       CAN BE FOUND.
20136 ;          8997  !--
20137 ;          8998
20138 ;          8999  local
20139 ;          9000      DSA_ADRS,                       !DSA ADRS COUNTER
20140 ;          9001      FND_GD_CHK;                     !FOUND GOOD 16K/64K CHUNK FLAG
20141 ;          9002
20142 ;          9003  IO_BUF = ONES;                      !LOAD FIRST IO_BUF WORD WITH ONES
20143 ;          9004  DSA_ADRS = -.RAS_INC;              !REST DSA COUNT
20144 ;          9005  FND_GD_CHK = ZERO;                 !CLEAR FLAG
20145 ;          9006
20146 ;          9007  do                                !DO UNTIL FOUND GOOD CHUNK OR LBT
20147 ;          9008      begin
20148 ;          9009      CLR_MBUS;
20149 ;          9010      BAI = ONE;                      !SET ON FIRST IO BUF WORD
20150 ;          9011      ECC_DIS = ONE;                  !DISABLE ECC
```

```
20152 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
20153 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (92)
20154
20155 ;          9012      DSA_ADRS = .DSA_ADRS + .RAS_INC;      !INCREMENT DSA ADRS COUNTER
20156 ;          9013      MLWC = .W_C_SIZE;                     !16K OR 64K WORDS
20157 ;          9014      MLBA = IO_BUF;                        !LOAD UBUS ADRS
20158 ;          9015      MLDA = .DSA_ADRS;                     !LOAD DSA ADRS
20159 ;          9016      ML_FUNC = write;                      !DO A WRITE FUNCTION
20160 ;          9017      TIME_OUT_LOOP;
20161 ;          9018
20162 ;          9019      if .SC IS_NOT_SET                     !DID XFERR CAUSE AN SC
20163 ;          9020      then
20164 ;          9021          begin                             !XFERR WAS OK
20165 ;          9022          MLWC = .W_C_SIZE;                 !LOAD WORD COUNT
20166 ;          9023          MLBA = IO_BUF;                    !LOAD UBUS ADRS
20167 ;          9024          MLDA = .DSA_ADRS;                 !LOAD DSA ADRS
20168 ;          9025          ML_FUNC = WRT_CHK;                !DO A WRITE CHECK FUNCTION
20169 ;          9026          TIME_OUT_LOOP;
20170 ;          9027
20171 ;          9028          if .SC IS_NOT_SET                 !IS THIS CHUNK GOOD
20172 ;          9029          then
20173 ;          9030              FND_GD_CHK = ONE;             !YES SET FLG
20174 ;          9031
20175 ;          9032          end
20176 ;          9033
20177 ;          9034      end
20178 ;          9035  !
20179 ;          9036  !VER CZMLAD CHANGED TEST TO UNSIGNED TEST
20180 ;          9037  !
20181 ;          9038  until (.FND_GD_CHK IS_SET ) or (.DSA_ADRS eqlU .LST_ARR + .ARR_INC);
20182 ;          9039
20183 ;          9040                                            !REPEAT UNTIL FOUND GOOD CHUNCK OR AT LBT
20184 ;          9041
20185 ;          9042  !
20186 ;          9043  !VER CZMLAD CHANGED TEST TO UNSIGNED TEST
20187 ;          9044  !
20188 ;          9045  if .DSA_ADRS eqlU .LST_ARR + .ARR_INC
20189 ;          9046  then
20190 ;          9047      begin
20191 ;          9048      ERRDF (111, INTER, DUMPER);
20192 ;          9049      PRINTB (FIV_FMT, FNC_13, FNC_17, WRD_50, WRD_60, WRD_56);
20193 ;          9050      PRINTB (THR_FMT, WRD_14, PHR_10, FNC_15);
20194 ;          9051      end
20195 ;          9052  else                                      !A GOOD CHUNCK WAS FOUND CONTINUE TEST
20196 ;          9053      begin
20197 ;          9054      CLR_MBUS;
20198 ;          9055      BAI = ONE;
20199 ;          9056      ECC_DIS = ONE;                        !DISABLE ECC
20200 ;          9057      IO_BUF = ZEROES;                      !FIRST BLOCK IN CHUNCK GETS ZEROES
20201 ;          9058      MLDA = .DSA_ADRS;                     !LOAD DSA
20202 ;          9059      MLWC = not 255;                       !LOAD WORD COUNT
20203 ;          9060      MLBA = IO_BUF;                        !LOAD UBUS ADRS
20204 ;          9061      ML_FUNC = write;                      !DO A WRITE FUNCTION
20205 ;          9062      TIME_OUT_LOOP;
20206 ;          9063      CLR_MBUS;
```

```
 20208 ;MI4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 20209 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (92)
 20210
 20211 ;          9064      BAI = ONE;
 20212 ;          9065      IO_BUF = ONES;                       !READ REMAINING BLOCKS FOR ONES
 20213 ;          9066      ECC_DIS = ONE;
 20214 ;          9067      MLDA = .DSA_ADRS + 1;                !LOAD DSA WITH NEXT DSA INCREMENT
 20215 ;          9068      MLBA = IO_BUF;                       !LOAD  UBUS ADRS
 20216 ;          9069      MLWC = .W_C_SIZE + 256;              !LOAD WORD COUNT WITH LESS ONE BLOCK
 20217 ;          9070      ML_FUNC = WRT_CHK;                   !DO A WRITE CHECK FUNCTION
 20218 ;          9071      TIME_OUT_LOOP;
 20219 ;          9072
 20220 ;          9073      if .WCE IS_SET                       !WERE ANY BITS DISTURBED
 20221 ;          9074      then
 20222 ;          9075          begin                            !ERROR IF WRITE CHECK FOUND BAD DATA
 20223 ;          9076          ERRDF (112, ASYNC, DUMPER);
 20224 ;          9077          PRINTB (FOR_FMT, FNC_17, WRD_50, WRD_60, WRD_14);
 20225 ;          9078          PRINTB (FMT_9, (.MLDA - 1));
 20226 ;          9079          DODU (.ML_LUN);
 20227 ;          9080          DOCLN;
 20228 ;          9081          end;
 20229 ;          9082
 20230 ;          9083      end;
 20231 ;          9084
 20232 ;          9085  ENDTST;
 20236
 20237                                      .SBTTL  $T40 TEST CODE SECTION
 20241 072474  004167  112040          $T40: JSR     R1,$SAVE2                    ;                          8058
 20242 072500  012767  177777  121314        MOV     #-1,IO.BUF                   ;                          9003
 20243 072506  016702  122634                MOV     RAS.INC,R2                   ; *,DSA.ADRS               9004
 20244 072512  005402                        NEG     R2                           ; DSA.ADRS
 20245 072514  005000                        CLR     R0                           ; FND.GD.CHK               9005
 20246 072516  152777  000040  122720  1$:   BISB    #40,@ML.REG+40               ;                          9008
 20247 072524  016701  123302                MOV     ML.DUT,R1
 20248 072530  042701  177770                BIC     #177770,R1
 20249 072534  142777  000007  122702        BICB    #7,@ML.REG+40
 20250 072542  150177  122676                BISB    R1,@ML.REG+40
 20251 072546  152777  000010  122670        BISB    #10,@ML.REG+40               ;                          9010
 20252 072554  152777  000002  122742        BISB    #2,@ML.REG+120               ;                          9011
 20253 072562  066702  122560                ADD     RAS.INC,R2                   ; *,DSA.ADRS               9012
 20254 072566  016777  122552  122620        MOV     W.C.SIZE,@ML.REG+10          ;                          9013
 20255 072574  012777  014022  122622        MOV     #IO.BUF,@ML.REG+20           ;                          9014
 20256 072602  010277  122626                MOV     R2,@ML.REG+30                ; DSA.ADRS,*               9015
 20257 072606  142777  000077  122570        BICB    #77,@ML.REG                  ;                          9016
 20258 072614  152777  000061  122562        BISB    #61,@ML.REG
 20259 072622  105777  122626          2$:   TSTB    @ML.REG+50
 20260 072626  100375                        BPL     2$
 20261 072630  032777  100000  122546        BIT     #100000,@ML.REG              ;                          9019
 20262 072636  001027                        BNE     4$
```

```
                                                                                29-Mar-1982 16:23:04    TOPS
                                                                                29-Mar-1982 16:21:03    PA:<
                                ;ML4AD
                                ;               TEST CODE SECTION
20264
20265
20266
20267 072640 016777 122500 122546       MOV     W.C.SIZE,@ML.REG+10     ;                               9022
20268 072646 012777 014022 122550       MOV     #IO.BUF,@ML,REG+20      ;                               9023
20269 072654 010277 122554              MOV     R2,@ML.REG+30           ; DSA.ADRS,*                    9024
20270 072660 142777 000077 122516       BICB    #77,@ML.REG             ;                               9025
20271 072666 152777 000051 122510       BISB    #51,@ML.REG
20272 072674 105777 122554      3$:     TSTB    @ML.REG+50
20273 072700 100375                     BPL     3$
20274 072702 032777 100000 122474       BIT     #100000,@ML.REG         ;                               9028
20275 072710 001002                     BNE     4$
20276 072712 012700 000001              MOV     #1,R0                   ; *,FND.GD.CHK                  9030
20277 072716 020027 000001      4$:     CMP     R0,#1                   ; FND.GD.CHK,*                  9038
20278 072722 001406                     BEQ     5$
20279 072724 016701 121070              MOV     LST.ARR,R1
20280 072730 066701 121050              ADD     ARR.INC,R1
20281 072734 020201                     CMP     R2,R1                   ; DSA.ADRS,*
20282 072736 001267                     BNE     1$
20283 072740 016701 121054      5$:     MOV     LST.ARR,R1              ;                               9045
20284 072744 066701 121034              ADD     ARR.INC,R1
20285 072750 020201                     CMP     R2,R1                   ; DSA.ADRS,*
20286 072752 001043                     BNE     6$
20287 072754 104455                     TRAP    55                      ;                               9048
20288 072756 000157                     .WORD   157
20289 072760 013114                     .WORD   INTER
20290 072762 026302                     .WORD   DUMPER
20291 072764 012746 011344              MOV     #WRD.56,-(SP)           ;                               9049
20292 072770 012746 011400              MOV     #WRD.60,-(SP)
20293 072774 012746 011270              MOV     #WRD.50,-(SP)
20294 073000 012746 012404              MOV     #FNC.17,-(SP)
20295 073004 012746 012334              MOV     #FNC.13,-(SP)
20296 073010 012746 010304              MOV     #FIV.FMT,-(SP)
20297 073014 012746 000006              MOV     #6,-(SP)
20298 073020 010600                     MOV     SP,R0                   ; SP,*
20299 073022 104414                     TRAP    14
20300 073024 012716 012360              MOV     #FNC.15,(SP)            ;                               9050
20301 073030 012746 012006              MOV     #PHR.10,-(SP)
20302 073034 012746 010664              MOV     #WRD.14,-(SP)
20303 073040 012746 010256              MOV     #THR.FMT,-(SP)
20304 073044 012746 000004              MOV     #4,-(SP)
20305 073050 010600                     MOV     SP,R0                   ; SP,*
20306 073052 104414                     TRAP    14
20307 073054 062706 000026              ADD     #26,SP                  ;                               9047
20308 073060 000207                     RTS     PC                      ;                               9045
20309 073062 152777 000040 122354 6$:   BISB    #40,@ML.REG+40          ;                               9053
20310 073070 016701 122736              MOV     ML.DUT,R1
20311 073074 042701 177770              BIC     #177770,R1
20312 073100 142777 000007 122336       BICB    #7,@ML.REG+40
20313 073106 150177 122332              BISB    R1,@ML.REG+40
20314 073112 152777 000010 122324       BISB    #10,@ML.REG+40          ;                               9055
20315 073120 152777 000002 122376       BISB    #2,@ML.REG+120          ;                               9056
20316 073126 005067 120670              CLR     IO.BUF                  ;                               9057
20317 073132 010277 122276              MOV     R2,@ML.REG+30           ; DSA.ADRS,*                    9058
20318 073136 012777 177400 122250       MOV     #-400,@ML.REG+10        ;                               9059
```

SEQ 0424


```
20320                                   ;ML4AD
20321                                   ;               TEST CODE SECTION
20322
20323 073144  012777  014022  122252          MOV     #IO.BUF,@ML.REG+20      ;                       9060
20324 073152  142777  000077  122224          BICB    #77,@ML.REG             ;                       9061
20325 073160  152777  000061  122216          BISB    #61,@ML.REG
20326 073166  105777  122262           7$:    TSTB    @ML.REG+50
20327 073172  100375                          BPL     7$
20328 073174  152777  000040  122242          BISB    #40,@ML.REG+40          ;                       9062
20329 073202  016701  122624                  MOV     ML.DUT,R1
20330 073206  042701  177770                  BIC     #177770,R1
20331 073212  142777  000007  122224          BICB    #7,@ML.REG+40
20332 073220  150177  122220                  BISB    R1,@ML.REG+40
20333 073224  152777  000010  122212          BISB    #10,@ML.REG+40          ;                       9064
20334 073232  012767  177777  120562          MOV     #-1,IO.BUF              ;                       9065
20335 073240  152777  000002  122256          BISB    #2,@ML.REG+120          ;                       9066
20336 073246  010201                          MOV     R2,R1                   ; DSA.ADRS,*            9067
20337 073250  005201                          INC     R1
20338 073252  010177  122156                  MOV     R1,@ML.REG+30
20339 073256  012777  014022  122140          MOV     #IO.BUF,@ML.REG+20      ;                       9068
20340 073264  016702  122054                  MOV     W.C.SIZE,R2             ;                       9069
20341 073270  062702  000400                  ADD     #400,R2
20342 073274  010277  122114                  MOV     R2,@ML.REG+10
20343 073300  142777  000077  122076          BICB    #77,@ML.REG             ;                       9070
20344 073306  152777  000051  122070          BISB    #51,@ML.REG
20345 073314  105777  122134           8$:    TSTB    @ML.REG+50
20346 073320  100375                          BPL     8$
20347 073322  032777  040000  122114          BIT     #40000,@ML.REG+40       ;                       9073
20348 073330  001441                          BEQ     9$
20349 073332  104455                          TRAP    55                      ;                       9076
20350 073334  000160                          .WORD   160
20351 073336  012706                          .WORD   ASYNC
20352 073340  026302                          .WORD   DUMPER
20353 073342  012746  010664                  MOV     #WRD.14,-(SP)           ;                       9077
20354 073346  012746  011400                  MOV     #WRD.60,-(SP)
20355 073352  012746  011270                  MOV     #WRD.50,-(SP)
20356 073356  012746  012404                  MOV     #FNC.17,-(SP)
20357 073362  012746  010270                  MOV     #FCS.FMT,-(SP)
20358 073366  012746  000005                  MOV     #5,-(SP)
20359 073372  010600                          MOV     SP,(SP)                 ; SP,*
20360 073374  104414                          TRAP    14
20361 073376  017716  122032                  MOV     @ML.REG+30,(SP)         ;                       9078
20362 073402  005316                          DEC     (SP)
20363 073404  012746  007064                  MOV     #FMT.9,-(SP)
20364 073410  012746  000002                  MOV     #2,-(SP)
20365 073414  010600                          MOV     SP,R0                   ; SP,*
20366 073416  104414                          TRAP    14
20367 073420  016700  122404                  MOV     ML.LUN,R0               ;                       9079
20368 073424  104451                          TRAP    51
20369 073426  104444                          TRAP    44
20370 073430  062706  000020                  ADD     #20,SP                  ;                       9075
20371 073434  000207             9$:    RTS     PC                      ;                       8958
20372
20373                                   ; Routine Size:  241 words
20374                                   ; Maximum stack depth per invocation:  14 words
```

```
20376                                   ;ML4AD
20377                                   ;               TEST CODE SECTION
20378
20383
20384
20388
20389                                           .SBTTL  T40 TEST CODE SECTION
20393 073436                            T40::
20394 073436  004767  177032            1$:     JSR     PC,$T40                 ;
20395 073442  104466                            TRAP    66
20396 073444  006000                            ROR     R0
20397 073446  103773                            BLO     1$
20398 073450  000207                            RTS     PC
20399
20400                                   ; Routine Size:  6 words
20401                                   ; Maximum stack depth per invocation:  0 words
20406
20407
20408 ;          9086  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

9083

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (93)

```
20410 ;ML4AD
20411 ;                 TEST CODE SECTION
20412 ;
20413 ;          9087  !
20414 ;          9088  BGNTST;
20415 ;          9089
20416 ;          9090  !++
20417 ;          9091  ! TEST NUMBER:  TST 41
20418 ;          9092  !
20419 ;          9093  ! TEST NAME:    NIBBLE OFFSET TEST
20420 ;          9094  !
20421 ;          9095  ! TEST DESCRIPTION:
20422 ;          9096  !       TEST NIBBLE OFFSET COUNTERS TO OFFSET GOOD NIBBLE DATA A MAX
20423 ;          9097  !       OF 14 WORDS ON DETECTION OF ALL BAD NIBBLES BY:
20424 ;          9098  !
20425 ;          9099  !       1.  LOADING FIRST 2 1/4 WORDS OF THE IO_BUF WITH ZEROES AND THE
20426 ;          9100  !           REMAINING OF BUFFER WITH ONES.
20427 ;          9101  !
20428 ;          9102  !       2.  VIA DAT_DM MODE WRITE THE GOOD BLOCK WITH BACKGROUND
20429 ;          9103  !           ON ONES.
20430 ;          9104  !
20431 ;          9105  !       3.  VIA PROM R/W MODE FORCE ALL ARRAY NIBBLES BAD.
20432 ;          9106  !
20433 ;          9107  !       4.  VIA A MBUS WRITE FUNCTION LOAD IO_BUF INTO THE GOOD BLOCK.
20434 ;          9108  !
20435 ;          9109  !       5.  VIA DAT_DM MODE READ FIRST 15 ARRAY WORDS FOR ZEROES AND THE
20436 ;          9110  !           REMAINING WORDS FOR ONES.
20437 ;          9111  !
20438 ;          9112  ! IMPLICIT INPUTS:
20439 ;          9113  !       PD_TEMP
20440 ;          9114  !       A BIT VECTOR OF 16 BITS WHERE THE READ PROM DATA IS STORED AND
20441 ;          9115  !       ACCESSED FROM.
20442 ;          9116  !
20443 ;          9117  !       IO_BUF
20444 ;          9118  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE
20445 ;          9119  !       FUNCTIONS ARE FOUND.
20446 ;          9120  !--
20447 ;          9121
20448 ;          9122  local
20449 ;          9123      DODU_FLG,                       !DROP UNIT FLAG
20450 ;          9124      TST_PAT,                        !TEST PATTERN
20451 ;          9125      START,                          !STARTING WORD
20452 ;          9126      FINISH,                         !ENDING WORD
20453 ;          9127      ERR_FLG;                        !ERROR FLAG
20454 ;          9128
20455 ;          9129  CLR_THRESHOLD;                      !CLEAR ERROR PRINT THRESHOLD
20456 ;          9130  BGNSUB;
20457 ;          9131  CLR_MBUS;
20458 ;          9132  DODU_FLG = ZERO;
20459 ;          9133
20460 ;          9134  incr WD_CNT from 0 to 255 do       !LOAD IO_BUF WITH ONES
20461 ;          9135      IO_BUF [.WD_CNT] = ONES;
20462 ;          9136
20463 ;          9137  IO_BUF [0] = ZEROES;              !LOAD FIRST 2 1/4 WORDS WITH ZEROES
20464 ;          9138  IO_BUF [1] = ZEROES;
```

```
20466 ;ML4AD
20467 ;                 TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 BLiss-16 V2(212)
20468 ;                                                      29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (93)
20469 ;          9139   IO_BUF [2] = %o'177760';
20470 ;          9140   MLD1 = ONES;                         !LOAD DATA DIAG REGISTERS WITH ONES
20471 ;          9141   MLD2 = ONES;
20472 ;          9142   MLE2 = ONES;
20473 ;          9143   DAT_DM_XFER ();                      !SET UP A DATA DIAG MODE XFERR
20474 ;          9144   MLCS1 = write;                       !DO A WRITE XFERR
20475 ;          9145
20476 ;          9146   incr WD_CNT from 0 to 127 do         !LOAD BLOCK WITH BG PAT
20477 ;          9147       begin
20478 ;          9148       DELAY (ONE_US);
20479 ;          9149       DAT_CLK = ONE;
20480 ;          9150       end;
20481 ;          9151
20482 ;          9152   CLR_MBUS;
20483 ;          9153   WRT_PD (ONES, 19);                   !FORCE ALL NIBBLES BAD
20484 ;          9154   PROM_RW = ONE;                       !SET PROM READ WRITE
20485 ;          9155   GD_BLK_XFER ();                      !SET UP A GOOD BLOCK XFERR
20486 ;          9156   MLCS1 = write;                       !WRITE ZEROES IN FIRST BLOCK WORD LOCATION
20487 ;          9157                                       !OFFSETTING 14 NIBBLES WITH ZEROES ALSO.
20488 ;          9158   TIME_OUT_LOOP;
20489 ;          9159   CLR_MBUS;
20490 ;          9160   START = ZERO;                        !START AT THE FIRST BLOCK WORD
20491 ;          9161   FINISH = 14;                         !END AT THE 14'TH BLOCK WORD
20492 ;          9162   TST_PAT = ZEROES;                    !TEST FOR ZEROES IN FIRST 14 WORDS
20493 ;          9163   DAT_DM_XFER ();                      !SET UP A DATA DIAG MODE XFERR
20494 ;          9164   MLCS1 = read;                        !DO A READ FUNCTION
20495 ;          9165   DELAY (ONE_US);
20496 ;          9166
20497 ;          9167   incr TWICE from 0 to 1 do            !READ WORDS 0-14 FOR 0'S AND 15-126 FOR 1'S
20498 ;          9168       begin
20499 ;          9169
20500 ;          9170       incr WRD_CNT from .START to .FINISH do    !READ BLOCK WORDS FORM START TO FINISH
20501 ;          9171           begin
20502 ;          9172           PD_TEMP = .MLPD;             !GET PROM DATA
20503 ;          9173           DAT_CLK = ONE;               !CLOCK OUT THE DATA WORD
20504 ;          9174           DELAY (ONE_US);
20505 ;          9175           RD_LNG_WRD;                  !READ THE DATA DIAG REGISTERS
20506 ;          9176
20507 ;          9177           incr NIB_PTR from 0 to 8 do  !LOOK AT 9 NIBBLES
20508 ;          9178
20509 ;          9179               if .PD_TEMP [.NIB_PTR] IS_NOT_SET   !FIND GOOD NIBBLES
20510 ;          9180               then
20511 ;          9181                   begin
20512 ;          9182                   TST_LNG_WRD (.NIB_PTR, .TST_PAT, ERR_FLG);      !COMPARE NIBBLE WITH TST PAT
20513 ;          9183
20514 ;          9184                   if .ERR_FLG IS_SET           !SEE IF COMPARE FOUND AN ERROR
20515 ;          9185                   then
20516 ;          9186                       begin                    !ERROR AND SET DODU_FLG IF ERROR FLG SET
20517 ;          9187                       CMP_THRESHOLD;           !COMPARE ERROR PRINT THRESHOLD
20518 ;          9188                       ERRDF (94, ARR_DAT, DUMPER);
20519 ;          9189                       PRINTB (THR_FMT, WRD_41, WRD_46, WRD_10);
20520 ;          9190                       PRINTB (FMT_6, .NIB_PTR);
```

```
20522 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
20523 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (93)
20524
20525 ;          9191                       DODU_FLG = ONE;
20526 ;          9192                       end;
20527 ;          9193
20528 ;          9194                   end;
20529 ;          9195
20530 ;          9196               end;
20531 ;          9197
20532 ;          9198           TST_PAT = not .TST_PAT;          !NOW READ FOR ONES
20533 ;          9199           START = 15;                      !START A 15
20534 ;          9200           FINISH = 126;                    !END AT 126
20535 ;          9201           end
20536 ;          9202
20537 ;          9203   ENDSUB;
20538 ;          9204
20539 ;          9205   if .DODU_FLG IS_SET                      !DROP THIS UNIT IF DODU_FLG SET
20540 ;          9206   then
20541 ;          9207       begin
20542 ;          9208       DODU (.ML_LUN);
20543 ;          9209       DOCLN;
20544 ;          9210       end;
20545 ;          9211
20546 ;          9212   ENDTST;
20550
20551                                      .SBTTL  $T41 TEST CODE SECTION
20555 073452  004167  111134        $T41: JSR     R1,$SAVE5                    ;                    9085
20556 073456  162706  000012              SUB     #12,SP
20557 073462  005067  121710              CLR     P.CNT                        ;                    9127
20558 073466  104402              1$:     TRAP    2                            ;                    9129
20559 073470  152777  000040  121746      BISB    #40,@ML.REG+40               ;                    9130
20560 073476  016704  122330              MOV     ML.DUT,R4
20561 073502  042704  177770              BIC     #177770,R4
20562 073506  142777  000007  121730      BICB    #7,@ML.REG+40
20563 073514  150477  121724              BISB    R4,@ML.REG+40
20564 073520  005066  000004              CLR     4(SP)                        ; DODU.FLG           9132
20565 073524  005002                      CLR     R2                           ; WD.CNT             9134
20566 073526  010203              2$:     MOV     R2,R3                        ; WD.CNT,*           9135
20567 073530  006303                      ASL     R3
20568 073532  012763  177777  014022      MOV     #-1,IO.BUF(R3)
20569 073540  005202                      INC     R2                           ; WD.CNT             9134
20570 073542  020227  000377              CMP     R2,#377                      ; WD.CNT,*
20571 073546  003767                      BLE     2$
20572 073550  005067  120246              CLR     IO.BUF                       ;                    9137
20573 073554  005067  120244              CLR     IO.BUF+2                     ;                    9138
20574 073560  012767  177760  120240      MOV     #-20,IO.BUF+4                ;                    9139
20575 073566  012777  177777  122000      MOV     #-1,@ML.REG+170              ;                    9140
20576 073574  012777  177777  122002      MOV     #-1,@ML.REG+200              ;                    9141
```

```
20578                                   ;ML4AD
20579                                   ;                       TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
20580                                                                                                               29-Mar-1982 16:21:03    PA:<
20581 073602  012777  177777  121754          MOV     #-1,@ML.REG+160         ;                                                           9142
20582 073610  004767  122576                  JSR     PC,DAT.DM.XFER          ;                                                           9143
20583 073614  012777  000061  121562          MOV     #61,@ML.REG             ;                                                           9144
20584 073622  005001                          CLR     R1                      ; WD.CNT                                                    9146
20585 073624  012702  000001          3$:     MOV     #1,R2                   ; *,$$TMP2                                                  9148
20586 073630  001411          4$:     BEQ     7$
20587 073632  016703  106260                  MOV     L$DLY,R3                ; *,$$TMP1
20588 073636  001404                          BEQ     6$
20589 073640  005066  000010          5$:     CLR     10(SP)                  ; $$TMP
20590 073644  005303                          DEC     R3                      ; $$TMP1
20591 073646  001374                          BNE     5$
20592 073650  005302                  6$:     DEC     R2                      ; $$TMP2
20593 073652  000766                          BR      4$
20594 073654  152777  000020  121642  7$:     BISB    #20,@ML.REG+120         ;                                                           9149
20595 073662  005201                          INC     R1                      ; WD.CNT                                                    9146
20596 073664  020127  000177                  CMP     R1,#177                 ; WD.CNT,*
20597 073670  003755                          BLE     3$
20598 073672  152777  000040  121544          BISB    #40,@ML.REG+40          ;                                                           9150
20599 073700  016704  122126                  MOV     ML.DUT,R4
20600 073704  042704  177770                  BIC     #177770,R4
20601 073710  142777  000007  121526          BICB    #7,@ML.REG+40
20602 073716  150477  121522                  BISB    R4,@ML.REG+40
20603 073722  012746  177777                  MOV     #-1,-(SP)               ;                                                           9153
20604 073726  012746  000023                  MOV     #23,-(SP)
20605 073732  004767  127512                  JSR     PC,URT.PD
20606 073736  152777  000100  121560          BISB    #100,@ML.REG+120        ;                                                           9154
20607 073744  004767  124342                  JSR     PC,GD.BLK.XFER          ;                                                           9155
20608 073750  012777  000061  121426          MOV     #61,@ML.REG             ;                                                           9156
20609 073756  105777  121472          8$:     TSTB    @ML.REG+50
20610 073762  100375                          BPL     8$
20611 073764  152777  000040  121452          BISB    #40,@ML.REG+40          ;                                                           9158
20612 073772  016704  122034                  MOV     ML.DUT,R4
20613 073776  042704  177770                  BIC     #177770,R4
20614 074002  142777  000007  121434          BICB    #7,@ML.REG+40
20615 074010  150477  121430                  BISB    R4,@ML.REG+40
20616 074014  005066  000006                  CLR     6(SP)                   ; START                                                     9160
20617 074020  012766  000016  000004          MOV     #16,4(SP)               ; *,FINISH                                                  9161
20618 074026  005005                          CLR     R5                      ; TST.PAT                                                   9162
20619 074030  004767  122356                  JSR     PC,DAT.DM.XFER          ;                                                           9163
20620 074034  012777  000071  121342          MOV     #71,@ML.REG             ;                                                           9164
20621 074042  012702  000001                  MOV     #1,R2                   ; *,$$TMP2                                                  9165
20622 074046  001411          9$:     BEQ     12$
20623 074050  016703  106042                  MOV     L$DLY,R3                ; *,$$TMP1
20624 074054  001404                          BEQ     11$
20625 074056  005066  000014          10$:    CLR     14(SP)                  ; $$TMP
20626 074062  005303                          DEC     R3                      ; $$TMP1
20627 074064  001374                          BNE     10$
20628 074066  005302                  11$:    DEC     R2                      ; $$TMP2
20629 074070  000766                          BR      9$
20630 074072  005004                  12$:    CLR     R4                      ; TWICE                                                     9167
20631 074074  016601  000006          13$:    MOV     6(SP),R1                ; START,WRD.CNT                                             9170
20632 074100  005301                          DEC     R1                      ; WRD.CNT
```

```
20634                                          ;ML4AD
20635                                          ;              TEST CODE SECTION
20636
20637 074102  000545                                  BR      23$
20638 074104  017767  121524  121230  14$:    MOV     @ML.REG+230,PD.TEMP                    ;                                  9172
20639 074112  152777  000020  121404          BISB    #20,@ML.REG+120                        ;                                  9173
20640 074120  012702  000001                  MOV     #1,R2                                  ; *,$$TMP2                         9174
20641 074124  001411                  15$:    BEQ     18$
20642 074126  016703  105764                  MOV     L$DLY,R3                               ; *,$$TMP1
20643 074132  001404                          BEQ     17$
20644 074134  005066  000014          16$:    CLR     14(SP)                                 ; $$TMP
20645 074140  005303                          DEC     R3                                     ; $$TMP1
20646 074142  001374                          BNE     16$
20647 074144  005302                  17$:    DEC     R2                                     ; $$TMP2
20648 074146  000766                          BR      15$
20649 074150  017767  121420  117216  18$:    MOV     @ML.REG+170,D1.TEMP
20650 074156  017767  121422  117212          MOV     @ML.REG+200,D2.TEMP
20651 074164  017767  121374  117206          MOV     @ML.REG+160,E2.TEMP
20652 074172  005002                          CLR     R2                                     ; NIB.PTR                          9177
20653 074174  010203                  19$:    MOV     R2,R3                                  ; NIB.PTR,-                        9179
20654 074176  006203                          ASR     R3
20655 074200  006203                          ASR     R3
20656 074202  006203                          ASR     R3
20657 074204  062703  015342                  ADD     #PD.TEMP,R3
20658 074210  010346                          MOV     R3,-(SP)
20659 074212  010246                          MOV     R2,-(SP)                               ; NIB.PTR,*
20660 074214  042716  177770                  BIC     #177770,(SP)
20661 074220  012746  000001                  MOV     #1,-(SP)
20662 074224  005046                          CLR     -(SP)
20663 074226  004767  107402                  JSR     PC,BL$GT2
20664 074232  062706  000010                  ADD     #10,SP
20665 074236  005700                          TST     R0
20666 074240  001062                          BNE     22$
20667 074242  010246                          MOV     R2,-(SP)                               ; NIB.PTR,*                        9182
20668 074244  010546                          MOV     R5,-(SP)                               ; TST.PAT,*
20669 074246  012746  000020                  MOV     #20,-(SP)
20670 074252  060616                          ADD     SP,(SP)                                ; ERR.FLG,*
20671 074254  004767  124116                  JSR     PC,TST.LNG.WRD
20672 074260  026627  000020  000001          CMP     20(SP),#1                              ; ERR.FLG,*                        9184
20673 074266  001045                          BNE     21$
20674 074270  005267  121102                  INC     P.CNT                                  ;                                  9186
20675 074274  026767  121076  121076          CMP     P.CNT,LIMIT
20676 074302  003403                          BLE     20$
20677 074304  062706  000006                  ADD     #6,SP
20678 074310  000442                          BR      23$
20679 074312  104455                  20$:    TRAP    55                                     ;                                  9188
20680 074314  000136                          .WORD   136
20681 074316  013012                          .WORD   ARR.DAT
20682 074320  026302                          .WORD   DUMPER
20683 074322  012746  010630                  MOV     #WRD.10,-(SP)                          ;                                  9189
20684 074326  012746  011226                  MOV     #WRD.46,-(SP)
20685 074332  012746  011154                  MOV     #WRD.41,-(SP)
20686 074336  012746  010256                  MOV     #THR.FMT,-(SP)
20687 074342  012746  000004                  MOV     #4,-(SP)
20688 074346  010600                          MOV     SP,R0                                  ; SP,*
```

```
                                        ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
                                                TRAP    14
074350  104414                                  MOV     R2,(SP)                 ; NIB.PTR,*                             9190
074352  010216                                  MOV     #FMT.6,-(SP)
074354  012746  006752                          MOV     #2,-(SP)
074360  012746  000002                          MOV     SP,R0                   ; SP,*
074364  010600                                  TRAP    14
074366  104414
074370  012766  000001  000034                  MOV     #1,34(SP)               ; *,DODU.FLG                            9191
074376  062706  000016                          ADD     #16,SP                  ;                                       9186
074402  062706  000006          21$:            ADD     #6,SP                   ;                                       9181
074406  005202                  22$:            INC     R2                      ; NIB.PTR                               9177
074410  020227  000010                          CMP     R2,#10                  ; NIB.PTR,*
074414  003667                                  BLE     19$
074416  005201                  23$:            INC     R1                      ; WRD.CNT                               9170
074420  020166  000004                          CMP     R1,4(SP)                ; WRD.CNT,FINISH
074424  003627                                  BLE     14$
074426  005105                                  COM     R5                      ; TST.PAT                               9198
074430  012766  000017  000006                  MOV     #17,6(SP)               ; *,START                               9199
074436  012766  000176  000004                  MOV     #176,4(SP)              ; *,FINISH                              9200
074444  005204                                  INC     R4                      ; TWICE                                 9167
074446  020427  000001                          CMP     R4,#1                   ; TWICE,*
074452  003610                                  BLE     13$
074454  022626                                  CMP     (SP)+,(SP)+             ;                                       9129
074456  104467                                  TRAP    67                      ;                                       9201
074460  006000                                  ROR     R0
074462  103002                                  BHIS    24$
074464  000167  176776                          JMP     1$
074470  026627  000004  000001  24$:            CMP     4(SP),#1                ; DODU.FLG,*                            9205
074476  001004                                  BNE     25$
074500  016700  121324                          MOV     ML.LUN,R0               ;                                       9208
074504  104451                                  TRAP    51
074506  104444                                  TRAP    44
074510  062706  000012          25$:            ADD     #12,SP                  ;                                       9085
074514  000207                                  RTS     PC
                                ; Routine Size:  274 words
                                ; Maximum stack depth per invocation:  23 words
                                        .SBTTL  T41 TEST CODE SECTION
074516                          T41::
074516  004767  176730          1$:     JSR     PC,$T41                         ;                                       9210
```

```
20746                                   ;ML4AD
20747                                   ;                TEST CODE SECTION
20748
20749 074522  104466                              TRAP    66
20750 074524  006000                              ROR     R0
20751 074526  103773                              BLO     1$
20752 074530  000207                              RTS     PC
20753
20754                                   ; Routine Size:  6 words
20755                                   ; Maximum stack depth per invocation:  0 words
20760
20761
20762 ;          9213   !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (94)

```
20764 ;ML4AD
20765 ;                 TEST CODE SECTION
20766
20767 ;          9214   !
20768 ;          9215   BGNTST;
20769 ;          9216
20770 ;          9217   !++
20771 ;          9218   ! TEST NUMBER:  TST 42
20772 ;          9219   !
20773 ;          9220   ! TEST NAME:    CS1 FUNCTION ABORT TEST
20774 ;          9221   !
20775 ;          9222   ! TEST DESCRIPTION:
20776 ;          9223   !       TEST CS1 FUNCTION ABORTS ON DETECTION OF CLASS 'A' & 'B' ERRORS BY:
20777 ;          9224   !
20778 ;          9225   !       1.  VIA MBUS WRITE FUNCTION LOAD THE GOOD BLOCK WITH BACKGROUND
20779 ;          9226   !           PATTERN OF ONES.
20780 ;          9227   !
20781 ;          9228   !       2.  CLEAR THE IO_BUF
20782 ;          9229   !
20783 ;          9230   !       3.  DO A MBUS READ FUNCTION.  WHILE THE READ IS IN PROGRESS WRITE TO
20784 ;          9231   !           MLDA (CLASS 'A' ERROR) READ THE IO_BUF FOR ONES.
20785 ;          9232   !
20786 ;          9233   !       4.  CLEAR THE IO_BUF
20787 ;          9234   !
20788 ;          9235   !       5.  VIA PROM R/W MODE FORCE UV ERROR TO THE UV ADRS ERROR PROM
20789 ;          9236   !           (CLASS 'B' ERROR).  DO A MBUS READ.
20790 ;          9237   !
20791 ;          9238   !       6.  READ IO_BUF FOR ZEROES
20792 ;          9239   !
20793 ;          9240   ! IMPLICIT INPUTS:
20794 ;          9241   !       IO_BUF
20795 ;          9242   !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND
20796 ;          9243   !       WRITE FUNCTION ARE FOUND.
20797 ;          9244   !--
20798 ;          9245
20799 ;          9246   CLR_THRESHOLD;                          !CLEAR ERROR PRINT THRESHOLD
20800 ;          9247   CLR_MBUS;
20801 ;          9248   BAI = ONE;                              !SET ON FIRST IO_BUF ADRS
20802 ;          9249   IO_BUF = ONES;                          !LOAD FIRST IO_BUF ADRS
20803 ;          9250   GD_BLK_XFER ();                         !SET UP A GOOD BLOCK XFERR
20804 ;          9251   MLCS1 = write;                          !WRITE BACKGROUND PATTERN
20805 ;          9252   TIME_OUT_LOOP;
20806 ;          9253
20807 ;          9254   incr TWICE from 0 to 1 do               !FORCE CLASS 'A' AND CLASS 'B' ERRORS
20808 ;          9255       begin
20809 ;          9256       BGNSUB;
20810 ;          9257       CLR_MBUS;
20811 ;          9258
20812 ;          9259       incr CNT from 0 to 255 do
20813 ;          9260           IO_BUF [.CNT] = ZEROES;         !CLEAR OUT IO_BUF
20814 ;          9261
20815 ;          9262       if .TWICE eql 1                     !IF 2ND PASS THEN FORCE 'B' ERROR
20816 ;          9263       then
20817 ;          9264           begin
20818 ;          9265           PROM_RW = ONE;
```

```
20820 ;ML4AD
20821 ;                 TEST CODE SECTION
20822
20823 ;          9266           MLPD = %o'777';
20824 ;          9267           end;
20825 ;          9268
20826 ;          9269       GD_BLK_XFER ();                          !SET UP A GOOD BLOCK XFERR
20827 ;          9270       MLCS1 = read;                            !DO A READ FUNCTION
20828 ;          9271
20829 ;          9272       if .TWICE eql 0 then MLDA = ONES;        !IF FIRST PASS THEN FORCE AN 'A' ERROR
20830 ;          9273
20831 ;          9274       TIME_OUT_LOOP;
20832 ;          9275
20833 ;          9276       if .TWICE eql 0                          !SEE WHICH PASS WE'RE ON
20834 ;          9277       then
20835 ;          9278           begin                                !CLASS 'A' ERROR
20836 ;          9279
20837 ;          9280           incr WRD_CNT from 0 to 64 do         !SEE IF XFERR WAY ALLOWED TO CONTINUE
20838 ;          9281               begin
20839 ;          9282
20840 ;          9283               if .IO_BUF [.WRD_CNT] neq ONES   !READ IO_BUF FOR BG PAT
20841 ;          9284               then
20842 ;          9285                   begin                        !ERROR AND EXIT LOOP IF ZEROES
20843 ;          9286                   CMP_THRESHOLD;               !COMPARE ERROR PRINT THRESHOLD
20844 ;          9287                   ERRDF (95, SYNC, DUMPER);
20845 ;          9288                   PRINTB (FOR_FMT, PHR_8, FNC_13, WRD_19, WRD_10);
20846 ;          9289                   exitloop;
20847 ;          9290                   end;
20848 ;          9291
20849 ;          9292               end;
20850 ;          9293
20851 ;          9294           end
20852 ;          9295       else                                     !CLASS 'B' ERROR
20853 ;          9296           begin
20854 ;          9297
20855 ;          9298           incr WRD_CNT from 0 to 64 do         !SEE IF XFERR WAS ABORTED
20856 ;          9299               begin
20857 ;          9300
20858 ;          9301               if .IO_BUF [.WRD_CNT] neq ZEROES !READ IO_BUF CLEARED DATA
20859 ;          9302               then
20860 ;          9303                   begin                        !ERROR IF ONES AND EXIT LOOP
20861 ;          9304                   CMP_THRESHOLD;               !COMPARE ERROR PRINT THRESHOLD
20862 ;          9305                   ERRDF (96, SYNC, DUMPER);
20863 ;          9306                   PRINTB (FOR_FMT, PHR_9, FNC_13, WRD_19, WRD_10);
20864 ;          9307                   exitloop
20865 ;          9308                   end;
20866 ;          9309
20867 ;          9310               end;
20868 ;          9311
20869 ;          9312           end;
20870 ;          9313
20871 ;          9314       if .SC IS_NOT_SET                        !SEE IF SC BIT SET
20872 ;          9315       then
20873 ;          9316           begin                                !ERROR IF NOT SET
20874 ;          9317           ERRDF (104, SYNC, DUMPER);
```

```
20876 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
20877 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (94)
20878
20879 ;          9318          PRINTB (FIV_FMT, WRD_59, PHR_1, WRD_11, WRD_19, FNC_13)
20880 ;          9319          end;
20881 ;          9320
20882 ;          9321      ENDSUB;
20883 ;          9322      end;
20884 ;          9323
20885 ;          9324  ENDTST;
20889
20890                                        .SBTTL  $T42 TEST CODE SECTION
20894 074532  004167  110016          $T42:  JSR     R1,$SAVE3                        ;                              9212
20895 074536  005067  120634                 CLR     P.CNT                            ;                              9215
20896 074542  152777  000040  120674        BISB    #40,@ML.REG+40                   ;                              9246
20897 074550  016703  121256                MOV     ML.DUT,R3
20898 074554  042703  177770                BIC     #177770,R3
20899 074560  142777  000007  120656        BICB    #7,@ML.REG+40
20900 074566  150377  120652                BISB    R3,@ML.REG+40
20901 074572  152777  000010  120644        BISB    #10,@ML.REG+40                   ;                              9248
20902 074600  012767  177777  117214        MOV     #-1,IO.BUF                       ;                              9249
20903 074606  004767  123500                JSR     PC,GD.BLK.XFER                   ;                              9250
20904 074612  012777  000061  120564        MOV     #61,@ML.REG                      ;                              9251
20905 074620  105777  120630          1$:   TSTB    @ML.REG+50
20906 074624  100375                       BPL     1$
20907 074626  005003                       CLR     R3                               ; TWICE                        9254
20908 074630  104402                 2$:   TRAP    2                                ;                              9255
20909 074632  152777  000040  120604        BISB    #40,@ML.REG+40                   ;                              9256
20910 074640  016702  121166                MOV     ML.DUT,R2
20911 074644  042702  177770                BIC     #177770,R2
20912 074650  142777  000007  120566        BICB    #7,@ML.REG+40
20913 074656  150277  120562                BISB    R2,@ML.REG+40
20914 074662  005000                       CLR     R0                               ; CNT                          9259
20915 074664  010001                 3$:   MOV     R0,R1                            ; CNT,*                        9260
20916 074666  006301                       ASL     R1
20917 074670  005061  014022                CLR     IO.BUF(R1)
20918 074674  005200                       INC     R0                               ; CNT                          9259
20919 074676  020027  000377                CMP     R0,#377                          ; CNT,*
20920 074702  003770                       BLE     3$
20921 074704  020327  000001                CMP     R3,#1                            ; TWICE,*                      9262
20922 074710  001006                       BNE     4$
20923 074712  152777  000100  120604        BISB    #100,@ML.REG+120                 ;                              9265
20924 074720  012777  000777  120706        MOV     #777,@ML.REG+230                 ;                              9266
20925 074726  004767  123360          4$:   JSR     PC,GD.BLK.XFER                   ;                              9269
20926 074732  012777  000071  120444        MOV     #71,@ML.REG                      ;                              9270
20927 074740  005002                       CLR     R2                               ;                              9272
20928 074742  005703                       TST     R3                               ; TWICE
20929 074744  001004                       BNE     5$
20930 074746  005202                       INC     R2
```

```
                                    ;ML4AD
                                    ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04   TOPS
                                                                                                        29-Mar-1982 16:21:03   PA:<
20935 074750  012777  177777  120456          MOV     #-1,@ML.REG+30
20936 074756  105777  120472          5$:     TSTB    @ML.REG+50
20937 074762  100375                          BPL     5$
20938 074764  006002                          ROR     R2                                                                      9276
20939 074766  103045                          BCC     8$
20940 074770  005002                          CLR     R2                                 ; WRD.CNT                            9280
20941 074772  010201                  6$:     MOV     R2,R1                              ; WRD.CNT,*                          9283
20942 074774  006301                          ASL     R1
20943 074776  026127  014022  177777          CMP     IO.BUF(R1),#-1
20944 075004  001431                          BEQ     7$
20945 075006  005267  120364                  INC     P.CNT                                                                   9285
20946 075012  026767  120360  120360          CMP     P.CNT,LIMIT
20947 075020  003075                          BGT     12$
20948 075022  104455                          TRAP    55                                                                      9287
20949 075024  000137                          .WORD   137
20950 075026  012750                          .WORD   SYNC
20951 075030  026302                          .WORD   DUMPER
20952 075032  012746  010630                  MOV     #WRD.10,-(SP)                                                           9288
20953 075036  012746  010730                  MOV     #WRD.19,-(SP)
20954 075042  012746  012334                  MOV     #FNC.13,-(SP)
20955 075046  012746  011762                  MOV     #PHR.8,-(SP)
20956 075052  012746  010270                  MOV     #FOR.FMT,-(SP)
20957 075056  012746  000005                  MOV     #5,-(SP)
20958 075062  010600                          MOV     SP,R0                              ; SP,*
20959 075064  104414                          TRAP    14
20960 075066  000443                          BR      10$                                                                     9289
20961 075070  005202                  7$:     INC     R2                                 ; WRD.CNT                            9280
20962 075072  020227  000100                  CMP     R2,#100                            ; WRD.CNT,*
20963 075076  003735                          BLE     6$
20964 075100  000445                          BR      12$                                                                     9276
20965 075102  005002                  8$:     CLR     R2                                 ; WRD.CNT                            9298
20966 075104  010201                  9$:     MOV     R2,R1                              ; WRD.CNT,*                          9301
20967 075106  006301                          ASL     R1
20968 075110  005761  014022                  TST     IO.BUF(R1)
20969 075114  001433                          BEQ     11$
20970 075116  005267  120254                  INC     P.CNT                                                                   9303
20971 075122  026767  120250  120250          CMP     P.CNT,LIMIT
20972 075130  003031                          BGT     12$
20973 075132  104455                          TRAP    55                                                                      9305
20974 075134  000140                          .WORD   140
20975 075136  012750                          .WORD   SYNC
20976 075140  026302                          .WORD   DUMPER
20977 075142  012746  010630                  MOV     #WRD.10,-(SP)                                                           9306
20978 075146  012746  010730                  MOV     #WRD.19,-(SP)
20979 075152  012746  012334                  MOV     #FNC.13,-(SP)
20980 075156  012746  011774                  MOV     #PHR.9,-(SP)
20981 075162  012746  010270                  MOV     #FOR.FMT,-(SP)
20982 075166  012746  000005                  MOV     #5,-(SP)
20983 075172  010600                          MOV     SP,R0                              ; SP,*
20984 075174  104414                          TRAP    14
20985 075176  062706  000014          10$:    ADD     #14,SP                                                                  9307
20986 075202  000404                          BR      12$
```

```
                                        ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
                                        11$:    INC     R2                      ; WRD.CNT                               9298
                                                CMP     R2,#100                 ; WRD.CNT,*
                                                BLE     9$
                                        12$:    BIT     #100000,@ML.REG         ;                                       9314
                                                BNE     13$
                                                TRAP    55                      ;                                       9317
                                                .WORD   150
                                                .WORD   SYNC
                                                .WORD   DUMPER
                                                MOV     #FNC.13,-(SP)           ;                                       9318
                                                MOV     #WRD.19,-(SP)
                                                MOV     #WRD.11,-(SP)
                                                MOV     #PHR.1,-(SP)
                                                MOV     #WRD.59,-(SP)
                                                MOV     #FIV.FMT,-(SP)
                                                MOV     #6,-(SP)
                                                MOV     SP,R0                   ; SP,*
                                                TRAP    14
                                                ADD     #16,SP                  ;                                       9316
                                        13$:    TRAP    67                      ;                                       9319
                                                ROR     R0
                                                BHIS    15$
                                        14$:    JMP     2$
                                        15$:    INC     R3                      ; TWICE                                 9254
                                                CMP     R3,#1                   ; TWICE,*
                                                BLE     14$
                                                RTS     PC                      ;                                       9212

                                        ; Routine Size:  189 words
                                        ; Maximum stack depth per invocation:  11 words


                                                .SBTTL  T42 TEST CODE SECTION
                                        T42::
                                        1$:     JSR     PC,$T42                 ;                                       9322
                                                TRAP    66
                                                ROR     R0
                                                BLO     1$
                                                RTS     PC

                                        ; Routine Size:  6 words
```

```
21044                                   ;ML4AD
21045                                   ;                    TEST CODE SECTION
21046
21047                                   ; Maximum stack depth per invocation:  0 words
21052
21053
21054 ;          9325  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (95)

```
21056 ;ML4AD
21057 ;                 TEST CODE SECTION
21058
21059 ;          9326  !
21060 ;          9327  !
21061 ;          9328  BGNTST;
21062 ;          9329
21063 ;          9330  !++
21064 ;          9331  ! TEST NUMBER:  TST 43
21065 ;          9332  !
21066 ;          9333  ! TEST NAME:    LAST BLOCK TRANSFER TEST
21067 ;          9334  !
21068 ;          9335  ! TEST DESCRIPTION:
21069 ;          9336  !
21070 ;          9337  !       TEST THE LAST BLOCK INDICATOR BIT
21071 ;          9338  !       FOR SETTING/NOT SETTING AND THE
21072 ;          9339  !       DSA REGISTER FOR INCREMENTING BY:
21073 ;          9340  !
21074 ;          9341  !       1.  DOING MBUS TRANSFERRS AT EACH
21075 ;          9342  !           BLOCK FROM BLOCK ZERO TO
21076 ;          9343  !           LAST BLOCK -1 AND TEST LBT
21077 ;          9344  !           CLEAR AND DSA REGISTER TO BE
21078 ;          9345  !           INCREMENTED.
21079 ;          9346  !
21080 ;          9347  !       2.  DO A MBUS TRANSFERR AT
21081 ;          9348  !           THE LAST BLOCK.
21082 ;          9349  !           TEST LBT TO BE SET AND
21083 ;          9350  !           TEST FOR CLEAR.
21084 ;          9351  !           TEST DSA REG TO BE INCREMENTED.
21085 ;          9352  !
21086 ;          9353  ! IMPLICIT INPUTS:
21087 ;          9354  !       IO_BUF
21088 ;          9355  !       A VECTOR OF 256 WORDS WHERE
21089 ;          9356  !       DATA FOR MBUS READS AND WRITE
21090 ;          9357  !       FUNCTION ARE FOUND.
21091 ;          9358  !
21092 ;          9359  !--
21093 ;          9360
21094 ;          9361  CLR_THRESHOLD;                          !CLEAR ERROR PRINT THRESHOLD
21095 ;          9362
21096 ;          9363  !
21097 ;          9364  !VERSION CZMLAD CHANGED TEST TO UNSIGNED TEST
21098 ;          9365  !
21099 ;          9366  incrU DSA_CNT from 0 to .LST_BLK - 1 do          !DO XFERRS UP TO THE LAST BLOCK
21100 ;          9367      begin
21101 ;          9368      BGNSUB;
21102 ;          9369      CLR_MBUS;
21103 ;          9370      ECC_DIS = ONE;                       !DISABLE ECC
21104 ;          9371      MLWC = not 255;                      !LOAD WORD COUNT
21105 ;          9372      MLBA = IO_BUF;                       !LOAD UBUS ADRS
21106 ;          9373      MLDA = .DSA_CNT;                     !LOAD DSA WITH DSA_CNT
21107 ;          9374      MLCS1 = write;                       !DO A WRITE FUNCTION
21108 ;          9375      TIME_OUT_LOOP;
21109 ;          9376
21110 ;          9377      if .LBT IS_SET                       !SEE IF THE LAST BLOCK XFERR BIT SET
```

```
21112 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
21113 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (95)
21114 ;
21115 ;          9378      then
21116 ;          9379          begin
21117 ;          9380          CMP_THRESHOLD;                          !COMPARE ERROR PRINT THRESHOLD
21118 ;          9381          ERRDF (97, ASYNC, DUMPER);
21119 ;          9382          PRINTB (THR_FMT, WRD_27, PHR_5, WRD_29);
21120 ;          9383          PRINTB (FMT_7, .DSA_CNT);
21121 ;          9384          end;
21122 ;          9385
21123 ;          9386      if .MLDA neq .DSA_CNT + 1                   !SEE IF THE DSA REG INCREMENTED
21124 ;          9387      then
21125 ;          9388          begin
21126 ;          9389          CMP_THRESHOLD;                          !COMPARE ERROR PRINT THRESHOLD
21127 ;          9390          ERRDF (98, ASYNC, DUMPER);
21128 ;          9391          PRINTB (THR_FMT, REG_6, WRD_31, WRD_14);
21129 ;          9392          PRINTB (FMT_7, .DSA_CNT);
21130 ;          9393          end;
21131 ;          9394
21132 ;          9395      ENDSUB;
21133 ;          9396      end;
21134 ;          9397
21135 ;          9398  BGNSUB;
21136 ;          9399  CLR_MBUS;
21137 ;          9400  ECC_DIS = ONE;                                  !DISABLE ECC
21138 ;          9401  LAST_BLK_XFER ();                               !SET UP A LAST BLOCK XFERR
21139 ;          9402  MLCST = write;                                  !DO A WRITE FUNCTION
21140 ;          9403  TIME_OUT_LOOP;
21141 ;          9404
21142 ;          9405  if .MLDA neq .LST_BLK + 1                       !SEE IF DSA REGISTER INCREMENTED
21143 ;          9406  then
21144 ;          9407      begin
21145 ;          9408      ERRDF (101, ASYNC, DUMPER);
21146 ;          9409      PRINTB (THR_FMT, REG_6, WRD_31, WRD_14);
21147 ;          9410      PRINTB (FMT_7, .LST_BLK);
21148 ;          9411      end;
21149 ;          9412
21150 ;          9413  if .LBT IS_SET                                  !SEE IF LBT BIT SET
21151 ;          9414  then
21152 ;          9415      begin
21153 ;          9416      MLDA = ONES;                                !IF SET THEN TRY TO CLEAR IT
21154 ;          9417
21155 ;          9418      if .LBT IS_SET                              !SEE IF BIT CLEARED
21156 ;          9419      then
21157 ;          9420          begin                                   !ERROR IF NOT
21158 ;          9421          ERRDF (99, ASYNC, DUMPER);
21159 ;          9422          PRINTB (FIV_FMT, WRD_27, PHR_2, WRD_11, WRD_17, REG_6);
21160 ;          9423          end;
21161 ;          9424
21162 ;          9425      end
21163 ;          9426  else                                            !LBT NOT SET
21164 ;          9427      begin
21165 ;          9428      ERRDF (100, ASYNC, DUMPER);
21166 ;          9429      PRINTB (FOR_FMT, WRD_27, PHR_1, WRD_11, WRD_27);
```

```
21168 ;ML4AD
21169 ;                 TEST CODE SECTION
21170
21171 ;          9430       end;
21172 ;          9431
21173 ;          9432   ENDSUB;
21174 ;          9433   ENDTST;
21178
21179                                    .SBTTL  $T43 TEST CODE SECTION
21183 075340 004167 107210          $T43: JSR     R1,$SAVE3                    ;                    9324
21184 075344 005067 120026                CLR     P.CNT                        ;                    9328
21185 075350 016703 116440                MOV     LST.BLK,R3                   ;                    9366
21186 075354 005303                       DEC     R3
21187 075356 005001                       CLR     R1                           ; DSA.CNT
21188 075360 000551                       BR      7$
21189 075362 010102                 1$:   MOV     R1,R2                        ; DSA.CNT,*          9386
21190 075364 005202                       INC     R2
21191 075366 104402                 2$:   TRAP    2                            ;                    9367
21192 075370 152777 000040 120046         BISB    #40,@ML.REG+40               ;                    9368
21193 075376 016700 120430                MOV     ML.DUT,R0
21194 075402 042700 177770                BIC     #177770,R0
21195 075406 142777 000007 120030         BICB    #7,@ML.REG+40
21196 075414 150077 120024                BISB    R0,@ML.REG+40
21197 075420 152777 000002 120076         BISB    #2,@ML.REG+120               ;                    9370
21198 075426 012777 177400 117760         MOV     #-400,@ML.REG+10             ;                    9371
21199 075434 012777 014022 117762         MOV     #IO.BUF,@ML.REG+20           ;                    9372
21200 075442 010177 117766                MOV     R1,@ML.REG+30                ; DSA.CNT,*          9373
21201 075446 012777 000061 117730         MOV     #61,@ML.REG                  ;                    9374
21202 075454 105777 117774          3$:   TSTB    @ML.REG+50
21203 075460 100375                       BPL     3$
21204 075462 032777 002000 117764         BIT     #2000,@ML.REG+50             ;                    9377
21205 075470 001437                       BEQ     4$
21206 075472 005267 117700                INC     P.CNT                        ;                    9379
21207 075476 026767 117674 117674         CMP     P.CNT,LIMIT
21208 075504 003076                       BGT     6$
21209 075506 104455                       TRAP    55                           ;                    9381
21210 075510 000141                       .WORD   141
21211 075512 012706                       .WORD   ASYNC
21212 075514 026302                       .WORD   DUMPER
21213 075516 012746 011032                MOV     #WRD.29,-(SP)                ;                    9382
21214 075522 012746 011714                MOV     #PHR.5,-(SP)
21215 075526 012746 011024                MOV     #WRD.27,-(SP)
21216 075532 012746 010256                MOV     #THR.FMT,-(SP)
21217 075536 012746 000004                MOV     #4,-(SP)
21218 075542 010600                       MOV     SP,R0                        ; SP,*
21219 075544 104414                       TRAP    14
21220 075546 010116                       MOV     R1,(SP)                      ; DSA.CNT,*          9383
21221 075550 012746 007002                MOV     #FMT.7,-(SP)
21222 075554 012746 000002                MOV     #2,-(SP)
```

```
21224                                   ;ML4AD
21225                                   ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
21226                                                                                                       29-Mar-1982 16:21:03    PA:<
21227 075560  010600                             MOV     SP,R0           ; SP,*
21228 075562  104414                             TRAP    14
21229 075564  062706  000016                     ADD     #16,SP          ;                                                       9379
21230 075570  027702  117640              4$:    CMP     @ML.REG+30,R2   ;                                                       9386
21231 075574  001437                             BEQ     5$
21232 075576  005267  117574                     INC     P.CNT           ;                                                       9388
21233 075602  026767  117570  117570             CMP     P.CNT,LIMIT
21234 075610  003034                             BGT     6$
21235 075612  104455                             TRAP    55              ;                                                       9390
21236 075614  000142                             .WORD   142
21237 075616  012706                             .WORD   ASYNC
21238 075620  026302                             .WORD   DUMPER
21239 075622  012746  010664                     MOV     #WRD.14,-(SP)   ;                                                       9391
21240 075626  012746  011054                     MOV     #WRD.31,-(SP)
21241 075632  012746  012540                     MOV     #REG.6,-(SP)
21242 075636  012746  010256                     MOV     #THR.FMT,-(SP)
21243 075642  012746  000004                     MOV     #4,-(SP)
21244 075646  010600                             MOV     SP,R0           ; SP,*
21245 075650  104414                             TRAP    14
21246 075652  010116                             MOV     R1,(SP)         ; DSA.CNT,*                                             9392
21247 075654  012746  007002                     MOV     #FMT.7,-(SP)
21248 075660  012746  000002                     MOV     #2,-(SP)
21249 075664  010600                             MOV     SP,R0           ; SP,*
21250 075666  104414                             TRAP    14
21251 075670  062706  000016                     ADD     #16,SP          ;                                                       9388
21252 075674  104467              5$:    TRAP    67              ;                                                       9393
21253 075676  006000                             ROR     R0
21254 075700  103632                             BLO     2$
21255 075702  005201              6$:    INC     R1              ; DSA.CNT                                               9366
21256 075704  020103              7$:    CMP     R1,R3           ; DSA.CNT,*
21257 075706  101625                             BLOS    1$
21258 075710  104402              8$:    TRAP    2               ;                                                       9396
21259 075712  152777  000040  117524             BISB    #40,@ML.REG+40  ;                                               9398
21260 075720  016703  120106                     MOV     ML.DUT,R3
21261 075724  042703  177770                     BIC     #177770,R3
21262 075730  142777  000007  117506             BICB    #7,@ML.REG+40
21263 075736  150377  117502                     BISB    R3,@ML.REG+40
21264 075742  152777  000002  117554             BISB    #2,@ML.REG+120  ;                                               9400
21265 075750  004767  122370                     JSR     PC,LAST.BLK.XFER ;                                              9401
21266 075754  012777  000061  117422             MOV     #61,@ML.REG     ;                                               9402
21267 075762  105777  117466              9$:    TSTB    @ML.REG+50
21268 075766  100375                             BPL     9$
21269 075770  016701  116020                     MOV     LST.BLK,R1      ;                                               9405
21270 075774  005201                             INC     R1
21271 075776  027701  117432                     CMP     @ML.REG+30,R1
21272 076002  001432                             BEQ     10$
21273 076004  104455                             TRAP    55              ;                                               9408
21274 076006  000145                             .WORD   145
21275 076010  012706                             .WORD   ASYNC
21276 076012  026302                             .WORD   DUMPER
21277 076014  012746  010664                     MOV     #WRD.14,-(SP)   ;                                               9409
21278 076020  012746  011054                     MOV     #WRD.31,-(SP)
```

SEQ 0443

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
21280                                   ;ML4AD
21281                                   ;               TEST CODE SECTION
21282
21283 076024  012746  012540                      MOV     #REG.6,-(SP)
21284 076030  012746  010256                      MOV     #THR.FMT,-(SP)
21285 076034  012746  000004                      MOV     #4,-(SP)
21286 076040  010600                              MOV     SP,R0           ; SP,*
21287 076042  104414                              TRAP    14
21288 076044  016716  115744                      MOV     LST.BLK,(SP)    ;                       9410
21289 076050  012746  007002                      MOV     #FMT.7,-(SP)
21290 076054  012746  000002                      MOV     #2,-(SP)
21291 076060  010600                              MOV     SP,R0           ; SP,*
21292 076062  104414                              TRAP    14
21293 076064  062706  000016                      ADD     #16,SP          ;                       9407
21294 076070  032777  002000  117356  10$:         BIT     #2000,@ML.REG+50 ;                      9413
21295 076076  001436                              BEQ     11$
21296 076100  012777  177777  117326              MOV     #-1,@ML.REG+30  ;                       9416
21297 076106  032777  002000  117340              BIT     #2000,@ML.REG+50 ;                      9418
21298 076114  001453                              BEQ     12$
21299 076116  104455                              TRAP    55              ;                       9421
21300 076120  000143                              .WORD   143
21301 076122  012706                              .WORD   ASYNC
21302 076124  026302                              .WORD   DUMPER
21303 076126  012746  012540                      MOV     #REG.6,-(SP)    ;                       9422
21304 076132  012746  010712                      MOV     #WRD.17,-(SP)
21305 076136  012746  010640                      MOV     #WRD.11,-(SP)
21306 076142  012746  011626                      MOV     #PHR.2,-(SP)
21307 076146  012746  011024                      MOV     #WRD.27,-(SP)
21308 076152  012746  010304                      MOV     #FIV.FMT,-(SP)
21309 076156  012746  000006                      MOV     #6,-(SP)
21310 076162  010600                              MOV     SP,R0           ; SP,*
21311 076164  104414                              TRAP    14
21312 076166  062706  000016                      ADD     #16,SP          ;                       9420
21313 076172  000424                              BR      12$             ;                       9413
21314 076174  104455                  11$:        TRAP    55              ;                       9428
21315 076176  000144                              .WORD   144
21316 076200  012706                              .WORD   ASYNC
21317 076202  026302                              .WORD   DUMPER
21318 076204  012746  011024                      MOV     #WRD.27,-(SP)   ;                       9429
21319 076210  012746  010640                      MOV     #WRD.11,-(SP)
21320 076214  012746  011610                      MOV     #PHR.1,-(SP)
21321 076220  012746  011024                      MOV     #WRD.27,-(SP)
21322 076224  012746  010270                      MOV     #FOR.FMT,-(SP)
21323 076230  012746  000005                      MOV     #5,-(SP)
21324 076234  010600                              MOV     SP,R0           ; SP,*
21325 076236  104414                              TRAP    14
21326 076240  062706  000014                      ADD     #14,SP          ;                       9427
21327 076244  104467                  12$:        TRAP    67              ;                       9430
21328 076246  006000                              ROR     R0
21329 076250  103617                              BLO     8$
21330 076252  000207                              RTS     PC              ;                       9324
21331
21332
21333                                   ; Routine Size:  230 words
                                        ; Maximum stack depth per invocation:  11 words
```

```
21342
21343
21347
21348                                   .SBTTL  T43 TEST CODE SECTION
21352 076254                    T43::
21353 076254  004767  177060    1$:     JSR     PC,$T43                         ;       9432
21354 076260  104466                    TRAP    66
21355 076262  006000                    ROR     R0
21356 076264  103773                    BLO     1$
21357 076266  000207                    RTS     PC
21358
21359                           ; Routine Size:  6 words
21360                           ; Maximum stack depth per invocation:  0 words
21365
21366
21367 ;         9434  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (96)

```
21369 ;ML4AD
21370 ;                 TEST CODE SECTION
21371
21372 ;          9435  !
21373 ;          9436  !
21374 ;          9437  BGNTST;
21375 ;          9438
21376 ;          9439  !++
21377 ;          9440  ! TEST NUMBER:  TST 44
21378 ;          9441  !
21379 ;          9442  ! TEST NAME:    INVALID ADRS TEST
21380 ;          9443  !
21381 ;          9444  ! TEST DESCRIPTION:
21382 ;          9445  !
21383 ;          9446  !       TEST THE DETECTION OF ILLEGAL DSA
21384 ;          9447  !       ADDRESSES BY:
21385 ;          9448  !
21386 ;          9449  !       1.  DOING A MBUS WRITE FUNCTION
21387 ;          9450  !           AT ALL POSSIBLE ILLEGAL DSA
21388 ;          9451  !           ADDRESSES AND TEST THE
21389 ;          9452  !           IAE BIT SET.
21390 ;          9453  !
21391 ;          9454  ! IMPLICIT INPUTS:
21392 ;          9455  !
21393 ;          9456  !       IO_BUF
21394 ;          9457  !       A VECTOR OF 256 WORDS WHERE
21395 ;          9458  !       DATA FOR MBUS READS AND WRITE
21396 ;          9459  !       FUNCTION ARE FOUND.
21397 ;          9460  !
21398 ;          9461  !
21399 ;          9462  !--
21400 ;          9463
21401 ;          9464  local
21402 ;          9465      IAE_CNT;                          !INVLID ADRS COUNT
21403 ;          9466
21404 ;          9467  CLR_THRESHOLD;                        !CLEAR ERROR PRINT THRESHOLD
21405 ;          9468  IAE_CNT = .LST_BLK;                   !START AT LAST  BLOCK + 1
21406 ;          9469
21407 ;          9470  do                                    !TEST FOR ALL INVALID ADDRESSES
21408 ;          9471      begin
21409 ;          9472      IAE_CNT = .IAE_CNT + 1;           !INCREMENT IAE_CNT
21410 ;          9473      BGNSUB;
21411 ;          9474      CLR_MBUS;
21412 ;          9475      MLDA = .IAE_CNT;                  !LOAD DSA
21413 ;          9476      MLWC = not 255;                   !LOAD WORD COUNT
21414 ;          9477      MLBA = IO_BUF;                    !LOAD UBUS ADRS
21415 ;          9478      MLCS1 = write;                    !DO A WRITE FUNCTION
21416 ;          9479
21417 ;          9480      if .IAE IS_NOT_SET                !SET IF IAE SET
21418 ;          9481      then
21419 ;          9482          begin                         !ERROR IF NOT SET
21420 ;          9483          CMP_THRESHOLD;                !COMPARE ERROR PRINT THRESHOLD
21421 ;          9484          ERRDF (102, ASYNC, DUMPER);
21422 ;          9485          PRINTB (FIV_FMT, WRD_30, PHR_1, WRD_11, WRD_30, WRD_10);
21423 ;          9486          end;
```

```
21425 ;ML4AD
21426 ;                 TEST CODE SECTION
21427
21428 ;          9487
21429 ;          9488       ENDSUB;
21430 ;          9489       end
21431 ;          9490  !
21432 ;          9491  !VERSION CZMLAD CHANGED TEST TO UNSIGNED TEST
21433 ;          9492  !
21434 ;          9493  until .IAE_CNT eqlU %o'177777';                !REPEAT UNTIL ALL TESTED
21435 ;          9494
21436 ;          9495  ENDTST;
21440
21441                                        .SBTTL  ST44 TEST CODE SECTION
21445 076270  010146                 ST44:  MOV     R1,-(SP)                ;                                  9433
21446 076272  005067  117100               CLR     P.CNT                   ;                                  9465
21447 076276  016701  115512               MOV     LST.BLK,R1              ; *,IAE.CNT                        9468
21448 076302  005201                 1$:   INC     R1                      ; IAE.CNT                          9472
21449 076304  104402                 2$:   TRAP    2
21450 076306  152777  000040  117130       BISB    #40,@ML.REG+40          ;                                  9473
21451 076314  016700  117512               MOV     ML.DUT,R0
21452 076320  042700  177770               BIC     #177770,R0
21453 076324  142777  000007  117112       BICB    #7,@ML.REG+40
21454 076332  150077  117106               BISB    R0,@ML.REG+40
21455 076336  010177  117072               MOV     R1,@ML.REG+30           ; IAE.CNT,*                        9475
21456 076342  012777  177400  117044       MOV     #-400,@ML.REG+10        ;                                  9476
21457 076350  012777  014022  117046       MOV     #IO.BUF,@ML.REG+20      ;                                  9477
21458 076356  012777  000061  117020       MOV     #61,@ML.REG             ;                                  9478
21459 076364  032777  002000  117072       BIT     #2000,@ML.REG+60        ;                                  9480
21460 076372  001034                       BNE     3$
21461 076374  005267  116776               INC     P.CNT                   ;                                  9482
21462 076400  026767  116772  116772       CMP     P.CNT,LIMIT
21463 076406  003031                       BGT     4$
21464 076410  104455                       TRAP    55                      ;                                  9484
21465 076412  000146                       .WORD   146
21466 076414  012706                       .WORD   ASYNC
21467 076416  026302                       .WORD   DUMPER
21468 076420  012746  010630               MOV     #WRD.10,-(SP)           ;                                  9485
21469 076424  012746  011046               MOV     #WRD.30,-(SP)
21470 076430  012746  010640               MOV     #WRD.11,-(SP)
21471 076434  012746  011610               MOV     #PHR.1,-(SP)
21472 076440  012746  011046               MOV     #WRD.30,-(SP)
21473 076444  012746  010304               MOV     #FIV.FMT,-(SP)
21474 076450  012746  000006               MOV     #6,-(SP)
21475 076454  010600                       MOV     SP,R0                   ; SP,*
21476 076456  104414                       TRAP    14
21477 076460  062706  000016               ADD     #16,SP                  ;                                  9482
21478 076464  104467                 3$:   TRAP    67                      ;                                  9486
21479 076466  006000                       ROR     R0
```

```
21481                                   ;ML4AD
21482                                   ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
21483                                                                                                       29-Mar-1982 16:21:03    PA:<
21484 076470  103705                            BLO     2$
21485 076472  020127  177777        4$:        CMP     R1,#-1                  ; IAE.CNT,*                                 9493
21486 076476  001301                            BNE     1$
21487 076500  012601                            MOV     (SP)+,R1                ;                                           9433
21488 076502  000207                            RTS     PC
21489
21490                                   ; Routine Size:  70 words
21491                                   ; Maximum stack depth per invocation:  8 words
21496
21497
21501
21502                                           .SBTTL  T44 TEST CODE SECTION
21506 076504                            T44::
21507 076504  004767  177560        1$:        JSR     PC,ST44                 ;                                           9493
21508 076510  104466                            TRAP    66
21509 076512  006000                            ROR     R0
21510 076514  103773                            BLO     1$
21511 076516  000207                            RTS     PC
21512
21513                                   ; Routine Size:  6 words
21514                                   ; Maximum stack depth per invocation:  0 words
21519
21520
21521 ;          9496  !<BLF/PAGE>
```

```
21523 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
21524 ;                 TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (97)
21525
21526 ;          9497   !
21527 ;          9498   BGNTST;
21528 ;          9499
21529 ;          9500   !++
21530 ;          9501   ! TEST NUMBER:  TST 45
21531 ;          9502   !
21532 ;          9503   ! TEST NAME:    ADRS OVERFLOW BIT TEST
21533 ;          9504   !
21534 ;          9505   ! TEST DESCRIPTION:
21535 ;          9506   !       TEST THE DETECTION OF ADDRESS OVERFLOWS BY:
21536 ;          9507   !
21537 ;          9508   !       1.  STARTING AT THE LAST BLOCK DO A TWO BLOCK TRANSFER.
21538 ;          9509   !
21539 ;          9510   !       2.  READ THE AOE BIT SET.
21540 ;          9511   !
21541 ;          9512   ! IMPLICIT INPUTS:
21542 ;          9513   !       IO_BUF
21543 ;          9514   !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE
21544 ;          9515   !       FUNCTIONS ARE FOUND.
21545 ;          9516   !--
21546 ;          9517
21547 ;          9518   CLR_MBUS;
21548 ;          9519   MLWC = not 511;                          !DO TWO BLOCK XFERR
21549 ;          9520   MLBA = IO_BUF;                           !LOAD UBUS ADRS
21550 ;          9521   MLDA = .LST_BLK;                         !START AT LAST BLOCK
21551 ;          9522   MLCS1 = write;                           !DO A WRITE FUNCTION
21552 ;          9523   TIME_OUT_LOOP;
21553 ;          9524
21554 ;          9525   if .AOE IS_NOT_SET                       !SEE IF AOE SET
21555 ;          9526   then
21556 ;          9527       begin                                !ERROR IF NOT SET
21557 ;          9528       ERRDF (103, SYNC, DUMPER);
21558 ;          9529       PRINTB (FOR_FMT, WRD_26, PHR_1, WRD_11, FNC_19);
21559 ;          9530       end;
21560 ;          9531
21561 ;          9532   ENDTST;
21565
21566                                          .SBTTL  ST45 TEST CODE SECTION
21570 076520 152777 000040  116716  ST45:    BISB    #40,@ML.REG+40                  ;                9498
21571 076526 016700 117300                   MOV     ML.DUT,R0
21572 076532 042700 177770                   BIC     #177770,R0
21573 076536 142777 000007  116700           BICB    #7,@ML.REG+40
21574 076544 150077 116674                   BISB    R0,@ML.REG+40
21575 076550 012777 177000  116636           MOV     #-1000,@ML.REG+10               ;                9519
21576 076556 012777 014022  116640           MOV     #IO.BUF,@ML.REG+20              ;                9520
21577 076564 016777 115224  116642           MOV     LST.BLK,@ML.REG+30              ;                9521
```

```
21579                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
21580                                   ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
21581
21582 076572  012777  000061  116604            MOV     #61,@ML.REG                     ;                               9522
21583 076600  105777  116650           1$:      TSTB    @ML.REG+50
21584 076604  100375                            BPL     1$
21585 076606  032777  001000  116650            BIT     #1000,@ML.REG+60                ;                               9525
21586 076614  001024                            BNE     2$
21587 076616  104455                            TRAP    55                              ;                               9528
21588 076620  000147                            .WORD   147
21589 076622  012750                            .WORD   SYNC
21590 076624  026302                            .WORD   DUMPER
21591 076626  012746  012426                    MOV     #FNC.19,-(SP)                   ;                               9529
21592 076632  012746  010640                    MOV     #WRD.11,-(SP)
21593 076636  012746  011610                    MOV     #PHR.1,-(SP)
21594 076642  012746  011016                    MOV     #WRD.26,-(SP)
21595 076646  012746  010270                    MOV     #FOR.FMT,-(SP)
21596 076652  012746  000005                    MOV     #5,-(SP)
21597 076656  010600                            MOV     SP,R0                           ; SP,*
21598 076660  104414                            TRAP    14
21599 076662  062706  000014                    ADD     #14,SP                          ;                               9527
21600 076666  000207           2$:      RTS     PC                                      ;                               9495
21601
21602                                   ; Routine Size:  52 words
21603                                   ; Maximum stack depth per invocation:  6 words
21608
21609
21613
21614                                           .SBTTL  T45 TEST CODE SECTION
21618 076670                            T45::
21619 076670  004767  177624           1$:      JSR     PC,$T45                         ;                               9530
21620 076674  104466                            TRAP    66
21621 076676  006000                            ROR     R0
21622 076700  103773                            BLO     1$
21623 076702  000207                            RTS     PC
21624
21625                                   ; Routine Size:  6 words
21626                                   ; Maximum stack depth per invocation:  0 words
21631
21632
21633 ;          9533  !<BLF/PAGE>
```

```
21635 ;ML4AD
21636 ;                 TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
21637 ;                                                      29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (98)
21638 ;          9534   !
21639 ;          9535   BGNTST;
21640 ;          9536
21641 ;          9537   !++
21642 ;          9538   ! TEST NUMBER:  TST 46
21643 ;          9539   !
21644 ;          9540   ! TEST NAME:    SYNC BUS PARITY TEST
21645 ;          9541   !
21646 ;          9542   ! TEST DESCRIPTION:
21647 ;          9543   !       TEST ABILITY OF SYNC BUS TO DETECT AND GENERATE
21648 ;          9544   !       GOOD PARITY BY:
21649 ;          9545   !
21650 ;          9546   !       1.  VIA MBUS WRITE FUNCTION WRITE ALTERNATING ONES AND ZEROES TO THE
21651 ;          9547   !           DEVICE AND READ THE DPAR BIT CLEARED.
21652 ;          9548   !
21653 ;          9549   !       2.  REPEAT WITH SHIFTED DATA
21654 ;          9550   !
21655 ;          9551   !       3.  VIA MBUS READ FUNCTION READ THE ALTERNATING PATTERN AND MDPE CLEARED.
21656 ;          9552   !
21657 ;          9553   ! IMPLICIT INPUTS:
21658 ;          9554   !       IO_BUF
21659 ;          9555   !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITES
21660 ;          9556   !       FUNCTIONS ARE FOUND.
21661 ;          9557   !--
21662 ;          9558
21663 ;          9559   if .PAR_DIS IS_SET                   !SEE IF PARITY IS DISABLED
21664 ;          9560   then
21665 ;          9561       begin                            !PRINT MESSAGE AND EXIT TST IF YES
21666 ;          9562       PRINTB (THR_FMT, FNC_3, WRD_7, WRD_37);
21667 ;          9563       EXIT_TST;
21668 ;          9564       end;
21669 ;          9565
21670 ;          9566   CLR_MBUS;
21671 ;          9567   BAI = ONE;                           !SET ON FIRST IO_BUF ADRS
21672 ;          9568   IO_BUF = %o'125252';                 !ALTERNATE 1, 0 PATTERN
21673 ;          9569
21674 ;          9570   incr TWICE from 0 to 1 do            !REPEAT LOOP TWICE
21675 ;          9571       begin
21676 ;          9572       BGNSUB;
21677 ;          9573       GD_BLK_XFER ();                  !SET UP A GOOD BLOCK XFERR
21678 ;          9574       MLCS1 = write;                   !DO A WRITE FUNCTION
21679 ;          9575       TIME_OUT_LOOP;
21680 ;          9576
21681 ;          9577       if .DPAR IS_SET                  !SEE IF DPAR GOT SET
21682 ;          9578       then
21683 ;          9579           begin                        !ERROR IF SET
21684 ;          9580           ERRDF (105, SYNC, DUMPER);
21685 ;          9581           PRINTB (FOR_FMT, WRD_23, WRD_6, WRD_7, WRD_9);
21686 ;          9582           end;
21687 ;          9583
21688 ;          9584       IO_BUF = .IO_BUF^ONE;            !SHIFT THE IO_BUF & REPEAT
21689 ;          9585       MLER = ZEROES;                   !CLEAR OUT ERROR REG & REPEAT
```

```
;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (98)

;          9586      ENDSUB;
;          9587      end;
;          9588
;          9589  incr TWICE from 0 to 1 do                !REPEATE LOOP TWICE
;          9590      begin
;          9591      BGNSUB;
;          9592      CLR_MBUS;
;          9593      BAI = ONE;                           !SET ON FIRST IO BUF ADRS
;          9594      GD_BLK_XFER ();                      !SET UP A GOOD BLOCK XFERR
;          9595      MLCS1 = read;                        !DO A READ XFERR
;          9596      TIME_OUT_LOOP;
;          9597
;          9598      if .MDPE IS_SET                      !SEE IF READ GENERATED BAD PARITY
;          9599      then
;          9600          begin                            !ERROR IF MDPE SET
;          9601          ERRDF (106, SYNC, DUMPER);
;          9602          PRINTB (FOR_FMT, WRD_23, WRD_6, WRD_7, WRD_8);
;          9603          end;
;          9604
;          9605      ENDSUB;
;          9606      IO_BUF = .IO_BUF^ONE;                !MAKE DATA PATTERN HAVE ONE LESS ONE
;          9607      GD_BLK_XFER ();                      !SET UP A GOOD BLK XFER
;          9608      MLCS1 = write;                       !CHANGE THE BACKGROUND IN MEMORY
;          9609      TIME_OUT_LOOP;
;          9610      end;
;          9611
;          9612  ENDTST;
21724
21725                                        .SBTTL  $T46 TEST CODE SECTION
21729 076704 010146                  $T46:   MOV     R1,-(SP)                ;                       9532
21730 076706 026727 115076 000001            CMP     PAR.DIS,#1              ;                       9559
21731 076714 001021                          BNE     1$
21732 076716 012746 011122                   MOV     #WRD.37,-(SP)           ;                       9562
21733 076722 012746 010566                   MOV     #WRD.7,-(SP)
21734 076726 012746 012172                   MOV     #FNC.3,-(SP)
21735 076732 012746 010256                   MOV     #THR.FMT,-(SP)
21736 076736 012746 000004                   MOV     #4,-(SP)
21737 076742 010600                          MOV     SP,R0                   ; SP,*
21738 076744 104414                          TRAP    14
21739 076746 104463                          TRAP    63
21740 076750 062706 000012                   ADD     #12,SP                  ;                       9559
21741 076754 000167 000402                   JMP     9$                      ;                       9561
21742 076760 152777 000040 116456    1$:     BISB    #40,@ML.REG+40          ;                       9564
21743 076766 016701 117040                   MOV     ML.DUT,R1
21744 076772 042701 177770                   BIC     #177770,R1
21745 076776 142777 000007 116440            BICB    #7,@ML.REG+40
```

```
21747                                   ;ML4AD
21748                                   ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
21749                                                                                                       29-Mar-1982 16:21:03    PA:<
21750 077004  150177  116434                    BISB    R1,@ML.REG+40
21751 077010  152777  000010  116426            BISB    #10,@ML.REG+40          ;                                               9567
21752 077016  012767  125252  114776            MOV     #-52526,IO.BUF          ;                                               9568
21753 077024  005001                            CLR     R1                      ; TWICE                                         9570
21754 077026  104402                    2$:     TRAP    2                       ;                                               9571
21755 077030  004767  121256                    JSR     PC,GD.BLK.XFER          ;                                               9573
21756 077034  012777  000061  116342            MOV     #61,@ML.REG             ;                                               9574
21757 077042  105777  116406            3$:     TSTB    @ML.REG+50
21758 077046  100375                            BPL     3$
21759 077050  132777  000040  116406            BITB    #40,@ML.REG+60          ;                                               9577
21760 077056  001424                            BEQ     4$
21761 077060  104455                            TRAP    55                      ;                                               9580
21762 077062  000151                            .WORD   151
21763 077064  012750                            .WORD   SYNC
21764 077066  026302                            .WORD   DUMPER
21765 077070  012746  010616                    MOV     #WRD.9,-(SP)            ;                                               9581
21766 077074  012746  010566                    MOV     #WRD.7,-(SP)
21767 077100  012746  010560                    MOV     #WRD.6,-(SP)
21768 077104  012746  010766                    MOV     #WRD.23,-(SP)
21769 077110  012746  010270                    MOV     #FOR.FMT,-(SP)
21770 077114  012746  000005                    MOV     #5,-(SP)
21771 077120  010600                            MOV     SP,R0                   ; SP,*
21772 077122  104414                            TRAP    14
21773 077124  062706  000014                    ADD     #14,SP                  ;                                               9579
21774 077130  006367  114666            4$:     ASL     IO.BUF                                                                  9584
21775 077134  005077  116324                    CLR     @ML.REG+60              ;                                               9585
21776 077140  104467                            TRAP    67
21777 077142  006000                            ROR     R0
21778 077144  103730                            BLO     2$
21779 077146  005201                            INC     R1                      ; TWICE                                         9570
21780 077150  020127  000001                    CMP     R1,#1                   ; TWICE,*
21781 077154  003724                            BLE     2$
21782 077156  005001                            CLR     R1                      ; TWICE                                         9589
21783 077160  104402                    5$:     TRAP    2                       ;                                               9590
21784 077162  152777  000040  116254            BISB    #40,@ML.REG+40          ;                                               9591
21785 077170  016700  116636                    MOV     ML.DUT,R0
21786 077174  042700  177770                    BIC     #177770,R0
21787 077200  142777  000007  116236            BICB    #7,@ML.REG+40
21788 077206  150077  116232                    BISB    R0,@ML.REG+40
21789 077212  152777  000010  116224            BISB    #10,@ML.REG+40          ;                                               9593
21790 077220  004767  121066                    JSR     PC,GD.BLK.XFER          ;                                               9594
21791 077224  012777  000071  116152            MOV     #71,@ML.REG             ;                                               9595
21792 077232  105777  116216            6$:     TSTB    @ML.REG+50
21793 077236  100375                            BPL     6$
21794 077240  032777  000400  116176            BIT     #400,@ML.REG+40         ;                                               9598
21795 077246  001424                            BEQ     7$
21796 077250  104455                            TRAP    55                      ;                                               9601
21797 077252  000152                            .WORD   152
21798 077254  012750                            .WORD   SYNC
21799 077256  026302                            .WORD   DUMPER
21800 077260  012746  010602                    MOV     #WRD.8,-(SP)            ;                                               9602
21801 077264  012746  010566                    MOV     #WRD.7,-(SP)
```

```
21803                                   ;ML4AD
21804                                   ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
21805                                                                                                       29-Mar-1982 16:21:03    PA:<
21806 077270  012746  010560                     MOV     #WRD.6,-(SP)
21807 077274  012746  010766                     MOV     #WRD.23,-(SP)
21808 077300  012746  010270                     MOV     #FOR.FMT,-(SP)
21809 077304  012746  000005                     MOV     #5,-(SP)
21810 077310  010600                             MOV     SP,R0                          ; SP,*
21811 077312  104414                             TRAP    14
21812 077314  062706  000014                     ADD     #14,SP                         ;                                                   9600
21813 077320  104467                     7$:     TRAP    67                             ;                                                   9603
21814 077322  006000                             ROR     R0
21815 077324  103715                             BLO     5$
21816 077326  006367  114470                     ASL     IO.BUF                         ;                                                   9606
21817 077332  004767  120754                     JSR     PC,GD.BLK.XFER                 ;                                                   9607
21818 077336  012777  000061  116040             MOV     #61,@ML.REG                    ;                                                   9608
21819 077344  105777  116104              8$:    TSTB    @ML.REG+50
21820 077350  100375                             BPL     8$
21821 077352  005201                             INC     R1                             ; TWICE                                             9589
21822 077354  020127  000001                     CMP     R1,#1                          ; TWICE,*
21823 077360  003677                             BLE     5$
21824 077362  012601                     9$:     MOV     (SP)+,R1                       ;                                                   9532
21825 077364  000207                             RTS     PC
21826
21827                                   ; Routine Size:  153 words
21828                                   ; Maximum stack depth per invocation:  7 words
21833
21834
21838
21839                                           .SBTTL  T46 TEST CODE SECTION
21843 077366                             T46::
21844 077366  004767  177312             1$:    JSR     PC,$T46                        ;                                                   9610
21845 077372  104466                             TRAP    66
21846 077374  006000                             ROR     R0
21847 077376  103773                             BLO     1$
21848 077400  000207                             RTS     PC
21849
21850                                   ; Routine Size:  6 words
21851                                   ; Maximum stack depth per invocation:  0 words
21856
21857 ;          9613  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (99)

```
21859 ;ML4AD
21860 ;                 TEST CODE SECTION
21861
21862 ;          9614  !
21863 ;          9615  !
21864 ;          9616  BGNTST;
21865 ;          9617
21866 ;          9618  !++
21867 ;          9619  ! TEST NUMBER:  TST 47
21868 ;          9620  !
21869 ;          9621  ! TEST NAME:    WRITE READ ML11
21870 ;          9622  !
21871 ;          9623  ! TEST DESCRIPTION:
21872 ;          9624  !
21873 ;          9625  !       PROVIDES A MBUS READ/WRITE
21874 ;          9626  !       DATA TRANSFER TROUBLE SHOOTING
21875 ;          9627  !       LOOP BY:
21876 ;          9628  !
21877 ;          9629  !       1.  LOAD APPROPRIATE RH REGISTERS.
21878 ;          9630  !           DO A WRITE FUNCTION.
21879 ;          9631  !
21880 ;          9632  !       2.  LOAD APPROPRIATE RH REGISTERS
21881 ;          9633  !           DO A WRITE CHECK FUNCTION.
21882 ;          9634  !
21883 ;          9635  !       3.  COMPLIMENT DATA AND
21884 ;          9636  !           REPEAT.
21885 ;          9637  !
21886 ;          9638  ! IMPLICIT INPUTS:
21887 ;          9639  !
21888 ;          9640  !       IO_BUF
21889 ;          9641  !       A VECTOR OF 256 WORDS WHERE
21890 ;          9642  !       DATA FOR MBUS READ AND WRITE
21891 ;          9643  !       FUNCTIONS IS FOUND.
21892 ;          9644  !
21893 ;          9645  !--
21894 ;          9646
21895 ;          9647  local
21896 ;          9648      TST_PAT;                          !TEST PATTERN
21897 ;          9649
21898 ;          9650  TST_PAT = %o'052525';
21899 ;          9651
21900 ;          9652  incr TWICE from 0 to 1 do             !WRITE READ 1'S AND 0'S ON MBUS
21901 ;          9653      begin
21902 ;          9654      CLR_MBUS;
21903 ;          9655      BAI = ONE;                        !SET ON FIRST IO_BUF ADRS
21904 ;          9656      IO_BUF = .TST_PAT;                !LOAD FIRST IO_BUF ADRS
21905 ;          9657      GD_BLK_XFER ();                   !SET UP A GOOD BLOCK XFERR
21906 ;          9658      MLCS1 = write;                    !DO A WRITE FUNCTION
21907 ;          9659      TIME_OUT_LOOP;
21908 ;          9660      CLR_MBUS;
21909 ;          9661      BAI = ONE;                        !SET ON FIRST IO_BUF ADRS
21910 ;          9662      GD_BLK_XFER ();                   !SET UP A GOOD BLOCK XFERR
21911 ;          9663      MLCS1 = WRT_CHK;                  !DO A WRITE CHECK FUNCTION
21912 ;          9664      TIME_OUT_LOOP;
21913 ;          9665
```

```
 21915 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 21916 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (99)
 21917
 21918 ;          9666      if .WCE IS_SET                      !SEE IF WRITE CHECK ERROR SET
 21919 ;          9667      then
 21920 ;          9668          begin                           !ERROR IF SET
 21921 ;          9669          ERRDF (109, TRBLE_LOOP, DUMPER);
 21922 ;          9670          PRINTB (SIX_FMT, FNC_4, WRD_10, WRD_12, FNC_5, FNC_6, FNC_3);
 21923 ;          9671          end;
 21924 ;          9672
 21925 ;          9673      TST_PAT = not .TST_PAT;             !COMPLIMENT TST_PAT AND REPEAT
 21926 ;          9674      end;
 21927 ;          9675
 21928 ;          9676  ENDTST;
 21932
 21933                                        .SBTTL  $T47 TEST CODE SECTION
 21937 077402  004167  105132          $T47:  JSR     R1,$SAVE2                    ;                          9612
 21938 077406  012701  052525                 MOV     #52525,R1                    ; *,TST.PAT                9650
 21939 077412  005002                         CLR     R2                           ; TWICE                    9652
 21940 077414  152777  000040  116022  1$:    BISB    #40,@ML.REG+40               ;                          9653
 21941 077422  016700  116404                 MOV     ML.DUT,R0
 21942 077426  042700  177770                 BIC     #177770,R0
 21943 077432  142777  000007  116004         BICB    #7,@ML.REG+40
 21944 077440  150077  116000                 BISB    R0,@ML.REG+40
 21945 077444  152777  000010  115772         BISB    #10,@ML.REG+40               ;                          9655
 21946 077452  010167  114344                 MOV     R1,IO.BUF                    ; TST.PAT,*                9656
 21947 077456  004767  120630                 JSR     PC,GD.BLK.XFER               ;                          9657
 21948 077462  012777  000061  115714         MOV     #61,@ML.REG                  ;                          9658
 21949 077470  105777  115760          2$:    TSTB    @ML.REG+50
 21950 077474  100375                         BPL     2$
 21951 077476  152777  000040  115740         BISB    #40,@ML.REG+40               ;                          9659
 21952 077504  016700  116322                 MOV     ML.DUT,R0
 21953 077510  042700  177770                 BIC     #177770,R0
 21954 077514  142777  000007  115722         BICB    #7,@ML.REG+40
 21955 077522  150077  115716                 BISB    R0,@ML.REG+40
 21956 077526  152777  000010  115710         BISB    #10,@ML.REG+40               ;                          9661
 21957 077534  004767  120552                 JSR     PC,GD.BLK.XFER               ;                          9662
 21958 077540  012777  000051  115636         MOV     #51,@ML.REG                  ;                          9663
 21959 077546  105777  115702          3$:    TSTB    @ML.REG+50
 21960 077552  100375                         BPL     3$
 21961 077554  032777  040000  115662         BIT     #40000,@ML.REG+40            ;                          9666
 21962 077562  001430                         BEQ     4$
 21963 077564  104455                         TRAP    55                           ;                          9669
 21964 077566  000155                         .WORD   155
 21965 077570  013250                         .WORD   TRBLE.LOOP
 21966 077572  026702                         .WORD   DUMPER
 21967 077574  012746  012172                 MOV     #FNC.3,-(SP)                 ;                          9670
 21968 077600  012746  012226                 MOV     #FNC.6,-(SP)
 21969 077604  012746  012216                 MOV     #FNC.5,-(SP)
```

```
21971                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
21972                                   ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
21973
21974 077610  012746  010650                       MOV     #WRD.12,-(SP)
21975 077614  012746  010630                       MOV     #WRD.10,-(SP)
21976 077620  012746  012200                       MOV     #FNC.4,-(SP)
21977 077624  012746  010322                       MOV     #SIX.FMT,-(SP)
21978 077630  012746  000007                       MOV     #7,-(SP)
21979 077634  010600                              MOV     SP,R0                           ; SP,*
21980 077636  104414                              TRAP    14
21981 077640  062706  000020                       ADD     #20,SP                          ;                                       9668
21982 077644  005101                        4$:    COM     R1                              ; TST.PAT                               9673
21983 077646  005202                              INC     R2                              ; TWICE                                 9652
21984 077650  020227  000001                       CMP     R2,#1                           ; TWICE,*
21985 077654  003657                              BLE     1$
21986 077656  000207                              RTS     PC                              ;                                       9612
21987
21988                                   ; Routine Size:  87 words
21989                                   ; Maximum stack depth per invocation:  11 words
21994
21995
21999
22000                                           .SBTTL  T47 TEST CODE SECTION
22004 077660                            T47::
22005 077660  004767  177516            1$:     JSR     PC,$T47                         ;                                       9674
22006 077664  104466                              TRAP    66
22007 077666  006000                              ROR     R0
22008 077670  103773                              BLO     1$
22009 077672  000207                              RTS     PC
22010
22011                                   ; Routine Size:  6 words
22012                                   ; Maximum stack depth per invocation:  0 words
22017
22018
22019 ;          9677  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (100)

```
22021 ;ML4AD
22022 ;                 TEST CODE SECTION
22023
22024 ;          9678  !
22025 ;          9679  BGNTST;
22026 ;          9680
22027 ;          9681  !++
22028 ;          9682  ! TEST NUMBER:  TST 48
22029 ;          9683  !
22030 ;          9684  ! TEST NAME:    CRC DATA BUS TEST
22031 ;          9685  !
22032 ;          9686  ! TEST DESCRIPTION:
22033 ;          9687  !       TEST THE CRC DATA BUS BETWEEN THE CRC
22034 ;          9688  !       GENERATORS AND THE CRC/MBUS DATA
22035 ;          9689  !       MUX FOR CONTINUITY AND BIT UNIQUENESS
22036 ;          9690  !       BY:
22037 ;          9691  !
22038 ;          9692  !       1.  WHILE IN ECC DIAGNOSTIC MODE DO
22039 ;          9693  !           GOOD BLOCK TRANSFERS WITH VARIOUS
22040 ;          9694  !           CRC_A,CRC_B AND PAR_CRC_WRD WHICH
22041 ;          9695  !           WILL EXERCISE BUS CONTINUITY AND
22042 ;          9696  !           UNIQUENESS.
22043 ;          9697  !
22044 ;          9698  !       2.  THEN IN DATA DIAGNOSTEC MODE READ
22045 ;          9699  !           THE CRC NIBBLES FROM ONE CRC GROUP
22046 ;          9700  !           FOR CORRECT CRC_A,CRC_B AND PAR_CRC_WRD
22047 ;          9701  !           DATA PATTERNS.
22048 ;          9702  !
22049 ;          9703  ! IMPLICIT INPUTS:
22050 ;          9704  !       PD_TEMP
22051 ;          9705  !       A BIT VECTOR OF 16 BITS WHERE THE READ
22052 ;          9706  !       PROM DATA IS STORED OAND ACCESSED FROM.
22053 ;          9707  !
22054 ;          9708  !
22055 ;          9709  !--
22056 ;          9710
22057 ;          9711  local
22058 ;          9712      TMP_E1,                     !TEMP STORAGE FOR E1 DATA WORD
22059 ;          9713      DODO_FLG,                   !DROP UNIT FLAG
22060 ;          9714      CRC_NIB,                    !POINTER TO THE CRC NIBBLE
22061 ;          9715      CRC_TSTED,                  !STORES HOW MANY CRC NIBBLES WE HAVE TESTED
22062 ;          9716      ERR_FLG,                    !ERROR FLAG
22063 ;          9717      NIB_PAT;                    !TEST DATA FOR THE CRC NIBBLE
22064 ;          9718
22065 ;          9719  CLR_THRESHOLD;                  !CLEAR ERROR PRINT THRESHOLD
22066 ;          9720  CRC_NIB = 9;                    !SET CRC NIBBLE POINTER
22067 ;          9721  DODO_FLG = ZERO;                !CLEAR THE DROP UNIT FLAG
22068 ;          9722
22069 ;          9723  incr LOOP from 0 to 4 do        !REPEAT THE TEST WITH FIVE NIBBLE PATTERNS
22070 ;          9724      begin
22071 ;          9725      BGNSUB;                     !START OF THE SCOPE LOOP
22072 ;          9726      CLR_MBUS;
22073 ;          9727      ECC_DM = ONE;               !SET ECC DIAG MOD
22074 ;          9728
22075 ;          9729      case .LOOP from 0 to 4 of   !SELECT ONE OF THE FIVE NIBBLE PATTERNS
```

```
22077 ;ML4AD
22078 ;                 TEST CODE SECTION
22079
22080 ;          9730           set
22081 ;          9731
22082 ;          9732           [0] :                                   !PATTERN 0
22083 ;          9733               NIB_PAT = %b'110';
22084 ;          9734
22085 ;          9735           [1] :                                   !PATTERN 1
22086 ;          9736               begin
22087 ;          9737               MLE1 = ONES;
22088 ;          9738               MLE2 = ONES;
22089 ;          9739               NIB_PAT = %b'001';
22090 ;          9740               end;
22091 ;          9741
22092 ;          9742           [2] :                                   !PATTERN 2
22093 ;          9743               begin
22094 ;          9744               CRC_A = ONES;
22095 ;          9745               CRC_B = ONES;
22096 ;          9746               NIB_PAT = %b'000';
22097 ;          9747               end;
22098 ;          9748
22099 ;          9749           [3] :                                   !PATTERN 3
22100 ;          9750               begin
22101 ;          9751               PAR_CRC_WRD = ONES;
22102 ;          9752               CRC_B = ONES;
22103 ;          9753               NIB_PAT = %b'011';
22104 ;          9754               end;
22105 ;          9755
22106 ;          9756           [4] :                                   !PATTERN 4
22107 ;          9757               begin
22108 ;          9758           !+
22109 ;          9759           ! THE FOLLOWING ASSIGNMENT:
22110 ;          9760           !
22111 ;          9761           !       MLE1 = .TMP_E1
22112 ;          9762           !
22113 ;          9763           ! IS EQUIVALENT TO THE FOLLOWING
22114 ;          9764           ! TWO ASSIGNMENTS:
22115 ;          9765           !
22116 ;          9766           !   PAR_CRC_WRD = ONES;          !LOAD PAR_CRC_WRD WITH REM_TBL
22117 ;          9767           !   CRC_A = ONES;                !LOAD CRC_A WITH REM_TBL
22118 ;          9768           !
22119 ;          9769           ! THIS IS NECESSARY DUE TO THE FACT THAT
22120 ;          9770           ! THE E1 REGISTER IS WORD ORIENTATED AND
22121 ;          9771           ! THE BLISS COMPILER GENERATES BYTE INST
22122 ;          9772           ! TO ACCESS THE REGISTER.
22123 ;          9773           !-
22124 ;          9774
22125 ;          9775               TMP_E1 = ZEROES;                    !CLEAR OUT THE TEMP WORD
22126 ;          9776               TMP_E1<0, 6> = ONES;         !LOAD CRC_A WITH REM_TBL
22127 ;          9777               TMP_E1<8, 6> = ONES;         !LOAD PAR_CRC_WRD WITH REM_TBL
22128 ;          9778               MLE1 = .TMP_E1;                     !LOAD THE E1 REGISTER WITH DATA
22129 ;          9779
22130 ;          9780               NIB_PAT = %b'101';
22131 ;          9781               end;
```

```
;ML4AD
;                    TEST CODE SECTION

;          9782          tes;
;          9783
;          9784      GD_BLK_XFER ();                    !SET UP A GOOD BLOCK TRANSFER
;          9785      MLCS1 = write;                     !DO A WRITE TRANSFER
;          9786      TIME_OUT_LOOP;                     !WAIT UNTIL DRIVE READY
;          9787      DAT_DM = ONE;                      !SET DATA DIAG MODE
;          9788      GD_BLK_XFER ();                    !SET UP A GOOD BLOCK TRANSFER
;          9789      MLCS1 = read;                      !DO A READ TRANSFER
;          9790      CRC_TSTED = ZERO;
;          9791      DELAY (ONE_US);
;          9792
;          9793      do                                 !TEST ONE CRC GROUP '6 CRC NIBBLES'
;          9794          begin
;          9795          PD_TEMP = .MLPD;               !READ THE PROM DATA FOR THIS ARRAY WORD
;          9796          DAT_CLK = ONE;                 !CLOCK OUT AN ARRAY WORD INTO THE DATA DIAG REG'S
;          9797          DELAY (ONE_US);                !DELAY ONE MICRO SECOND
;          9798          RD_LNG_WRD;                    !READ THE DATA DIAG REG'S INTO MEMORY
;          9799
;          9800          if .PD_TEMP [.CRC_NIB] IS_NOT_SET   !TEST THIS CRC NIBBLE IF GOOD
;          9801          then
;          9802              begin
;          9803              CRC_TSTED = .CRC_TSTED + 1;     !INCREMENT THE GOOD NIBBLE COUNTER
;          9804              TST_LNG_WRD (.CRC_NIB, .NIB_PAT, ERR_FLG);  !TEST THE CRC NIBBLE
;          9805
;          9806              if .ERR_FLG IS_SET              !SEE IF THE TEST FOUND AN ERROR
;          9807              then
;          9808                  begin                       !REPORT ERROR IF FLAG IS SET
;          9809                  CMP_THRESHOLD;              !COMPARE ERROR PRINT THRESHOLD
;          9810                  ERRDF (131, SYNC, DUMPER);
;          9811                  PRINTB (FOR_FMT, WRD_64, WRD_24, WRD_73, WRD_10);
;          9812                  PRINTB (FMT_5, .NIB_PAT, .NIB_SAVE [NIB_9], 8);
;          9813                  DODU_FLG = ONE;             !SET THE DROP UNIT FLAG
;          9814                  end;
;          9815
;          9816              end;
;          9817
;          9818          end
;          9819      until .CRC_TSTED eql 6;             !REPEAT UNTIL ONE CRC GROUP HAS BEEN TESTED
;          9820
;          9821      ENDSUB;                             !END OF SCOPE LOOP
;          9822      end;
;          9823
;          9824  if .DODU_FLG IS_SET
;          9825  then
;          9826      begin
;          9827      DODU (.ML_LUN);
;          9828      DOCLN;
;          9829      end;
;          9830
;          9831  ENDTST;
```

```
22193
22194                                          .SBTTL  $T48 TEST CODE SECTION
22198 077674  004167  104712          $T48:   JSR     R1,$SAVE5               ;                       9676
22199 077700  162706  000012                  SUB     #12,SP
22200 077704  005067  115466                  CLR     P.CNT                   ;                       9717
22201 077710  012746  000011                  MOV     #11,-(SP)               ; *,CRC.NIB             9720
22202 077714  005066  000006                  CLR     6(SP)                   ; DODU.FLG              9721
22203 077720  005001                          CLR     R1                      ; LOOP                  9723
22204 077722  104402                  1$:     TRAP    2                       ;                       9724
22205 077724  152777  000040  115512          BISB    #40,@ML.REG+40          ;                       9725
22206 077732  016705  116074                  MOV     ML.DUT,R5
22207 077736  042705  177770                  BIC     #177770,R5
22208 077742  142777  000007  115474          BICB    #7,@ML.REG+40
22209 077750  150577  115470                  BISB    R5,@ML.REG+40
22210 077754  152777  000001  115542          BISB    #1,@ML.REG+120          ;                       9727
22211 077762  010105                          MOV     R1,R5                   ; LOOP,*                9729
22212 077764  006305                          ASL     R5
22213 077766  066507  077772                  ADD     2$(R5),PC
22214 077772  000012                  2$:     .WORD   3$-2$
22215 077774  000020                          .WORD   4$-2$
22216 077776  000042                          .WORD   5$-2$
22217 100000  000062                          .WORD   6$-2$
22218 100002  000104                          .WORD   7$-2$
22219 100004  012703  000006          3$:     MOV     #6,R3                   ; *,NIB.PAT             9733
22220 100010  000444                          BR      8$                      ;                       9729
22221 100012  012777  177777  115534  4$:     MOV     #-1,@ML.REG+150         ;                       9737
22222 100020  012777  177777  115536          MOV     #-1,@ML.REG+160         ;                       9738
22223 100026  012703  000001                  MOV     #1,R3                   ; *,NIB.PAT             9739
22224 100032  000433                          BR      8$                      ;                       9729
22225 100034  152777  000077  115512  5$:     BISB    #77,@ML.REG+150         ;                       9744
22226 100042  152777  000077  115514          BISB    #77,@ML.REG+160         ;                       9745
22227 100050  005003                          CLR     R3                      ; NIB.PAT               9746
22228 100052  000423                          BR      8$                      ;                       9729
22229 100054  052777  037400  115472  6$:     BIS     #37400,@ML.REG+150      ;                       9751
22230 100062  052777  000077  115474          BISB    #77,@ML.REG+160         ;                       9752
22231 100070  012703  000003                  MOV     #3,R3                   ; *,NIB.PAT             9753
22232 100074  000412                          BR      8$                      ;                       9729
22233 100076  005066  000002          7$:     CLR     2(SP)                   ; TMP.E1                9775
22234 100102  052766  037477  000002          BIS     #37477,2(SP)            ; *,TMP.E1              9777
22235 100110  016677  000002  115436          MOV     2(SP),@ML.REG+150       ; TMP.E1,*              9778
22236 100116  012703  000005                  MOV     #5,R3                   ; *,NIB.PAT             9780
22237 100122  004767  120164          8$:     JSR     PC,GD.BLK.XFER          ;                       9784
22238 100126  012777  000061  115250          MOV     #61,@ML.REG             ;                       9785
22239 100134  105777  115314          9$:     TSTB    @ML.REG+50
22240 100140  100375                          BPL     9$
22241 100142  152777  000010  115354          BISB    #10,@ML.REG+120         ;                       9787
22242 100150  004767  120136                  JSR     PC,GD.BLK.XFER          ;                       9788
22243 100154  012777  000071  115222          MOV     #71,@ML.REG             ;                       9789
```

```
22245                                   ;ML4AD                                                  29-Mar-1982 16:23:04   TOPS
22246                                   ;           TEST CODE SECTION                           29-Mar-1982 16:21:03   PA:<
22247
22248 100162 005066 000004                    CLR     4(SP)                   ; CRC.TSTED                                 9790
22249 100166 012704 000001                    MOV     #1,R4                   ; *,$$TMP2                                  9791
22250 100172 001411                    10$:   BEQ     13$
22251 100174 016705 101716                    MOV     L$DLY,R5                ; *,$$TMP1
22252 100200 001404                           BEQ     12$
22253 100202 005066 000012             11$:   CLR     12(SP)                  ; $$TMP
22254 100206 005305                           DEC     R5                      ; $$TMP1
22255 100210 001374                           BNE     11$
22256 100212 005304                    12$:   DEC     R4                      ; $$TMP2
22257 100214 000766                           BR      10$
22258 100216 011602                    13$:   MOV     (SP),R2                 ; CRC.NIB,*                                 9800
22259 100220 006202                           ASR     R2
22260 100222 006202                           ASR     R2
22261 100224 006202                           ASR     R2
22262 100226 062702 015342                    ADD     #PD.TEMP,R2
22263 100232 017767 115376 115102      14$:   MOV     @ML.REG+230,PD.TEMP     ;                                           9795
22264 100240 152777 000020 115256             BISB    #20,@ML.REG+120         ;                                           9796
22265 100246 012704 000001                    MOV     #1,R4                   ; *,$$TMP2                                  9797
22266 100252 001411                    15$:   BEQ     18$
22267 100254 016705 101636                    MOV     L$DLY,R5                ; *,$$TMP1
22268 100260 001404                           BEQ     17$
22269 100262 005066 000012             16$:   CLR     12(SP)                  ; $$TMP
22270 100266 005305                           DEC     R5                      ; $$TMP1
22271 100270 001374                           BNE     16$
22272 100272 005304                    17$:   DEC     R4                      ; $$TMP2
22273 100274 000766                           BR      15$
22274 100276 017767 115272 113070      18$:   MOV     @ML.REG+170,D1.TEMP
22275 100304 017767 115274 113064             MOV     @ML.REG+200,D2.TEMP
22276 100312 017767 115246 113060             MOV     @ML.REG+160,E2.TEMP
22277 100320 010246                           MOV     R2,-(SP)                ;                                           9800
22278 100322 016646 000002                    MOV     2(SP),-(SP)             ; CRC.NIB,*
22279 100326 042716 177770                    BIC     #177770,(SP)
22280 100332 012746 000001                    MOV     #1,-(SP)
22281 100336 005046                           CLR     -(SP)
22282 100340 004767 103270                    JSR     PC,BL$GT2
22283 100344 062706 000010                    ADD     #10,SP
22284 100350 005700                           TST     R0
22285 100352 001102                           BNE     21$
22286 100354 005266 000004                    INC     4(SP)                   ; CRC.TSTED                                 9803
22287 100360 011646                           MOV     (SP),-(SP)              ; CRC.NIB,*                                 9804
22288 100362 010346                           MOV     R3,-(SP)                ; NIB.PAT,*
22289 100364 012746 000016                    MOV     #16,-(SP)
22290 100370 060616                           ADD     SP,(SP)                 ; ERR.FLG,*
22291 100372 004767 120000                    JSR     PC,TST.LNG.WRD
22292 100376 026627 000016 000001             CMP     16(SP),#1               ; ERR.FLG,*                                 9806
22293 100404 001063                           BNE     20$
22294 100406 005267 114764                    INC     P.CNT                   ;                                           9808
22295 100412 026767 114760 114760             CMP     P.CNT,LIMIT
22296 100420 003403                           BLE     19$
22297 100422 062706 000006                    ADD     #6,SP
22298 100426 000460                           BR      22$
22299 100430 104455                    19$:   TRAP    55                      ;                                           9810
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
                                        .WORD   203
100434  012750                          .WORD   SYNC
100436  026302                          .WORD   DUMPER
100440  012746  010630                  MOV     #WRD.10,-(SP)           ;                                                       9811
100444  012746  011532                  MOV     #WRD.73,-(SP)
100450  012746  010774                  MOV     #WRD.24,-(SP)
100454  012746  011446                  MOV     #WRD.64,-(SP)
100460  012746  010270                  MOV     #FOR.FMT,-(SP)
100464  012746  000005                  MOV     #5,-(SP)
100470  010600                          MOV     SP,R0                   ; SP,*
100472  104414                          TRAP    14
100474  012716  000010                  MOV     #10,(SP)                ;                                                       9812
100500  016705  112674                  MOV     NIB.SAVE+4,R5
100504  006205                          ASR     R5
100506  006205                          ASR     R5
100510  006205                          ASR     R5
100512  006205                          ASR     R5
100514  000305                          SWAB    R5
100516  042705  177770                  BIC     #177770,R5
100522  010546                          MOV     R5,-(SP)
100524  010346                          MOV     R3,-(SP)                ; NIB.PAT,*
100526  012746  006650                  MOV     #FMT.5,-(SP)
100532  012746  000004                  MOV     #4,-(SP)
100536  010600                          MOV     SP,R0                   ; SP,*
100540  104414                          TRAP    14
100542  012766  000001  000040          MOV     #1,40(SP)               ; *,DODU.FLG                                            9813
100550  062706  000024                  ADD     #24,SP                  ;                                                       9808
100554  062706  000006          20$:    ADD     #6,SP                   ;                                                       9802
100560  026627  000004  000006  21$:    CMP     4(SP),#6                ; CRC.TSTED,*                                           9819
100566  001221                          BNE     14$
100570  104467                  22$:    TRAP    67
100572  006000                          ROR     R0
100574  103002                          BHIS    24$
100576  000167  177120          23$:    JMP     1$
100602  005201                  24$:    INC     R1                      ; LOOP                                                  9723
100604  020127  000004                  CMP     R1,#4                   ; LOOP,*
100610  003772                          BLE     23$
100612  026627  000006  000001          CMP     6(SP),#1                ; DODU.FLG,*                                            9824
100620  001004                          BNE     25$
100622  016700  115202                  MOV     ML.LUN,R0               ;                                                       9827
100626  104451                          TRAP    51
100630  104444                          TRAP    44
100632  062706  000014          25$:    ADD     #14,SP                  ;                                                       9676
100636  000207                          RTS     PC

                                ; Routine Size:  242 words
                                ; Maximum stack depth per invocation:  25 words
```

SEQ 0463

```
 22362
 22363                                          .SBTTL  T48 TEST CODE SECTION
 22367 100640                           T48::
 22368 100640  004767  177030           1$:     JSR     PC,$T48                         ;                9829
 22369 100644  104466                           TRAP    66
 22370 100646  006000                           ROR     R0
 22371 100650  103773                           BLO     1$
 22372 100652  000207                           RTS     PC
 22373
 22374                                  ; Routine Size:  6 words
 22375                                  ; Maximum stack depth per invocation:  0 words
 22380
 22381
 22382 ;          9832  !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (101)

```
;ML4AD
;                    TEST CODE SECTION

                !
BGNTST;

                !++
                ! TEST NUMBER:  TST 49
                !
                ! TEST NAME:    UNIQUE NIBBLE CRC GENERATION TEST
                !
                ! TEST DESCRIPTION:
                !       TEST CRC CODES GENERATED FOR ONE CRC
                !       GROUP (52 UNIQUE UNIBUS NIBBLES) BY:
                !
                !       1.  DOING MASS BUS WRITE TRANSFERSS WITH THE
                !       FIRST IO_BUF CRC GROUP HAVING 51 NIBBLES
                !       OF %B'0000' PATTERN AND THE REMAINING NIBBLE
                !           WITH PATTERNS OF %B'0000' TO %B'1111'.
                !
                !       2.  THEN IN DATA DIAGNOSTIC MODE CALCULATE
                !           THE CRC CODE FOR THE FIRST CRC
                !           GROUP AND COMPARE IT AGAINST THE
                !           HARDWARE GENERATED CRC CODE.
                !
                !       3.  REPEAT PATTERN OF %B'0000' TO %B'1111'
                !           FOR EACH OF THE 52 NIBBLES.
                !
                ! IMPLICIT INPUTS:
                !       IO_BUF
                !       A VECTOR OF 256 WORDS WHERE DATA
                !       FOR MBUS READS AND WRITES TRANSFERS
                !       IS FOUND.
                !
                !
                !--

                local
                    DODU_FLG,                   !DROP UNIT FLAG
                    TEMP,                       !TEMPORARY STORAGE LOCATION
                    NIB_SEL;                    !SELECTS 4 BITS 'ONE NIBBLE' FROM THE IO_BUF

                DODU_FLG = ZEROES;              !CLEAR THE DROP UNIT FLAG
                CLR_THRESHOLD;                  !CLEAR ERROR PRINT THRESHOLD

                incr CNT from 0 to 13 do        !CLEAR THE FIRST 14 IO_BUF WORDS
                    IO_BUF [.CNT] = ZEROES;

                incr WRD_CNT from 0 to 12 do    !DO THIS TEST ON 13 IO_BUF WORDS
                    begin
                    NIB_SEL = -4;               !RESET THE NIBBLE SELECTOR

                    incr NIB_TST from 0 to 3 do !DO THIS TEST ON 4 NIBBLES PER WORD
                        begin
```

```
22440 ;ML4AD
22441 ;                 TEST CODE SECTION
22442 ;
22443 ;          9885           if (.WRD_CNT eql 12) and (.NIB_TST eql 2) then exitloop;        !THE LAST 2 NIBBLES ARE NOT TESTED
22444 ;          9886
22445 ;          9887           NIB_SEL = .NIB_SEL + 4;                                         !INCREMENT THE NIBBLE SELECTED
22446 ;          9888           TEMP = ZEROES;                                                  !CLEAR  TEMP LOCATION
22447 ;          9889
22448 ;          9890           incr NIB_PAT from %b'0000' to %b'1111' by %b'0001' do   !RUN PATTERNS ON THEIS NIBBLE
22449 ;          9891               begin
22450 ;          9892               BGNSUB;                                                     !START OF SCOPE LOOP
22451 ;          9893               TEMP<.NIB_SEL, 4> = .NIB_PAT;                               !LOAD PATTERN INTO TEMP AT THIS NIBBLE
22452 ;          9894               IO_BUF [.WRD_CNT] = .TEMP;                                  !LOAD THE GENERATED PATTERN INTO THE IO_BUF
22453 ;          9895               CLR_MBUS;
22454 ;          9896               GD_BLK_XFER ();                                             !SET UP A GOOD BLOCK TRANSFER
22455 ;          9897               MLCS1 = write;                                              !DO A WRITE TRANSFER
22456 ;          9898               TIME_OUT_LOOP;                                              !WAIT FOR THE TRANSFER TO COMPLETE
22457 ;          9899               A_GEN = ZEROES;                                             !CLEAR THE GEN & CAL STORAGE LOCATIONS
22458 ;          9900               B_GEN = ZEROES;
22459 ;          9901               P_GEN = ZEROES;
22460 ;          9902               A_CAL = ZEROES;
22461 ;          9903               B_CAL = ZEROES;
22462 ;          9904               P_CAL = ZEROES;
22463 ;          9905               CAL_CRC ();                                                 !CALL ROUTINE TO CALCULATE THE CRC CODE FOR THIS PATTERN
22464 ;          9906
22465 ;          9907               if ERR_CHK_CRC ()                                           !NOW SEE IF THE GEN CRC IS SAME AS CAL CRC
22466 ;          9908               then
22467 ;          9909                   begin                                                   !REPORT THE ERROR IF NOT THE SAME
22468 ;          9910                   CMP_THRESHOLD;                                          !COMPARE ERROR PRINT THRESHOLD
22469 ;          9911                   ERRDF (132, SYNC, DUMPER);
22470 ;          9912                   PRINTB (THR_FMT, WRD_5, WRD_64, WRD_8);
22471 ;          9913                   PRINTB (FMT_19, .B_GEN, .A_GEN, .P_GEN);
22472 ;          9914                   PRINTB (FMT_20, .B_CAL, .A_CAL, .P_CAL);
22473 ;          9915                   DODU_FLG = ONE;
22474 ;          9916                   end;
22475 ;          9917
22476 ;          9918               ENDSUB;                                                     !END OF SCOPE LOOP
22477 ;          9919               end;
22478 ;          9920
22479 ;          9921           IO_BUF [.WRD_CNT] = ZEROES;                                     !CLEAR THIS IO_BUF WORD FO NEXT NIBBLE
22480 ;          9922           end;
22481 ;          9923
22482 ;          9924       end;
22483 ;          9925
22484 ;          9926   if .DODU_FLG IS_SET                                                     !DROP THIS UNIT IF THE DODU FLAG GOT SET
22485 ;          9927   then
22486 ;          9928       begin
22487 ;          9929       DODU (.ML_LUN);
22488 ;          9930       DOCLN;
22489 ;          9931       end;
22490 ;          9932
22491 ;          9933   ENDTST;
```

```
                                                                                29-Mar-1982 16:23:04   TOPS
                                                                                29-Mar-1982 16:21:03   PA:<
 22496                          ;ML4AD
 22497                          ;                TEST CODE SECTION
 22498
 22499
 22500                                  .SBTTL  $T49 TEST CODE SECTION
 22504 100654 004167 103732     $T49:   JSR     R1,$SAVE5                                                       9831
 22505 100660 024646                    CMP     -(SP),-(SP)
 22506 100662 005066 000002             CLR     2(SP)                   ; DODU.FLG                              9872
 22507 100666 005067 114504             CLR     P.CNT
 22508 100672 005000                    CLR     R0                      ; CNT                                   9875
 22509 100674 010001            1$:     MOV     R0,R1                   ; CNT,*                                 9876
 22510 100676 006301                    ASL     R1
 22511 100700 005061 014022             CLR     IO.BUF(R1)
 22512 100704 005200                    INC     R0                      ; CNT                                   9875
 22513 100706 020027 000015             CMP     R0,#15                  ; CNT,*
 22514 100712 003770                    BLE     1$
 22515 100714 005002                    CLR     R2                      ; WRD.CNT                               9878
 22516 100716 012716 177774     2$:     MOV     #-4,(SP)                ; *,NIB.SEL                             9880
 22517 100722 010201                    MOV     R2,R1                   ; WRD.CNT,*                             9894
 22518 100724 006301                    ASL     R1
 22519 100726 005004                    CLR     R4                      ; NIB.TST                               9882
 22520 100730 020227 000014     3$:     CMP     R2,#14                  ; WRD.CNT,*                             9885
 22521 100734 001003                    BNE     4$
 22522 100736 020427 000002             CMP     R4,#2                   ; NIB.TST,*
 22523 100742 001575                    BEQ     10$
 22524 100744 062716 000004     4$:     ADD     #4,(SP)                 ; *,NIB.SEL                             9887
 22525 100750 005005                    CLR     R5                      ; TEMP                                  9888
 22526 100752 005003                    CLR     R3                      ; NIB.PAT                               9890
 22527 100754 104402            5$:     TRAP    2                       ;                                       9891
 22528 100756 010546                    MOV     R5,-(SP)                ; TEMP,*                                9893
 22529 100760 016646 000002             MOV     2(SP),-(SP)             ; NIB.SEL,*
 22530 100764 012746 000004             MOV     #4,-(SP)
 22531 100770 010346                    MOV     R3,-(SP)                ; NIB.PAT,*
 22532 100772 004767 103000             JSR     PC,BL$PU1
 22533 100776 010005                    MOV     R0,R5                   ; *,TEMP                                9891
 22534 101000 010561 014022             MOV     R5,IO.BUF(R1)           ; TEMP,*                                9894
 22535 101004 152777 000040 114432      BISB    #40,@ML.REG+40
 22536 101012 016700 115014             MOV     ML.DUT,R0
 22537 101016 042700 177770             BIC     #177770,R0
 22538 101022 142777 000007 114414      BICB    #7,@ML.REG+40
 22539 101030 150077 114410             BISB    R0,@ML.REG+40
 22540 101034 004767 117252             JSR     PC,GD.BLK.XFER
 22541 101040 012777 000061 114336      MOV     #61,@ML.REG             ;                                       9896
 22542 101046 105777 114402     6$:     TSTB    @ML.REG+50              ;                                       9897
 22543 101052 100375                    BPL     6$
 22544 101054 005067 114310             CLR     A.GEN                   ;                                       9899
 22545 101060 005067 114306             CLR     B.GEN                   ;                                       9900
 22546 101064 005067 114304             CLR     P.GEN                   ;                                       9901
 22547 101070 005067 114266             CLR     A.CAL                   ;                                       9902
 22548 101074 005067 114264             CLR     B.CAL                   ;                                       9903
 22549 101100 005067 114262             CLR     P.CAL                   ;                                       9904
 22550 101104 004767 115732             JSR     PC,CAL.CRC              ;                                       9905
```

```
                                        ;ML4AD                                                                  29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                                       29-Mar-1982 16:21:03    PA:<
101110  004767  116326                          JSR     PC,ERR.CHK.CRC          ;                                                       9907
101114  006000                                  ROR     R0
101116  103066                                  BCC     8$
101120  005267  114252                          INC     P.CNT                   ;                                                       9909
101124  026767  114246  114246                  CMP     P.CNT,LIMIT
101132  003403                                  BLE     7$
101134  062706  000010                          ADD     #10,SP
101140  000462                                  BR      9$
101142  104455                          7$:     TRAP    55                      ;                                                       9911
101144  000204                                  .WORD   204
101146  012750                                  .WORD   SYNC
101150  026302                                  .WORD   DUMPER
101152  012746  010602                          MOV     #WRD.8,-(SP)            ;                                                       9912
101156  012746  011446                          MOV     #WRD.64,-(SP)
101162  012746  010552                          MOV     #WRD.5,-(SP)
101166  012746  010256                          MOV     #THR.FMT,-(SP)
101172  012746  000004                          MOV     #4,-(SP)
101176  010600                                  MOV     SP,R0                   ; SP,*
101200  104414                                  TRAP    14
101202  016716  114166                          MOV     P.GEN,(SP)              ;                                                       9913
101206  016746  114156                          MOV     A.GEN,-(SP)
101212  016746  114154                          MOV     B.GEN,-(SP)
101216  012746  007632                          MOV     #FMT.19,-(SP)
101222  012746  000004                          MOV     #4,-(SP)
101226  010600                                  MOV     SP,R0                   ; SP,*
101230  104414                                  TRAP    14
101232  016716  114130                          MOV     P.CAL,(SP)              ;                                                       9914
101236  016746  114120                          MOV     A.CAL,-(SP)
101242  016746  114116                          MOV     B.CAL,-(SP)
101246  012746  007702                          MOV     #FMT.20,-(SP)
101252  012746  000004                          MOV     #4,-(SP)
101256  010600                                  MOV     SP,R0                   ; SP,*
101260  104414                                  TRAP    14
101262  012766  000001  000044                  MOV     #1,44(SP)               ; *,DODU.FLG                                            9915
101270  062706  000032                          ADD     #32,SP                  ;                                                       9909
101274  062706  000010                  8$:     ADD     #10,SP                  ;                                                       9891
101300  104467                                  TRAP    67                      ;                                                       9916
101302  006000                                  ROR     R0
101304  103623                                  BLO     5$
101306  005203                          9$:     INC     R3                      ; NIB.PAT                                               9890
101310  020327  000017                          CMP     R3,#17                  ; NIB.PAT,*
101314  003617                                  BLE     5$
101316  005061  014022                          CLR     10.BUF(R1)              ;                                                       9921
101322  005204                                  INC     R4                      ; NIB.TST                                               9882
101324  020427  000003                          CMP     R4,#3                   ; NIB.TST,*
101330  003002                                  BGT     10$
101332  000167  177372                          JMP     3$
101336  005202                          10$:    INC     R2                      ; WRD.CNT                                               9878
101340  020227  000014                          CMP     R2,#14                  ; WRD.CNT,*
101344  003002                                  BGT     11$
101346  000167  177344                          JMP     2$
101352  026627  000002  000001  11$:    CMP     2(SP),#1                ; DODU.FLG,*                                            9926
```

```
                                ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
                                ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<

101360  001004                          BNE     12$
101362  016700  114442                  MOV     ML.LUN,R0                       ;                               9929
101366  104451                          TRAP    51
101370  104444                          TRAP    44
101372  022626                  12$:    CMP     (SP)+,(SP)+                     ;                               9831
101374  000207                          RTS     PC

                                ; Routine Size:  169 words
                                ; Maximum stack depth per invocation:  25 words


                                        .SBTTL  T49 TEST CODE SECTION
101376                          T49::
101376  004767  177252          1$:     JSR     PC,$T49                         ;                               9931
101402  104466                          TRAP    66
101404  006000                          ROR     R0
101406  103773                          BLO     1$
101410  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


;       9934    !
;       9935    !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (102)

```
22652 ;ML4AD
22653 ;                 TEST CODE SECTION
22654
22655 ;          9936  !
22656 ;          9937  BGNTST;
22657 ;          9938
22658 ;          9939  !++
22659 ;          9940  ! TEST NUMBER:  TST 50
22660 ;          9941  !
22661 ;          9942  ! TEST NAME:            UNIQUE WORD CRC GENERATION TEST
22662 ;          9943  !
22663 ;          9944  ! TEST DESCRIPTION:
22664 ;          9945  !       TEST CRC CODES GENERATED FOR ONE
22665 ;          9946  !       CRC GROUP (13 UNIBUS WORDS) BY:
22666 ;          9947  !
22667 ;          9948  !       1.  DOING MASS BUS WRITE TRANSFERS
22668 ;          9949  !           USING VARIOUS WORD DATA PATTERENS.
22669 ;          9950  !
22670 ;          9951  !       2.  THEN IN DATA DIAGNOSTIC MODE
22671 ;          9952  !           CALCULATE THE CRC CODE FOR THE
22672 ;          9953  !           FIRST CRC GROUP AND COMPARE IT TO
22673 ;          9954  !           THE HARDWARE GENERATED CRC CODE.
22674 ;          9955  !
22675 ;          9956  ! IMPLICIT INPUTS:
22676 ;          9957  !       IO_BUF
22677 ;          9958  !       A  VECTOR OF 256 WORDS WHERE DATA
22678 ;          9959  !       FOR MBUS READS AND WRITES TRANSFERS
22679 ;          9960  !       IS FOUND.
22680 ;          9961  !
22681 ;          9962  !--
22682 ;          9963
22683 ;          9964  local
22684 ;          9965      DODU_FLG,                   !DROP UNIT FLAG
22685 ;          9966      TST_PAT;                    !TEST PATTERN SAVE LOCATION
22686 ;          9967
22687 ;          9968  DODU_FLG = ZEROES;              !CLEAR THE DROP UNIT FLAG
22688 ;          9969
22689 ;          9970  incr PAT_SEL from 0 to 3 do     !RUN THE TEST ON 4 PATTERNS
22690 ;          9971      begin
22691 ;          9972      BGNSUB;                     !START OF SCOPE LOOP
22692 ;          9973      CLR_MBUS;
22693 ;          9974
22694 ;          9975      case .PAT_SEL from 0 to 3 of    !SELECT ONE OF THE 4 PATTERNS
22695 ;          9976          set
22696 ;          9977
22697 ;          9978          [0] :
22698 ;          9979              TST_PAT = ONES;
22699 ;          9980
22700 ;          9981          [1] :
22701 ;          9982              TST_PAT = %o'052525';
22702 ;          9983
22703 ;          9984          [2] :
22704 ;          9985              TST_PAT = %o'125252';
22705 ;          9986
22706 ;          9987          [3] :
```

```
;ML4AD                                                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (102)
;
;       9988            TST_PAT = %o'070707';
;       9989        tes;
;       9990
;       9991    incr CNT from 0 to 255 do                       !LOAD THE FIRST 13 IO_BUF WORDS WITH TST_PAT
;       9992        IO_BUF [.CNT] = .TST_PAT;
;       9993
;       9994    GD_BLK_XFER ();                                 !SET UP A GOOD BLOCK TRANSFER
;       9995    ML.CS1 = write;                                 !DO A WRITE TRANSFER
;       9996    TIME_OUT_LOOP;                                  !WAIT FOR THE TRANSFER TO COMPLETE
;       9997    A_GEN = ZEROES;                                 !CLEAR THE GEN & CAL SAVE LOCATIONS
;       9998    B_GEN = ZEROES;
;       9999    P_GEN = ZEROES;
;       10000   A_CAL = ZEROES;
;       10001   B_CAL = ZEROES;
;       10002   P_CAL = ZEROES;
;       10003   CAL_CRC ();                                     !CALCULATE WHAT THE CRC CODE SHOULD BE
;       10004
;       10005   if ERR_CHK_CRC ()                               !NOW SEE IF THE GENERATED CRC CODE IS THE SAME
;       10006   then
;       10007       begin                                       !REPORT THE ERROR IF NOT THE SAME
;       10008       ERRDF (133, SYNC, DUMPER);
;       10009       PRINTB (THR_FMT, WRD_5, WRD_64, WRD_8);
;       10010       PRINTB (FMT_19, .B_GEN, .A_GEN, .P_GEN);
;       10011       PRINTB (FMT_20, .B_CAL, .A_CAL, .P_CAL);
;       10012       DODU_FLG = ONE;
;       10013       end;
;       10014
;       10015   ENDSUB;                                         !END OF SCOPE LOOP
;       10016   end;
;       10017
;       10018 if .DODU_FLG IS_SET                               !DROP THIS UNIT IF THE DODU FLAG GOT SET
;       10019 then
;       10020     begin
;       10021     DODU (.ML_LUN);
;       10022     DOCLN;
;       10023     end;
;       10024
;       10025 ENDTST;
;
;                                       .SBTTL  ST50 TEST CODE SECTION
;101412  004167  103174          ST50:  JSR     R1,$SAVE5                                                       ;9933
;101416  005005                         CLR     R5                      ;DODU.FLG                               ;996R
;101420  005004                         CLR     R4                      ;PAT.SEL                                ;9970
;101422  104402                  1$:    TRAP    2                       ;                                       ;9971
;101424  152777  000040  114012         BISB    #40,@ML.REG+40          ;                                       ;9972
;101432  016703  114374                 MOV     ML.DUT,R3
```

```
                                        :ML4AD                                                                  29-Mar-1982 16:23:04    TOPS
                                        :               TEST CODE SECTION                                       29-Mar-1982 16:21:03    PA:<
                                        BIC     #177770,R3
                                        BICB    #7,@ML.REG+40
                                        BISB    R3,@ML.REG+40
                                        MOV     R4,R3                   ; PAT.SEL,*                                             9975
                                        ASL     R3
                                        ADD     2$(R3),PC
                                2$:     .WORD   3$-2$
                                        .WORD   4$-2$
                                        .WORD   5$-2$
                                        .WORD   6$-2$
                                3$:     MOV     #-1,R1                  ; *,TST.PAT                                             9979
                                        BR      7$                      ;                                                       9975
                                4$:     MOV     #52525,R1               ; *,TST.PAT                                             9982
                                        BR      7$                      ;                                                       9975
                                5$:     MOV     #-52526,R1              ; *,TST.PAT                                             9985
                                        BR      7$                      ;                                                       9975
                                6$:     MOV     #70707,R1               ; *,TST.PAT                                             9988
                                7$:     CLR     R2                      ; CNT                                                   9991
                                8$:     MOV     R2,R3                   ; CNT,*                                                 9992
                                        ASL     R3
                                        MOV     R1,IO.BUF(R3)           ; TST.PAT,*
                                        INC     R2                      ; CNT                                                   9991
                                        CMP     R2,#377                 ; CNT,*
                                        BLE     8$
                                        JSR     PC,GD.BLK.XFER          ;                                                       9994
                                        MOV     #61,@ML.REG             ;                                                       9995
                                9$:     TSTB    @ML.REG+50
                                        BPL     9$
                                        CLR     A.GEN                   ;                                                       9997
                                        CLR     B.GEN                   ;                                                       9998
                                        CLR     P.GEN                   ;                                                       9999
                                        CLR     A.CAL                   ;                                                       1000
                                        CLR     B.CAL                   ;                                                       1000
                                        CLR     P.CAL                   ;                                                       1000
                                        JSR     PC,CAL.CRC              ;                                                       1000
                                        JSR     PC,ERR.CHK.CRC          ;                                                       1000
                                        ROR     R0
                                        BCC     10$
                                        TRAP    55                      ;                                                       1000
                                        .WORD   205
                                        .WORD   SYNC
                                        .WORD   DUMPER
                                        MOV     #WRD.8,-(SP)            ;                                                       1000
                                        MOV     #WRD.64,-(SP)
                                        MOV     #WRD.5,-(SP)
                                        MOV     #THR.FMT,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        MOV     P.GEN,(SP)              ;                                                       1001
                                        MOV     A.GEN,-(SP)
                                        MOV     B.GEN,-(SP)
```

```
                                        ;ML4AD
                                        ;               TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS
                                                                                                         29-Mar-1982 16:21:03    PA:<
22820
22821
22822
22823 101704  012746  007632                    MOV     #FMT.19,-(SP)
22824 101710  012746  000004                    MOV     #4,-(SP)
22825 101714  010600                            MOV     SP,R0                           ; SP,*
22826 101716  104414                            TRAP    14
22827 101720  016716  113442                    MOV     P.CAL,(SP)                      ;                                        1001
22828 101724  016746  113432                    MOV     A.CAL,-(SP)
22829 101730  016746  113430                    MOV     B.CAL,-(SP)
22830 101734  012746  007702                    MOV     #FMT.20,-(SP)
22831 101740  012746  000004                    MOV     #4,-(SP)
22832 101744  010600                            MOV     SP,R0                           ; SP,*
22833 101746  104414                            TRAP    14
22834 101750  012705  000001                    MOV     #1,R5                           ; *,DODU.FLG                             1001
22835 101754  062706  000032                    ADD     #32,SP                          ;                                        1000
22836 101760  104467              10$:          TRAP    67                              ;                                        1001
22837 101762  006000                            ROR     R0
22838 101764  103616                            BLO     1$
22839 101766  005204                            INC     R4                              ; PAT.SEL                                9970
22840 101770  020427  000003                    CMP     R4,#3                           ; PAT.SEL,*
22841 101774  003612                            BLE     1$
22842 101776  005305                            DEC     R5                              ; DODU.FLG                               1001
22843 102000  001004                            BNE     11$
22844 102002  016700  114022                    MOV     ML.LUN,R0                       ;                                        1002
22845 102006  104451                            TRAP    51
22846 102010  104444                            TRAP    44
22847 102012  000207              11$:          RTS     PC                              ;                                        9933
22848
22849                                   ; Routine Size:  129 words
22850                                   ; Maximum stack depth per invocation:  19 words
22855
22856
22860
22861                                           .SBTTL  T50 TEST CODE SECTION
22865 102014                            T50::
22866 102014  004767  177372            1$:     JSR     PC,$T50                         ;                                        1002
22867 102020  104466                            TRAP    66
22868 102022  006000                            ROR     R0
22869 102024  103773                            BLO     1$
22870 102026  000207                            RTS     PC
22871
22872                                   ; Routine Size:  6 words
22873                                   ; Maximum stack depth per invocation:  0 words
```

```
22882
22883
22884 ;          10026 !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (103)

```
;ML4AD
;                TEST CODE SECTION

!
BGNTST;

!++
! TEST NUMBER:  TST 51
!
! TEST NAME:            CORRECTABLE ERROR SYNDRONE DECODE TEST
!
! TEST DESCRIPTION:
!       TEST SYNDRONE DECODE AND ERROR
!       CORRECTION TO DECODE AND CORRECT
!       SINGLE BIT AND MULTIPLE BIT
!       CHANNEL ERRORS BY:
!
!       1.  VIA ECC DIAG MODE FORCE SYNDRONE
!           BITS TO DECODE SINGLE BIT AND
!           MULTIPLE BIT CHANNEL ERRORS.
!
!       2.  THEN DO A MASS BUS READ TRANSFER
!           AND EXAM THE IO BUF FOR CORRECT
!           BIT COMPLIMENTING.
!
! IMPLICIT INPUTS:
!       IO_BUF
!       A  VECTOR OF 256 WORDS WHERE DATA
!       FOR MBUS READS AND WRITES TRANSFERS
!       IS FOUND.
!
!
!--

local
    BITS_XFERED;                        !POINTS TO THE CHANNEL BEING TESTED

CLR_THRESHOLD;                          !CLEAR ERROR PRINT THRESHOLD
BAI = ONE;                              !BEFORE WE START LETS CLEAR OUT THE ML11'S
IO_BUF = ZEROES;                        !GOOD BLOCK BY WRITTING ZEROES TO IT
GD_BLK_XFER ();
MLCS1 = write;
TIME_OUT_LOOP;
BITS_XFERED = -1;                       !RESET THE CHANNEL POINTER

incr PLOG from 0 to 5 do                !TEST ONE CRC GROUP '6 ARRAY WORDS'
    begin

    incr CHANNEL from 0 to 35 do        !TEST 36 BITS IN EACH ARRAY WORD
        begin
        BGNSUB;                         !START OF SCOPE LOOP
        CLR_MBUS;                       !CLEAR THE DRIVE
        BITS_XFERED = .BITS_XFERED + 1; !INCREMENT THE CHANNEL POINTER

        incr CNT from 0 to 255 do       !CLEAR THE FIRST CRC GROUP IN THE IO_BUF
```

```
22942 ;ML4AD
22943 ;                 TEST CODE SECTION
22944
22945 ;          10079              IO_BUF [.CNT] = ZEROES;
22946 ;          10080
22947 ;          10081          FORCE_REM (.PLOG, .CHANNEL);          !FORCE THIS BIT IN THIS WORD IN ERROR
22948 ;          10082          GD_BLK_XFER ();                       !SET UP A GOOD BLOCK TRANSFER
22949 ;          10083          ECC_DIS = ZERO;                       !CLEAR ECC DISABLE
22950 ;          10084          MLCS1 = read;                         !DO A READ TRANSFER
22951 ;          10085          TIME_OUT_LOOP;                        !WAIT FOR THE TRANSFER TO COMPLETE
22952 ;          10086
22953 ;          10087          if not FIND_COMP_BIT (.BITS_XFERED)   !SEARCH THE IO_BUF FOR THE COMP BIT
22954 ;          10088          then
22955 ;          10089              begin                             !REPORT AN ERROR IF THIS BIT IS NOT COMP
22956 ;          10090              CMP_THRESHOLD;                    !COMPARE ERROR PRINT THRESHOLD
22957 ;          10091              ERRDF (134, SYNC, DUMPER);
22958 ;          10092              PRINTB (THR_FMT, WRD_74, WRD_75, PHR_1);
22959 ;          10093              PRINTB (FMT_21, .PLOG, .CHANNEL);
22960 ;          10094              end;
22961 ;          10095
22962 ;          10096          incr WRD_CNT from 0 to 12 do          !SEE IF ANY OTHER BITS GOT COMP'ED
22963 ;          10097
22964 ;          10098              if .IO_BUF [.WRD_CNT] neq ZEROES  !TEST THE IO_BUF FOR ZEROES
22965 ;          10099              then
22966 ;          10100                  begin                         !REPORT AN ERROR IF ANY OTHERS ARE SET
22967 ;          10101                  CMP_THRESHOLD;                !COMPARE ERROR PRINT THRESHOLD
22968 ;          10102                  ERRDF (135, SYNC, DUMPER);
22969 ;          10103                  PRINTB (FOR_FMT, WRD_61, WRD_74, WRD_75, PHR_5);
22970 ;          10104                  PRINTB (FMT_21, .PLOG, .CHANNEL);
22971 ;          10105                  end;
22972 ;          10106
22973 ;          10107              ENDSUB;                           !END OF SCOPE LOOP
22974 ;          10108              end;
22975 ;          10109
22976 ;          10110      end;
22977 ;          10111
22978 ;          10112 BITS_XFERED = -1;                              !RESET THE CHANNEL POINTER
22979 ;          10113
22980 ;          10114 incr CHANNEL from 0 to 35 do                   !TEST 36 CHANNELS FOR MULTIPLE BIT ERRORS
22981 ;          10115      begin
22982 ;          10116      BGNSUB;                                   !START OF SCOPE LOOP
22983 ;          10117      CLR_MBUS;                                 !CLEAR THE DRIVE
22984 ;          10118      BITS_XFERED = .BITS_XFERED + 1;           !INCREMENT THE CHANNEL POINTER
22985 ;          10119
22986 ;          10120      incr CNT from 0 to 20 do                  !CLEAR THE FIRST CRC GROUP IN THE IO_BUF
22987 ;          10121          IO_BUF [.CNT] = ZEROES;
22988 ;          10122
22989 ;          10123      FORCE_REM (58, .CHANNEL);                 !FORCE ALL 6 BITS IN THIS CHANNEL IN ERROR
22990 ;          10124      GD_BLK_XFER ();                           !SET UP A GOOD BLOCK TRANSFER
22991 ;          10125      ECC_DIS = ZERO;                           !CLEAR ECC DISABLE
22992 ;          10126      MLCS1 = read;                             !DO A READ TRANSFER
22993 ;          10127      TIME_OUT_LOOP;                            !WAIT FOR THE TRANSFER TO CMOPLETE
22994 ;          10128
22995 ;          10129      incr BIT_OFFSET from 0 to 180 by 36 do    !SEE IF EVERY 36TH BIT GOT COMP'ED
22996 ;          10130
```

```
;ML4AD                                                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (103)

;          10131          if not FIND_COMP_BIT (.BITS_XFERED + .BIT_OFFSET)       !SEARCH THE IO_BUF FOR THE COMP'ED BITS
;          10132          then
;          10133              begin                                               !REPORT AN ERROR IF ANY OF THE BITS ARE NOT COMP'ED
;          10134              CMP_THRESHOLD;                                      !COMPARE ERROR PRINT THRESHOLD
;          10135              ERRDF (134, SYNC, DUMPER);
;          10136              PRINTB (THR_FMT, WRD_74, WRD_75, PHR_1);
;          10137              PRINTB (FMT_21, 58, .CHANNEL);
;          10138              end;
;          10139
;          10140      incr WRD_CNT from 0 to 12 do                                !ALSO SEE IF ANY OTHER BIT GOT COMP'ED
;          10141
;          10142          if .IO_BUF [.WRD_CNT] neq ZEROES                        !TEST THE IO_BUF FOR ALL ZEROES
;          10143          then
;          10144              begin                                               !REPORT AN ERROR IF ANY OTHERS ARE COMP'ED
;          10145              CMP_THRESHOLD;                                      !COMPARE ERROR PRINT THRESHOLD
;          10146              ERRDF (135, SYNC, DUMPER);
;          10147              PRINTB (FIV_FMT, WRD_61, WRD_74, WRD_75, WRD_12, WRD_67);
;          10148              PRINTB (FMT_21, 58, .CHANNEL);
;          10149              end;
;          10150
;          10151      ENDSUB;                                                     !END OF SCOPE LOOP
;          10152      end;
;          10153
;          10154 ENDTST;
23028
23029                                       .SBTTL  $T51 TEST CODE SECTION
23033 102030  004167  102556        $T51:   JSR     R1,$SAVE5                   ;                                   1002
23034 102034  005746                        TST     -(SP)
23035 102036  005067  113334                CLR     P.CNT                       ;                                   1005
23036 102042  152777  000010  113374        BISB    #10,@ML.REG+40              ;                                   1006
23037 102050  005067  111746                CLR     IO.BUF                      ;                                   1006
23038 102054  004767  116232                JSR     PC,GD.BLK.XFER              ;                                   1006
23039 102060  012777  000061  113316        MOV     #61,@ML.REG                 ;                                   1006
23040 102066  105777  113362          1$:   TSTB    @ML.REG+50                  ;                                   1006
23041 102072  100375                        BPL     1$
23042 102074  012716  177777                MOV     #-1,(SP)                    ; *,BITS.XFERED                     1006
23043 102100  005005                        CLR     R5                          ; PLOG                              1006
23044 102102  005004                  2$:   CLR     R4                          ; CHANNEL                           1007
23045 102104  104402                  3$:   TRAP    2                           ;                                   1007
23046 102106  152777  000040  113330        BISB    #40,@ML.REG+40              ;                                   1007
23047 102114  016703  113712                MOV     ML.DUT,R3
23048 102120  042703  177770                BIC     #177770,R3
23049 102124  142777  000007  113312        BICB    #7,@ML.REG+40
23050 102132  150377  113306                BISB    R3,@ML.REG+40
23051 102136  005216                        INC     (SP)                        ; BITS.XFERED                       1007
23052 102140  005001                        CLR     R1                          ; CNT                               1007
```

```
23054                                   ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
23055                                   ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
23056
23057 102142  010102                    4$:     MOV     R1,R2                   ; CNT,*                                         1007
23058 102144  006302                            ASL     R2
23059 102146  005062  014022                    CLR     IO.BUF(R2)
23060 102152  005201                            INC     R1                      ; CNT                                           1007
23061 102154  020127  000377                    CMP     R1,#377                 ; CNT,*
23062 102160  003770                            BLE     4$
23063 102162  010546                            MOV     R5,-(SP)                ; PLOG,*                                        1008
23064 102164  010446                            MOV     R4,-(SP)                ; CHANNEL,*
23065 102166  004767  115640                    JSR     PC,FORCE.REM
23066 102172  004767  116114                    JSR     PC,GD.BLK.XFER          ;                                               1008
23067 102176  142777  000002  113320            BICB    #2,@ML.REG+120          ;                                               1008
23068 102204  012777  000071  113172            MOV     #71,@ML.REG             ;                                               1008
23069 102212  105777  113236            5$:     TSTB    @ML.REG+50
23070 102216  100375                            BPL     5$
23071 102220  016646  000004                    MOV     4(SP),-(SP)             ; BITS.XFERED,*                                 1008
23072 102224  004767  115430                    JSR     PC,FIND.COMP.BIT
23073 102230  005726                            TST     (SP)+
23074 102232  006000                            ROR     R0
23075 102234  103442                            BLO     7$
23076 102236  005267  113134                    INC     P.CNT                   ;                                               1008
23077 102242  026767  113130  113130            CMP     P.CNT,LIMIT
23078 102250  003402                            BLE     6$
23079 102252  022626                            CMP     (SP)+,(SP)+
23080 102254  000512                            BR      11$
23081 102256  104455                    6$:     TRAP    55                      ;                                               1009
23082 102260  000206                            .WORD   206
23083 102262  012750                            .WORD   SYNC
23084 102264  026302                            .WORD   DUMPER
23085 102266  012746  011610                    MOV     #PHR.1,-(SP)            ;                                               1009
23086 102272  012746  011552                    MOV     #WRD.75,-(SP)
23087 102276  012746  011540                    MOV     #WRD.74,-(SP)
23088 102302  012746  010256                    MOV     #THR.FMT,-(SP)
23089 102306  012746  000004                    MOV     #4,-(SP)
23090 102312  010600                            MOV     SP,R0                   ; SP,*
23091 102314  104414                            TRAP    14
23092 102316  010416                            MOV     R4,(SP)                 ; CHANNEL,*                                     1009
23093 102320  010546                            MOV     R5,-(SP)                ; PLOG,*
23094 102322  012746  007754                    MOV     #FMT.21,-(SP)
23095 102326  012746  000003                    MOV     #3,-(SP)
23096 102332  010600                            MOV     SP,R0                   ; SP,*
23097 102334  104414                            TRAP    14
23098 102336  062706  000020                    ADD     #20,SP                  ;                                               1008
23099 102342  005003                    7$:     CLR     R3                      ; WRD.CNT                                       1009
23100 102344  010302                    8$:     MOV     R3,R2                   ; WRD.CNT,*                                     1009
23101 102346  006302                            ASL     R2
23102 102350  005762  014022                    TST     IO.BUF(R2)
23103 102354  001442                            BEQ     9$
23104 102356  005267  113014                    INC     P.CNT                   ;                                               1010
23105 102362  026767  113010  113010            CMP     P.CNT,LIMIT
23106 102370  003040                            BGT     10$
23107 102372  104455                            TRAP    55                      ;                                               1010
23108 102374  000207                            .WORD   207
```

SEQ 0478

```
23110                                   ;ML4AD
23111                                   ;               TEST CODE SECTION                                29-Mar-1982 16:23:04    TOPS
23112                                                                                                    29-Mar-1982 16:21:03    PA:<
23113 102376  012750                            .WORD   SYNC
23114 102400  026302                            .WORD   DUMPER
23115 102402  012746  011714                    MOV     #PHR.5,-(SP)            ;                                               1010
23116 102406  012746  011552                    MOV     #WRD.75,-(SP)
23117 102412  012746  011540                    MOV     #WRD.74,-(SP)
23118 102416  012746  011416                    MOV     #WRD.61,-(SP)
23119 102422  012746  01027C                    MOV     #FOR.FMT,-(SP)
23120 102426  012746  000005                    MOV     #5,-(SP)
23121 102432  010600                            MOV     SP,R0                   ; SP,*
23122 102434  104414                            TRAP    14
23123 102436  010416                            MOV     R4,(SP)                 ; CHANNEL,*                                     1010
23124 102440  010546                            MOV     R5,-(SP)                ; PLOG,*
23125 102442  012746  007754                    MOV     #FMT.21,-(SP)
23126 102446  012746  000003                    MOV     #3,-(SP)
23127 102452  010600                            MOV     SP,R0                   ; SP,*
23128 102454  104414                            TRAP    14
23129 102456  062706  000022                    ADD     #22,SP                  ;                                               1010
23130 102462  005203                    9$:     INC     R3                      ; WRD.CNT                                       1009
23131 102464  020327  000014                    CMP     R3,#14                  ; WRD.CNT,*
23132 102470  003725                            BLE     8$
23133 102472  022626                    10$:    CMP     (SP)+,(SP)+             ;                                               1007
23134 102474  104467                            TRAP    67                      ;                                               1010
23135 102476  006000                            ROR     R0
23136 102500  103601                            BLO     3$
23137 102502  005204                    11$:    INC     R4                      ; CHANNEL                                       1007
23138 102504  020427  000043                    CMP     R4,#43                  ; CHANNEL,*
23139 102510  003002                            BGT     12$
23140 102512  000167  177366                    JMP     3$
23141 102516  005205                    12$:    INC     R5                      ; PLOG                                          1006
23142 102520  020527  000005                    CMP     R5,#5                   ; PLOG,*
23143 102524  003002                            BGT     13$
23144 102526  000167  177350                    JMP     2$
23145 102532  012716  177777            13$:    MOV     #-1,(SP)                ; *,BITS.XFERED                                 1011
23146 102536  005003                            CLR     R3                      ; CHANNEL                                       1011
23147 102540  104402                    14$:    TRAP    2                       ;                                               1011
23148 102542  152777  000040  112674            BISB    #40,@ML.REG+40          ;                                               1011
23149 102550  016705  113256                    MOV     ML.DUT,R5
23150 102554  042705  177770                    BIC     #177770,R5
23151 102560  142777  000007  112656            BICB    #7,@ML.REG+40
23152 102566  150577  112652                    BISB    R5,@ML.REG+40
23153 102572  005216                            INC     (SP)                    ; BITS.XFERED                                   1011
23154 102574  005001                            CLR     R1                      ; CNT                                           1012
23155 102576  010102                    15$:    MOV     R1,R2                   ; CNT,*                                         1012
23156 102600  006302                            ASL     R2
23157 102602  005062  014022                    CLR     IO.BUF(R2)
23158 102606  005201                            INC     R1                      ; CNT                                           1012
23159 102610  020127  000024                    CMP     R1,#24                  ; CNT,*
23160 102614  003770                            BLE     15$
23161 102616  012746  000072                    MOV     #72.,-(SP)              ;                                               1012
23162 102622  010346                            MOV     R3,-(SP)                ; CHANNEL,*
23163 102624  004767  115202                    JSR     PC,FORCE.REM
23164 102630  004767  115456                    JSR     PC,GD.BLK.XFER          ;                                               1012
```

```
23166                                   ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
23167                                   ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
23168
23169 102634  142777  000002  112662            BICB    #2,@ML.REG+120          ;                                       1012
23170 102642  012777  000071  112534            MOV     #71,@ML.REG             ;                                       1012
23171 102650  105777  112600           16$:     TSTB    @ML.REG+50
23172 102654  100375                            BPL     16$
23173 102656  005002                            CLR     R2                      ; BIT.OFFSET                            1012
23174 102660  010246                   17$:     MOV     R2,-(SP)                ; BIT.OFFSET,*                          1013
23175 102662  066616  000006                    ADD     6(SP),(SP)              ; BITS.XFERED,*
23176 102666  004767  114766                    JSR     PC,FIND.COMP.BIT
23177 102672  005726                            TST     (SP)+
23178 102674  006000                            ROR     R0
23179 102676  103441                            BLO     18$
23180 102700  005267  112472                    INC     P.CNT                   ;                                       1013
23181 102704  026767  112466  112466            CMP     P.CNT,LIMIT
23182 102712  003040                            BGT     19$
23183 102714  104455                            TRAP    55                      ;                                       1013
23184 102716  000206                            .WORD   206
23185 102720  012750                            .WORD   SYNC
23186 102722  026302                            .WORD   DUMPER
23187 102724  012746  011610                    MOV     #PHR.1,-(SP)            ;                                       1013
23188 102730  012746  011552                    MOV     #WRD.75,-(SP)
23189 102734  012746  011540                    MOV     #WRD.74,-(SP)
23190 102740  012746  010256                    MOV     #THR.FMT,-(SP)
23191 102744  012746  000004                    MOV     #4,-(SP)
23192 102750  010600                            MOV     SP,R0                   ; SP,*
23193 102752  104414                            TRAP    14
23194 102754  010316                            MOV     R3,(SP)                 ; CHANNEL,*                             1013
23195 102756  012746  000072                    MOV     #72,-(SP)
23196 102762  012746  007754                    MOV     #FMT.21,-(SP)
23197 102766  012746  000003                    MOV     #3,-(SP)
23198 102772  010600                            MOV     SP,R0                   ; SP,*
23199 102774  104414                            TRAP    14
23200 102776  062706  000020                    ADD     #20,SP                  ;                                       1013
23201 103002  062702  000044           18$:     ADD     #44,R2                  ; *,BIT.OFFSET                          1012
23202 103006  020227  000264                    CMP     R2,#264                 ; BIT.OFFSET,*
23203 103012  003722                            BLE     17$
23204 103014  005001                   19$:     CLR     R1                      ; WRD.CNT                               1014
23205 103016  010102                   20$:     MOV     R1,R2                   ; WRD.CNT,*                             1014
23206 103020  006302                            ASL     R2
23207 103022  005762  014022                    TST     IO.BUF(R2)
23208 103026  001445                            BEQ     21$
23209 103030  005267  112342                    INC     P.CNT                   ;                                       1014
23210 103034  026767  112336  112336            CMP     P.CNT,LIMIT
23211 103042  003043                            BGT     22$
23212 103044  104455                            TRAP    55                      ;                                       1014
23213 103046  000207                            .WORD   207
23214 103050  012750                            .WORD   SYNC
23215 103052  026302                            .WORD   DUMPER
23216 103054  012746  011462                    MOV     #WRD.67,-(SP)           ;                                       1014
23217 103060  012746  010650                    MOV     #WRD.12,-(SP)
23218 103064  012746  011552                    MOV     #WRD.75,-(SP)
23219 103070  012746  011540                    MOV     #WRD.74,-(SP)
23220 103074  012746  011416                    MOV     #WRD.61,-(SP)
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
23222
23223
23224
23225 103100  012746  010304                    MOV     #FIV.FMT,-(SP)
23226 103104  012746  000006                    MOV     #6,-(SP)
23227 103110  010600                            MOV     SP,R0                           ; SP,*
23228 103112  104414                            TRAP    14
23229 103114  010316                            MOV     R3,(SP)                         ; CHANNEL,*                             1014
23230 103116  012746  000072                    MOV     #72,-(SP)
23231 103122  012746  007754                    MOV     #FMT.21,-(SP)
23232 103126  012746  000003                    MOV     #3,-(SP)
23233 103132  010600                            MOV     SP,R0                           ; SP,*
23234 103134  104414                            TRAP    14
23235 103136  062706  000024                    ADD     #24,SP                          ;                                       1014
23236 103142  005201                21$:        INC     R1                              ; WRD.CNT                               1014
23237 103144  020127  000014                    CMP     R1,#14                          ; WRD.CNT,*
23238 103150  003722                            BLE     20$
23239 103152  022626                22$:        CMP     (SP)+,(SP)+                     ;                                       1011
23240 103154  104467                            TRAP    67                              ;                                       1014
23241 103156  006000                            ROR     R0
23242 103160  103002                            BHIS    24$
23243 103162  000167  177352        23$:        JMP     14$
23244 103166  005203                24$:        INC     R3                              ; CHANNEL                               1011
23245 103170  020327  000043                    CMP     R3,#43                          ; CHANNEL,*
23246 103174  003772                            BLE     23$
23247 103176  005726                            TST     (SP)+                           ;                                       1002
23248 103200  000207                            RTS     PC
23249
23250                                   ; Routine Size:  309 words
23251                                   ; Maximum stack depth per invocation:  19 words
23256
23257
23261
23262                                           .SBTTL  T51 TEST CODE SECTION
23266 103202                            T51::
23267 103202  004767  176622            1$:     JSR     PC,$T51                         ;                                       1015
23268 103206  104466                            TRAP    66
23269 103210  006000                            ROR     R0
23270 103212  103773                            BLO     1$
23271 103214  000207                            RTS     PC
23272
23273                                   ; Routine Size:  6 words
23274                                   ; Maximum stack depth per invocation:  0 words
```

```
23283
23284
23285 ;          10155 !<BLF/PAGE>
```

```
23287 ;ML4AD
23288 ;                 TEST CODE SECTION
23289 ;
23290 ;          10156 !
23291 ;          10157 BGNTST;
23292 ;          10158
23293 ;          10159 !++
23294 ;          10160 ! TEST NUMBER:  TST 52
23295 ;          10161 !
23296 ;          10162 ! TEST NAME:            UNCORRECTABLE ERROR SYNDRONE DECODE TEST AT CHANNELS > 35
23297 ;          10163 !
23298 ;          10164 ! TEST DESCRIPTION:
23299 ;          10165 !       TEST SYNDRONE DECODE TO DETECT BUT
23300 ;          10166 !       NOT CORRECT UNCORRECTABLE CHANNEL
23301 ;          10167 !       ERRORS BY:
23302 ;          10168 !
23303 ;          10169 !       1.  VIA ECC DIAG MODE FORCE SYNDRONE
23304 ;          10170 !           BITS TO INDICATE CHANNEL ERRORS
23305 ;          10171 !           AT WORDS OF A CRC GROUP BUT AT CHANNELS
23306 ;          10172 !           GREATER THAN 35.
23307 ;          10173 !
23308 ;          10174 !       2.  DO A MASS BUS READ TRANSFER
23309 ;          10175 !
23310 ;          10176 !       3.  THEN EXAMIN THE IO_BUF FOR NO
23311 ;          10177 !           BITS COMPLIMENTED AND ECH  AND
23312 ;          10178 !           BITS SET
23313 ;          10179 !
23314 ;          10180 ! IMPLICIT INPUTS:
23315 ;          10181 !       IO_BUF
23316 ;          10182 !       A  VECTOR OF 256 WORDS WHERE DATA
23317 ;          10183 !       FOR MBUS READS AND WRITES TRANSFERS
23318 ;          10184 !       IS FOUND.
23319 ;          10185 !
23320 ;          10186 !
23321 ;          10187 !--
23322 ;          10188
23323 ;          10189 local
23324 ;          10190     PLOG;                                   !INDEX INTO REMAINDER TABLE 'REM_TBL'
23325 ;          10191
23326 ;          10192 CLR_THRESHOLD;                              !CLEAR ERROR PRINT THRESHOLD
23327 ;          10193 BAI = ONE;                                  !FIRST LETS CLEAR THE ML11'S GOOD BLOCK
23328 ;          10194 IO_BUF = ZEROES;                            !BY WRITING ZEROES TO IT
23329 ;          10195 GD_BLK_XFER ();
23330 ;          10196 MLCS1 = write;
23331 ;          10197 TIME_OUT_LOOP;
23332 ;          10198
23333 ;          10199 incr PLOG_SEL from 0 to 6 do                !TEST SYNDRONE DECODE AT 7 DIFFERENT PLOG VALUES
23334 ;          10200     begin
23335 ;          10201
23336 ;          10202     case .PLOG_SEL from 0 to 6 of           !SELECT A PLOG VALUE
23337 ;          10203         set
23338 ;          10204
23339 ;          10205         [0] :                               !POINTS TO WORD 0 IN THE CRC GROUP
23340 ;          10206             PLOG = 0;
23341 ;          10207
```

```
23343 ;ML4AD
23344 ;                    TEST CODE SECTION
23345 ;
23346 ;          10208          [1] :
23347 ;          10209              PLOG = 1;
23348 ;          10210
23349 ;          10211          [2] :
23350 ;          10212              PLOG = 2;
23351 ;          10213
23352 ;          10214          [3] :
23353 ;          10215              PLOG = 3;
23354 ;          10216
23355 ;          10217          [4] :
23356 ;          10218              PLOG = 4;
23357 ;          10219
23358 ;          10220          [5] :
23359 ;          10221              PLOG = 5;
23360 ;          10222
23361 ;          10223          [6] :
23362 ;          10224              PLOG = 58;
23363 ;          10225          tes;
23364 ;          10226
23365 ;          10227      incr CHANNEL from 36 to 62 do          !TEST AT CHANNELS > 35 TO FORCE UNCORRECTABLE ERROR
23366 ;          10228          begin
23367 ;          10229          BGNSUB;                            !START OF SCOPE LOOP
23368 ;          10230          CLR_MBUS;                          !CLEAR THE DRIVE
23369 ;          10231
23370 ;          10232          incr COUNT from 0 to 255 do
23371 ;          10233              IO_BUF [.COUNT] = ZEROES;      !CLEAR THE FIRST CRC GROUP IN THE IO_BUF
23372 ;          10234
23373 ;          10235          FORCE_REM (.PLOG, .CHANNEL);       !FORCE UNCORRECTABLE ERROR AT THIS WORD AND CHANNEL
23374 ;          10236          GD_BLK_XFER ();                    !SET UP A GOOD BLOCK TRANSFER
23375 ;          10237          ECC_DIS = ZERO;                    !CLEAR ECC DISABLE
23376 ;          10238          MLCS1 = read;                      !DO A READ TRANSFER
23377 ;          10239          TIME_OUT_LOOP;                     !WAIT FOR THE  TRANSFER TO COMPLETE
23378 ;          10240
23379 ;          10241          if not ((.ECH_ERR) and (.UNC_ERR))  !SEE IF THE ERROR WAS DETECTED
23380 ;          10242          then
23381 ;          10243              begin                          !ERROR IF NOT DECTECTED
23382 ;          10244              CMP_THRESHOLD;                 !COMPARE ERROR PRINT THRESHOLD
23383 ;          10245              ERRDF (136, SYNC, DUMPER);
23384 ;          10246              PRINTB (FOR_FMT, WRD_67, WRD_10, WRD_76, WRD_9);
23385 ;          10247              PRINTB (FMT_21, .PLOG, .CHANNEL);
23386 ;          10248              end;
23387 ;          10249
23388 ;          10250          incr CNT from 0 to 12 do           !SEE IF ANY CORRECTION WAS DONE
23389 ;          10251
23390 ;          10252              if .IO_BUF [.CNT] neq ZEROES   !SEARCH THE IO_BUF FOR ALL ZEROES
23391 ;          10253              then
23392 ;          10254                  begin                      !REPORT AN ERROR IF ANY CORRECTION WAS DONE
23393 ;          10255                  CMP_THRESHOLD;             !COMPARE ERROR PRINT THRESHOLD
23394 ;          10256                  ERRDF (137, SYNC, DUMPER);
23395 ;          10257                  PRINTB (FIV_FMT, WRD_61, WRD_74, WRD_75, WRD_12, WRD_67);
23396 ;          10258                  PRINTB (FMT_21, .PLOG, .CHANNEL);
23397 ;          10259                  end;
```

```
23399 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
23400 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (104)
23401
23402 ;          10260
23403 ;          10261          ENDSUB;                          !END OF SCOPE LOOP
23404 ;          10262          end;
23405 ;          10263
23406 ;          10264      end;
23407 ;          10265
23408 ;          10266 ENDTST;
23412
23413                                          .SBTTL  $T52 TEST CODE SECTION
23417 103216  004167  101370          $T52:   JSR     R1,$SAVE5                ;                              1015
23418 103222  005067  112150                  CLR     P.CNT                    ;                              1019
23419 103226  152777  000010  112210          BISB    #10,@ML.REG+40           ;                              1019
23420 103234  005067  110562                  CLR     IO.BUF                   ;                              1019
23421 103240  004767  115046                  JSR     PC,GD.BLK.XFER           ;                              1019
23422 103244  012777  000061  112132          MOV     #61,@ML.REG              ;                              1019
23423 103252  105777  112176            1$:   TSTB    @ML.REG+50
23424 103256  100375                          BPL     1$
23425 103260  005005                          CLR     R5                       ; PLOG.SEL                     1019
23426 103262  010504                    2$:   MOV     R5,R4                    ; PLOG.SEL,*                   1020
23427 103264  006304                          ASL     R4
23428 103266  066407  103272                  ADD     3$(R4),PC
23429 103272  000016                    3$:   .WORD   4$-3$
23430 103274  000022                          .WORD   5$-3$
23431 103276  000030                          .WORD   6$-3$
23432 103300  000036                          .WORD   7$-3$
23433 103302  000044                          .WORD   8$-3$
23434 103304  000052                          .WORD   9$-3$
23435 103306  000060                          .WORD   10$-3$
23436 103310  005003                    4$:   CLR     R3                       ; PLOG                         1020
23437 103312  000421                          BR      11$                      ;                              1020
23438 103314  012703  000001            5$:   MOV     #1,R3                    ; *,PLOG                       1020
23439 103320  000416                          BR      11$                      ;                              1020
23440 103322  012703  000002            6$:   MOV     #2,R3                    ; *,PLOG                       1021
23441 103326  000413                          BR      11$                      ;                              1020
23442 103330  012703  000003            7$:   MOV     #3,R3                    ; *,PLOG                       1021
23443 103334  000410                          BR      11$                      ;                              1020
23444 103336  012703  000004            8$:   MOV     #4,R3                    ; *,PLOG                       1021
23445 103342  000405                          BR      11$                      ;                              1020
23446 103344  012703  000005            9$:   MOV     #5,R3                    ; *,PLOG                       1022
23447 103350  000402                          BR      11$                      ;                              1020
23448 103352  012703  000072           10$:   MOV     #72,R3                   ; *,PLOG                       1022
23449 103356  012704  000044           11$:   MOV     #44,R4                   ; *,CHANNEL                    1022
23450 103362  104402                   12$:   TRAP    2                        ;                              1022
23451 103364  152777  000040  112052          BISB    #40,@ML.REG+40           ;                              1022
23452 103372  016702  112434                  MOV     ML.DUT,R2
23453 103376  042702  177770                  BIC     #177770,R2
```

SEQ 0485

```
23455                                   ;ML4AD
23456                                   ;                       TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
23457                                                                                                               29-Mar-1982 16:21:03    PA:<
23458 103402  142777  000007  112034            BICB    #7,@ML.REG+40
23459 103410  150277  112030                    BISB    R2,@ML.REG+40
23460 103414  005001                            CLR     R1                              ; COUNT                                 1023
23461 103416  010102                    13$:    MOV     R1,R2                           ; COUNT,*                               1023
23462 103420  006302                            ASL     R2
23463 103422  005062  014022                    CLR     I0.BUF(R2)
23464 103426  005201                            INC     R1                              ; COUNT                                 1023
23465 103430  020127  000377                    CMP     R1,#377                         ; COUNT,*
23466 103434  003770                            BLE     13$
23467 103436  010346                            MOV     R3,-(SP)                        ; PLOG,*                                1023
23468 103440  010446                            MOV     R4,-(SP)                        ; CHANNEL,*
23469 103442  004767  114364                    JSR     PC,FORCE.REM
23470 103446  004767  114640                    JSR     PC,GD.BLK.XFER                  ;                                       1023
23471 103452  142777  000002  112044            BICB    #2,@ML.REG+120                  ;                                       1023
23472 103460  012777  000071  111716            MOV     #71,@ML.REG                     ;                                       1023
23473 103466  105777  111762            14$:    TSTB    @ML.REG+50
23474 103472  100375                            BPL     14$
23475 103474  132777  000100  111762            BITB    #100,@ML.REG+60                 ;                                       1024
23476 103502  001403                            BEQ     15$
23477 103504  005777  112104                    TST     @ML.REG+210
23478 103510  100444                            BMI     17$
23479 103512  005267  111660            15$:    INC     P.CNT                           ;                                       1024
23480 103516  026767  111654  111654            CMP     P.CNT,LIMIT
23481 103524  003402                            BLE     16$
23482 103526  022626                            CMP     (SP)+,(SP)+
23483 103530  000520                            BR      22$
23484 103532  104455                    16$:    TRAP    55                              ;                                       1024
23485 103534  000210                            .WORD   210
23486 103536  012750                            .WORD   SYNC
23487 103540  026302                            .WORD   DUMPER
23488 103542  012746  010616                    MOV     #WRD.9,-(SP)                    ;                                       1024
23489 103546  012746  011562                    MOV     #WRD.76,-(SP)
23490 103552  012746  010630                    MOV     #WRD.10,-(SP)
23491 103556  012746  011462                    MOV     #WRD.67,-(SP)
23492 103562  012746  010270                    MOV     #FOR.FMT,-(SP)
23493 103566  012746  000005                    MOV     #5,-(SP)
23494 103572  010600                            MOV     SP,R0                           ; SP,*
23495 103574  104414                            TRAP    14
23496 103576  010416                            MOV     R4,(SP)                         ; CHANNEL,*                             1024
23497 103600  010346                            MOV     R3,-(SP)                        ; PLOG,*
23498 103602  012746  007754                    MOV     #FMT.21,-(SP)
23499 103606  012746  000003                    MOV     #3,-(SP)
23500 103612  010600                            MOV     SP,R0                           ; SP,*
23501 103614  104414                            TRAP    14
23502 103616  062706  000022                    ADD     #22,SP                          ;                                       1024
23503 103622  005001                    17$:    CLR     R1                              ; CNT                                   1025
23504 103624  010102                    18$:    MOV     R1,R2                           ; CNT,*                                 1025
23505 103626  006302                            ASL     R2
23506 103630  005762  014022                    TST     I0.BUF(R2)
23507 103634  001,44                           BEQ     19$
23508 103636  005267  111534                    INC     P.CNT                           ;                                       1025
23509 103642  026767  111530  111530            CMP     P.CNT,LIMIT
```

```
23511                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
23512                                   ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
23513
23514 103650  003042                            BGT     20$
23515 103652  104455                            TRAP    55                      ;                                       1025
23516 103654  000211                            .WORD   211
23517 103656  012750                            .WORD   SYNC
23518 103660  026302                            .WORD   DUMPER
23519 103662  012746  011462                    MOV     #WRD.67,-(SP)           ;                                       1025
23520 103666  012746  010650                    MOV     #WRD.12,-(SP)
23521 103672  012746  011552                    MOV     #WRD.75,-(SP)
23522 103676  012746  011540                    MOV     #WRD.74,-(SP)
23523 103702  012746  011416                    MOV     #WRD.61,-(SP)
23524 103706  012746  010304                    MOV     #FIV.FMT,-(SP)
23525 103712  012746  000006                    MOV     #6,-(SP)
23526 103716  010600                            MOV     SP,R0                   ; SP,*
23527 103720  104414                            TRAP    14
23528 103722  010416                            MOV     R4,(SP)                 ; CHANNEL,*                             1025
23529 103724  010346                            MOV     R3,-(SP)                ; PLOG,*
23530 103726  012746  007754                    MOV     #FMT.21,-(SP)
23531 103732  012746  000003                    MOV     #3,-(SP)
23532 103736  010600                            MOV     SP,R0                   ; SP,*
23533 103740  104414                            TRAP    14
23534 103742  062706  000024                    ADD     #24,SP                  ;                                       1025
23535 103746  005201                    19$:    INC     R1                      ; CNT                                   1025
23536 103750  020127  000014                    CMP     R1,#14                  ; CNT,*
23537 103754  003723                            BLE     18$
23538 103756  022626                    20$:    CMP     (SP)+,(SP)+             ;                                       1022
23539 103760  104467                            TRAP    67                      ;                                       1025
23540 103762  006000                            ROR     R0
23541 103764  103002                            BHIS    22$
23542 103766  000167  177370            21$:    JMP     12$
23543 103772  005204                    22$:    INC     R4                      ; CHANNEL                               1022
23544 103774  020427  000076                    CMP     R4,#76                  ; CHANNEL,*
23545 104000  003772                            BLE     21$
23546 104002  005205                            INC     R5                      ; PLOG.SEL                              1019
23547 104004  020527  000006                    CMP     R5,#6                   ; PLOG.SEL,*
23548 104010  003002                            BGT     23$
23549 104012  000167  177244                    JMP     2$
23550 104016  000207                    23$:    RTS     PC                      ;                                       1015
23551
23552                                   ; Routine Size:  193 words
23553                                   ; Maximum stack depth per invocation:  18 words
23558
23559
23563
23564                                           .SBTTL  T52 TEST CODE SECTION
```

SEQ 0487

```
 23572 104020                           T52::
 23573 104020  004767  177172          1$:     JSR     PC,$T52                 ;                1026
 23574 104024  104466                          TRAP    66
 23575 104026  006000                          ROR     R0
 23576 104030  103773                          BLO     1$
 23577 104032  000207                          RTS     PC
 23578
 23579                                  ; Routine Size:  6 words
 23580                                  ; Maximum stack depth per invocation:  0 words
 23585
 23586
 23587 ;          10267 !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (105)

```
23589 ;ML4AD
23590 ;                 TEST CODE SECTION
23591 ;
23592 ;          10268 !
23593 ;          10269 BGNTST;
23594 ;          10270
23595 ;          10271 !++
23596 ;          10272 ! TEST NUMBER:  TST 53
23597 ;          10273 !
23598 ;          10274 ! TEST NAME:              MULTIPLE CHANNEL ERROR SYNDRONE DECODE TEST
23599 ;          10275 !
23600 ;          10276 ! TEST DESCRIPTION:
23601 ;          10277 !       TEST SYNDRONE DECODE TO DETECT BUT NOT
23602 ;          10278 !       CORRECT UNCORRECTABLE MULTIPLE CHANNEL
23603 ;          10279 !       ERRORS BY:
23604 ;          10280 !
23605 ;          10281 !       1.  VIA ECC DIAGNOSTIC MODE FORCE SYNDRONE
23606 ;          10282 !           BITS TO INDICATE MULTIPLE CHANNEL ERRORS
23607 ;          10283 !           BY ASSIGNING CRC_A EQUAL TO CRC_B
23608 ;          10284 !
23609 ;          10285 !       2.  DO A MASS BUS READ TRANSFER
23610 ;          10286 !
23611 ;          10287 !       3.  THEN EXAMIN THE IO_BUF FOR  NO BITS
23612 ;          10288 !           COMPLIMENTED AND ECH AND UNC BITS SET
23613 ;          10289 !
23614 ;          10290 ! IMPLICIT INPUTS:
23615 ;          10291 !       IO_BUF
23616 ;          10292 !       A  VECTOR OF 256 WORDS WHERE DATA
23617 ;          10293 !       FOR MBUS READS AND WRITES TRANSFERS
23618 ;          10294 !       IS FOUND.
23619 ;          10295 !
23620 ;          10296 !
23621 ;          10297 !--
23622 ;          10298
23623 ;          10299 local
23624 ;          10300     PLOG;                         !INDEX INTO REMAINDER TABLE 'REM_TBL'
23625 ;          10301
23626 ;          10302 CLR_THRESHOLD;                    !CLEAR ERROR PRINT THRESHOLD
23627 ;          10303 BAI = ONE;                        !FIRST LETS CLEAR THE ML11'S GOOD
23628 ;          10304 IO_BUF = ZEROES;                  !BLOCK BY WRITING ZEROES TO IT
23629 ;          10305 GO_BLK_XFER ();
23630 ;          10306 MLCS1 = write;
23631 ;          10307 TIME_OUT_LOOP;
23632 ;          10308
23633 ;          10309 incr PLOG_SEL from 0 to 6 do     !TEST SYNDRONE DECODE AT 7 DIFFERENT PLOG VALUES
23634 ;          10310     begin
23635 ;          10311
23636 ;          10312     case .PLOG_SEL from 0 to 6 of   !SELECT A PLOG VALUE
23637 ;          10313         set
23638 ;          10314
23639 ;          10315         [0] :                       !POINTS TO CRC WORD 0
23640 ;          10316             PLOG = 0;
23641 ;          10317
23642 ;          10318         [1] :                       !POINTS TO CRC WORD 1
23643 ;          10319             PLOG = 1;
```

```
23645 ;ML4AD
23646 ;                 TEST CODE SECTION
23647 ;
23648 ;          10320
23649 ;          10321          [2] :
23650 ;          10322              PLOG = 2;
23651 ;          10323
23652 ;          10324          [3] :
23653 ;          10325              PLOG = 3;
23654 ;          10326
23655 ;          10327          [4] :
23656 ;          10328              PLOG = 4;
23657 ;          10329
23658 ;          10330          [5] :
23659 ;          10331              PLOG = 5;
23660 ;          10332
23661 ;          10333          [6] :
23662 ;          10334              PLOG = 58;
23663 ;          10335          tes;
23664 ;          10336
23665 ;          10337      incr CHANNEL from 1 to 35 do
23666 ;          10338          begin
23667 ;          10339          RGNSUB;
23668 ;          10340          CLR_BUS;
23669 ;          10341
23670 ;          10342          incr COUNT from 0 to 255 do
23671 ;          10343              IO_BUF [.COUNT] = ZEROES;
23672 ;          10344
23673 ;          10345          FORCE_REM (.PLOG, .CHANNEL);
23674 ;          10346          CRC_A = .CRC_B;
23675 ;          10347          GD_BLK_XFER ();
23676 ;          10348          ECC_DIS = ZERO;
23677 ;          10349          MLCS1 = read;
23678 ;          10350          TIME_OUT_LOOP;
23679 ;          10351
23680 ;          10352          if not ((.ECH_ERR) and (.UNC_ERR))
23681 ;          10353          then
23682 ;          10354              begin
23683 ;          10355              CMP_THRESHOLD;
23684 ;          10356              ERRDF (138, SYNC, DUMPER);
23685 ;          10357              PRINTB (FOR_FMT, WRD_77, WRD_10, WRD_76, WRD_9);
23686 ;          10358              PRINTB (FMT_21, .PLOG, .CHANNEL);
23687 ;          10359              end;
23688 ;          10360
23689 ;          10361          incr CNT from 0 to 12 do
23690 ;          10362
23691 ;          10363              if .IO_BUF [.CNT] neq ZEROES
23692 ;          10364              then
23693 ;          10365                  begin
23694 ;          10366                  CMP_THRESHOLD;
23695 ;          10367                  ERRDF (139, SYNC, DUMPER);
23696 ;          10368                  PRINTB (FIV_FMT, WRD_61, WRD_74, WRD_75, WRD_12, WRD_67);
23697 ;          10369                  PRINTB (FMT_21, .PLOG, .CHANNEL);
23698 ;          10370                  end;
23699 ;          10371
```

```
!POINTS TO CRC WORD 2
!POINTS TO CRC WORD 3
!POINTS TO CRC WORD 4
!POINTS TO CRC WORD 5
!POINTS TO ALL 6 WORDS IN CRC GROUP
!TEST 35 CHANNELS STARTING AT CHANNEL 1
!START OF SCOPE LOOP
!CLEAR THE DRIVE
!CLEAR FIRST CRC GROUP IN THE IO_BUF
!FORCE ERR AT THIS WORD AND CHANNEL
!CRC_A EQL TO CRC_B CAUSES THE UNC ERROR
!SET UP A GOOD BLOCK TRANSFER
!CLEAR ECC DISABLE
!DO A READ TRANSFER
!WAIR UNTIL THE TRANSFER IS COMPLETE
!SEE IF THE ERROR WAS DETECTED
!REPORT ERROR IF NOT DETECTED
!COMPARE ERROR PRINT THRESHOLD
!SEE IF ANY ERROR CORRECTION WAS DONE
!TEST THE IO_BUF FOR ALL ZEROES
!ERROR IF ANY BITS GOT FLIPPED
!COMPARE ERROR PRINT THRESHOLD
```

```
23701 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
23702 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (105)
23703
23704 ;          10372          ENDSUB;                         !END OF SCOPE LOOP
23705 ;          10373          end;
23706 ;          10374
23707 ;          10375      end;
23708 ;          10376
23709 ;          10377 ENDTST;
23713
23714                                          .SBTTL  $T53 TEST CODE SECTION
23718 104034  004167  100552          $T53:   JSR     R1,$SAVE5                       ;                       1026
23719 104040  005067  111332                  CLR     P.CNT                           ;                       1030
23720 104044  152777  000010  111372          BISB    #10,@ML.REG+40                  ;                       1030
23721 104052  005067  107744                  CLR     IO.BUF                          ;                       1030
23722 104056  004767  114230                  JSR     PC,GD.BLK.XFER                  ;                       1030
23723 104062  012777  000061  111314          MOV     #61,@ML.REG                     ;                       1030
23724 104070  105777  111360          1$:     TSTB    @ML.REG+50
23725 104074  100375                          BPL     1$
23726 104076  005005                          CLR     R5                              ; PLOG.SEL              1030
23727 104100  010504          2$:     MOV     R5,R4                           ; PLOG.SEL,*            1031
23728 104102  006304                          ASL     R4
23729 104104  066407  104110                  ADD     3$(R4),PC
23730 104110  000016          3$:     .WORD   4$-3$
23731 104112  000022                  .WORD   5$-3$
23732 104114  000030                  .WORD   6$-3$
23733 104116  000036                  .WORD   7$-3$
23734 104120  000044                  .WORD   8$-3$
23735 104122  000052                  .WORD   9$-3$
23736 104124  000060                  .WORD   10$-3$
23737 104126  005003          4$:     CLR     R3                              ; PLOG                  1031
23738 104130  000421                  BR      11$                             ;                       1031
23739 104132  012703  000001    5$:     MOV     #1,R3                           ; *,PLOG                1031
23740 104136  000416                  BR      11$                             ;                       1031
23741 104140  012703  000002    6$:     MOV     #2,R3                           ; *,PLOG                1032
23742 104144  000413                  BR      11$                             ;                       1031
23743 104146  012703  000003    7$:     MOV     #3,R3                           ; *,PLOG                1032
23744 104152  000410                  BR      11$                             ;                       1031
23745 104154  012703  000004    8$:     MOV     #4,R3                           ; *,PLOG                1032
23746 104160  000405                  BR      11$                             ;                       1031
23747 104162  012703  000005    9$:     MOV     #5,R3                           ; *,PLOG                1033
23748 104166  000402                  BR      11$                             ;                       1031
23749 104170  012703  000072    10$:    MOV     #72,R3                          ; *,PLOG                1033
23750 104174  012704  000001    11$:    MOV     #1,R4                           ; *,CHANNEL             1033
23751 104200  104402            12$:    TRAP    2                               ;                       1033
23752 104202  152777  000040  111234          BISB    #40,@ML.REG+40                  ;                       1033
23753 104210  016702  111616                  MOV     ML.OUT,R2
23754 104214  042702  177770                  BIC     #177770,R2
23755 104220  142777  000007  111216          BICB    #7,@ML.REG+40
```

```
23757                                   ;ML4AD
23758                                   ;                       TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
23759                                                                                                               29-Mar-1982 16:21:03    PA:<
23760 104226  150277  111212                      BISB    R2,@ML.REG+40
23761 104232  005001                      CLR     R1                          ; COUNT                                                     1034
23762 104234  010102                13$:  MOV     R1,R2                       ; COUNT,*                                                   1034
23763 104236  006302                      ASL     R2
23764 104240  005062  014022              CLR     IO.BUF(R2)
23765 104244  005201                      INC     R1                          ; COUNT                                                     1034
23766 104246  020127  000377              CMP     R1,#377                     ; COUNT,*
23767 104252  003770                      BLE     13$
23768 104254  010346                      MOV     R3,-(SP)                    ; PLOG,*                                                    1034
23769 104256  010446                      MOV     R4,-(SP)                    ; CHANNEL,*
23770 104260  004767  113546              JSR     PC,FORCE.REM
23771 104264  117702  111274              MOVB    @ML.REG+160,R2              ;                                                           1034
23772 104270  042702  177700              BIC     #177700,R2
23773 104274  142777  000077  111252      BICB    #77,@ML.REG+150
23774 104302  150277  111246              BISB    R2,@ML.REG+150
23775 104306  004767  114000              JSR     PC,GD.BLK.XFER              ;                                                           1034
23776 104312  142777  000002  111204      BICB    #2,@ML.REG+120              ;                                                           1034
23777 104320  012777  000071  111056      MOV     #71,@ML.REG                 ;                                                           1034
23778 104326  105777  111122        14$:  TSTB    @ML.REG+50
23779 104332  100375                      BPL     14$
23780 104334  132777  000100  111122      BITB    #100,@ML.REG+60             ;                                                           1035
23781 104342  001403                      BEQ     15$
23782 104344  005777  111244              TST     @ML.REG+210
23783 104350  100444                      BMI     17$
23784 104352  005267  111020        15$:  INC     P.CNT                       ;                                                           1035
23785 104356  026767  111014  111014      CMP     P.CNT,LIMIT
23786 104364  003402                      BLE     16$
23787 104366  022626                      CMP     (SP)+,(SP)+
23788 104370  000520                      BR      22$
23789 104372  104455        16$:  TRAP    55                          ;                                                           1035
23790 104374  000212                      .WORD   212
23791 104376  012750                      .WORD   SYNC
23792 104400  026302                      .WORD   DUMPER
23793 104402  012746  010616              MOV     #WRD.9,-(SP)                ;                                                           1035
23794 104406  012746  011562              MOV     #WRD.76,-(SP)
23795 104412  012746  010630              MOV     #WRD.10,-(SP)
23796 104416  012746  011570              MOV     #WRD.77,-(SP)
23797 104422  012746  010270              MOV     #FOR.FMT,-(SP)
23798 104426  012746  000005              MOV     #5,-(SP)
23799 104432  010600                      MOV     SP,R0                       ; SP,*
23800 104434  104414                      TRAP    14
23801 104436  010416                      MOV     R4,(SP)                     ; CHANNEL,*                                                 1035
23802 104440  010346                      MOV     R3,-(SP)                    ; PLOG,*
23803 104442  012746  007754              MOV     #FMT.21,-(SP)
23804 104446  012746  000003              MOV     #3,-(SP)
23805 104452  010600                      MOV     SP,R0                       ; SP,*
23806 104454  104414                      TRAP    14
23807 104456  062706  000022              ADD     #22,SP                      ;                                                           1035
23808 104462  005001        17$:  CLR     R1                          ; CNT                                                       1036
23809 104464  010102        18$:  MOV     R1,R2                       ; CNT,*                                                     1036
23810 104466  006302                      ASL     R2
23811 104470  005762  014022              TST     IO.BUF(R2)
```

```
23813                                   ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
23814                                   ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
23815
23816 104474  001444                            BEQ     19$
23817 104476  005267  110674                    INC     P.CNT
23818 104502  026767  110670  110670            CMP     P.CNT,LIMIT                     ;               1036
23819 104510  003042                            BGT     20$
23820 104512  104455                            TRAP    55                              ;               1036
23821 104514  000213                            .WORD   213
23822 104516  012750                            .WORD   SYNC
23823 104520  026302                            .WORD   DUMPER
23824 104522  012746  011462                    MOV     #WRD.67,-(SP)                   ;               1036
23825 104526  012746  010650                    MOV     #WRD.12,-(SP)
23826 104532  012746  011552                    MOV     #WRD.75,-(SP)
23827 104536  012746  011540                    MOV     #WRD.74,-(SP)
23828 104542  012746  011416                    MOV     #WRD.61,-(SP)
23829 104546  012746  010304                    MOV     #FIV.FMT,-(SP)
23830 104552  012746  000006                    MOV     #6,-(SP)
23831 104556  010600                            MOV     SP,R0                           ; SP,*
23832 104560  104414                            TRAP    14
23833 104562  010416                            MOV     R4,(SP)                         ; CHANNEL,*     1036
23834 104564  010346                            MOV     R3,-(SP)                        ; PLOG,*
23835 104566  012746  007754                    MOV     #FMT.21,-(SP)
23836 104572  012746  000003                    MOV     #3,-(SP)
23837 104576  010600                            MOV     SP,R0                           ; SP,*
23838 104600  104414                            TRAP    14
23839 104602  062706  000024                    ADD     #24,SP                          ;               1036
23840 104606  005201                    19$:    INC     R1                              ; CNT           1036
23841 104610  020127  000014                    CMP     R1,#14                          ; CNT,*
23842 104614  003723                            BLE     18$
23843 104616  022626                    20$:    CMP     (SP)+,(SP)+                     ;               1033
23844 104620  104467                            TRAP    67                              ;               1037
23845 104622  006000                            ROR     R0
23846 104624  103002                            BHIS    22$
23847 104626  000167  177346            21$:    JMP     12$
23848 104632  005204                    22$:    INC     R4                              ; CHANNEL       1033
23849 104634  020427  000043                    CMP     R4,#43                          ; CHANNEL,*
23850 104640  003772                            BLE     21$
23851 104642  005205                            INC     R5                              ; PLOG.SEL      1030
23852 104644  020527  000006                    CMP     R5,#6                           ; PLOG.SEL,*
23853 104650  003002                            BGT     23$
23854 104652  000167  177222                    JMP     2$
23855 104656  000207                    23$:    RTS     PC                              ;               1026
23856
23857                                   ; Routine Size:  202 words
23858                                   ; Maximum stack depth per invocation:  18 words
23863
23864
```

```
                                ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<


                                        .SBTTL  T53 TEST CODE SECTION
104660                          T53::
104660  004767  177150          1$:     JSR     PC,$T53                 ;                       1037
104664  104466                          TRAP    66
104666  006000                          ROR     R0
104670  103773                          BLO     1$
104672  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


;       10378 !<BLF/PAGE>
```

```
23894 ;ML4AD
23895 ;                 TEST CODE SECTION
23896 ;
23897 ;          10379 !
23898 ;          10380 BGNTST;
23899 ;          10381
23900 ;          10382 !++
23901 ;          10383 ! TEST NUMBER:  TST 54
23902 ;          10384 !
23903 ;          10385 ! TEST NAME:              SINGLE BIT ERROR SYNDRONE GENERATION & DECODE TEST
23904 ;          10386 !
23905 ;          10387 ! TEST DESCRIPTION:
23906 ;          10388 !       TEST SYNDRONE GENERATION, SYNDRONE DECODE
23907 ;          10389 !       AND ERROR CORRECTION FOR SINGLE BIT CHANNEL
23908 ;          10390 !       ERRORS BY:
23909 ;          10391 !
23910 ;          10392 !       1.  CLEAR THE FIRST CRC GROUP IN THE
23911 ;          10393 !           IO_BUF AND GOOD BLOCK TO ZEROES
23912 ;          10394 !
23913 ;          10395 !       2.  VIA DATA DIAGNOSTIC MODE FLIP A BIT
23914 ;          10396 !           TO A ONE
23915 ;          10397 !
23916 ;          10398 !       3.  THEN DO A MASS BUS READ TRANSFER AND
23917 ;          10399 !           EXAMIN THE IO_BUF FOR ZEROES INDICATING
23918 ;          10400 !           THE CORRECTION WAS PERFORMED
23919 ;          10401 !
23920 ;          10402 !       4.  REPEAT WITH ALL BITS IN THE CRC GROUP
23921 ;          10403 !
23922 ;          10404 ! IMPLICIT INPUTS:
23923 ;          10405 !       IO_BUF
23924 ;          10406 !       A  VECTOR OF 256 WORDS WHERE DATA
23925 ;          10407 !       FOR MBUS READS AND WRITES TRANSFERS
23926 ;          10408 !       IS FOUND.
23927 ;          10409 !
23928 ;          10410 !       PD_TEMP
23929 ;          10411 !       A BIT VECTOR OF 16 BITS WHERE THE READ
23930 ;          10412 !       PROM DATA IS STORED AND ACCESSED FROM.
23931 ;          10413 !
23932 ;          10414 !
23933 ;          10415 !--
23934 ;          10416
23935 ;          10417 local
23936 ;          10418     DONE,                                 !DONE FLAG
23937 ;          10419     NIB_PAT : bitvector [4],              !STORAGE FOR SBE DATA GENERATED
23938 ;          10420     NIB_SEL,                              !POINTER WHERE SBE IS TO BE WRITTEN
23939 ;          10421     GD_WRD_CNT;                           !COUNT OF WHERE GOOD 'NIB_SEL' ARE FOUND
23940 ;          10422
23941 ;          10423 CLR_THRESHOLD;                            !CLEAR ERROR PRINT THRESHOLD
23942 ;          10424
23943 ;          10425 incr WRD_CNT from 0 to 5 do               !FORCE SINGLE BIT ERRORS IN ONE CRC GROUP
23944 ;          10426     begin
23945 ;          10427
23946 ;          10428     incr BIT_CNT from 0 to 35 do          !FORCE SBE'S AT EACH BIT OF A WORD
23947 ;          10429         begin
23948 ;          10430
```

```
23950 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
23951 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (106)
23952
23953 ;          10431          incr CNT from 0 to 255 do                !CLEAR FIRST CRC GROUP OF IO_BUF
23954 ;          10432              IO_BUF [.CNT] = ZEROES;
23955 ;          10433
23956 ;          10434          CLR_MBUS;                                !CLEAR THE DRIVE
23957 ;          10435          GD_BLK_XFER ();                          !SET UP A GOOF BLOCK TRANSFER
23958 ;          10436          MLCS1 = write;                           !DO A WRITE TRANSFER
23959 ;          10437          TIME_OUT_LOOP;                           !WAIT UNTIL THE TRANSFER IS COMPLETE
23960 ;          10438          CLR_MBUS;                                !CLEAR THE DRIVE AGAIN
23961 ;          10439          DONE = ZERO;                             !CLEAR THE DONE FLAG
23962 ;          10440          GD_WRD_CNT = -1;                         !RESET THE COUNT
23963 ;          10441          MLD1 = ZEROES;                           !LOAD THE DATA DIAG REG (BITS 0-38)
23964 ;          10442          MLD2 = ZEROES;                           !WITH ZEROES PAT AND CRC BIT FOR
23965 ;          10443          MLE2 = %o'060000';                       !ZEROES PAT
23966 ;          10444          NIB_SEL = .BIT_CNT/4;                    !CALCULATE THE NIBBLE WHERE THE SBE IS TO GO
23967 ;          10445          DAT_DM_XFER ();                          !SET UP A DATA DIAG MODE TRANSFER
23968 ;          10446          MLCS1 = write;                           !DO A WRITE TRANSFER
23969 ;          10447          DELAY (ONE_US);                          !GIVE THE PROM DATA TIME TO GET OUT
23970 ;          10448
23971 ;          10449          do                                       !FIND THE GOOD NIB AT THE TESTED WORD
23972 ;          10450              begin
23973 ;          10451              PD_TEMP = .MLPD;                     !READ THE PROM DATA FOR THIS ARRAY WORD
23974 ;          10452
23975 ;          10453              if .PD_TEMP [.NIB_SEL] IS_SET        !IS THIS A GOOD NIBBLE
23976 ;          10454              then
23977 ;          10455                  DAT_CLK = ONE                    !IF NOT THEN GET THE NEXT ARRAY WORD
23978 ;          10456              else
23979 ;          10457                  begin                            !ELSE SEE IF WE ARE AT THE TESTED WORD
23980 ;          10458                  GD_WRD_CNT = .GD_WRD_CNT + 1;    !UP THE COUNT
23981 ;          10459                                                   !AND SEE IF WE ARE DONE
23982 ;          10460
23983 ;          10461                  if .GD_WRD_CNT eql .WRD_CNT then DONE = ONE else DAT_CLK = ONE;
23984 ;          10462
23985 ;          10463                  end;
23986 ;          10464
23987 ;          10465              end
23988 ;          10466          until .DONE IS_SET;                      !REPEAT UNTIL WE ARE DONE
23989 ;          10467
23990 ;          10468          NIB_PAT = ZEROES;                        !CLEAR THE SAVE LOCATION
23991 ;          10469          NIB_PAT [.BIT_CNT mod 4] = ONE;          !AND SET THE SBE IN THIS NIBBLE
23992 ;          10470          D1_TEMP = ZEROES;                        !CLEAR THE OTHER NIBBLES IN THIS ARRAY WORD
23993 ;          10471          D2_TEMP = ZEROES;                        !TO BE ZEROES PATTERN AND ZEROES CRC PATTERN
23994 ;          10472          E2_TEMP = %o'060000';
23995 ;          10473          LD_LNG_WRD (.NIB_SEL, .NIB_PAT);         !LOAD THE SBE INTO THE SAVE LOCATION
23996 ;          10474          WRT_LNG_WRD;                             !AND WRITE THE SBE INTO THE DATA DIAG RES'S
23997 ;          10475          DAT_CLK = ONE;                           !CLOCK THE SBE INTO THE ARRAYS MEMORY
23998 ;          10476          DELAY (ONE_US);                          !GIVE IT TIME TO WRITE INTO THE MEMORY
23999 ;          10477          BGNSUB;                                  !START OF SCOPE LOOP
24000 ;          10478          CLR_MBUS;                                !CLEAR OUT THE DATA DIAG MODE XFERR
24001 ;          10479          GD_BLK_XFER ();                          !SET UP A GOOD BLOCK TRANSFER
24002 ;          10480          ECC_DIS = ZERO;                          !CLEAR ECC DISABLE
24003 ;          10481          MLCS1 = read;                            !AND READ OUT THE SBE
24004 ;          10482          TIME_OUT_LOOP;
```

```
24006 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
24007 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (106)
24008
24009 ;          10483
24010 ;          10484          incr CNT from 0 to 13 do                !SEE IF THE SBE WAS CORRECTED
24011 ;          10485
24012 ;          10486              if .IO_BUF [.CNT] neq ZEROES        !TEST THE IO_BUF FOR ALL ZEROES
24013 ;          10487              then
24014 ;          10488                  begin                           !ERROR IF THE ERROR WAS NOT CORRECTED
24015 ;          10489                  CMP_THRESHOLD;                  !COMPARE ERROR PRINT THRESHOLD
24016 ;          10490                  ERRDF (140, SYNC, DUMPER);
24017 ;          10491                  PRINTB (THR_FMT, PHR_12, WRD_76, WRD_9);
24018 ;          10492                  PRINTB (FMT_22, .WRD_CNT, .BIT_CNT);
24019 ;          10493                  end;
24020 ;          10494
24021 ;          10495          ENDSUB;                                 !END OF SCOPE LOOP
24022 ;          10496          end;
24023 ;          10497
24024 ;          10498      end;
24025 ;          10499
24026 ;          10500 ENDTST;
24030
24031                                      .SBTTL  $T54 TEST CODE SECTION
24035 104674  004167  077712        $T54:  JSR     R1,$SAVE5                ;                              1037
24036 104700  162706  000010               SUB     #10,SP
24037 104704  005067  110466               CLR     P.CNT                    ;                              1042
24038 104710  005001                       CLR     R1                       ; WRD.CNT                      1042
24039 104712  005002               1$:     CLR     R2                       ; BIT.CNT                      1042
24040 104714  005003               2$:     CLR     R3                       ; CNT                          1043
24041 104716  010304               3$:     MOV     R3,R4                    ; CNT,*                        1043
24042 104720  006304                       ASL     R4
24043 104722  005064  014022               CLR     IO.BUF(R4)
24044 104726  005203                       INC     R3                       ; CNT                          1043
24045 104730  020327  000377               CMP     R3,#377                  ; CNT,*
24046 104734  003770                       BLE     3$
24047 104736  152777  000040  110500       BISB    #40,@ML.REG+40           ;                              1043
24048 104744  016704  111062               MOV     ML.DUT,R4
24049 104750  042704  177770               BIC     #177770,R4
24050 104754  142777  000007  110462       BICB    #7,@ML.REG+40
24051 104762  150477  110456               BISB    R4,@ML.REG+40
24052 104766  004767  113320               JSR     PC,GD.BLK.XFER           ;                              1043
24053 104772  012777  000061  110404       MOV     #61,@ML.REG              ;                              1043
24054 105000  105777  110450        4$:    TSTB    @ML.REG+50
24055 105004  100375                       BPL     4$
24056 105006  152777  000040  110430       BISB    #40,@ML.REG+40           ;                              1043
24057 105014  016704  111012               MOV     ML.DUT,R4
24058 105020  042704  177770               BIC     #177770,R4
24059 105024  142777  000007  110412       BICB    #7,@ML.REG+40
24060 105032  150477  110406               BISB    R4,@ML.REG+40
```

```
24062                                          ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS
24063                                          ;       TEST CODE SECTION                                 29-Mar-1982 16:21:03    PA:<
24064
24065 105036 005066 000002                    CLR     2(SP)                   ; DONE                                          1043
24066 105042 012716 177777                    MOV     #-1,(SP)                ; *,GD.WRD.CNT                                  1044
24067 105046 005077 110522                    CLR     @ML.REG+170             ;                                               1044
24068 105052 005077 110526                    CLR     @ML.REG+200             ;                                               1044
24069 105056 012777 060000 110500             MOV     #60000,@ML.REG+160      ;                                               1044
24070 105064 010246                           MOV     R2,-(SP)                ; BIT.CNT,*                                     1044
24071 105066 012746 000004                    MOV     #4,-(SP)
24072 105072 004767 077350                    JSR     PC,BL$DIV
24073 105076 010005                           MOV     R0,R5                   ; *,NIB.SEL
24074 105100 004767 111306                    JSR     PC,DAT.DM.XFER          ;                                               1044
24075 105104 012777 000061 110272             MOV     #61,@ML.REG             ;                                               1044
24076 105112 012703 000001                    MOV     #1,R3                   ; *,$STMP2                                      1044
24077 105116 001411                    5$:    BEQ     8$
24078 105120 016704 074772                    MOV     L$DLY,R4                ; *,$STMP1
24079 105124 001404                           BEQ     7$
24080 105126 005066 000012             6$:    CLR     12(SP)                  ; $STMP
24081 105132 005304                           DEC     R4                      ; $STMP1
24082 105134 001374                           BNE     6$
24083 105136 005303                    7$:    DEC     R3                      ; $STMP2
24084 105140 000766                           BR      5$
24085 105142 010504                    8$:    MOV     R5,R4                   ; NIB.SEL,*                                     1045
24086 105144 006204                           ASR     R4
24087 105146 006204                           ASR     R4
24088 105150 006204                           ASR     R4
24089 105152 062704 015342                    ADD     #PD.TEMP,R4
24090 105156 017767 110452 110156      9$:    MOV     @ML.REG+230,PD.TEMP     ;                                               1045
24091 105164 010446                           MOV     R4,-(SP)                ;                                               1045
24092 105166 010546                           MOV     R5,-(SP)                ; NIB.SEL,*
24093 105170 042716 177770                    BIC     #177770,(SP)
24094 105174 012746 000001                    MOV     #1,-(SP)
24095 105200 005046                           CLR     -(SP)
24096 105202 004767 076426                    JSR     PC,BL$GT2
24097 105206 062706 000010                    ADD     #10,SP
24098 105212 005300                           DEC     R0
24099 105214 001411                           BEQ     10$                     ;                                               1045
24100 105216 005266 000004                    INC     4(SP)                   ; GD.WRD.CNT                                    1045
24101 105222 026601 000004                    CMP     4(SP),R1                ; GD.WRD.CNT,WRD.CNT                            1046
24102 105226 001004                           BNE     10$
24103 105230 012766 000001 000006             MOV     #1,6(SP)                ; *,DONE
24104 105236 000403                           BR      11$
24105 105240 152777 000020 110256      10$:   BISB    #20,@ML.REG+120
24106 105246 026627 000006 000001      11$:   CMP     6(SP),#1                ; DONE,*                                        1046
24107 105254 001340                           BNE     9$
24108 105256 005066 000010                    CLR     10(SP)                  ; NIB.PAT                                       1046
24109 105262 010246                           MOV     R2,-(SP)                ; BIT.CNT,*                                     1046
24110 105264 012746 000004                    MOV     #4,-(SP)
24111 105270 004767 077164                    JSR     PC,BL$MOD
24112 105274 010004                           MOV     R0,R4
24113 105276 006200                           ASR     R0
24114 105300 006200                           ASR     R0
24115 105302 006200                           ASR     R0
24116 105304 012703 000014                    MOV     #14,R3
```

```
                                        :ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        :                TEST CODE SECTION                              29-Mar-1982 16:21:03    PA:<
                                        :
24121 105310 060603                             ADD     SP,R3                   ; NIB.PAT,*
24122 105312 060300                             ADD     R3,R0
24123 105314 010016                             MOV     R0,(SP)
24124 105316 010446                             MOV     R4,-(SP)
24125 105320 042716 177770                      BIC     #177770,(SP)
24126 105324 012746 000001                      MOV     #1,-(SP)
24127 105330 011646                             MOV     (SP),-(SP)
24128 105332 004767 076534                      JSR     PC,BL$PU2
24129 105336 005067 106032                      CLR     D1.TEMP                 ;                       1047
24130 105342 005067 106030                      CLR     D2.TEMP                 ;                       1047
24131 105346 012767 060000 106024               MOV     #60000,E2.TEMP          ;                       1047
24132 105354 010516                             MOV     R5,(SP)                 ; NIB.SEL,*             1047
24133 105356 016646 000022                      MOV     22(SP),-(SP)            ; NIB.PAT,*
24134 105362 004767 113766                      JSR     PC,LD.LNG.WRD
24135 105366 016777 106002 110200               MOV     D1.TEMP,@ML.REG+170
24136 105374 016777 105776 110202               MOV     D2.TEMP,@ML.REG+200
24137 105402 016777 105772 110154               MOV     E2.TEMP,@ML.REG+160
24138 105410 152777 000020 110106               BISB    #20,@ML.REG+120         ;                       1047
24139 105416 012703 000001                      MOV     #1,R3                   ; *,$STMP2              1047
24140 105422 001411                     12$:    BEQ     15$
24141 105424 016704 074466                      MOV     L$DLY,R4                ; *,$STMP1
24142 105430 001404                             BEQ     14$
24143 105432 005066 000026              13$:    CLR     26(SP)                  ; $STMP
24144 105436 005304                             DEC     R4                      ; $STMP1
24145 105440 001374                             BNE     13$
24146 105442 005303                     14$:    DEC     R3                      ; $STMP2
24147 105444 000766                             BR      12$
24148 105446 104402                     15$:    TRAP    2
24149 105450 152777 000040 107766               BISB    #40,@ML.REG+40          ;                       1047
24150 105456 016704 110350                      MOV     ML.DUT,R4               ;                       1047
24151 105462 042704 177770                      BIC     #177770,R4
24152 105466 142777 000007 107750               BICB    #7,@ML.REG+40
24153 105474 150477 107744                      BISB    R4,@ML.REG+40
24154 105500 004767 112606                      JSR     PC,GD.BLK.XFER          ;                       1047
24155 105504 142777 000002 110012               BICB    #2,@ML.REG+120          ;                       1048
24156 105512 012777 000071 107664               MOV     #71,@ML.REG             ;                       1048
24157 105520 105777 107730              16$:    TSTB    @ML.REG+50
24158 105524 100375                             BPL     16$
24159 105526 005003                             CLR     R3                      ; CNT                   1048
24160 105530 010304                     17$:    MOV     R3,R4                   ; CNT,*                 1048
24161 105532 006304                             ASL     R4
24162 105534 005764 014022                      TST     IO.BUF(R4)
24163 105540 001440                             BEQ     18$
24164 105542 005267 107630                      INC     P.CNT                   ;                       1048
24165 105546 026767 107624 107624               CMP     P.CNT,LIMIT
24166 105554 003036                             BGT     19$
24167 105556 104455                             TRAP    55                      ;                       1049
24168 105560 000214                             .WORD   214
24169 105562 012750                             .WORD   SYNC
24170 105564 026302                             .WORD   DUMPER
24171 105566 012746 010616                      MOV     #WRD.9,-(SP)            ;                       1049
24172 105572 012746 011562                      MOV     #WRD.76,-(SP)
```

```
24174                                   ;ML4AD
24175                                   ;               TEST CODE SECTION
24176
24177 105576  012746  012036                      MOV     #PHR.12,-(SP)
24178 105602  012746  010256                      MOV     #THR.FMT,-(SP)
24179 105606  012746  000004                      MOV     #4,-(SP)
24180 105612  010600                              MOV     SP,R0                   ; SP,*
24181 105614  104414                              TRAP    14
24182 105616  010216                              MOV     R2,(SP)                 ; BIT.CNT,*                     1049
24183 105620  010146                              MOV     R1,-(SP)                ; WRD.CNT,*
24184 105622  012746  010026                      MOV     #FMT.22,-(SP)
24185 105626  012746  000003                      MOV     #3,-(SP)
24186 105632  010600                              MOV     SP,R0                   ; SP,*
24187 105634  104414                              TRAP    14
24188 105636  062706  000020                      ADD     #20,SP                  ;                               1048
24189 105642  005203                      18$:    INC     R3                      ; CNT                           1048
24190 105644  020327  000015                      CMP     R3,#15                  ; CNT,*
24191 105650  003727                              BLE     17$
24192 105652  104467                      19$:    TRAP    67                      ;                               1049
24193 105654  006000                              ROR     R0
24194 105656  103673                              BLO     15$
24195 105660  062706  000020                      ADD     #20,SP                  ;                               1042
24196 105664  005202                              INC     R2                      ; BIT.CNT                       1042
24197 105666  020227  000043                      CMP     R2,#43                  ; BIT.CNT,*
24198 105672  003002                              BGT     20$
24199 105674  000167  177014                      JMP     2$
24200 105700  005201                      20$:    INC     R1                      ; WRD.CNT                       1042
24201 105702  020127  000005                      CMP     R1,#5                   ; WRD.CNT,*
24202 105706  003002                              BGT     21$
24203 105710  000167  176776                      JMP     1$
24204 105714  062706  000010                      ADD     #10,SP                  ;                               1037
24205 105720  000207                      21$:    RTS     PC
24206
24207                                   ; Routine Size:  267 words
24208                                   ; Maximum stack depth per invocation:  26 words
24213
24214
24218
24219                                           .SBTTL  T54 TEST CODE SECTION
24223 105722                            T54::
24224 105722  004767  176746              1$:     JSR     PC,$T54                 ;                               1049
24225 105726  104466                              TRAP    66
24226 105730  006000                              ROR     R0
24227 105732  103773                              BLO     1$
24228 105734  000207                              RTS     PC
```

```
24230                                   ;ML4AD
24231                                   ;                TEST CODE SECTION
24232
24233
24234                                   ; Routine Size:  6 words
24235                                   ; Maximum stack depth per invocation:  0 words
24240
24241
24242 ;          10501 !<BLF/PAGE>
```

```
10502 !
10503 BGNTST;
10504
10505 !++
10506 ! TEST NUMBER:  TST 55
10507 !
10508 ! TEST NAME:            MULTIPLE BIT ERROR SYNDRONE GENERATION AND DECODE TEST
10509 !
10510 ! TEST DESCRIPTION:
10511 !       TEST SYNDRONE GENERATION, SYNDRONE DECODE
10512 !       AND ERROR CORRECTION FOR MULTIPLE BIT
10513 !       CHANNEL ERRORS BY:
10514 !
10515 !       1.  CLEAR THE FIRST CRC GROUP IN THE
10516 !           IO_BUF AND GOOD BLOCK TO ZEROES
10517 !
10518 !       2.  VIA DATA DIAG MODE FLIP ALL BITS IN
10519 !           A CHANNEL TO ONES
10520 !
10521 !       3.  THEN DO A MASS BUS READ TRANSFER AND
10522 !           EXAMIN THE IO_BUF FOR ZEROES INDICATING
10523 !           THE CORRECTION WAS PERFORMED
10524 !
10525 !       4.  REPEAT FOR ALL CHANNELS IN THE CRC GROUP
10526 !
10527 ! IMPLICIT INPUTS:
10528 !       PD_TEMP
10529 !       A BIT VECTOR OF 16 BITS WHERE THE READ
10530 !       PROM DATA IS STORED OAND ACCESSED FROM.
10531 !
10532 !       IO_BUF
10533 !       A  VECTOR OF 256 WORDS WHERE DATA
10534 !       FOR MBUS READS AND WRITES TRANSFERS
10535 !       IS FOUND.
10536 !
10537 !
10538 !--
10539
10540 local
10541     NIB_PAT : bitvector [4],                !STORAGE FOR SBE TO BE GENERATED
10542     NIB_SEL,                                !POINTER TO WHERE ERROR IS TO BE WRITTEN
10543     GD_WRD_CNT;                             !COUNT OF WHERE GOOD 'NIB_SEL' ARE FOUND
10544
10545 CLR_THRESHOLD;                              !CLEAR ERROR PRINT THRESHOLD
10546 E2_TEMP<12, 3> = %b'110';                   !LOAD DATA STRUCTURE WITH ZEROES CRC PATTERN
10547
10548 incr BIT_CNT from 0 to 35 do               !TEST FOR MULTILPLE ERRORS AT 36 CHANNELS
10549     begin
10550     CLR_MBUS;                               !CLEAR THE DRIVE
10551
10552     incr CNT from 0 to 255 do               !CLEAR FIRST CRC GROUP IN THE IO_BUF
10553         IO_BUF [.CNT] = ZEROES;
```

```
24300 ;ML4AD
24301 ;                 TEST CODE SECTION
24302 ;
24303 ;          10554
24304 ;          10555      GD_BLK_XFER ();                          !ALSO CLEAR THTE FIRST CRC GROUP
24305 ;          10556      MLCS1 = write;                           !IN THE ML11'S GOOD BLOCK
24306 ;          10557      TIME_OUT_LOOP;
24307 ;          10558      GD_WRD_CNT = ZERO;                       !CLEAR COUNT
24308 ;          10559      D1_TEMP = ZEROES;                        !CLEAR D1_TEMP
24309 ;          10560      D2_TEMP = ZEROES;                        !CLEAR D2_TEMP
24310 ;          10561      NIB_PAT = ZEROES;                        !CLEAR NIBBLE PATTERN
24311 ;          10562      NIB_SEL = .BIT_CNT/4;                    !CALCULATE THE NIBBLE TO BE IN ERROR
24312 ;          10563      NIB_PAT [.BIT_CNT mod 4] = ONE;          !CALCULATE AND THE BIT TO BE IN ERROR
24313 ;          10564      LD_LNG_WRD (.NIB_SEL, .NIB_PAT);         !LOAD THE ERROR IN D1_TEMP OR D2_TEMP
24314 ;          10565      WRT_LNG_WRD;                             !LOAD THE REGISTER WITH THE ERROR
24315 ;          10566      DAT_DM_XFER ();                          !SET UP A DATA DIAG MODE TRANSFER
24316 ;          10567      MLCS1 = write;                           !DO A WRITE TRANSFER
24317 ;          10568      DELAY (ONE_US);                          !GIVE THE PROM DATA TIME TO COME OUT
24318 ;          10569
24319 ;          10570      do                                       !LOAD THIS CHANNEL WITH MULTIPLE ERRORS
24320 ;          10571          begin
24321 ;          10572          PD_TEMP = .MLPD;                     !READ THIS ARRAY WORDS PROM_DATA
24322 ;          10573
24323 ;          10574          if .PD_TEMP [.NIB_SEL] IS_SET        !SEE IF THIS A GOOD NIBBLE
24324 ;          10575          then
24325 ;          10576              DAT_CLK = ONE                    !CLOCK OUT ANOTHER ARRAY WORD IF BAD
24326 ;          10577          else
24327 ;          10578              begin                            !ELSE CLOCK IN THE ERROR INTO A GOOD NIBBLE
24328 ;          10579              GD_WRD_CNT = .GD_WRD_CNT + 1;
24329 ;          10580              DAT_CLR = ONE;
24330 ;          10581              end;
24331 ;          10582
24332 ;          10583          end
24333 ;          10584      until .GD_WRD_CNT eql 6;                 !REPEAT UNTIL 6 WORDS ARE WRITTEN
24334 ;          10585
24335 ;          10586      BGNSUB;                                  !START OF SCOPE LOOP
24336 ;          10587      CLR_MBUS;                                !CLEAR OUT THE DATA DISG MODE TRANSFER
24337 ;          10588      GD_BLK_XFER ();                          !SET UP A GOOD BLOCK TRANSFER
24338 ;          10589      ECC_DIS = 0;                             !ENABLE ERROR CORRECTION
24339 ;          10590      MLCS1 = read;                            !READ OUT THE ERRORS
24340 ;          10591      TIME_OUT_LOOP;                           !WAIT FOR THE TRANSFER TO COMPLETE
24341 ;          10592
24342 ;          10593      incr WRD_CNT from 0 to 13 do             !SEE IF THE CORRECTIONS WHERE DONE
24343 ;          10594
24344 ;          10595          if .IO_BUF [.WRD_CNT] neq ZEROES     !TEST THE IO_BUF FOR ALL ZEROES
24345 ;          10596          then
24346 ;          10597              begin                            !REPORT THE ERROR IF ANY LOCATIONS ARE NOT ZEROES
24347 ;          10598              CMP_THRESHOLD;                   !COMPARE ERROR PRINT THRESHOLD
24348 ;          10599              ERRDF (141, SYNC, DUMPER);
24349 ;          10600              PRINTB (THR_FMT, PHR_13, WRD_76, WRD_9);
24350 ;          10601              PRINTB (FMT_22, .WRD_CNT, .BIT_CNT);
24351 ;          10602              end;
24352 ;          10603
24353 ;          10604      ENDSUB;                                  !END OF SCOPE LOOP
24354 ;          10605      end;
```

```
24356 ;ML4AD
24357 ;                 TEST CODE SECTION
24358
24359 ;          10606
24360 ;          10607 ENDTST;
24364
24365                                          .SBTTL  $T55 TEST CODE SECTION
24369 105736 004167 076650          $T55:      JSR     R1,$SAVE5                       ;                       1050
24370 105742 024646                            CMP     -(SP),-(SP)
24371 105744 005067 107426                     CLR     P.CNT                           ;                       1054
24372 105750 042767 070000 105422              BIC     #70000,E2.TEMP                  ;                       1054
24373 105756 052767 060000 105414              BIS     #60000,E2.TEMP
24374 105764 005003                            CLR     R3                              ; BIT.CNT               1054
24375 105766 152777 000040 107450  1$:         BISB    #40,@ML.REG+40                  ;                       1054
24376 105774 016702 110032                     MOV     ML.DUT,R2
24377 106000 042702 177770                     BIC     #177770,R2
24378 106004 142777 000007 107432              BICB    #7,@ML.REG+40
24379 106012 150277 107426                     BISB    R2,@ML.REG+40
24380 106016 005001                            CLR     R1                              ; CNT                   1055
24381 106020 010102                 2$:        MOV     R1,R2                           ; CNT,*                 1055
24382 106022 006302                            ASL     R2
24383 106024 005062 014022                     CLR     IO.BUF(R2)
24384 106030 005201                            INC     R1                              ; CNT                   1055
24385 106032 020127 000377                     CMP     R1,#377                         ; CNT,*
24386 106036 003770                            BLE     2$
24387 106040 004767 112246                     JSR     PC,GD.BLK.XFER                  ;                       1055
24388 106044 012777 000061 107332              MOV     #61,@ML.REG                     ;                       1055
24389 106052 105777 107376          3$:        TSTB    @ML.REG+50
24390 106056 100375                            BPL     3$
24391 106060 005005                            CLR     R5                              ; GD.WRD.CNT            1055
24392 106062 005067 105306                     CLR     D1.TEMP                         ;                       1055
24393 106066 005067 105304                     CLR     D2.TEMP                         ;                       1056
24394 106072 005016                            CLR     (SP)                            ; NIB.PAT               1056
24395 106074 010346                            MOV     R3,-(SP)                        ; BIT.CNT,*             1056
24396 106076 012746 000004                     MOV     #4,-(SP)
24397 106102 004767 076340                     JSR     PC,BL$DIV
24398 106106 010004                            MOV     R0,R4                           ; *,NIB.SEL
24399 106110 010346                            MOV     R3,-(SP)                        ; BIT.CNT,*             1056
24400 106112 012746 000004                     MOV     #4,-(SP)
24401 106116 004767 076336                     JSR     PC,BL$MOD
24402 106122 010002                            MOV     R0,R2
24403 106124 006200                            ASR     R0
24404 106126 006200                            ASR     R0
24405 106130 006200                            ASR     R0
24406 106132 012701 000010                     MOV     #10,R1
24407 106136 060601                            ADD     SP,R1                           ; NIB.PAT,*
24408 106140 060100                            ADD     R1,R0
24409 106142 010016                            MOV     R0,(SP)
24410 106144 010246                            MOV     R2,-(SP)
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
                                        ;
24415 106146 042716 177770              BIC     #177770,(SP)
24416 106152 012746 000001              MOV     #1,-(SP)
24417 106156 011646                     MOV     (SP),-(SP)
24418 106160 004767 075706              JSR     PC,BL$PU2
24419 106164 010416                     MOV     R4,(SP)                 ; NIB.SEL,*                     1056
24420 106166 016646 000016              MOV     16(SP),-(SP)            ; NIB.PAT,*
24421 106172 004767 113156              JSR     PC,LD.LNG.WRD
24422 106176 016777 105172 107370       MOV     D1.TEMP,@ML.REG+170
24423 106204 016777 105166 107372       MOV     D2.TEMP,@ML.REG+200
24424 106212 016777 105162 107344       MOV     E2.TEMP,@ML.REG+160
24425 106220 004767 110166              JSR     PC,DAT.DM.XFER          ;                               1056
24426 106224 012777 000061 107152       MOV     #61,@ML.REG             ;                               1056
24427 106232 012701 000001              MOV     #1,R1                   ; *,$$TMP2                      1056
24428 106236 001411                     BEQ     7$
24429 106240 016702 073652              MOV     L$DLY,R2                ; *,$$TMP1
24430 106244 001404                     BEQ     6$
24431 106246 005066 000022      5$:     CLR     22(SP)                  ; $$TMP
24432 106252 005302                     DEC     R2                      ; $$TMP1
24433 106254 001374                     BNE     5$
24434 106256 005301             6$:     DEC     R1                      ; $$TMP2
24435 106260 000766                     BR      4$
24436 106262 010402             7$:     MOV     R4,R2                   ; NIB.SEL,*                     1057
24437 106264 006202                     ASR     R2
24438 106266 006202                     ASR     R2
24439 106270 006202                     ASR     R2
24440 106272 062702 015342              ADD     #PD.TEMP,R2
24441 106276 017767 107332 107036 8$:   MOV     @ML.REG+230,PD.TEMP     ;                               1057
24442 106304 010246                     MOV     R2,-(SP)                ;                               1057
24443 106306 010446                     MOV     R4,-(SP)                ; NIB.SEL,*
24444 106310 042716 177770              BIC     #177770,(SP)
24445 106314 012746 000001              MOV     #1,-(SP)
24446 106320 005046                     CLR     -(SP)
24447 106322 004767 075306              JSR     PC,BL$GT2
24448 106326 062706 000010              ADD     #10,SP
24449 106332 005300                     DEC     R0
24450 106334 001401                     BEQ     9$                      ;                               1057
24451 106336 005205                     INC     R5                      ; GD.WRD.CNT                    1057
24452 106340 152777 000020 107156 9$:   BISB    #20,@ML.REG+120         ;                               1058
24453 106346 020527 000006              CMP     R5,#6                   ; GD.WRD.CNT,*                  1058
24454 106352 001351                     BNE     8$
24455 106354 104402             10$:    TRAP    2
24456 106356 152777 000040 107060       BISB    #40,@ML.REG+40          ;                               1058
24457 106364 016702 107442              MOV     ML.DUT,R2
24458 106370 042702 177770              BIC     #177770,R2
24459 106374 142777 000007 107042       BICB    #7,@ML.REG+40
24460 106402 150277 107036              BISB    R2,@ML.REG+40
24461 106406 004767 111700              JSR     PC,GD.BLK.XFER          ;                               1058
24462 106412 142777 000002 107104       BICB    #2,@ML.REG+120          ;                               1058
24463 106420 012777 000071 106756       MOV     #71,@ML.REG             ;                               1059
24464 106426 105777 107022      11$:    TSTB    @ML.REG+50
24465 106432 100375                     BPL     11$
24466 106434 005001                     CLR     R1                      ; WRD.CNT                       1059
```

```
24468                                   ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
24469                                   ;                       TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
24470
24471 106436  010102                    12$:    MOV     R1,R2                   ; WRD.CNT,*                                     1059
24472 106440  006302                            ASL     R2
24473 106442  005762  014022                    TST     IO.BUF(R2)
24474 106446  00144C                            BEQ     13$
24475 106450  005267  106722                    INC     P.CNT                   ;                                               1059
24476 106454  026767  106716  106716            CMP     P.CNT,LIMIT
24477 106462  003036                            BGT     14$
24478 106464  104455                            TRAP    55                      ;                                               1059
24479 106466  000215                            .WORD   215
24480 106470  012750                            .WORD   SYNC
24481 106472  026302                            .WORD   DUMPER
24482 106474  012746  010616                    MOV     #WRD.9,-(SP)            ;                                               1060
24483 106500  012746  011562                    MOV     #WRD.76,-(SP)
24484 106504  012746  012060                    MOV     #PHR.13,-(SP)
24485 106510  012746  010256                    MOV     #THR.FMT,-(SP)
24486 106514  012746  000004                    MOV     #4,-(SP)
24487 106520  010600                            MOV     SP,R0                   ; SP,*
24488 106522  104414                            TRAP    14
24489 106524  010316                            MOV     R3,(SP)                 ; BIT.CNT,*                                     1060
24490 106526  010146                            MOV     R1,-(SP)                ; WRD.CNT,*
24491 106530  012746  010026                    MOV     #FMT.22,-(SP)
24492 106534  012746  000003                    MOV     #3,-(SP)
24493 106540  010600                            MOV     SP,R0                   ; SP,*
24494 106542  104414                            TRAP    14
24495 106544  062706  000020                    ADD     #20,SP                  ;                                               1059
24496 106550  005201                    13$:    INC     R1                      ; WRD.CNT                                       1059
24497 106552  020127  000015                    CMP     R1,#15                  ; WRD.CNT,*
24498 106556  003727                            BLE     12$
24499 106560  104467                    14$:    TRAP    67                      ;                                               1060
24500 106562  006000                            ROR     R0
24501 106564  103673                            BLO     10$
24502 106566  062706  000020                    ADD     #20,SP                  ;                                               1054
24503 106572  005203                            INC     R3                      ; BIT.CNT                                       1054
24504 106574  020327  000043                    CMP     R3,#43                  ; BIT.CNT,*
24505 106600  003002                            BGT     15$
24506 106602  000167  177160                    JMP     1$
24507 106606  022626                    15$:    CMP     (SP)+,(SP)+             ;                                               1050
24508 106610  000207                            RTS     PC
24509
24510                                   ; Routine Size:  214 words
24511                                   ; Maximum stack depth per invocation:  24 words
24516
24517
24521
24522                                           .SBTTL  T55 TEST CODE SECTION
```

```
24524                                   ;ML4AD
24525                                   ;               TEST CODE SECTION
24526
24530 106612                            T55::
24531 106612  004767  177120            1$:     JSR     PC,$T55                 ;
24532 106616  104466                            TRAP    66
24533 106620  006000                            ROR     R0
24534 106622  103773                            BLO     1$
24535 106624  000207                            RTS     PC
24536
24537                                   ; Routine Size:  6 words
24538                                   ; Maximum stack depth per invocation:  0 words
24543
24544
24545 ;          10608 !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (108)

```
24547 ;ML4AD
24548 ;                 TEST CODE SECTION
24549 ;
24550 ;          10609 !
24551 ;          10610 BGNTST;
24552 ;          10611
24553 ;          10612 !++
24554 ;          10613 ! TEST NUMBER:  TST 56
24555 ;          10614 !
24556 ;          10615 ! TEST NAME:              ECC ERROR REGISTER TEST
24557 ;          10616 !
24558 ;          10617 ! TEST DESCRIPTION:
24559 ;          10618 !       TEST THE ECC ERROR REGISTER FOR
24560 ;          10619 !       CLEARING AND LATCHING OF ECC ERROR
24561 ;          10620 !       INFORMATION ON DETTECTION
24562 ;          10621 !       OF ECC ERRORS BY:
24563 ;          10622 !
24564 ;          10623 !       1.  THIS TEST IS TABLE DRIVEN.  DATA TABLE (DT_1)
24565 ;          10624 !           CONTAINS ERROR FORCING DATA AND EXPECTED
24566 ;          10625 !           MLEE DATA AS A RESULT OF THE FORCED ERROR
24567 ;          10626 !
24568 ;          10627 !       2.  THE TEST THEREFOR INDEXES INTO DT_1 TO FORCE
24569 ;          10628 !           AN ECC ERROR
24570 ;          10629 !
24571 ;          10630 !       3.  A MASS BUS READ TRANSFER IS DONE
24572 ;          10631 !
24573 ;          10632 !       4.  DT_1 IS AGAIN INDEXED AND ITS CONTENTS
24574 ;          10633 !           IS COMPARED AGAINST THE RESULTING
24575 ;          10634 !           MLEE REGISTER  CONTENTS
24576 ;          10635 !
24577 ;          10636 !       5.  A MASS BUS CLEAR IS DONE AN MLEE IS CHECKED FOR
24578 ;          10637 !           ZEROES
24579 ;          10638 !
24580 ;          10639 !       6.  THE ECH BIT IS ALSO EXAMINED FOR CORRECT
24581 ;          10640 !           LATCHING INFORMATION
24582 ;          10641 !
24583 ;          10642 !
24584 ;          10643 !
24585 ;          10644 !--
24586 ;          10645
24587 ;          10646 local
24588 ;          10647     TMP_E1,                          !TEMP STORAGE FOR E1 DATA WORD
24589 ;          10648     FINISH,                          !VARIABLE ENDING CONDITION FOR LATCH_LOOP
24590 ;          10649     ERR_FLG;                         !ERROR FLAG
24591 ;          10650
24592 ;          10651 CLR_MBUS;                            !CLEAR THE DRIVE
24593 ;          10652 FINISH = 4;                          !MAKE LATCH_LOOP DO 5 LOOPS AT FIRST
24594 ;          10653
24595 ;          10654 incr CLR_LOOP from 0 to 1 do         !THE FIRST PASS TESTS MLEE FOR LATCHING
24596 ;          10655     !THE SECOND PASS TESTS MLEE FOR CLEARING
24597 ;          10656     begin
24598 ;          10657
24599 ;          10658     incr LATCH_LOOP from 0 to .FINISH do   !TEST MLEE FOR LATCHING BY USING DT_1 DATA
24600 ;          10659         begin
24601 ;          10660         BGNSUB;                      !START OF SCOPE LOOP
```

```
24603 ;ML4AD
24604 ;                    TEST CODE SECTION
24605 ;
24606 ;          10661          ERR_FLG = ZERO;                          !CLEAR THE ERROR FLAG
24607 ;          10662          MLER = ZEROES;                           !CLEAR THE ERROR REGISTER
24608 ;          10663          ECC_DM = ONE;                            !ENABLE ECC DIAG MODE
24609 ;          10664          !+
24610 ;          10665          ! THE FOLLOWING ASSIGNMENT:
24611 ;          10666          !
24612 ;          10667          !       MLE1 = .TMP_E1
24613 ;          10668          !
24614 ;          10669          ! IS EQUIVALENT TO THE FOLLOWING
24615 ;          10670          ! TWO ASSINGMENTS:
24616 ;          10671          !
24617 ;          10672          !   PAR_CRC_WRD = .DT_1 [.LATCH_LOOP, P0_5];          !LOAD PAR_CRC_WRD WITH REM_TBL
24618 ;          10673          !   CRC_A = .DT_1 [.LATCH_LOOP, A0_5];                !LOAD CRC_A WITH REM_TBL
24619 ;          10674          !
24620 ;          10675          ! THIS IS NECESSARY DUE TO THE FACT THAT
24621 ;          10676          ! THE E1 REGISTER IS WORD ORIENTATED AND
24622 ;          10677          ! THE BLISS COMPILER GENERATES BYTE INST
24623 ;          10678          ! TO ACCESS THE REGISTER.
24624 ;          10679          !-
24625 ;          10680
24626 ;          10681          TMP_E1 = ZEROES;                                  !CLEAR OUT THE TEMP WORD
24627 ;          10682          TMP_E1<0, 6> = .DT_1 [.LATCH_LOOP, A0_5];         !LOAD CRC_A WITH REM_TBL
24628 ;          10683          TMP_E1<8, 6> = .DT_1 [.LATCH_LOOP, P0_5];         !LOAD PAR_CRC_WRD WITH REM_TBL
24629 ;          10684          MLE1 = .TMP_E1;                                   !LOAD THE E1 REGISTER WITH DATA
24630 ;          10685
24631 ;          10686          CRC_B = .DT_1 [.LATCH_LOOP, B0_5];                !GET CRC_B FROM DT_1
24632 ;          10687          GD_BLK_XFER ();                                   !SET UP A GOOD BLOCK TRANSFER
24633 ;          10688          ECC_DIS = ZERO;                                   !DISABLE ECC DIABLE
24634 ;          10689          MLCS1 = read;                                     !READ THE ERROR AND LACTH MLEE
24635 ;          10690          TIME_OUT_LOOP;
24636 ;          10691
24637 ;          10692          if .CRC_ERR neq .DT_1 [.LATCH_LOOP, CRC_DATA] then ERR_FLG = ONE;       !TEST CRC_ERR
24638 ;          10693
24639 ;          10694          if .SGL_ERR neq .DT_1 [.LATCH_LOOP, SGL_DATA] then ERR_FLG = ONE;       !TEST SGL_ERR
24640 ;          10695
24641 ;          10696          if .UNC_ERR neq .DT_1 [.LATCH_LOOP, UNC_DATA] then ERR_FLG = ONE;       !TEST UNC
24642 ;          10697
24643 ;          10698          if .BIT_IN_ERR neq .DT_1 [.LATCH_LOOP, E0_5] then ERR_FLG = ONE;        !TEST BIT IN ERROR
24644 ;          10699
24645 ;          10700          if .CHAN_IN_ERR neq .DT_1 [.LATCH_LOOP, C0_5] then ERR_FLG = ONE;       !TEST CHANNEL IN ERROR
24646 ;          10701
24647 ;          10702          if .ECH_ERR neq .DT_1 [.LATCH_LOOP, ECH_DATA]   !TEST IF ECH GOT SET
24648 ;          10703          then
24649 ;          10704              begin                                       !REPORT ERROR IF NOT SET
24650 ;          10705              ERRDF (123, ASYNC, DUMPER);
24651 ;          10706              PRINTB (FOR_FMT, WRD_67, WRD_10, WRD_69, WRD_14);
24652 ;          10707              end;
24653 ;          10708
24654 ;          10709          if .ERR_FLG IS_SET                              !TEST IF THE ERROR FLAG GOT SET
24655 ;          10710          then
24656 ;          10711              begin                                       !REPORT ERROR IF SET
24657 ;          10712              ERRDF (124, SYNC, DUMPER);
```

```
 24659 ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 24660 ;                TEST CODE SECTION                              29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (108)
 24661
 24662 ;          10713            PRINTB (THR_FMT, REG_14, WRD_52, PHR_11);
 24663 ;          10714            PRINTB (FMT_1, .DT_1 [.LATCH_LOOP, EE_DATA], .MLEE);
 24664 ;          10715            PRINTB (FMT_19, .DT_1 [.LATCH_LOOP, B0_5], .DT_1 [.LATCH_LOOP, A0_5], .DT_1 [.LATCH_LOOP, P0_5])
 24665 ;          10716            end;
 24666 ;          10717
 24667 ;          10718        ENDSUB;                                 !END OF SCOPE LOOP
 24668 ;          10719        end;
 24669 ;          10720
 24670 ;          10721    CLR_MBUS;                                   !CLEAR THE REGISTER
 24671 ;          10722
 24672 ;          10723    if .MLEE neq ZERO                           !SEE IF THE REGISTER GOT CLEARED
 24673 ;          10724    then
 24674 ;          10725        begin                                   !REPORT ERROR IF NOT CLEARED
 24675 ;          10726        ERRDF (125, SYNC, DUMPER);
 24676 ;          10727        PRINTB (FOR_FMT, REG_14, WRD_52, FNC_7, WRD_14);
 24677 ;          10728        end;
 24678 ;          10729
 24679 ;          10730    ECC_DM = ONE;                               !ENABLE ECC DISABLE
 24680 ;          10731    FINISH = ZERO;                              !MAKE LATCH_LOOP LOOP ONCE TO CLEAR MLEE AGAIN
 24681 ;          10732    end;
 24682 ;          10733
 24683 ;          10734 ENDTST;
 24687
 24688                                        .SBTTL  ST56 TEST CODE SECTION
 24692 106626  004167  075760          ST56: JSR     R1,$SAVE5                 ;                                        1060
 24693 106632  162706  000006                SUB     #6,SP
 24694 106636  152777  000040  106600        BISB    #40,@ML.REG+40            ;                                        1064
 24695 106644  016705  107162                MOV     ML.DUT,R5
 24696 106650  042705  177770                BIC     #177770,R5
 24697 106654  142777  000007  106562        BICB    #7,@ML.REG+40
 24698 106662  150577  106556                BISB    R5,@ML.REG+40
 24699 106666  011766  000004                MOV     (PC),4(SP)                ; *,FINISH                               1065
 24700 106672  005066  000002                CLR     2(SP)                     ; CLR.LOOP                               1065
 24701 106676  005004                  1$:   CLR     R4                        ; LATCH.LOOP                             1065
 24702 106700  000167  000656                JMP     16$
 24703 106704  010446                  2$:   MOV     R4,-(SP)                  ; LATCH.LOOP,*                           1068
 24704 106706  012746  000006                MOV     #6,-(SP)
 24705 106712  004767  075304                JSR     PC,BL$MUL
 24706 106716  010005                        MOV     R0,R5
 24707 106720  012701  015764                MOV     #DT.1,R1                  ;                                        1069
 24708 106724  060501                        ADD     R5,R1
 24709 106726  012702  015766                MOV     #DT.1+2,R2                ;                                        1068
 24710 106732  060502                        ADD     R5,R2
 24711 106734  104402                  3$:   TRAP    2                         ;                                        1065
 24712 106736  005003                        CLR     R3                        ; ERR.FLG                                1066
 24713 106740  005077  106520                CLR     @ML.REG+60                ;                                        1066
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
                                        ;
24718 106744 152777 000001 106552       BISB    #1,@ML.REG+120          ;                                                       1066
24719 106752 005066 000004              CLR     4(SP)                   ; TMP.E1                                                1068
24720 106756 111200                     MOVB    (R2),R0                 ;                                                       1068
24721 106760 042700 177700              BIC     #177700,R0
24722 106764 142766 000077 000004       BICB    #77,4(SP)               ; *,TMP.E1
24723 106772 050066 000004              BIS     R0,4(SP)                ; *,TMP.E1
24724 106776 011200                     MOV     (R2),R0                 ;                                                       1068
24725 107000 006300                     ASL     R0
24726 107002 006300                     ASL     R0
24727 107004 042700 140377              BIC     #140377,R0
24728 107010 042766 037400 000004       BIC     #37400,4(SP)            ; *,TMP.E1
24729 107016 050066 000004              BIS     R0,4(SP)                ; *,TMP.E1
24730 107022 016677 000004 106524       MOV     4(SP),@ML.REG+150       ; TMP.E1,*                                              1068
24731 107030 116500 015770              MOVB    DT.1+4(R5),R0           ;                                                       1068
24732 107034 042700 177700              BIC     #177700,R0
24733 107040 142777 000077 106516       BICB    #77,@ML.REG+160
24734 107046 150077 106512              BISB    R0,@ML.REG+160
24735 107052 004767 111234              JSR     PC,GD.BLK.XFER          ;                                                       1068
24736 107056 142777 000002 106440       BICB    #2,@ML.REG+120          ;                                                       1068
24737 107064 012777 000071 106312       MOV     #71,@ML.REG             ;                                                       1068
24738 107072 105777 106356      4$:     TSTB    @ML.REG+50
24739 107076 100375                     BPL     4$
24740 107100 011146                     MOV     (R1),-(SP)              ;                                                       1069
24741 107102 042716 157777              BIC     #157777,(SP)
24742 107106 017700 106502              MOV     @ML.REG+210,R0
24743 107112 042700 157777              BIC     #157777,R0
24744 107116 020026                     CMP     R0,(SP)+
24745 107120 001402                     BEQ     5$
24746 107122 012703 000001              MOV     #1,R3                   ; *,ERR.FLG
24747 107126 011146             5$:     MOV     (R1),-(SP)              ;                                                       1069
24748 107130 042716 137777              BIC     #137777,(SP)
24749 107134 017700 106454              MOV     @ML.REG+210,R0
24750 107140 042700 137777              BIC     #137777,R0
24751 107144 020026                     CMP     R0,(SP)+
24752 107146 001402                     BEQ     6$
24753 107150 012703 000001              MOV     #1,R3                   ; *,ERR.FLG
24754 107154 011146             6$:     MOV     (R1),-(SP)              ;                                                       1069
24755 107156 042716 077777              BIC     #77777,(SP)
24756 107162 017700 106426              MOV     @ML.REG+210,R0
24757 107166 042700 077777              BIC     #77777,R0
24758 107172 020026                     CMP     R0,(SP)+
24759 107174 001402                     BEQ     7$
24760 107176 012703 000001              MOV     #1,R3                   ; *,ERR.FLG
24761 107202 111146             7$:     MOVB    (R1),-(SP)              ;                                                       1069
24762 107204 042716 177700              BIC     #177700,(SP)
24763 107210 117700 106400              MOVB    @ML.REG+210,R0
24764 107214 042700 177700              BIC     #177700,R0
24765 107220 020026                     CMP     R0,(SP)+
24766 107222 001402                     BEQ     8$
24767 107224 012703 000001              MOV     #1,R3                   ; *,ERR.FLG
24768 107230 011146             8$:     MOV     (R1),-(SP)              ;                                                       1070
24769 107232 042716 170077              BIC     #170077,(SP)
```

```
24771                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
24772                                   ;       TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
24773
24774 107236  017700  106352                    MOV     @ML.REG+210,R0
24775 107242  042700  170077                    BIC     #170077,R0
24776 107246  020026                            CMP     R0,(SP)+
24777 107250  001402                            BEQ     9$
24778 107252  012703  000001                    MOV     #1,R3                   ; *,ERR.FLG
24779 107256  011246              9$:           MOV     (R2),-(SP)              ;                                       1070
24780 107260  005046                            CLR     -(SP)
24781 107262  032766  010000  000002            BIT     #10000,2(SP)
24782 107270  001401                            BEQ     10$
24783 107272  005216                            INC     (SP)
24784 107274  005000              10$:          CLR     R0
24785 107276  132777  000100  106160            BITB    #100,@ML.REG+60
24786 107304  001401                            BEQ     11$
24787 107306  005200                            INC     R0
24788 107310  020026              11$:          CMP     R0,(SP)+
24789 107312  001002                            BNE     12$
24790 107314  005726                            TST     (SP)+
24791 107316  000425                            BR      13$
24792 107320  005726              12$:          TST     (SP)+
24793 107322  104455                            TRAP    55                      ;                                       1070
24794 107324  000173                            .WORD   173
24795 107326  012706                            .WORD   ASYNC
24796 107330  026302                            .WORD   DUMPER
24797 107332  012746  010664                    MOV     #WRD.14,-(SP)           ;                                       1070
24798 107336  012746  011474                    MOV     #WRD.69,-(SP)
24799 107342  012746  010630                    MOV     #WRD.10,-(SP)
24800 107346  012746  011462                    MOV     #WRD.67,-(SP)
24801 107352  012746  010270                    MOV     #FOR.FMT,-(SP)
24802 107356  012746  000005                    MOV     #5,-(SP)
24803 107362  010600                            MOV     SP,R0                   ; SP,*
24804 107364  104414                            TRAP    14
24805 107366  062706  000014                    ADD     #14,SP                  ;                                       1070
24806 107372  020327  000001        13$:        CMP     R3,#1                   ; ERR.FLG,*                             1070
24807 107376  001062                            BNE     14$
24808 107400  104455                            TRAP    55                      ;                                       1071
24809 107402  000174                            .WORD   174
24810 107404  012750                            .WORD   SYNC
24811 107406  026302                            .WORD   DUMPER
24812 107410  012746  012020                    MOV     #PHR.11,-(SP)           ;                                       1071
24813 107414  012746  011310                    MOV     #WRD.52,-(SP)
24814 107420  012746  012620                    MOV     #REG.14,-(SP)
24815 107424  012746  010256                    MOV     #THR.FMT,-(SP)
24816 107430  012746  000004                    MOV     #4,-(SP)
24817 107434  010600                            MOV     SP,R0                   ; SP,*
24818 107436  104414                            TRAP    14
24819 107440  017716  106150                    MOV     @ML.REG+210,(SP)        ;                                       1071
24820 107444  011146                            MOV     (R1),-(SP)
24821 107446  012746  006442                    MOV     #FMT.1,-(SP)
24822 107452  012746  000003                    MOV     #3,-(SP)
24823 107456  010600                            MOV     SP,R0                   ; SP,*
24824 107460  104414                            TRAP    14
24825 107462  011200                            MOV     (R2),R0                 ;                                       1071
```

```
24827                                   ;ML4AD
24828                                   ;               TEST CODE SECTION                                   29-Mar-1982 16:23:04    TOPS
24829                                                                                                       29-Mar-1982 16:21:03    PA:<
24830 107464  006200                            ASR     R0
24831 107466  006200                            ASR     R0
24832 107470  006200                            ASR     R0
24833 107472  006200                            ASR     R0
24834 107474  006200                            ASR     R0
24835 107476  006200                            ASR     R0
24836 107500  042700  177700                    BIC     #177700,R0
24837 107504  010016                            MOV     R0,(SP)
24838 107506  111246                            MOVB    (R2),-(SP)
24839 107510  042716  177700                    BIC     #177700,(SP)
24840 107514  116546  015770                    MOVB    DT.1+4(R5),-(SP)
24841 107520  042716  177700                    BIC     #177700,(SP)
24842 107524  012746  007632                    MOV     #FMT.19,-(SP)
24843 107530  012746  000004                    MOV     #4,-(SP)
24844 107534  010600                            MOV     SP,R0                   ; SP,*
24845 107536  104414                            TRAP    14
24846 107540  062706  000030                    ADD     #30,SP                  ;                               1071
24847 107544  104467                  14$:      TRAP    67                      ;                               1071
24848 107546  006000                            ROR     R0
24849 107550  103002                            BHIS    15$
24850 107552  000167  177156                    JMP     3$
24851 107556  022626                  15$:      CMP     (SP)+,(SP)+             ;                               1065
24852 107560  005204                            INC     R4                      ; LATCH.LOOP                    1065
24853 107562  020466  000004            16$:    CMP     R4,4(SP)                ; LATCH.LOOP,FINISH
24854 107566  003002                            BGT     17$
24855 107570  000167  177110                    JMP     2$
24856 107574  152777  000040  105642  17$:      BISB    #40,@ML.REG+40          ;                               1071
24857 107602  016705  106224                    MOV     ML.DUT,R5
24858 107606  042705  177770                    BIC     #177770,R5
24859 107612  142777  000007  105624            BICB    #7,@ML.REG+40
24860 107620  150577  105620                    BISB    R5,@ML.REG+40
24861 107624  005777  105764                    TST     @ML.REG+210             ;                               1072
24862 107630  001424                            BEQ     18$
24863 107632  104455                            TRAP    55                      ;                               1072
24864 107634  000175                            .WORD   175
24865 107636  012750                            .WORD   SYNC
24866 107640  026302                            .WORD   DUMPER
24867 107642  012746  010664                    MOV     #WRD.14,-(SP)           ;                               1072
24868 107646  012746  012234                    MOV     #FNC.7,-(SP)
24869 107652  012746  011310                    MOV     #WRD.52,-(SP)
24870 107656  012746  012620                    MOV     #REG.14,-(SP)
24871 107662  012746  010270                    MOV     #FOR.FMT,-(SP)
24872 107666  012746  000005                    MOV     #5,-(SP)
24873 107672  010600                            MOV     SP,R0                   ; SP,*
24874 107674  104414                            TRAP    14
24875 107676  062706  000014                    ADD     #14,SP                  ;                               1072
24876 107702  152777  000001  105614  18$:      BISB    #1,@ML.REG+120          ;                               1073
24877 107710  005066  000004                    CLR     4(SP)                   ; FINISH                        1075
24878 107714  005266  000002                    INC     2(SP)                   ; CLR.LOOP                      1065
24879 107720  026627  000002  000001            CMP     2(SP),#1                ; CLR.LOOP,*
24880 107726  003002                            BGT     19$
24881 107730  000167  176742                    JMP     1$
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
24886 107734  062706  000006            19$:    ADD     #6,SP                           ;                                       1060
24887 107740  000207                            RTS     PC

                                        ; Routine Size:  294 words
                                        ; Maximum stack depth per invocation:  23 words


                                                .SBTTL  T56 TEST CODE SECTION
24905 107742                            T56::
24906 107742  004767  176660            1$:     JSR     PC,$T56                         ;                                       1073
24907 107746  104466                            TRAP    66
24908 107750  006000                            ROR     R0
24909 107752  103773                            BLO     1$
24910 107754  000207                            RTS     PC

                                        ; Routine Size:  6 words
                                        ; Maximum stack depth per invocation:  0 words


24920 ;         10735 !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (109)

```
24922 ;ML4AD
24923 ;                 TEST CODE SECTION
24924
24925 ;          10736 !
24926 ;          10737 BGNTST;
24927 ;          10738
24928 ;          10739 !++
24929 ;          10740 ! TEST NUMBER:  TST 57
24930 ;          10741 !
24931 ;          10742 ! TEST NAME:            ECC ERROR LOCATION REGISTER TEST
24932 ;          10743 !
24933 ;          10744 ! TEST DESCRIPTION:
24934 ;          10745 !       TEST THE ECC ERROR LOCATION REGISTER FOR CLOCKING, BIT UNIQUENESS
24935 ;          10746 !       CLEARING AND LATCHING OF DSA ADDRESSES ON DETECTION OF ECC ERROR BY:
24936 ;          10747 !
24937 ;          10748 !       1.  TEST FOR CLOCKING BY:
24938 ;          10749 !           A. DO A MASS BUS CLEAR FORCING REGISTER BITS TO ZEROES
24939 ;          10750 !           B. FORCE AN ECC ERROR AT THE LAST BLOCK AND DO A READ TRANSFER
24940 ;          10751 !           C. CHECK MLEL FOR LAST BLOCK ADDRESS
24941 ;          10752 !
24942 ;          10753 !       2.  TEST FOR BIT UNIQUENESS BY:
24943 ;          10754 !           A. WITH FORCED ECC ERRORS DO READ TRANSFERS AT VARIOUS DSA
24944 ;          10755 !              ADDRESSES AND CHECK MLEL FOR LATCHED DSA ADDRESSES
24945 ;          10756 !           B. DO CLEAR AND REPEAT
24946 ;          10757 !
24947 ;          10758 !       3.  TEST FOR LATCHING AND CLEARING BY:
24948 ;          10759 !           A. WITH A FORCED SINGLE BIT ERROR DO A READ TRANSFER AT THE
24949 ;          10760 !              LAST BLOCK. (LATCHES MLEL WITH THE LAST BLOCK ADDRESS TO START THE TEST OFF)
24950 ;          10761 !           B. AGAIN WITH A FORCED SINGLE BIT ERROR DO A READ TRANSFER AT DSA
24951 ;          10762 !              ZERO AND CHECK MLEL FOR NOT LATCHING.
24952 ;          10763 !              NO CLEAR DONE
24953 ;          10764 !           C. WITH FORCED UNCORRECTABLE ERRORS DO A READ TRANSFER AT DSA
24954 ;          10765 !              ZERO  CHECK MLEL FOR LATCHED DSA ZERO
24955 ;          10766 !              NO CLEAR DONE
24956 ;          10767 !           D. WITH FORCED SINGLE BIT ERRORS DO A READ TRANSFER AT THE LAST
24957 ;          10768 !              BLOCK AND CHECK MLEL FOR NOT LATCHING
24958 ;          10769 !              NO CLEAR DONE
24959 ;          10770 !           E. WITH A FORCED UNCORRECTABLE ERROR DO A READ TRANSFER AT
24960 ;          10771 !              THE LAST BLOCK AN CHECK MLEL FOR NOT LATCHING
24961 ;          10772 !           F. DO A MASS BUS CLEAR
24962 ;          10773 !           G. WITH A FORCED SINGLE BIT ERROR DO A READ TRANSFER AT THE
24963 ;          10774 !              LAST BLOCK
24964 ;          10775 !           H. DO A MASS BUS CLEAR AGAIN AND CHECK MLEL FOR ZEROES
24965 ;          10776 !
24966 ;          10777 ! IMPLICIT INPUTS:
24967 ;          10778 !       IO_BUF
24968 ;          10779 !       A  VECTOR OF 256 WORDS WHERE DATA
24969 ;          10780 !       FOR MBUS READS AND WRITES TRANSFERS
24970 ;          10781 !       IS FOUND.
24971 ;          10782 !
24972 ;          10783 !
24973 ;          10784 !--
24974 ;          10785
24975 ;          10786 local
24976 ;          10787     SGL,                                  !SINGLE BIT ERROR PARRAMETER
```

```
24978 ;ML4AD
24979 ;                 TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
24980 ;                                                      29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (109)
24981 ;          10788      UNC,                             !UNCORRECTABLE ERR PARRAMETER
24982 ;          10789      SAVE,                            !TEMPORY SAVE LOCATION
24983 ;          10790      SHF_DATA,                        !SHIFTING DATA PATTERN SAVE LOCATION
24984 ;          10791      DSA_PAT;                         !EXPECTED DSA PATTERN LATCHED IN MLEL
24985 ;          10792
24986 ;          10793 CLR_THRESHOLD;                        !CLEAR ERROR PRINT THRESHOLD
24987 ;          10794 BGNSUB;                               !START OF SCOPE LOOP
24988 ;          10795 CLR_MBUS;                             !CLEAR THE MLEL REGISTER
24989 ;          10796 FOR_ECC_ERR (SGL = 1, UNC = 0);       !FORCE A SBE
24990 ;          10797 MLDA = .LST_BLK;                      !LATCH THE LAST BLK ADRS INTO THE MLEL
24991 ;          10798 MLWC = not 255;
24992 ;          10799 MLBA = IO_BUF;
24993 ;          10800 MLCS1 = read;                         !DO A READ AND LATCH THE MLEL
24994 ;          10801 TIME_OUT_LOOP;
24995 ;          10802
24996 ;          10803 if .MLEL eql ZEROES                   !SEE IF THE MLEL GOT CLOCKED
24997 ;          10804 then
24998 ;          10805     begin                             !REPORT AN ERROR IF NOT CLOCKED
24999 ;          10806     ERRDF (126, SYNC, DUMPER);
25000 ;          10807     PRINTB (THR_FMT, REG_15, WRD_52, PHR_11);
25001 ;          10808     EXIT_TST;
25002 ;          10809     end;
25003 ;          10810
25004 ;          10811 ENDSUB;                               !END OF SCOPE LOOP
25005 ;          10812 SHF_DATA = .LST_ARR;                  !LOAD  THE SHIFT DATA
25006 ;          10813
25007 ;          10814 incr LOOP from 0 to 16 do             !FIRST TEST FOR 1'S AND 0'S THEN SHIFT DATA
25008 ;          10815     begin
25009 ;          10816     BGNSUB;                           !START OF SCOPE LOOP
25010 ;          10817     CLR_MBUS;                         !CLEAR THE DRIVE
25011 ;          10818     FOR_ECC_ERR (SGL = 1, UNC = 0);   !FORCE A SBE
25012 ;          10819
25013 ;          10820     selectone .LOOP of                !SELECT THE TEST PATTERN
25014 ;          10821         set
25015 ;          10822
25016 ;          10823         [0] :                         !ZEROES PATTERN
25017 ;          10824             begin
25018 ;          10825             DSA_PAT = ZEROES;
25019 ;          10826             MLDA = .DSA_PAT;
25020 ;          10827             end;
25021 ;          10828
25022 ;          10829         [1] :                         !MOSTLY ONES PATTERN
25023 ;          10830             begin
25024 ;          10831             DSA_PAT = .LST_BLK;
25025 ;          10832             MLDA = .DSA_PAT;
25026 ;          10833             end;
25027 ;          10834
25028 ;          10835         [otherwise] :                 !SHIFT PATTERN
25029 ;          10836             begin
25030 ;          10837             DSA_PAT = .SHF_DATA;
25031 ;          10838             MLDA = .SHF_DATA;
25032 ;          10839             SHF_DATA = .SHF_DATA^-ONE;
```

```
10840                end;
10841            tes;
10842
10843        MLWC = not 255;                    !SET UP THE TRANSFER
10844        MLBA = IO_BUF;
10845        MLCS1 = read;                      !DO THE TRANSFER AND LATCH MLEL
10846        TIME_OUT_LOOP;
10847        SAVE = .MLEL;                      !READ AND SAVE THE MLEL REGISTER
10848
10849        if .SAVE neq .DSA_PAT              !SEE IF SAVE IS WHAT WE EXPECTED
10850        then
10851            begin                          !REPORT AN ERROR IF NOT THE SAME
10852            CMP_THRESHOLD;                 !COMPARE ERROR PRINT THRESHOLD
10853            ERRDF (127, ASYNC, DUMPER);
10854            PRINTB (TWO_FMT, REG_15, PHR_4);
10855            PRINTB (FMT_2, .DSA_PAT, .SAVE, (.DSA_PAT xor .SAVE));
10856            end;
10857
10858        ENDSUB;                            !END OF SCOPE LOOP
10859        end;
10860
10861 CLR_MBUS;                                 !CLEAR THE DRIVE
10862 FOR_ECC_ERR (SGL = 1, UNC = 0);           !FORCE A SBE
10863 MLWC = not 255;                           !SET UP A LAST BLOCK TRANSFER
10864 MLBA = IO_BUF;                            !TO LATCH THE MLEL WITH T
10865 MLDA = .LST_BLK;
10866 MLCS1 = read;
10867 TIME_OUT_LOOP;
10868
10869 incr LOOP from 0 to 4 do                  !NOW TEST FOR CORRECT LATCHING
10870     begin
10871     BGNSUB;                               !START OF SCOPE LOOP
10872     MLER = ZEROES;                        !CLEAR THE ERROR REGISTER
10873
10874     case .LOOP from 0 to 4 of             !SELECT TEST DATA
10875         set
10876
10877         [0] :                             !FORCE A SBE AND CHECK FOR THE LST BLK ADRS
10878             begin
10879             SGL = 1;
10880             UNC = 0;
10881             DSA_PAT = .LST_BLK;
10882             MLDA = ZEROES;
10883             end;
10884
10885         [1] :                             !FORCE A UNC ERROR AND CHECK FOR ADRS ZERO
10886             begin
10887             SGL = 0;
10888             UNC = 1;
10889             DSA_PAT = ZEROES;
10890             MLDA = ZEROES;
10891             end;
```

```
25090 ;ML4AD
25091 ;                    TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
25092                                                           29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (109)
25093 ;          10892
25094 ;          10893          [2] :                           !FORCE SBE AND CHECK FOR ADRS ZERO
25095 ;          10894              begin
25096 ;          10895              SGL = 1;
25097 ;          10896              UNC = 0;
25098 ;          10897              DSA_PAT = ZEROES;
25099 ;          10898              MLDA = .LST_BLK;
25100 ;          10899              end;
25101 ;          10900
25102 ;          10901          [3] :                           !FORCE UNC ERROR AND CHECK FOR ADRS ZERO
25103 ;          10902              begin
25104 ;          10903              SGL = 0;
25105 ;          10904              UNC = 1;
25106 ;          10905              DSA_PAT = ZEROES;
25107 ;          10906              MLDA = .LST_BLK;
25108 ;          10907              end;
25109 ;          10908
25110 ;          10909          [4] :                           !FORCE SBE AND CHECK FOR ADRS ZERO
25111 ;          10910              begin
25112 ;          10911              CLR_MBUS;
25113 ;          10912              SGL = 1;
25114 ;          10913              UNC = 0;
25115 ;          10914              DSA_PAT = ZEROES;
25116 ;          10915              MLDA = .LST_BLK;
25117 ;          10916              end;
25118 ;          10917          tes;
25119 ;          10918
25120 ;          10919      FOR_ECC_ERR (.SGL, .UNC);           !FORCE SELECTED FORCED ERROR INTO ML11
25121 ;          10920      MLWC = not 255;                     !SET UP A TRANSFER
25122 ;          10921      MLBA = IO_BUF;
25123 ;          10922      MLCS1 = read;                       !READ THE ERROR OUT AND LATCH MLEL
25124 ;          10923      TIME_OUT_LOOP;
25125 ;          10924
25126 ;          10925      if .LOOP neq 4                      !TEST THE REG FOR LATCH DATA IF LOOP NEQ 4
25127 ;          10926      then
25128 ;          10927          SAVE = .MLEL
25129 ;          10928      else
25130 ;          10929          begin                           !ELSE TEST THE REG FOR CLEAR
25131 ;          10930          CLR_MBUS;
25132 ;          10931          SAVE = .MLEL;
25133 ;          10932          end;
25134 ;          10933
25135 ;          10934      if .SAVE neq .DSA_PAT               !SEE IF THE REG LATCHED  OK
25136 ;          10935      then
25137 ;          10936          begin                           !REPORT ERROR IF NOT LATCHED
25138 ;          10937          CMP_THRESHOLD;                  !COMPARE ERROR PRINT THRESHOLD
25139 ;          10938          ERRDF (128, SYNC, DUMPER);
25140 ;          10939          PRINTB (THR_FMT, REG_15, WRD_71, WRD_10);
25141 ;          10940          PRINTB (FMT_1, .DSA_PAT, .SAVE)
25142 ;          10941          end;
25143 ;          10942
25144 ;          10943      ENDSUB;                             !END OF SCOPE LOOP
```

```
25146 ;ML4AD
25147 ;                    TEST CODE SECTION
25148
25149 ;          10944      end;
25150 ;          10945
25151 ;          10946 ENDTST;
25155
25156                                          .SBTTL  ST57 TEST CODE SECTION
25160 107756  004167  074630          ST57:   JSR     R1,$SAVE5                       ;                       1073
25161 107762  005746                          TST     -(SP)
25162 107764  005067  105406                  CLR     P.CNT                           ;                       1079
25163 107770  104402                  1$:     TRAP    2                               ;                       1079
25164 107772  152777  000040  105444          BISB    #40,@ML.REG+40                  ;                       1079
25165 110000  016705  106026                  MOV     ML.DUT,R5
25166 110004  042705  177770                  BIC     #177770,R5
25167 110010  142777  000007  105426          BICB    #7,@ML.REG+40
25168 110016  150577  105422                  BISB    R5,@ML.REG+40
25169 110022  012703  000001                  MOV     #1,R3                           ; #,SGL                 1079
25170 110026  010346                          MOV     R3,-(SP)                        ; SGL,#
25171 110030  005001                          CLR     R1                              ; UNC
25172 110032  005046                          CLR     -(SP)
25173 110034  004767  110150                  JSR     PC,FOR.ECC.ERR
25174 110040  016777  103750  105366          MOV     LST.BLK,@ML.REG+30              ;                       1079
25175 110046  012777  177400  105340          MOV     #-400,@ML.REG+10                ;                       1079
25176 110054  012777  014022  105342          MOV     #IO.BUF,@ML.REG+20              ;                       1079
25177 110062  012777  000071  105314          MOV     #71,@ML.REG                     ;                       1080
25178 110070  105777  105360          2$:     TSTB    @ML.REG+50
25179 110074  100375                          BPL     2$
25180 110076  005777  105522                  TST     @ML.REG+220                     ;                       1080
25181 110102  001025                          BNE     3$
25182 110104  104455                          TRAP    55                              ;                       1080
25183 110106  000176                          .WORD   176
25184 110110  012750                          .WORD   SYNC
25185 110112  026302                          .WORD   DUMPER
25186 110114  012746  012020                  MOV     #PHR.11,-(SP)                   ;                       1080
25187 110120  012746  011310                  MOV     #WRD.52,-(SP)
25188 110124  012746  012626                  MOV     #REG.15,-(SP)
25189 110130  012746  010256                  MOV     #THR.FMT,-(SP)
25190 110134  012746  000004                  MOV     #4,-(SP)
25191 110140  010600                          MOV     SP,R0                           ; SP,#
25192 110142  104414                          TRAP    14
25193 110144  104463                          TRAP    63
25194 110146  062706  000016                  ADD     #16,SP                          ;                       1080
25195 110152  000167  001052                  JMP     31$                             ;                       1080
25196 110156  022626                  3$:     CMP     (SP)+,(SP)+                     ;                       1079
25197 110160  104467                          TRAP    67                              ;                       1080
25198 110162  006000                          ROR     R0
25199 110164  103701                          BLO     1$
25200 110166  016705  103626                  MOV     LST.ARR,R5                      ; @,SHF.DATA            1081
```

```
25202                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
25203                                   ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
25204
25205 110172  005002                            CLR     R2                      ; LOOP                                  1081
25206 110174  104402                    4$:     TRAP    2                       ;                                       1081
25207 110176  152777  000040  105240            BISB    #40,@ML.REG+40          ;                                       1081
25208 110204  016700  105622                    MOV     ML.DUT,R0
25209 110210  042700  177770                    BIC     #177770,R0
25210 110214  142777  000007  105222            BICB    #7,@ML.REG+40
25211 110222  150077  105216                    BISB    R0,@ML.REG+40
25212 110226  012703  000001                    MOV     #1,R3                                                           1081
25213 110232  010346                            MOV     R3,-(SP)                ; *,SGL
25214 110234  005001                            CLR     R1                      ; SGL,*
25215 110236  005046                            CLR     -(SP)                   ; UNC
25216 110240  004767  107744                    JSR     PC,FOR.ECC.ERR
25217 110244  005702                            TST     R2                      ; LOOP                                  1082
25218 110246  001002                            BNE     5$
25219 110250  005004                            CLR     R4                      ; DSA.PAT                               1082
25220 110252  000405                            BR      6$                      ;                                       1082
25221 110254  020227  000001            5$:     CMP     R2,#1                   ; LOOP,*                                1082
25222 110260  001005                            BNE     7$
25223 110262  016704  103526                    MOV     LST.BLK,R4              ; *,DSA.PAT                             1083
25224 110266  010477  105142            6$:     MOV     R4,@ML.REG+30           ; DSA.PAT,*                             1083
25225 110272  000404                            BR      8$                      ;                                       1082
25226 110274  010504                    7$:     MOV     R5,R4                   ; SHF.DATA,DSA.PAT                      1083
25227 110276  010577  105132                    MOV     R5,@ML.REG+30           ; SHF.DATA,*                            1083
25228 110302  006205                            ASR     R5                      ; SHF.DATA                              1083
25229 110304  012777  177400  105102    8$:     MOV     #-400,@ML.REG+10        ;                                       1084
25230 110312  012777  014022  105104            MOV     #10.BUF,@ML.REG+20      ;                                       1084
25231 110320  012777  000071  105056            MOV     #71,@ML.REG             ;                                       1084
25232 110326  105777  105122            9$:     TSTB    @ML.REG+50
25233 110332  100375                            BPL     9$
25234 110334  017766  105264  000004            MOV     @ML.REG+220,4(SP)       ; *,SAVE                                1084
25235 110342  026604  000004                    CMP     4(SP),R4                ; SAVE,DSA.PAT                          1084
25236 110346  001451                            BEQ     11$
25237 110350  005267  105022                    INC     P.CNT                   ;                                       1085
25238 110354  026767  105016  105016            CMP     P.CNT,LIMIT
25239 110362  003402                            BLE     10$
25240 110364  022626                            CMP     (SP)+,(SP)+
25241 110366  000445                            BR      12$
25242 110370  104455            10$:    TRAP    55                      ;                                               1085
25243 110372  000177                    .WORD   177
25244 110374  012706                    .WORD   ASYNC
25245 110376  026302                    .WORD   DUMPER
25246 110400  012746  011676                    MOV     #PHR.4,-(SP)            ;                                       1085
25247 110404  012746  012626                    MOV     #REG.15,-(SP)
25248 110410  012746  010246                    MOV     #TWO.FMT,-(SP)
25249 110414  012746  000003                    MOV     #3,-(SP)
25250 110420  010600                            MOV     SP,R0                   ; SP,*
25251 110422  104414                            TRAP    14
25252 110424  016616  000014                    MOV     14(SP),(SP)             ; SAVE,*                                1085
25253 110430  010446                            MOV     R4,-(SP)                ; DSA.PAT,*
25254 110432  046616  000002                    BIC     2(SP),(SP)
25255 110436  040466  000002                    BIC     R4,2(SP)                ; DSA.PAT,*
25256 110442  052616                            BIS     (SP)+,(SP)
```

SEQ 0520

29-Mar-1982 16:23:04 TOPS
29-Mar-1982 16:21:03 PA:<

```
25258                                   ;ML4AD
25259                                   ;
25260                                   ;       TEST CODE SECTION
25261 110444 016646 000014              MOV     14(SP),-(SP)            ; SAVE,*
25262 110450 010446                     MOV     R4,-(SP)                ; DSA.PAT,*
25263 110452 012746 006506              MOV     #FMT.2,-(SP)
25264 110456 012746 000004              MOV     #4,-(SP)
25265 110462 010600                     MOV     SP,R0                   ; SP,*
25266 110464 104414                     TRAP    14                                              1085
25267 110466 062706 000020              ADD     #20,SP                  ;                       1081
25268 110472 022626             11$:    CMP     (SP)+,(SP)+             ;                       1085
25269 110474 104467                     TRAP    67                      ;
25270 110476 006000                     ROR     R0
25271 110500 103635                     BLO     4$
25272 110502 005202             12$:    INC     R2                      ; LOOP                  1081
25273 110504 020227 000020              CMP     R2,#20                  ; LOOP,*
25274 110510 003631                     BLE     4$
25275 110512 152777 000040 104724       BISB    #40,@ML.REG+40          ;                       1085
25276 110520 016705 105306              MOV     ML.DUT,R5
25277 110524 042705 177770              BIC     #177770,R5
25278 110530 142777 000007 104706       BICB    #7,@ML.REG+40
25279 110536 150577 104702              BISB    R5,@ML.REG+40
25280 110542 012703 000001              MOV     #1,R3                   ; *,SGL                 1086
25281 110546 010346                     MOV     R3,-(SP)                ; SGL,*
25282 110550 005001                     CLR     R1                      ; UNC
25283 110552 005046                     CLR     -(SP)
25284 110554 004767 107430              JSR     PC,FOR.ECC.ERR
25285 110560 012777 177400 104626       MOV     #-400,@ML.REG+10        ;                       1086
25286 110566 012777 014022 104630       MOV     #IO.BUF,@ML.REG+20      ;                       1086
25287 110574 016777 103214 104632       MOV     LST.BLK,@ML.REG+30      ;                       1086
25288 110602 012777 000071 104574       MOV     #71,@ML.REG             ;                       1086
25289 110610 105777 104640      13$:    TSTB    @ML.REG+50
25290 110614 100375                     BPL     13$
25291 110616 005002                     CLR     R2                      ; LOOP                  1086
25292 110620 104402             14$:    TRAP    2                       ;                       1087
25293 110622 005077 104636              CLR     @ML.REG+60              ;                       1087
25294 110626 010205                     MOV     R2,R5                   ; LOOP,*                1087
25295 110630 006305                     ASL     R5
25296 110632 066507 110636              ADD     15$(R5),PC
25297 110636 000012             15$:    .WORD   16$-15$
25298 110640 000026                     .WORD   17$-15$
25299 110642 000044                     .WORD   19$-15$
25300 110644 000046                     .WORD   20$-15$
25301 110646 000056                     .WORD   21$-15$
25302 110650 012703 000001      16$:    MOV     #1,R3                   ; *,SGL                 1087
25303 110654 005001                     CLR     R1                      ; UNC                   1088
25304 110656 016704 103132              MOV     LST.BLK,R4              ; *,DSA.PAT             1088
25305 110662 000404                     BR      18$                     ;                       1088
25306 110664 005003             17$:    CLR     R3                      ; SGL                   1088
25307 110666 012701 000001              MOV     #1,R1                   ; *,UNC                 1088
25308 110672 005004                     CLR     R4                      ; DSA.PAT               1088
25309 110674 005077 104534      18$:    CLR     @ML.REG+30              ;                       1089
25310 110700 000430                     BR      24$                     ;                       1087
25311 110702 000420             19$:    BR      22$                     ;                       1089
25312 110704 005003             20$:    CLR     R3                      ; SGL                   1090
```

```
25314                                          ;ML4AD
25315                                          ;                TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS
25316                                                                                               29-Mar-1982 16:21:03    PA:<
25317 110706  012701  000001                         MOV     #1,R1                   ; *,UNC                                 1090
25318 110712  000417                                 BR      23$                     ;                                       1090
25319 110714  152777  000040  104522  21$:           BISB    #40,@ML.REG+40          ;                                       1091
25320 110722  016705  105104                         MOV     ML.DUT,R5
25321 110726  042705  177770                         BIC     #177770,R5
25322 110732  142777  000007  104504                 BICB    #7,@ML.REG+40
25323 110740  150577  104500                         BISB    R5,@ML.REG+40
25324 110744  012703  000001          22$:           MOV     #1,R3                   ; *,SGL                                 1091
25325 110750  005001                                 CLR     R1                      ; UNC                                   1091
25326 110752  005004                 23$:           CLR     R4                      ; DSA.PAT                               1091
25327 110754  016777  103034  104452                 MOV     LST.BLK,@ML.REG+30      ;                                       1091
25328 110762  010346                 24$:           MOV     R3,-(SP)                ; SGL,*                                 1091
25329 110764  010146                                 MOV     R1,-(SP)                ; UNC,*
25330 110766  004767  107216                         JSR     PC,FOR.ECC.ERR
25331 110772  012777  177400  104414                 MOV     #-400,@ML.REG+10        ;                                       1092
25332 111000  012777  014022  104416                 MOV     #IO.BUF,@ML.REG+20      ;                                       1092
25333 111006  012777  000071  104370                 MOV     #71,@ML.REG             ;                                       1092
25334 111014  105777  104434          25$:           TSTB    @ML.REG+50
25335 111020  100375                                 BPL     25$
25336 111022  020227  000004                         CMP     R2,#4                   ; LOOP,*                                1092
25337 111026  001014                                 BNE     26$                     ;                                       1092
25338 111030  152777  000040  104406                 BISB    #40,@ML.REG+40          ;                                       1092
25339 111036  016705  104770                         MOV     ML.DUT,R5
25340 111042  042705  177770                         BIC     #177770,R5
25341 111046  142777  000007  104370                 BICB    #7,@ML.REG+40
25342 111054  150577  104364                         BISB    R5,@ML.REG+40
25343 111060  017766  104540  000010  26$:           MOV     @ML.REG+220,10(SP)      ; *,SAVE                                1093
25344 111066  026604  000010                         CMP     10(SP),R4               ; SAVE,DSA.PAT                          1093
25345 111072  001443                                 BEQ     28$
25346 111074  005267  104276                         INC     P.CNT                   ;                                       1093
25347 111100  026767  104272  104272                 CMP     P.CNT,LIMIT
25348 111106  003402                                 BLE     27$
25349 111110  022626                                 CMP     (SP)+,(SP)+
25350 111112  000437                                 BR      29$
25351 111114  104455                 27$:           TRAP    55                      ;                                       1093
25352 111116  000200                                 .WORD   200
25353 111120  012750                                 .WORD   SYNC
25354 111122  026302                                 .WORD   DUMPER
25355 111124  012746  010630                         MOV     #WRD.10,-(SP)           ;                                       1093
25356 111130  012746  011514                         MOV     #WRD.71,-(SP)
25357 111134  012746  012626                         MOV     #REG.15,-(SP)
25358 111140  012746  010256                         MOV     #THR.FMT,-(SP)
25359 111144  012746  000004                         MOV     #4,-(SP)
25360 111150  010600                                 MOV     SP,R0                   ; SP,*
25361 111152  104414                                 TRAP    14
25362 111154  016616  000022                         MOV     22(SP),(SP)             ; SAVE,*                                1094
25363 111160  010446                                 MOV     R4,-(SP)                ; DSA.PAT,*
25364 111162  012746  006442                         MOV     #FMT.1,-(SP)
25365 111166  012746  000003                         MOV     #3,-(SP)
25366 111172  010600                                 MOV     SP,R0                   ; SP,*
25367 111174  104414                                 TRAP    14
25368 111176  062706  000020                         ADD     #20,SP                  ;                                       1093
```

```
25370                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
25371                                   ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
25372
25373 111202  022626                    28$:    CMP     (SP)+,(SP)+                     ;                               1087
25374 111204  104467                            TRAP    67                              ;                               1094
25375 111206  006000                            ROR     R0
25376 111210  103603                            BLO     14$
25377 111212  005202                    29$:    INC     R2                              ; LOOP                          1086
25378 111214  020227  000004                    CMP     R2,#4                           ; LOOP,*
25379 111220  003002                            BGT     30$
25380 111222  000167  177372                    JMP     14$
25381 111226  022626                    30$:    CMP     (SP)+,(SP)+                     ;                               1073
25382 111230  005726                    31$:    TST     (SP)+
25383 111232  000207                            RTS     PC
25384
25385                                   ; Routine Size:  343 words
25386                                   ; Maximum stack depth per invocation:  19 words
25391
25392
25396
25397                                           .SBTTL  T57 TEST CODE SECTION
25401 111234                            T57::
25402 111234  004767  176516            1$:     JSR     PC,$T57                         ;                               1094
25403 111240  104466                            TRAP    66
25404 111242  006000                            ROR     R0
25405 111244  103773                            BLO     1$
25406 111246  000207                            RTS     PC
25407
25408                                   ; Routine Size:  6 words
25409                                   ; Maximum stack depth per invocation:  0 words
25414
25415
25416 ;          10947 !<BLF/PAGE>
```

```
25418 ;ML4AD
25419 ;                 TEST CODE SECTION
25420
25421 ;          10948 !
25422 ;          10949 BGNTST;
25423 ;          10950
25424 ;          10951 !++
25425 ;          10952 ! TEST NUMBER:  TST 58
25426 ;          10953 !
25427 ;          10954 ! TEST NAME:              ECC HARD ERROR BIT TEST
25428 ;          10955 !
25429 ;          10956 ! TEST DESCRIPTION:
25430 ;          10957 !       VIA ECC_DM AND ECC_DIS TEST THE
25431 ;          10958 !       ECH BIT FOR SETTING AND NOT SETTING BY:
25432 ;          10959 !
25433 ;          10960 !       1.  VIA ECC_DM AND A WRITE TRANSFER FORCE
25434 ;          10961 !           ECC ERRORS INTO THE ML11'S GOOD BLOCK
25435 ;          10962 !
25436 ;          10963 !       2.  WITH ECC_DM = 0 AND ECC_DIS = 0
25437 ;          10964 !           DO A READ TRANSFER AND CHECK ECH SET
25438 ;          10965 !
25439 ;          10966 !       3.  WITH ECC_DM = 0 AND ECC_DIS = 1
25440 ;          10967 !           DO A READ TRANSFER AND CHECK ECH NOT SET
25441 ;          10968 !
25442 ;          10969 ! IMPLICIT INPUTS:
25443 ;          10970 !       IO_BUF
25444 ;          10971 !       A VECTOR OF 256 WORDS WHERE DATA
25445 ;          10972 !       FOR MBUS READS AND WRITES TRANSFERS
25446 ;          10973 !       IS FOUND.
25447 ;          10974 !
25448 ;          10975 !
25449 ;          10976 !--
25450 ;          10977
25451 ;          10978 local
25452 ;          10979     EXPECTED;                    !EXPECTED ECH_ERR VALUE
25453 ;          10980
25454 ;          10981 CLR_MBUS;                        !CLEAR THE DRIVE
25455 ;          10982 ECC_DM = ONE;                    !SET ECC DIAG MODE
25456 ;          10983 PAR_CRC_WRD = ZEROES;            !FORCE UNCORRECTABLE ERRORS INTO THE GOOD BLOCK
25457 ;          10984 CRC_A = ZEROES;
25458 ;          10985 CRC_B = ZEROES;
25459 ;          10986 BAI = ONE;
25460 ;          10987 IO_BUF = ONES;
25461 ;          10988 GD_BLK_XFER ();
25462 ;          10989 MLCS1 = write;
25463 ;          10990 TIME_OUT_LOOP;
25464 ;          10991
25465 ;          10992 incr LOOP from 0 to 1 do       !TEST ECH BIT FOR SETTING / NOT SETTING
25466 ;          10993     begin
25467 ;          10994     BGNSUB;                    !START OF SCOPE LOOP
25468 ;          10995     CLR_MBUS;                  !CLEAR THE DRIVE
25469 ;          10996     GD_BLK_XFER ();            !SET UP A GOOD BLOCK TRANSFER
25470 ;          10997
25471 ;          10998     if .LOOP eql ZERO          !SELECT THE TEST DATA
25472 ;          10999     then
```

```
25474 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
25475 ;                 TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI 4 (110)
25476
25477 ;         11000           begin                           !EXPECT ECH BIT TO SET
25478 ;         11001           ECC_DIS = 0;
25479 ;         11002           EXPECTED = 1;
25480 ;         11003           end
25481 ;         11004       else
25482 ;         11005           begin                           !EXPECT ECH BIT NOT TO SET
25483 ;         11006           ECC_DIS = 1;
25484 ;         11007           EXPECTED = 0;
25485 ;         11008           end;
25486 ;         11009
25487 ;         11010       MLCS1 = read;                       !READ THE ERROR OUT
25488 ;         11011       TIME_OUT_LOOP;
25489 ;         11012
25490 ;         11013       if .ECH_ERR neq .EXPECTED           !IS ECH BIT WHAT WE EXPECTED
25491 ;         11014       then
25492 ;         11015           begin                           !ERROR IF NOT
25493 ;         11016           ERRDF (129, SYNC, DUMPER);
25494 ;         11017           PRINTB (FOR_FMT, WRD_67, WRD_10, WRD_69, WRD_14);
25495 ;         11018           PRINTB (FMT_1, .EXPECTED, .ECH_ERR);
25496 ;         11019           end;
25497 ;         11020
25498 ;         11021       ENDSUB;                             !END OF SCOPE LOOP
25499 ;         11022       end;
25500 ;         11023
25501 ;         11024 ENDTST;
25505
25506                                   .SBTTL  ST58 TEST CODE SECTION
25510 111250  004167  073300          ST58:   JSR     R1,$SAVE3                                       1094
25511 111254  152777  000040  104162          BISB    #40,@ML.REG+40          ;                       1097
25512 111262  016702  104544                  MOV     ML.DUT,R2
25513 111266  042702  177770                  BIC     #177770,R2
25514 111272  142777  000007  104144          BICB    #7,@ML.REG+40
25515 111300  150277  104140                  BISB    R2,@ML.REG+40
25516 111304  152777  000001  104212          BISB    #1,@ML.REG+120          ;                       1098
25517 111312  042777  037400  104234          BIC     #37400,@ML.REG+150      ;                       1098
25518 111320  142777  000077  104226          BICB    #77,@ML.REG+150         ;                       1098
25519 111326  142777  000077  104230          BICB    #77,@ML.REG+160         ;                       1098
25520 111334  152777  000010  104102          BISB    #10,@ML.REG+40          ;                       1098
25521 111342  012767  177777  102452          MOV     #-1,IO.BUF              ;                       1098
25522 111350  004767  106736                  JSR     PC,GD.BLK.XFER          ;                       1098
25523 111354  012777  000061  104022          MOV     #61,@ML.REG             ;                       1098
25524 111362  105777  104066          1$:     TSTB    @ML.REG+50
25525 111366  100375                  BPL     1$
25526 111370  005001                  CLR     R1                      ; LOOP                          1099
25527 111372  104402          2$:     TRAP    2                       ;                               1099
25528 111374  152777  000040  104042          BISB    #40,@ML.REG+40          ;                       1099
```

```
25530                                   ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
25531                                   ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
25532
25533 111402  016700  104424                    MOV     ML.DUT,R0
25534 111406  042700  177770                    BIC     #177770,R0
25535 111412  142777  C00007  104024            BICB    #7,@ML.REG+40
25536 111420  150077  104020                    BISB    R0,@ML.REG+40
25537 111424  004767  106662                    JSR     PC,GD.BLK.XFER          ;                                                       1099
25538 111430  005701                            TST     R1                      ; LOOP                                                  1099
25539 111432  001006                            BNE     3$                                                                              1099
25540 111434  142777  000002  104062            BICB    #2,@ML.REG+120          ;                                                       1100
25541 111442  012702  000001                    MOV     #1,R2                   ; *,EXPECTED                                            1100
25542 111446  000404                            BR      4$                      ;                                                       1099
25543 111450  152777  000002  104046  3$:       BISB    #2,@ML.REG+120          ;                                                       1100
25544 111456  005002                            CLR     R2                      ; EXPECTED                                              1100
25545 111460  012777  000071  103716  4$:       MOV     #71,@ML.REG             ;                                                       1101
25546 111466  105777  103762        5$:         TSTB    @ML.REG+50
25547 111472  100375                            BPL     5$
25548 111474  010203                            MOV     R2,R3                   ; EXPECTED,*                                            1101
25549 111476  005000                            CLR     R0
25550 111500  132777  000100  103756            BITB    #100,@ML.REG+60
25551 111506  001401                            BEQ     6$
25552 111510  005200                            INC     R0
25553 111512  020003                  6$:       CMP     R0,R3
25554 111514  001441                            BEQ     8$
25555 111516  104455                            TRAP    55                      ;                                                       1101
25556 111520  000201                            .WORD   201
25557 111522  012750                            .WORD   SYNC
25558 111524  026302                            .WORD   DUMPER
25559 111526  012746  010664                    MOV     #WRD.14,-(SP)           ;                                                       1101
25560 111532  012746  011474                    MOV     #WRD.69,-(SP)
25561 111536  012746  010630                    MOV     #WRD.10,-(SP)
25562 111542  012746  011462                    MOV     #WRD.67,-(SP)
25563 111546  012746  010270                    MOV     #FOR.FMT,-(SP)
25564 111552  012746  000005                    MOV     #5,-(SP)
25565 111556  010600                            MOV     SP,R0                   ; SP,*
25566 111560  104414                            TRAP    14
25567 111562  005016                            CLR     (SP)                    ;                                                       1101
25568 111564  132777  000100  103672            BITB    #100,@ML.REG+60
25569 111572  001401                            BEQ     7$
25570 111574  005216                            INC     (SP)
25571 111576  010246                  7$:       MOV     R2,-(SP)                ; EXPECTED,*
25572 111600  012746  006442                    MOV     #FMT.1,-(SP)
25573 111604  012746  000003                    MOV     #3,-(SP)
25574 111610  010600                            MOV     SP,R0                   ; SP,*
25575 111612  104414                            TRAP    14
25576 111614  062706  000022                    ADD     #22,SP                  ;                                                       1101
25577 111620  104467                  8$:       TRAP    67                      ;                                                       1101
25578 111622  006000                            ROR     R0
25579 111624  103662                            BLO     2$
25580 111626  005201                            INC     R1                      ; LOOP                                                  1099
25581 111630  020127  000001                    CMP     R1,#1                   ; LOOP,*
25582 111634  003656                            BLE     2$
25583 111636  000207                            RTS     PC                      ;                                                       1094
25584
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
                                        ; Routine Size:  124 words
                                        ; Maximum stack depth per invocation:  13 words
                                                .SBTTL  T58 TEST CODE SECTION
25605 111640                            T58::
25606 111640  004767  177404            1$:     JSR     PC,$T58                         ;               1102
25607 111644  104466                            TRAP    66
25608 111646  006000                            ROR     R0
25609 111650  103773                            BLO     1$
25610 111652  000207                            RTS     PC
                                        ; Routine Size:  6 words
                                        ; Maximum stack depth per invocation:  0 words
25620 ;         11025 !<BLF/PAGE>
```

29-Mar-1982 16:23:04 TOPS-20 Bliss-16 V2(212)
29-Mar-1982 16:21:03 PA:<NEALE>ML4AD.BLI.4 (111)

```
25622 ;ML4AD
25623 ;                 TEST CODE SECTION
25624 ;
25625 ;          11026 !
25626 ;          11027 BGNTST;
25627 ;          11028
25628 ;          11029 !++
25629 ;          11030 ! TEST NUMBER:  TST 59
25630 ;          11031 !
25631 ;          11032 ! TEST NAME:            DATA CHECK ERROR BIT TEST
25632 ;          11033 !
25633 ;          11034 ! TEST DESCRIPTION:
25634 ;          11035 !       VIA ECC_EN, ECC_DM AND ECC_DM TEST THE
25635 ;          11036 !       DCK BIT FOR SETTING AND NOT SETTING BY:
25636 ;          11037 !
25637 ;          11038 !       1.  VIA ECC_DM AND A WRITE TRANSFER FORCE
25638 ;          11039 !           ECC ERRORS INTO THE ML11'S GOOD BLOCK
25639 ;          11040 !
25640 ;          11041 !       2.  WITH DCK_EN = 1, ECC_DM = 0, ECC_DIS = 0
25641 ;          11042 !           DO A READ TRANSFER AND CHECK THE DCK BIT SET
25642 ;          11043 !
25643 ;          11044 !       3.  WITH DCK_EN = 0, ECC_DM = 0, ECC_DIS = 0
25644 ;          11045 !           DO A READ TRANSFER AND CHECK THE DCK BIT NOT SET
25645 ;          11046 !
25646 ;          11047 !       4.  WITH ECK_EN = 1, ECC_DM = 0, ECC_DIS = 1
25647 ;          11048 !           DO A READ TRANSFER AND CHECK THE DCK BIT SET
25648 ;          11049 !
25649 ;          11050 !
25650 ;          11051 ! IMPLICIT INPUTS:
25651 ;          11052 !       IO_BUF
25652 ;          11053 !       A  VECTOR OF 256 WORDS WHERE DATA
25653 ;          11054 !       FOR MBUS READS AND WRITES TRANSFERS
25654 ;          11055 !       IS FOUND.
25655 ;          11056 !
25656 ;          11057 !
25657 ;          11058 !--
25658 ;          11059
25659 ;          11060 local
25660 ;          11061     SGL,                          !SINGLE BIT ERROR PARRAMETER
25661 ;          11062     UNC,                          !UNCORRECTABLE ERROR PARRAMETER
25662 ;          11063     EXPECTED;                     !EXPECTED DATA CHECK BIT VALUE
25663 ;          11064
25664 ;          11065 CLR_MBUS;                         !CLEAR THE DRIVE
25665 ;          11066 FOR_ECC_ERR (SGL = 1, UNC = 0);   !FORCE  A SBE
25666 ;          11067 BAI = ONE;
25667 ;          11068 IO_BUF = ZEROES;
25668 ;          11069 GD_BLK_XFER ();                   !SET UP A GOOD BLOCK TRANSFER
25669 ;          11070 MLCS1 = write;                    !WRITE THE DRIVE WITH THE ERROR
25670 ;          11071 TIME_OUT_LOOP;
25671 ;          11072
25672 ;          11073 incr LOOP from 0 to 2 do          !TEST DCK BIT FOR 3 CONDITIONS
25673 ;          11074     begin
25674 ;          11075     BGNSUB;                       !START OF SCOPE LOOP
25675 ;          11076     CLR_MBUS;                     !CLEAR THE DRIVE
25676 ;          11077     GD_BLK_XFER ();               !SET UP A GOOD BLOCK TRANSFER
```

```
25678 ;ML4AD
25679 ;                 TEST CODE SECTION
25680
25681 ;          11078
25682 ;          11079      case .LOOP from 0 to 2 of                    !SELECT THE TEST DATA
25683 ;          11080          set
25684 ;          11081
25685 ;          11082          [0] :                                    !EXPECT THE DCK BIT TO SET
25686 ;          11083              begin
25687 ;          11084              DCK_EN = 1;
25688 ;          11085              ECC_DIS = 0;
25689 ;          11086              EXPECTED = 1;
25690 ;          11087              end;
25691 ;          11088
25692 ;          11089          [1] :                                    !EXPECT THE DCK BIT NOT TO SET
25693 ;          11090              begin
25694 ;          11091              DCK_EN = 0;
25695 ;          11092              ECC_DIS = 0;
25696 ;          11093              EXPECTED = 0;
25697 ;          11094              end;
25698 ;          11095
25699 ;          11096          [2] :                                    !EXPECT THE DCK BIT TO SET
25700 ;          11097              begin
25701 ;          11098              DCK_EN = 1;
25702 ;          11099              ECC_DIS = 1;
25703 ;          11100              EXPECTED = 1;
25704 ;          11101              end;
25705 ;          11102          tes;
25706 ;          11103
25707 ;          11104      MLCS1 = read;                                !READ THE ERROR AND CLOCK THE DCK BIT
25708 ;          11105      TIME_OUT_LOOP;
25709 ;          11106
25710 ;          11107      if .DCK neq .EXPECTED                        !IS THE DCK BIT WHAT WE EXPECTED
25711 ;          11108      then
25712 ;          11109          begin                                    !REPORT THE ERRRO IF NOT THE SAME
25713 ;          11110
25714 ;          11111          if .LOOP neq ONE then ERRDF (130, SYNC, DUMPER) else ERRDF (130, ASYNC, DUMPER);
25715 ;          11112
25716 ;          11113                                                   !CHOOSE WHICH THE ERROR IS ON
25717 ;          11114          PRINTB (THR_FMT, WRD_72, PHR_5, WRD_10);
25718 ;          11115          PRINTB (FMT_1, .EXPECTED, .DCK);
25719 ;          11116          end;
25720 ;          11117
25721 ;          11118      ENDSUB;                                      !END OF THE SCOPE LOOP
25722 ;          11119      end;
25723 ;          11120
25724 ;          11121 ENDTST;
25728
25729                                  .SBTTL  $T59 TEST CODE SECTION
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
                                        $T59:   JSR     R1,$SAVE3
                                                BISB    #40,@ML.REG+40          ;                                               1102
                                                MOV     ML.DUT,R2               ;                                               1106
                                                BIC     #177770,R2
                                                BICB    #7,@ML.REG+40
                                                BISB    R2,@ML.REG+40
                                                MOV     #1,R1
                                                MOV     R1,-(SP)                ; *,SGL                                         1106
                                                CLR     R1                      ; SGL,*
                                                CLR     -(SP)                   ; UNC
                                                JSR     PC,FOR.ECC.ERR
                                                BISB    #10,@ML.REG+40          ;                                               1106
                                                CLR     IO.BUF                  ;                                               1106
                                                JSR     PC,GD.BLK.XFER          ;                                               1106
                                                MOV     #61,@ML.REG             ;                                               1107
                                        1$:     TSTB    @ML.REG+50
                                                BPL     1$
                                                CLR     R1                      ; LOOP                                          1107
                                        2$:     TRAP    2                       ;                                               1107
                                                BISB    #40,@ML.REG+40          ;                                               1107
                                                MOV     ML.DUT,R0
                                                BIC     #177770,R0
                                                BICB    #7,@ML.REG+40
                                                BISB    R0,@ML.REG+40
                                                JSR     PC,GD.BLK.XFER          ;                                               1107
                                                MOV     R1,R0                   ; LOOP,*                                        1107
                                                ASL     R0
                                                ADD     3$(R0),PC
                                        3$:     .WORD   4$-3$
                                                .WORD   5$-3$
                                                .WORD   6$-3$
                                        4$:     BISB    #4,@ML.REG+120          ;                                               1108
                                                BICB    #2,@ML.REG+120          ;                                               1108
                                                BR      7$                      ;                                               1108
                                        5$:     BICB    #4,@ML.REG+120          ;                                               1109
                                                BICB    #2,@ML.REG+120          ;                                               1109
                                                CLR     R2                      ; EXPECTED                                      1109
                                                BR      8$                      ;                                               1107
                                        6$:     BISB    #4,@ML.REG+120          ;                                               1109
                                                BISB    #2,@ML.REG+120          ;                                               1109
                                        7$:     MOV     #1,R2                   ; *,EXPECTED                                    1110
                                        8$:     MOV     #71,@ML.REG             ;                                               1110
                                        9$:     TSTB    @ML.REG+50
                                                BPL     9$
                                                MOV     R2,R3                   ; EXPECTED,*                                    1110
                                                CLR     R0
                                                TST     @ML.REG+60
                                                BPL     10$
                                                INC     R0
                                        10$:    CMP     R0,R3
                                                BEQ     14$
                                                CMP     R1,#1                   ; LOOP,*                                        1111
```

```
25790                                   ;ML4AD
25791                                   ;                TEST CODE SECTION                                  29-Mar-1982 16:23:04    TOPS
25792                                                                                                      29-Mar-1982 16:21:03    PA:<
25793 112154  001405                            BEQ     11$
25794 112156  104455                            TRAP    55
25795 112160  000202                            .WORD   202
25796 112162  012750                            .WORD   SYNC
25797 112164  026302                            .WORD   DUMPER
25798 112166  000404                            BR      12$
25799 112170  104455                    11$:    TRAP    55
25800 112172  000202                            .WORD   202
25801 112174  012706                            .WORD   ASYNC
25802 112176  026302                            .WORD   DUMPER
25803 112200  012746  010630            12$:    MOV     #WRD.10,-(SP)           ;                          1111
25804 112204  012746  011714                    MOV     #PHR.5,-(SP)
25805 112210  012746  011524                    MOV     #WRD.72,-(SP)
25806 112214  012746  010256                    MOV     #THR.FMT,-(SP)
25807 112220  012746  000004                    MOV     #4,-(SP)
25808 112224  010600                            MOV     SP,R0                   ; SP,*
25809 112226  104414                            TRAP    14
25810 112230  005016                            CLR     (SP)                    ;                          1111
25811 112232  005777  103226                    TST     @ML.REG+60
25812 112236  100001                            BPL     13$
25813 112240  005216                            INC     (SP)
25814 112242  010246                    13$:    MOV     R2,-(SP)                ; EXPECTED,*
25815 112244  012746  006442                    MOV     #FMT.1,-(SP)
25816 112250  012746  000003                    MOV     #3,-(SP)
25817 112254  010600                            MOV     SP,R0                   ; SP,*
25818 112256  104414                            TRAP    14
25819 112260  062706  000020                    ADD     #20,SP                  ;                          1110
25820 112264  104467                    14$:    TRAP    67                      ;                          1111
25821 112266  006000                            ROR     R0
25822 112270  103634                            BLO     2$
25823 112272  005201                            INC     R1                      ; LOOP                     1107
25824 112274  020127  000002                    CMP     R1,#2                   ; LOOP,*
25825 112300  003630                            BLE     2$
25826 112302  022626                            CMP     (SP)+,(SP)+             ;                          1102
25827 112304  000207                            RTS     PC
25828
25829                                   ; Routine Size:  141 words
25830                                   ; Maximum stack depth per invocation:  14 words
25835
25836
25840
25841                                           .SBTTL  T59 TEST CODE SECTION
```

```
25846                                   ;ML4AD
25847                                   ;               TEST CODE SECTION
25848
25849 112306                            T59::
25850 112306  004767  177342            1$:     JSR     PC,$T59                 ;
25851 112312  104466                            TRAP    66
25852 112314  006000                            ROR     R0
25853 112316  103773                            BLO     1$
25854 112320  000207                            RTS     PC
25855
25856                                   ; Routine Size:  6 words
25857                                   ; Maximum stack depth per invocation:  0 words
25862
25863
25864 ;          11122 !<BLF/PAGE>
```

```
11123 !
11124 BGNTST;
11125
11126 !++
11127 ! TEST NUMBER:  TST 60
11128 !
11129 ! TEST NAME:    PROM DATA TEST
11130 !
11131 ! TEST DESCRIPTION:
11132 !
11133 !       VERIFY THAT CHECK SUM VALUES FOR
11134 !       ALL PRESENT MEMORY ARRAY UV PROM
11135 !       LOCATIONS ARE ERROR FREE BY:
11136 !
11137 !       1.  DOING MBUS TRANSFERS
11138 !           AT ALL PRESENT BLOCKS
11139 !           AND TESTING THE UNS BIT
11140 !           TO BE CLEARED.
11141 !
11142 ! IMPLICIT INPUTS:
11143 !
11144 !       IO_BUF
11145 !
11146 !       A VECTOR OF 256 WORDS WHERE
11147 !       DATA FOR MBUS READ AND WRITE
11148 !       FUNCTIONS IS FOUND.
11149 !
11150 !
11151 !       CHIP_SIZ
11152 !       INDICATED THE SIZE OF THE
11153 !       ARRAY MODULES MOS RAMS.
11154 !
11155 !
11156 !--
11157
11158 CLR_THRESHOLD;                                  !CLEAR ERROR PRINT THRESHOLD
11159
11160 !
11161 !VERSION CZMLAD CHANGED TEST TO UNSIGNED TEST
11162 !
11163 incrU DSA_CNT from 0 to .LST_BLK do             !WRITE TO ALL PRESENT BLK'S AND CHECK UNS BIT
11164     begin
11165     BGNSUB;
11166     CLR_MBUS;
11167     ECC_DIS = ONE;                              !DISABLE ECC
11168     MLWC = not 255;                             !LOAD WORD COUNT
11169     MLBA = IO_BUF;                              !LOAD UBUS ADRS
11170     MLDA = .DSA_CNT;                            !LOAD DSA
11171     MLCS1 = write;                              !DO A WRITE FUNCTION
11172     TIME_OUT_LOOP;
11173
11174     if .UNS IS_SET                              !SEE IF XFERR CAUSED AN UNS ERROR
```

```
25922 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
25923 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD BLI.4 (112)
25924
25925 ;          11175      then
25926 ;          11176          begin                          !ERROR IF SET
25927 ;          11177          CMP_THRESHOLD;                 !COMPARE ERROR PRINT THRESHOLD
25928 ;          11178          ERRDF (107, ARR_DAT, DUMPER);
25929 ;          11179          PRINTB (TWO_FMT, WRD_35, PHR_4);
25930 ;          11180          PRINTB (FMT_9, .DSA_CNT);
25931 ;          11181
25932 ;          11182          if .CHIP_SIZ eql 64            !NEED TO KNOW CHIP SIZE TO PRINT ARRAY NUMBER
25933 ;          11183          then
25934 ;          11184              begin
25935 ;          11185              PRINTB (FMT_8, ((.DSA_CNT<11, 4>) + 1));    !64K MOS RAM
25936 ;          11186              end
25937 ;          11187          else
25938 ;          11188              begin
25939 ;          11189              PRINTB (FMT_8, ((.DSA_CNT<9, 4>) + 1));     !16K MOS RAM
25940 ;          11190              end
25941 ;          11191
25942 ;          11192          end;
25943 ;          11193
25944 ;          11194      ENDSUB;
25945 ;          11195      end;
25946 ;          11196
25947 ;          11197 ENDTST;
25951
25952                                      .SBTTL  $T60 TEST CODE SECTION
25956 112322  004167  072212        $T60: JSR     R1,$SAVE2                ;                              1112
25957 112326  005067  103044              CLR     P.CNT                    ;                              1112
25958 112332  016702  101456              MOV     LST.BLK,R2               ;                              1116
25959 112336  005001                      CLR     R1                       ; DSA.CNT
25960 112340  000544                      BR      7$
25961 112342  104402                1$:   TRAP    2                        ;                              1116
25962 112344  152777  000040  103072      BISB    #40,@ML.REG+40           ;                              1116
25963 112352  016700  103454              MOV     ML.DUT,R0
25964 112356  042700  177770              BIC     #177770,R0
25965 112362  142777  000007  103054      BICB    #7,@ML.REG+40
25966 112370  150077  103050              BISB    R0,@ML.REG+40
25967 112374  152777  000002  103122      BISB    #2,@ML.REG+120           ;                              1116
25968 112402  012777  177400  103004      MOV     #-400,@ML.REG+10         ;                              1116
25969 112410  012777  014022  103006      MOV     #10.BUF,@ML.REG+20       ;                              1116
25970 112416  010177  103012              MOV     R1,@ML.REG+30            ; DSA.CNT,*                    1117
25971 112422  012777  000061  102754      MOV     #61,@ML.REG              ;                              1117
25972 112430  105777  103020          2$: TSTB    @ML.REG+50
25973 112434  100375                      BPL     2$
25974 112436  032777  040000  103020      BIT     #40000,@ML.REG+60        ;                              1117
25975 112444  001476                      BEQ     5$
25976 112446  005267  102724              INC     P.CNT                    ;                              1117
```

```
25978                                   ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
25979                                   ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<
25980
25981 112452  026767  102720  102720            CMP     P.CNT,LIMIT
25982 112460  003073                            BGT     6$
25983 112462  104455                            TRAP    55                              ;                               1117
25984 112464  000153                            .WORD   153
25985 112466  013012                            .WORD   ARR.DAT
25986 112470  026302                            .WORD   DUMPER
25987 112472  012746  011676                    MOV     #PHR.4,-(SP)                    ;                               1117
25988 112476  012746  011110                    MOV     #WRD.35,-(SP)
25989 112502  012746  010246                    MOV     #TWO.FMT,-(SP)
25990 112506  012746  000003                    MOV     #3,-(SP)
25991 112512  010600                            MOV     SP,R0                           ; SP,*
25992 112514  104414                            TRAP    14
25993 112516  010116                            MOV     R1,(SP)                         ; DSA.CNT,*                     1118
25994 112520  012746  007064                    MOV     #FMT.9,-(SP)
25995 112524  012746  000002                    MOV     #2,-(SP)
25996 112530  010600                            MOV     SP,R0                           ; SP,*
25997 112532  104414                            TRAP    14
25998 112534  026727  101252  000100            CMP     CHIP.SIZ,#100                   ;                               1118
25999 112542  001020                            BNE     3$
26000 112544  010100                            MOV     R1,R0                           ; DSA.CNT,*                     1118
26001 112546  006200                            ASR     R0
26002 112550  006200                            ASR     R0
26003 112552  006200                            ASR     R0
26004 112554  000300                            SWAB    R0
26005 112556  042700  177760                    BIC     #177760,R0
26006 112562  010046                            MOV     R0,-(SP)
26007 112564  005216                            INC     (SP)
26008 112566  012746  007030                    MOV     #FMT.8,-(SP)
26009 112572  012746  000002                    MOV     #2,-(SP)
26010 112576  010600                            MOV     SP,R0                           ; SP,*
26011 112600  104414                            TRAP    14
26012 112602  000415                            BR      4$                              ;                               1118
26013 112604  010100                    3$:     MOV     R1,R0                           ; DSA.CNT,*                     1118
26014 112606  006200                            ASR     R0
26015 112610  000300                            SWAB    R0
26016 112612  042700  177760                    BIC     #177760,R0
26017 112616  010046                            MOV     R0,-(SP)
26018 112620  005216                            INC     (SP)
26019 112622  012746  007030                    MOV     #FMT.8,-(SP)
26020 112626  012746  000002                    MOV     #2,-(SP)
26021 112632  010600                            MOV     SP,R0                           ; SP,*
26022 112634  104414                            TRAP    14
26023 112636  062706  000022            4$:     ADD     #22,SP                          ;                               1117
26024 112642  104467                    5$:     TRAP    67                              ;                               1119
26025 112644  006000                            ROR     R0
26026 112646  103635                            BLO     1$
26027 112650  005201                    6$:     INC     R1                              ; DSA.CNT                       1116
26028 112652  020102                    7$:     CMP     R1,R2                           ; DSA.CNT,*
26029 112654  101632                            BLOS    1$
26030 112656  000207                            RTS     PC                              ;                               1112
26031
26032

                                        ; Routine Size:  111 words
```

```
                                ;ML4AD                                          29-Mar-1982 16:23:04    TOPS
                                ;               TEST CODE SECTION               29-Mar-1982 16:21:3?    PA:<
                                ;
                                ; Maximum stack depth per invocation:  12 words
                                ;
                                ;
                                ;
                                        .SBTTL  T60 TEST CODE SECTION
112660                          T60::
112660  004767  177436          1$:     JSR     PC,$T60                 ;       1119
112664  104466                          TRAP    66
112666  006000                          ROR     R0
112670  103773                          BLO     1$
112672  000207                          RTS     PC
                                ;
                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words
                                ;
                                ;
;       11198 !<BLF/PAGE>
```

```
26069 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
26070 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (113)
26071
26072 ;          11199 !
26073 ;          11200 !
26074 ;          11201 BGNTST;                                  !BEGIN THE TEST
26075 ;          11202
26076 ;          11203 !++
26077 ;          11204 !  TEST NUMBER: TST 61
26078 ;          11205 !
26079 ;          11206 !  TEST NAME:   BATTERY BACK-UP TEST
26080 ;          11207 !
26081 ;          11208 !  TEST DESCRIPTION:
26082 ;          11209 !       THIS TEST IS DESIGNED SPECIFICALLY FOR THE
26083 ;          11210 !       MARLBORO MFG'ING VERIFY STATION.  ITS PURPOSE
26084 ;          11211 !       IS TO TEST THE ML-11 SYSTEM FOR BATTERY BACK-UP DURING
26085 ;          11212 !       POWER LOSS AND NO ECC INITIALIZE DURING POWER
26086 ;          11213 !       UP WITH GOOD BATTERY BACK-UP.
26087 ;          11214 !
26088 ;          11215 !--
26089 ;          11216
26090 ;          11217 local
26091 ;          11218     DSA_CNT,                             !COUNTS THE NUMBER OF BLOCKS TRANSFERED
26092 ;          11219     SIZE,                                !STORES THE SIZE OF MASS BUS TRANSFERS
26093 ;          11220     DST,                                 !STORES THE DESTINATION ADRS OF MASS BUS TRANSFERS
26094 ;          11221     SRC,                                 !STORES THE SOURCE ADRS OF MASS BUS TRANSFERS
26095 ;          11222     INIT_ERR,                            !COUNTS THE # OF WORDS FOUND TO HAVE ZEROES DATA
26096 ;          11223     BB_ERR;                              !COUNTS THE # OF WORDS FOUND TO HAVE NONE ONES DATA BUT NOT 0'S DATA
26097 ;          11224
26098 ;          11225 !+
26099 ;          11226 ! PERFORM THIS TEST ONLY IF THE OPERATOR HAS
26100 ;          11227 ! ENABLED MANUAL INTERVENTION TEST VIA THE S/W
26101 ;          11228 ! QUESTIONS.
26102 ;          11229 !-
26103 ;          11230
26104 ;          11231 if .INTERVEN                              !DO THIS TEST IF INTERVEN IS SET
26105 ;          11232 then
26106 ;          11233     begin
26107 ;          11234 !
26108 ;          11235 !
26109 ;          11236 ! PRINT WHICH TEST NUMBER THIS
26110 ;          11237 !
26111 ;          11238 !
26112 ;          11239     PRINTB (ONE_FMT, T_61);
26113 ;          11240 !
26114 ;          11241 !
26115 ;          11242 ! TO START THE TEST OFF WRITE THE ENTIRE ML-11 MEMORY
26116 ;          11243 ! WITH ALL ONES DATA.  THIS WILL BE THE BACK GROUND FOR
26117 ;          11244 ! THE ECC INITIALIZE PORTION OF THIS TEST.  THIS ALL ONES
26118 ;          11245 ! DATA PATTERN SHOULD STILL BE IN MEMORY AFTER THE POWER
26119 ;          11246 ! IS TURNED BACK ON WITH GOOD BATTERY BACK-UP.
26120 ;          11247 !
26121 ;          11248 !
26122 ;          11249     CLR_MBUS;                            !CLEAR THE MASS BUS
26123 ;          11250     IO_BUF = ONES;                       !LOAD THE FIRST IO BUF WORD WITH ONES DATA
```

```
26125 ;ML4AD
26126 ;                TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
26127 ;                                                     29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (113)
26128 ;          11251     DSA_CNT = -1;                    !RESET THE DSA COUNTER
26129 ;          11252
26130 ;          11253     do                               !LOAD ALL OF MEMORY WITH ONE'S DATA
26131 ;          11254         begin
26132 ;          11255         BREAK;                       !LOOK FOR CONTROL C
26133 ;          11256         DSA_CNT = .DSA_CNT + 1;      !UP THE DSA COUNTER
26134 ;          11257         WRT_TRANSFER (SIZE = -256, DST = .DSA_CNT, SRC = IO_BUF);      !DO THE WRITE XFER
26135 ;          11258         end
26136 ;          11259 !
26137 ;          11260 !VERSION CZMLAD CHANGED TEST TO UNSIGNED TEST
26138 ;          11261 !
26139 ;          11262     until .DSA_CNT eqlU .LST_BLK;    !REPEAT WRITING UNTIL THE LAST BLOCK IS DONE
26140 ;          11263
26141 ;          11264 !
26142 ;          11265 !
26143 ;          11266 ! SET THE VOLUMN VALID BIT 'VV'
26144 ;          11267 ! AND TELL THE OPERATOR TO TURN THE BOX
26145 ;          11268 ! POWER OFF WITH GOOD BATTERY BACK-UP.
26146 ;          11269 ! THE BACK GROUND PATTERN OF ALL ONES SHOULD
26147 ;          11270 ! BE MAINTAINED BY BATTERY BACK-UP AND ECC INIT
26148 ;          11271 ! ON POWER UP SHOULD NOT CLEAR THE PATTERN OUT.
26149 ;          11272 !
26150 ;          11273 !
26151 ;          11274     CLR_MBUS;                        !CLEAR THE MASS BUS
26152 ;          11275     MLCS1 = RD_IN_PRE;               !SET THE VV BIT via A RD_IN_PRE FUNC
26153 ;          11276     PRINTB (ONE_FMT, PWR_OFF);       !TELL OPERATOR TO SHUT POWER OFF
26154 ;          11277
26155 ;          11278     do                               !DO NOTHING
26156 ;          11279         begin
26157 ;          11280         BREAK;                       !LOOK FOR CONTROL C
26158 ;          11281         end
26159 ;          11282     until not (.MOL);                !UNTIL THE DRIVE POWER IS OFF
26160 ;          11283
26161 ;          11284     DELAY (TWO_TH_US);               !DELAY FOR BATTERY BACK-UP MAINTAIN MEMORY
26162 ;          11285 !
26163 ;          11286 !
26164 ;          11287 ! NOW TELL THE OPERATOR TO POWER UP THE BOX
26165 ;          11288 ! WITH GOOD BATTERY BACK-UP.  THEN TEST THE
26166 ;          11289 ! VV BIT TO BE STILL SET AND THE BACK GROUND
26167 ;          11290 ! PATTERN OF ALL ONES TO STILL BE IN MEMORY.
26168 ;          11291 !
26169 ;          11292 !
26170 ;          11293     PRINTB (ONE_FMT, PUP_BB);        !TELL OPERATOR TO TURN BOX POWER BACK ON
26171 ;          11294
26172 ;          11295     do                               !DO NOTHING
26173 ;          11296         begin
26174 ;          11297         BREAK;                       !LOOK FOR CONTROL C
26175 ;          11298         end
26176 ;          11299     until .DRY;                      !UNTIL THE DRIVE IS READY
26177 ;          11300
26178 ;          11301     DELAY (10);                      !WAIT FOR THE DRIVE TO SETILE
26179 ;          11302
```

```
26181 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
26182 ;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (113)
26183 ;
26184 ;          11303      if not (.VV)                        !DID RECOVERING UP WITH GOOD BB CLEAR THE VV BIT
26185 ;          11304      then
26186 ;          11305          begin                           !THE VV BIT GOT CLEARED
26187 ;          11306          ERRDF (154, ASYNC, DUMPER);     !REPORT THE ERROR TO THE OPERATOR
26188 ;          11307          PRINTB (ONE_FMT, BB_VV_ERR);    !TELL WHAT THE ERROR IS
26189 ;          11308          end;
26190 ;          11309
26191 ;          11310      INIT_ERR = ZERO;                    !CLEAR THE INIT COUNTER
26192 ;          11311      BB_ERR = ZERO;                      !CLEAR THE BATTERY BACK-UP COUNTER
26193 ;          11312      DSA_CNT = -1;                       !RESET THE DSA COUNTER
26194 ;          11313
26195 ;          11314      do                                  !WRITE CHECK MEMORY AND TEST FOR BAT BACK-UP AND/OR NO ECC INIT
26196 ;          11315          begin
26197 ;          11316          BREAK;                          !LOOK FOR CONTROL C
26198 ;          11317          DSA_CNT = .DSA_CNT + 1;         !UP THE DSA COUNTEER
26199 ;          11318          WRT_CHK_TRANSFER (SIZE = -256, DST = .DSA_CNT, SRC = IO_BUF);   !DO THE WRITE CHECK XFER
26200 ;          11319
26201 ;          11320          if .WCE                         !IS THIS BLOCK IN ERROR
26202 ;          11321          then
26203 ;          11322              begin                       !THIS BLOCK IS IN ERROR
26204 ;          11323
26205 ;          11324              if .MLDB eql ZERO           !SEE WHAT KIND OF AN ERROR IT IS
26206 ;          11325              then
26207 ;          11326                  INIT_ERR = .INIT_ERR + 1    !IT'S A INITIALIZE ERROR
26208 ;          11327              else
26209 ;          11328                  BB_ERR = .BB_ERR + 1;       !IT'S A BAT BACK-UP ERROR
26210 ;          11329
26211 ;          11330              end;
26212 ;          11331
26213 ;          11332          end
26214 ;          11333 !
26215 ;          11334 !VERSION CZMLAD CHANGED TEST TO UNSIGNED TEST
26216 ;          11335 !
26217 ;          11336      until .DSA_CNT eqlU .LST_BLK;       !REPEAT WRITE CHECKING UNTIL THE LAST BLOCK
26218 ;          11337
26219 ;          11338 !
26220 ;          11339 ! IF THERE ARE MORE THAN 127 SECTORS IN
26221 ;          11340 ! ERROR WITH ALL ZEORES DATA THEN THE
26222 ;          11341 ! PROBLEM IS PROBIBLY IN
26223 ;          11342 ! THE ECC INITIALIZE LOGIC AND NOT IN THE
26224 ;          11343 ! BATTERY BACK-UP OF MEMORY.
26225 ;          11344 !
26226 ;          11345 ! IF INIT_ERR IS NOT > 127 AND THERE ARE
26227 ;          11346 ! > 100 WORDS READ WITH DATA OTHER THAN
26228 ;          11347 ! ALL ONES OR ALL ZEROES THEN THE PROBLEM
26229 ;          11348 ! IS PROBIBLY CAUSED BY THE BATTERY BACK-UP
26230 ;          11349 ! LOGIC NOT MAINTAINING THE MEMORY DURING
26231 ;          11350 ! POWER FAILURE.
26232 ;          11351 !
26233 ;          11352
26234 ;          11353      if .INIT_ERR gtr 127                !IS THE ECC INIT LOGIC IN ERROR
26235 ;          11354      then
```

```
 26237 ;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
 26238 ;                 TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (113)
 26239
 26240 ;          11355          begin                           !THE ECC INIT LOGIC IS IN ERROR
 26241 ;          11356          ERRDF (155, ASYNC, DUMPER);     !REPORT THE ERROR TO THE OPERATOR
 26242 ;          11357          PRINTB (ONE_FMT, BB_INIT_ERR);  !TELL WHAT THE ERROR IS
 26243 ;          11358          end
 26244 ;          11359      else
 26245 ;          11360          begin                           !THE ECC INIT LOGIC IS NOT IN ERROR
 26246 ;          11361
 26247 ;          11362          if .BB_ERR gtr 100              !IS THE BATTERY BACK-UP LOGIC IN ERROR
 26248 ;          11363          then
 26249 ;          11364              begin                       !THE BATTERY BACK-UP LOGIC IS IN ERROR
 26250 ;          11365              ERRDF (156, ASYNC, DUMPER); !REPORT THE ERROR TO THE OPERATOR
 26251 ;          11366              PRINTB (ONE_FMT, BB_BB_ERR);!TELL WHAT THE ERROR IS
 26252 ;          11367              end;
 26253 ;          11368
 26254 ;          11369          end;
 26255 ;          11370
 26256 ;          11371      end;
 26257 ;          11372
 26258 ;          11373 ENDTST;                                   !END THE TEST
 26262
 26263                                          .SBTTL  $T61 TEST CODE SECTION
 26267 112674  004167  071712          $T61:   JSR     R1,$SAVE5                       ;                       1119
 26268 112700  024646                          CMP     -(SP),-(SP)
 26269 112702  032767  000001  067500          BIT     #1,INTERVEN                     ;                       1123
 26270 112710  001002                          BNE     1$
 26271 112712  000167  000600                  JMP     20$
 26272 112716  012746  005012          1$:     MOV     #T.61,-(SP)                     ;                       1123
 26273 112722  012746  010240                  MOV     #ONE.FMT,-(SP)
 26274 112726  012746  000002                  MOV     #2,-(SP)
 26275 112732  010600                          MOV     SP,R0                           ; SP,*
 26276 112734  104414                          TRAP    14
 26277 112736  152777  000040  102500          BISB    #40,@ML.REG+40
 26278 112744  016705  103062                  MOV     ML.DUT,R5
 26279 112750  042705  177770                  BIC     #177770,R5
 26280 112754  142777  000007  102462          BICB    #7,@ML.REG+40
 26281 112762  150577  102456                  BISB    R5,@ML.REG+40
 26282 112766  012767  177777  101026          MOV     #-1,IO.BUF                      ;                       1125
 26283 112774  012702  177777                  MOV     #-1,R2                          ; *,DSA.CNT             1125
 26284 113000  104422                  2$:     TRAP    22                              ;                       1125
 26285 113002  005202                          INC     R2                              ; DSA.CNT               1125
 26286 113004  012705  177400                  MOV     #-400,R5                        ; *,SIZE                1125
 26287 113010  010546                          MOV     R5,-(SP)                        ; SIZE,*
 26288 113012  010266  000010                  MOV     R2,10(SP)                       ; DSA.CNT,DST
 26289 113016  010246                          MOV     R2,-(SP)                        ; DSA.CNT,*
 26290 113020  012701  014022                  MOV     #IO.BUF,R1                      ; *,SRC
 26291 113024  010146                          MOV     R1,-(SP)                        ; SRC,*
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
                                        ;
26296 113026 004767 114374              JSR     PC,WRT.TRANSFER
26297 113032 062706 000006              ADD     #6,SP
26298 113036 020267 100752              CMP     R2,LST.BLK              ; DSA.CNT,*                             1125
26299 113042 001356                     BNE     2$                                                              1126
26300 113044 152777 000040 102372       BISB    #40,@ML.REG+40
26301 113052 016704 102754              MOV     ML.DUT,R4
26302 113056 042704 177770              BIC     #177770,R4
26303 113062 142777 000007 102354       BICB    #7,@ML.REG+40
26304 113070 150477 102350              BISB    R4,@ML.REG+40
26305 113074 012777 000021 102302       MOV     #21,@ML.REG             ;                               1127
26306 113102 012716 005112              MOV     #PWR.OFF,(SP)           ;                               1127
26307 113106 012746 010240              MOV     #ONE.FMT,-(SP)
26308 113112 012746 000002              MOV     #2,-(SP)
26309 113116 010600                     MOV     SP,R0                   ; SP,*
26310 113120 104414                     TRAP    14
26311 113122 104422             3$:     TRAP    22                      ;                               1127
26312 113124 032777 010000 102322       BIT     #10000,@ML.REG+50       ;                               1128
26313 113132 001373                     BNE     3$
26314 113134 012703 003720              MOV     #3720,R3                ; *,$$TMP2                      1128
26315 113140 001411             4$:     BEQ     7$
26316 113142 016704 066750              MOV     L$DLY,R4                ; *,$$TMP1
26317 113146 001404                     BEQ     6$
26318 113150 005066 000014      5$:     CLR     14(SP)                  ; $$TMP
26319 113154 005304                     DEC     R4                      ; $$TMP1
26320 113156 001374                     BNE     5$
26321 113160 005303             6$:     DEC     R3                      ; $$TMP2
26322 113162 000766                     BR      4$
26323 113164 012716 006060      7$:     MOV     #PUP.BB,(SP)            ;                               1129
26324 113170 012746 010240              MOV     #ONE.FMT,-(SP)
26325 113174 012746 000002              MOV     #2,-(SP)
26326 113200 010600                     MOV     SP,R0                   ; SP,*
26327 113202 104414                     TRAP    14
26328 113204 104422             8$:     TRAP    22                      ;                               1129
26329 113206 105777 102242              TSTB    @ML.REG+50              ;                               1129
26330 113212 100374                     BPL     8$
26331 113214 012703 000012              MOV     #12,R3                  ; *,$$TMP2                      1130
26332 113220 001411             9$:     BEQ     12$
26333 113222 016704 066670              MOV     L$DLY,R4                ; *,$$TMP1
26334 113226 001404                     BEQ     11$
26335 113230 005066 000020      10$:    CLR     20(SP)                  ; $$TMP
26336 113234 005304                     DEC     R4                      ; $$TMP1
26337 113236 001374                     BNE     10$
26338 113240 005303             11$:    DEC     R3                      ; $$TMP2
26339 113242 000766                     BR      9$
26340 113244 132777 000100 102202 12$:  BITB    #100,@ML.REG+50         ;                               1130
26341 113252 001016                     BNE     13$
26342 113254 104455                     TRAP    55                      ;                               1130
26343 113256 000232                     .WORD   232
26344 113260 012706                     .WORD   ASYNC
26345 113262 026302                     .WORD   DUMPER
26346 113264 012746 006150              MOV     #BB.VV.ERR,-(SP)        ;                               1130
26347 113270 012746 010240              MOV     #ONE.FMT,-(SP)
```

```
                                        ;ML4AD                                                          29-Mar-1982 16:23:04    TOPS
                                        ;               TEST CODE SECTION                               29-Mar-1982 16:21:03    PA:<
                                        
113274  012746  000002                        MOV     #2,-(SP)
113300  010600                                MOV     SP,R0           ; SP,*
113302  104414                                TRAP    14
113304  062706  000006                        ADD     #6,SP           ;                                                       1130
113310  005003                        13$:    CLR     R3              ; INIT.ERR                                              1131
113312  005004                                CLR     R4              ; BB.ERR                                                1131
113314  012702  177777                        MOV     #-1,R2          ; *,DSA.CNT                                             1131
113320  104422                        14$:    TRAP    22              ;                                                       1131
113322  005202                                INC     R2              ; DSA.CNT                                               1131
113324  012705  177400                        MOV     #-400,R5        ; *,SIZE                                                1131
113330  010546                                MOV     R5,-(SP)        ; SIZE,*
113332  010266  000020                        MOV     R2,20(SP)       ; DSA.CNT,DST
113336  010246                                MOV     R2,-(SP)        ; DSA.CNT,*
113340  012701  014022                        MOV     #IO.BUF,R1      ; *,SRC
113344  010146                                MOV     R1,-(SP)        ; SRC,*
113346  004767  113604                        JSR     PC,WRT.CHK.TRANSFE
113352  032777  040000  102064                BIT     #40000,@ML.REG+40       ;                                               1132
113360  001406                                BEQ     16$
113362  005777  102126                        TST     @ML.REG+110     ;                                                       1132
113366  001002                                BNE     15$
113370  005203                                INC     R3              ; INIT.ERR                                              1132
113372  000401                                BR      16$             ;                                                       1132
113374  005204                        15$:    INC     R4              ; BB.ERR                                                1132
113376  062706  000006                16$:    ADD     #6,SP           ;                                                       1131
113402  020267  100406                        CMP     R2,LST.BLK      ; DSA.CNT,*                                             1133
113406  001344                                BNE     14$
113410  020327  000177                        CMP     R3,#177         ; INIT.ERR,*                                            1135
113414  003415                                BLE     17$
113416  104455                                TRAP    55              ;                                                       1135
113420  000233                                .WORD   233
113422  012706                                .WORD   ASYNC
113424  026302                                .WORD   DUMPER
113426  012746  006242                        MOV     #BB.INIT.ERR,-(SP)      ;                                               1135
113432  012746  010240                        MOV     #ONE.FMT,-(SP)
113436  012746  000002                        MOV     #2,-(SP)
113442  010600                                MOV     SP,R0           ; SP,*
113444  104414                                TRAP    14
113446  000417                                BR      18$             ;                                                       1135
113450  020427  000144                17$:    CMP     R4,#144         ; BB.ERR,*                                              1136
113454  003416                                BLE     19$
113456  104455                                TRAP    55              ;                                                       1136
113460  000234                                .WORD   234
113462  012706                                .WORD   ASYNC
113464  026302                                .WORD   DUMPER
113466  012746  006334                        MOV     #BB.BB.ERR,-(SP)        ;                                               1136
113472  012746  010240                        MOV     #ONE.FMT,-(SP)
113476  012746  000002                        MOV     #2,-(SP)
113502  010600                                MOV     SP,R0           ; SP,*
113504  104414                                TRAP    14
113506  062706  000006                18$:    ADD     #6,SP           ;                                                       1136
113512  062706  000016                19$:    ADD     #16,SP          ;                                                       1123
113516  022626                        20$:    CMP     (SP)+,(SP)+     ;                                                       1119
```

```
                                ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
                                ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<

113520  000207                          RTS     PC

                                ; Routine Size:  203 words
                                ; Maximum stack depth per invocation:  18 words


                                        .SBTTL  T61 TEST CODE SECTION
113522                          T61::
113522  004767  177146          1$:     JSR     PC,$T61                         ;       1137
113526  104466                          TRAP    66
113530  006000                          ROR     R0
113532  103773                          BLO     1$
113534  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


;       11374 !<BLF/PAGE>
```

```
;ML4AD                                                    29-Mar-1982 16:23:04    TOPS-20 Bliss-16 V2(212)
;                TEST CODE SECTION                        29-Mar-1982 16:21:03    PA:<NEALE>ML4AD.BLI.4 (114)

;          11375 !
;          11376 ! CLEAN UP SECTION
;          11377 !
;          11378 !       THE CLEAN UP SECTION IS ENTERED WHEN:
;          11379 !       1.  A NORMAL END OF PASS IS DONE
;          11380 !           ie. WHEN ONE COMPLETE PASS OF
;          11381 !               THE DIAG IS DONE.
;          11382 !
;          11383 !       2.  WHEN A DRIVE ERROR IS DECTECTED AND
;          11384 !           AND THE UNIT IS DROPPED AND THE DOCLN
;          11385 !           MACRO IS CALLED.
;          11386 !
;          11387 !       3.  WHEN A ^C IS TYPE AT THE CONSOLE TERMINAL
;          11388 !           DURING EXECUTION OF A TEST.
;          11389 !
;          11390 !       IN ANY EVENT OF ENTERING THIS SECTION THE
;          11391 !       ML11 CONTROL BOARD TESTER'S POWER SUPPLIES
;          11392 !       ARE POWERED DOWN via THE ROUTINE CALL: ''
;          11393
;          11394 !++
;          11395 BGNCLN;                                  !BEGIN THE CLEAN UP CODE
;          11396 CLR = ONE;                              !CLEAR THE MASS BUS
;          11397 return;                                 !RETURN TO DRS>
;          11398 ENDCLN;                                 !END THE CLEAN UP CODE
26473
26474                                   .SBTTL  LCLEAN TEST CODE SECTION
26478 113536  152777  000040  101700  LCLEAN: BISB    #40,@ML.REG+40          ;                1139
26479 113544  000207                          RTS     PC                      ;                1137
26480
26481                                   ; Routine Size:  4 words
26482                                   ; Maximum stack depth per invocation:  0 words
26487
26488
26492
26493                                   .SBTTL  LSCLEAN TEST CODE SECTION
26497 113546                            LSCLEAN::
```

```
                                ;ML4AD                                                  29-Mar-1982 16:23:04    TOPS
                                ;               TEST CODE SECTION                       29-Mar-1982 16:21:03    PA:<

113546  004767  177764                 JSR     PC,LCLEAN                       ;                               1139
113552  104412                         TRAP    12
113554  000207                         RTS     PC

                                ; Routine Size:  4 words
                                ; Maximum stack depth per invocation:  0 words

;          11399 LASTAD;                                        !DEFINE THE DIAG'S LAST ADDRESS
;          11400 BGNSETUP (0);                                  !SET UP ZERO P-TABLES
;          11401 ENDSETUP;                                      !END THE SETUP

113556  113562                 BL$LAS::.WORD   T$FREE
113560  000000                         .WORD   <<T$FREE-<BL$LAS+4>>/2>
113562  000000                 T$FREE::.WORD   0

        113562                 L$LAST==            BL$LAS+4
        000000                 T$PTHV==            0

                                        .SBTTL  $END.LINK TEST CODE SECTION
113564                          $END.LINK::
113564  000207                         RTS     PC                              ;                               1139

                                ; Routine Size:  1 word
                                ; Maximum stack depth per invocation:  0 words

;          11402 end
;          11403
;          11404 eludom
```

SEQ 0545

```
26554                           ;ML4AD
26555                           ;                TEST CODE SECTION                    29-Mar-1982 16:23:04    TOPS
26556                                                                                 29-Mar-1982 16:21:03    PA:<
26557
26558                           ; OTS external references
26559                                   .GLOBL  BL$GT2, $SAVE5, $SAVE4, $SAVE3
26560                                   .GLOBL  $SAVE2, BL$PU2, BL$PU1, BL$GT1
26561                                   .GLOBL  BL$DIV, BL$MOD, BL$MUL
26562
26563
26564
26565
26566
26567                           ; Size:          15786 code + 2346 data words
26568                           ; Run Time:      02:46.0
26569                           ; Elapsed Time: 03:20.7
26570                           ; Memory Used:   124 pages
26571                           ; Compilation Complete
26572
26573        000001                     .END
```

SYMBOL TABLE

SEQ 0542

```
A)R    = 000020 G      C$CLCK= 000062      D2.TEM= 013376      FNC.19= 012426      HELP   = 000000
ARR.DA= 013012         C$CLEA= 000012      EF.CON= 000036 G    FNC.2  = 012164      HOE    = 100000 G
ARR.IN  014004         C$CLOS= 000035      EF.NEW= 000035 G    FNC.21= 012440      HW.OR.   013402
ARR.16  014016         C$CLP1= 000006      EF.PWR= 000034 G    FNC.22= 012452      IBE    = 010000 G
ASSEMB= 000010         C$CVEC= 000036      EF.RES= 000037 G    FNC.23= 012464      IDU    = 000040 G
ASYNC = 012706         C$DCLN= 000044      EF.STA= 000040 G    FNC.3  = 012172      IER    = 020000 G
A.CAL   015362         C$DODU= 000051      EIG.FM= 010364      FNC.4  = 012200      INTER = 013114
A.GEN   015370         C$DRPT= 000024      ELV.FM= 010466      FNC.5  = 012216      INTERV  002410 G
BB.BB.= 006334         C$DU   = 000053     ERRBLK  002160 G    FNC.6  = 012226      IO.BUF  014022
BB.INI= 006242         C$EDIT= 000003      ERRMSG  002156 G    FNC.7  = 012234      ISR    = 000100 G
BB.VV.= 006150         C$ERDF= 000055      ERRNBR  002154 G    FNC.8  = 012244      IXE    = 004000 G
BIT0   = 000001 G      C$ERHR= 000056      ERRTHR  002402 G    FNC.9  = 012260      I$AU   = 000041
BIT00  = 000001 G      C$ERRO= 000060      ERRTYP  002152 G    FORCE.   020032      I$AUTO= 000041
BIT01  = 000002 G      C$ERSF= 000054      ERR.CH  017442      FOR.EC   020210      I$CLN  = 000041
BIT02  = 000004 G      C$ERSO= 000057      EVL    = 000004 G   FOR.FM= 010270      I$DU   = 000041
BIT03  = 000010 G      C$ESCA= 000010      ESEND  = 002100      F$AU   = 000015      I$HRD  = 000041
BIT04  = 000020 G      C$ESEG= 000005      ESLOAD= 000035      F$AUTO= 000020      I$INIT= 000041
BIT05  = 000040 G      C$ESUB= 000003      E2.TEM= 013400      F$BGN  = 000040      I$MOD  = 000041
BIT06  = 000100 G      C$ETST= 000001      FIND.C  017660      F$CLEA= 000007      I$MSG  = 000041
BIT07  = 000200 G      C$EXIT= 000032      FIRST.   020262     F$DU   = 000016      I$PROT= 000040
BIT08  = 000400 G      C$GETB= 000026      FIV.FM= 010304      F$END  = 000041      I$PTAB= 000041
BIT09  = 001000 G      C$GETW= 000027      FMT.1  = 006442     F$HARD= 000004      I$PWR  = 000041
BIT1   = 000002 G      C$GMAN= 000043      FMT.10= 007116      F$HW   = 000013      I$RPT  = 000041
BIT10  = 002000 G      C$GPHR= 000042      FMT.11= 007170      F$INIT= 000006      I$SEG  = 000041
BIT11  = 004000 G      C$GPLO= 000030      FMT.12= 007224      F$JMP  = 000050      I$SETU= 000041
BIT12  = 010000 G      C$GPRI= 000040      FMT.13= 007254      F$MOD  = 000000      I$SFT  = 000041
BIT13  = 020000 G      C$INIT= 000011      FMT.14= 007330      F$MSG  = 000011      I$SRV  = 000041
BIT14  = 040000 G      C$INLP= 000020      FMT.15= 007370      F$PROT= 000021      I$SUB  = 000041
BIT15  = 100000 G      C$MANI= 000050      FMT.16= 007436      F$PWR  = 000017      I$TST  = 000041
BIT2   = 000004 G      C$MEM  = 000031     FMT.17= 007526      F$RPT  = 000012      J$JMP  = 000167
BIT3   = 000010 G      C$MSG  = 000023     FMT.18= 007562      F$SEG  = 000003      LAST.B   020344
BIT4   = 000020 G      C$OPEN= 000034      FMT.19= 007632      F$SOFT= 000005      LAU    004704
BIT5   = 000040 G      C$PNTB= 000014      FMT.2  = 006506     F$SRV  = 000010      LAUTO   004660
BIT6   = 000100 G      C$PNTF= 000017      FMT.20= 007702      F$SUB  = 000002      LCLEAN  113536
BIT7   = 000200 G      C$PNTS= 000016      FMT.21= 007754      F$SW   = 000014      LDU    004672
BIT8   = 000400 G      C$PNTX= 000015      FMT.22= 010026      F$TEST= 000001      LD.LNG  021354
BIT9   = 001000 G      C$QIO  = 000377     FMT.23= 010072      GD.BLK   020312      LIMIT   015400
BL$DIV  004446 G       C$RDBU= 000007      FMT.24= 010142      GOOD.B   014006      LINIT   027650
BL$GT1  003512 G       C$REFG= 000047      FMT.25= 010166      G$CNTO= 000200      LOAD.S  016034
BL$GT2  003634 G       C$RESE= 000033      FMT.26= 010206      G$DELM= 000372      LOE    = 040000 G
BL$LAS  113556 G       C$REVI= 000003      FMT.3  = 006574     G$DISP= 000003      LOT    = 000010 G
BL$MOD  004460 G       C$RFLA= 000021      FMT.4  = 006620     G$EXCP= 000400      LRPT    004646
BL$MUL  004222 G       C$RPT  = 000025     FMT.5  = 006650     G$HILI= 000002      LST.AR  014020
BL$PU1  003776 G       C$SEFG= 000046      FMT.6  = 006752     G$LOLI= 000001      LST.BL  014014
BL$PU2  004072 G       C$SPRI= 000041      FMT.7  = 007002     G$NO   = 000000      LST.DU  015356
BL$SH?  004472 G       C$SVEC= 000037      FMT.8  = 007030     G$OFFS= 000400      L$ACP   002110 G
BOE    = 000400 G      C$TPRI= 000013      FMT.9  = 007064     G$OFSI= 000376      L$APT   002036 G
B.CAL   015364         DATA.L= 013154      FNC.1  = 012150     G$PRMA= 000001      L$AU    004706 G
B.GEN   015372         DAT.DM  016412      FNC.10= 012272      G$PRMD= 000002      L$AUT   002070 G
CAL.CR  017042         DFPTBL  002360 G    FNC.11= 012302      G$PRML= 000000      L$AUTO  004662 G
CHIP.S  014012         DIAGMC= 000000      FNC.12= 012322      G$RADA= 000140      L$CCP   002106 G
C$AU   = 000052        DIVMOD  004264      FNC.13= 012334      G$RADB= 000000      L$CLEA  113546 G
C$AUTO= 000061         DRIVE.  015354      FNC.14= 012344      G$RADD= 000040      L$CO    002032 G
C$BRK  = 000022        DROP.C  015402      FNC.15= 012360      G$RADL= 000120      L$DEPO  002011 G
C$BSEG= 000004         DT.1    015764      FNC.16= 012372      G$RADO= 000020      L$DESC  002130 G
C$BSUB= 000002         DUMPER  026302 G    FNC.17= 012404      G$XFER= 000004      L$DESP  002076 G
C$CEFG= 000045         D1.TEM= 013374      FNC.18= 012414      G$YES  = 000010      L$DEVP  002060 G
```

SEQ

L$DISP	002164 G	MSGH7	003052	P.AAC	005112	P.ACH	010566	P.AEM	011454
L$DLY	002116 G	MSGS1	003160	P.AAD	005202	P.ACI	010602	P.AEN	011462
L$DTP	002040 G	MSGS2	003224	P.AAE	005272	P.ACJ	010616	P.AEO	011470
L$DTYP	002034 G	MSGS3	003270	P.AAF	005340	P.ACK	010630	P.AEP	011474
L$DU	004674 G	MSGS4	003334	P.AAG	005414	P.ACL	010640	P.AEQ	011502
L$DUT	002072 G	MSGS5	003400	P.AAH	005470	P.ACM	010650	P.AER	011514
L$DVTY	002122 G	MSDUMP	026312	P.AAI	005570	P.ACN	010660	P.AES	011524
L$EF	002052 G	NIB.SA	013374	P.AAJ	005652	P.ACO	010664	P.AET	011532
L$ENVI	002044 G	NIN.FM=	010410	P.AAK	005722	P.ACP	010676	P.AEU	011540
L$ERRT	002152 G	NO.INI=	005470	P.AAL	005764	P.ACQ	010704	P.AEV	011552
L$ETP	002102 G	ONEFIL=	000001	P.AAM	006060	P.ACR	010712	P.AEW	011562
L$EXP1	002046 G	ONEPAS	002406 G	P.AAN	006150	P.ACS	010724	P.AEX	011570
L$EXP4	002064 G	ONE.FM=	010240	P.AAO	006242	P.ACT	010730	P.AEY	011610
L$EXP5	002066 G	OP.NUM	014002	P.AAP	006334	P.ACU	010736	P.AEZ	011626
L$HARD	002414 G	O$APTS=	000001	P.AAQ	006442	P.ACV	010744	P.AFA	011644
L$HIME	002120 G	O$AU =	000001	P.AAR	006506	P.ACW	010752	P.AFB	011676
L$HPCP	002016 G	O$BGNR=	000001	P.AAS	006574	P.ACX	010766	P.AFC	011714
L$HPTP	002022 G	O$BGNS=	000001	P.AAT	006620	P.ACY	010774	P.AFD	011726
L$HW	002360 G	O$DU =	000001	P.AAU	006650	P.ACZ	011002	P.AFE	011740
L$ICP	002104 G	O$ERRT=	000001	P.AAV	006752	P.ADA	011016	P.AFF	011762
L$INIT	030714 G	O$GNSW=	000001	P.AAW	007002	P.ADB	011024	P.AFG	011774
L$LADP	002026 G	O$POIN=	000001	P.AAX	007030	P.ADC	011032	P.AFH	012006
L$LAST=	113562 G	O$SETU=	000001	P.AAY	007064	P.ADD	011046	P.AFI	012020
L$LOAD	002100 G	PAR.DI	014010	P.AAZ	007116	P.ADE	011054	P.AFJ	012036
L$LUN	002074 G	PD.TEM	015342	P.ABA	007170	P.ADF	011070	P.AFK	012060
L$MREV	002050 G	PHR.1 =	011610	P.ABB	007224	P.ADG	011076	P.AFL	012104
L$NAME	002000 G	PHR.10=	012006	P.ABC	007254	P.ADH	011102	P.AFM	012136
L$PRIO	002042 G	PHR.11=	012020	P.ABD	007330	P.ADI	011110	P.AFN	012150
L$PROT	003444 G	PHR.12=	012036	P.ABE	007370	P.ADJ	011116	P.AFO	012164
L$PRT	002112 G	PHR.13=	012060	P.ABF	007436	P.ADK	011122	P.AFP	012172
L$REPP	002062 G	PHR.14=	012104	P.ABG	007526	P.ADL	011132	P.AFQ	012200
L$REV	002010 G	PHR 15=	012136	P.ABH	007562	P.ADM	011140	P.AFR	012216
L$RPT	004650 G	PHR.2 =	011626	P.ABI	007632	P.ADN	011150	P.AFS	012226
L$SOFT	003122 G	PHR.3 =	011644	P.ABJ	007702	P.ADO	011154	P.AFT	012234
L$SPC	002056 G	PHR.4 =	011676	P.ABK	007754	P.ADP	011170	P.AFU	012244
L$SPCP	002020 G	PHR.5 =	011714	P.ABL	010026	P.ADQ	011176	P.AFV	012260
L$SPTP	002024 G	PHR.6 =	011726	P.ABM	010072	P.ADR	011206	P.AFW	012272
L$STA	002030 G	PHR.7 =	011740	P.ABN	010142	P.ADS	011214	P.AFX	012302
L$SW	002400 G	PHR.8 =	011762	P.ABO	010166	P.ADT	011226	P.AFY	012322
L$TEST	002114 G	PHR.9 =	011774	P.ABP	010206	P.ADU	011240	P.AFZ	012334
L$TIML	002014 G	PNT =	001000 G	P.ABQ	010240	P.ADV	011250	P.AGA	012344
L$UNIT	002012 G	PRI =	002000 G	P.ABR	010246	P.ADW	011260	P.AGB	012360
L10000	002376	PRI00 =	000000 G	P.ABS	010256	P.ADX	011270	P.AGC	012372
L10001	002412	PRI01 =	000040 G	P.ABT	010270	P.ADY	011276	P.AGD	012404
L10002	002510	PRI02 =	000100 G	P.ABU	010304	P.ADZ	011310	P.AGE	012414
L10003	003160	PRI03 =	000140 G	P.ABV	010322	P.AEA	011316	P.AGF	012426
MB.DIS=	005570	PRI04 =	000200 G	P.ABW	010342	P.AEB	011326	P.AGG	012440
MEM.AR=	013052	PRI05 =	000240 G	P.ABX	010364	P.AEC	011336	P.AGH	012452
ML.DUT	016032	PRI06 =	000300 G	P.ABY	010410	P.AED	011344	P.AGI	012464
ML.LUN	016030	PRI07 =	000340 G	P.ABZ	010436	P.AEE	011352	P.AGJ	012500
ML.REG	015404	PRSN	002400 G	P.ACA	010466	P.AEF	011362	P.AGK	012510
MSGH1	002510	PTBL.P	014000	P.ACB	010520	P.AEG	011374	P.AGL	012516
MSGH2	002556	PUP.BB=	006060	P.ACC	010524	P.AEH	011400	P.AGM	012524
MSGH3	002624	PWR.OF=	005112	P.ACD	010536	P.AEI	011416	P.AGN	012532
MSGH4	002670	PWR.ON=	005202	P.ACE	010544	P.AEJ	011432	P.AGO	012540
MSGH5	002736	P.AAA	004716	P.ACF	010552	P.AEK	011440	P.AGP	012546
MSGH6	003004	P.AAB	005012	P.ACG	010560	P.AEL	011446	P.AGQ	012554

```
P.AGR   012562          REG.20= 012666          T$TEMP= 000000          T51     103202 G        WRD.43= 011176
P.AGS   012570          REG.21= 012676          T$TEST= 000000          T52     104020 G        WRD.44= 011206
P.AGT   012576          REG.3 = 012516          T$TSTM= 177777          T53     104660 G        WRD.45= 011214
P.AGU   012604          REG.4 = 012524          T$TSTS= 000000          T54     105722 G        WRD.46= 011226
P.AGV   012612          REG.5 = 012532          T$$HAR= 010002          T55     106612 G        WRD.47= 011240
P.AGW   012620          REG.6 = 012540          T$$HW = 010000          T56     107742 G        WRD.48= 011250
P.AGX   012626          REG.7 = 012546          T$$PRO= 010004          T57     111234 G        WRD.49= 011260
P.AGY   012634          REG.8 = 012554          T$$SOF= 010003          T58     111640 G        WRD.5 = 010552
P.AGZ   012642          REG.9 = 012562          T$$SW = 010001          T59     112306 G        WRD.50= 011270
P.AHA   012652          REM.TB  015664          T.21  = 004716          T6      033476 G        WRD.51= 011276
P.AHB   012660          RE2     004640          T.61  = 005012          T60     112660 G        WRD.52= 011310
P.AHC   012666          RE3     004636          T1      031134 G        T61     113522 G        WRD.53= 011316
P.AHD   012676          RE4     004634          T10     036266 G        T7      034354 G        WRD.54= 011326
P.AHE   012706          RH.ADD  016022          T11     036510 G        T8      035370 G        WRD.55= 011336
P.AHF   012750          RH.ERR= 013302          T12     037066 G        T9      035624 G        WRD.56= 011344
P.AHG   013012          RH.TYP  016024          T13     040230 G        UAM   = 000200 G        WRD.57= 011352
P.AHH   013052          RH.VEC  016026          T14     041462 G        UNS.ER= 005272          WRD.58= 011362
P.AHI   013114          SC.SET= 013214          T15     042714 G        VV.CLE= 005414          WRD.59= 011374
P.AHJ   013154          SEV.FM= 010342          T16     043556 G        VV.NOT= 005340          WRD.6 = 010560
P.AHK   013214          SFPTBL  002400 G        T17     045044 G        VV.SET= 005764          WRD.60= 011400
P.AHL   013250          SIX.FM= 010322          T18     045260 G        WC.ERR= 005652          WRD.61= 011416
P.AHM   013302          STACK   015034          T19     046430 G        WRD.1 = 010520          WRD.62= 011432
P.AHN   013330          STK.OF  015022          T2      031460 G        WRD.10= 010630          WRD.63= 011440
P.CAL   015366          STRIPP  016444          T20     047430 G        WRD.11= 010640          WRD.64= 011446
P.CNT   015376          SVCGBL= 177777          T21     051176 G        WRD.12= 010650          WRD.65= 011454
P.GEN   015374          SVCINS= 177777          T22     051724 G        WRD.13= 010660          WRD.67= 011462
RAS.IN  015346          SVCSUB= 177777          T23     052262 G        WRD.14= 010664          WRD.68= 011470
RD.CS1  021762          SVCTAG= 177777          T24     053456 G        WRD.15= 010676          WRD.69= 011474
RD.DA   022316          SVCTST= 177777          T25     054270 G        WRD.16= 010704          WRD.7 = 010566
RD.DAT  015352          SYNC  = 012750          T26     055466 G        WRD.17= 010712          WRD.70= 011502
RD.DS   025216          S$LSYM= 010000          T27     056244 G        WRD.18= 010724          WRD.71= 011514
RD.D1   024104          TEN.FM= 010436          T28     057230 G        WRD.19= 010730          WRD.72= 011524
RD.D2   024416          THR.FM= 010256          T29     060342 G        WRD.2 = 010524          WRD.73= 011532
RD.D3   024744          TIME.O= 013330          T3      031666 G        WRD.20= 010736          WRD.74= 011540
RD.EE   023760          TRBLE.= 013250          T30     061136 G        WRD.21= 010744          WRD.75= 011552
RD.EL   023716          TST.LN  020376          T31     062534 G        WRD.22= 010752          WRD.76= 011562
RD.ER   022140          TWO.FM= 010246          T32     064272 G        WRD.23= 010766          WRD.77= 011570
RD.E1   023074          T$ARGC= 000002          T33     065304 G        WRD.24= 010774          WRD.8 = 010602
RD.E2   023314          T$CODE= 004130          T34     066266 G        WRD.25= 011002          WRD.9 = 010616
RD.MR   022474          T$ERRN= 000000          T35     066744 G        WRD.26= 011016          WRT.CH  027156
RD.PA   022666          T$EXCP= 000000          T36     070030 G        WRD.27= 011024          WRT.CS  021712
RD.PD   023556          T$FREE  113562 G        T37     070756 G        WRD.29= 011032          WRT.DA  022246
RD.REG  025712          T$GMAN= 000000          T38     072032 G        WRD.3 = 010536          WRT.DS  025214
REGDMP  002404 G        T$HILI= 000007          T39     072460 G        WRD.30= 011046          WRT.D1  024020
REG.IN  015360          T$LAST= 000000          T4      032336 G        WRD.31= 011054          WRT.D2  024332
REG.1 = 012500          T$LOLI= 000000          T40     073436 G        WRD.32= 011070          WRT.D3  024644
REG.10= 012570          T$LSYM= 010000          T41     074516 G        WRD.33= 011076          WRT.EE  023756
REG.11= 012576          T$NEST= 177777          T42     075324 G        WRD.34= 011102          WRT.EL  023714
REG.12= 012604          T$NS0 = 000000          T43     076254 G        WRD.35= 011110          WRT.ER  022070
REG.13= 012612          T$NS1 = 000021          T44     076504 G        WRD.36= 011116          WRT.E1  023010
REG.14= 012620          T$PTHV= 000000 G        T45     076670 G        WRD.37= 011122          WRT.E2  023216
REG.15= 012626          T$PTNU= 000000          T46     077366 G        WRD.38= 011132          WRT.MR  022424
REG.16= 012634          T$SAVL= 177777          T47     077660 G        WRD.39= 011140          WRT.PA  022602
REG.17= 012642          T$SEGL= 177777          T48     100640 G        WRD.4 = 010544          WRT.PD  023450
REG.18= 012652          T$SUBN= 000000          T49     101376 G        WRD.40= 011150          WRT.RE  025324
REG.19= 012660          T$TAGL= 177777          T5      033006 G        WRD.41= 011154          WRT.TR  027426
REG.2 = 012510          T$TAGN= 010005          T50     102014 G        WRD.42= 011170          WT.DAT  015350
```

```
W.C.SI  015774          $T11    036302          $T25    053472          $T39    072046          $T52    103216
W.ERR = 005,22          $T12    036524          $T26    054304          $T4     031702          $T53    104034
XOR.LN  021072          $T13    037102          $T27    055502          $T40    072474          $T54    104674
X$ALWA= 000000          $T14    040244          $T28    056260          $T41    073452          $T55    105736
X$FALS= 000040          $T15    041476          $T29    057244          $T42    074532          $T56    106626
X$OFFS= 000400          $T16    042730          $T3     031474          $T43    075340          $T57    107756
X$TRUE= 000020          $T17    043572          $T30    060356          $T44    076270          $T58    111250
$END.L  113564 G        $T18    045060          $T31    061152          $T45    076520          $T59    111654
$PATCH  003452 G        $T19    045274          $T32    062550          $T46    076704          $T6     033022
$SAVE2  004540 G        $T2     031150          $T33    064306          $T47    077402          $T60    112322
$SAVE3  004554 G        $T20    046444          $T34    065320          $T48    077674          $T61    112674
$SAVE4  004572 G        $T21    047444          $T35    066302          $T49    100654          $T7     033512
$SAVE5  004612 G        $T22    051212          $T36    066760          $T5     032352          $T8     034370
$T1     030724          $T23    051740          $T37    070044          $T50    101412          $T9     035404
$T10    035640          $T24    052276          $T38    070772          $T51    102030

. ABS.  113566      000
        000000      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  35594 WORDS  ( 140 PAGES)
DYNAMIC MEMORY:  21558 WORDS  ( 82 PAGES)
ELAPSED TIME:  00:16:01
ML11,ML11/-SP/CR:SYM=SVC/ML,CZMLAD.DOC,ML2AD,OTS,ML3AD,ML4AD
```

ML11 CREATED BY MACRO ON 30-MAR-82 AT 11:24 PAGE 1
SYMBOL CROSS REFERENCE CREF SEQ 0550

SYMBOL	VALUE		REFERENCES									
ADR	= 000020	G	#94-2561									
ARR.DA	= 013012		#98-2777	224-8947	230-9210	235-9514	243-9909	289-12284	290-12353	292-12438	336-14770	
			342-15047	348-15402	349-15426	349-15450	350-15474	350-15494	361-16098	418-19076	424-19406	
			447-20681	551-25985								
ARR.IN	014004		#88-2240	*205-8023	*205-8043	307-13243	308-13292	308-13294	359-15984	369-16514	369-16523	
			388-17530	389-17548	391-17695	392-17713	440-20280	440-20284				
ARR.16	014016		#89-2251	*205-8008	*205-8024	*205-8044	308-13293					
ASSEMB	= 000010		6-13	6-13								
ASYNC	= 012706		#98-2775	216-8527	220-8730	224-8929	229-9188	235-9494	236-9556	243-9900	249-10237	
			250-10261	250-10281	253-10418	256-10574	256-10590	257-10610	261-10817	261-10832	262-10853	
			262-10870	262-10886	263-10924	263-10943	264-10971	268-11190	269-11209	269-11226	269-11243	
			269-11259	270-11297	270-11316	271-11344	275-11587	276-11606	276-11623	276-11640	277-11660	
			277-11694	278-11717	278-11741	282-11937	283-11957	283-11975	283-11991	284-12012	284-12029	
			295-12602	299-12830	300-12851	300-12869	301-12905	301-12938	302-12984	306-13191	306-13207	
			307-13228	307-13261	308-13311	316-13715	316-13729	316-13742	316-13758	317-13778	318-13835	
			318-13855	319-13912	320-13958	324-14134	324-14156	325-14184	329-14369	355-15775	374-16800	
			383-17250	393-17797	441-20351	458-21211	459-21237	459-21275	460-21301	460-21316	463-21466	
			528-24795	536-25244	547-25801	557-26344	558-26382	558-26394				
A.CAL	015362		#89-2268	108-3283	*108-3284	*108-3285	111-3408	*483-22547	484-22582	*488-22798	489-22828	
A.GEN	015370		#89-2271	111-3384	111-3408	*483-22544	484-22575	*488-22795	488-22817			
BB.BB.	= 006334		#95-2592	558-26396								
BB.INI	= 006242		#95-2591	558-26384								
BB.VV.	= 006150		#95-2590	557-26346								
BIT0	= 000001	G	#94-2545									
BIT00	= 000001	G	#94-2535									
BIT01	= 000002	G	#94-2534									
BIT02	= 000004	G	#94-2533									
BIT03	= 000010	G	#94-2532									
BIT04	= 000020	G	#94-2531									
BIT05	= 000040	G	#94-2530									
BIT06	= 000100	G	#94-2529									
BIT07	= 000200	G	#94-2528									
BIT08	= 000400	G	#94-2527									
BIT09	= 001000	G	#93-2522									
BIT1	= 000002	G	#94-2544									
BIT10	= 002000	G	#93-2521									
BIT11	= 004000	G	#93-2520									
BIT12	= 010000	G	#93-2519									
BIT13	= 020000	G	#93-2518									
BIT14	= 040000	G	#93-2517									
BIT15	= 100000	G	#93-2516									
BIT2	= 000004	G	#94-2543									
BIT3	= 000010	G	#94-2542									
BIT4	= 000020	G	#94-2541									
BIT5	= 000040	G	#94-2540									
BIT6	= 000100	G	#94-2539									
BIT7	= 000200	G	#94-2538									
BIT8	= 000400	G	#94-2537									
BIT9	= 001000	G	#94-2536									
BL$DIV	004446	G	#35-1170	112-3480	514-24072	520-24397	562-26560					
BL$GT1	003512	G	#11-169	424-19395	562-26559							
BL$GT2	003634	G	#16-384	104-3098	108-3255	113-3506	299-12824	301-12899	302-12978	334-14654	336-14740	

SYMBOL	VALUE		REFERENCES								
			348-15353	361-16070	368-16472	374-16773	380-17106	382-17223	389-17573	392-17737	399-18095
			406-18446	417-19045	418-19058	447-20663	478-22282	514-24096	521-24447	562-26558	
BL$LAS	113556	G	#561-26522	561-26523	561-26527						
BL$MOD	004460	G	#36-1201	113-3488	514-24111	520-24401	562-26560				
BL$MUL	004222	G	#32-1022	526-24705	562-26560						
BL$PU1	003776	G	#21-592	483-22532	562-26559						
BL$PU2	004072	G	#27-829	111-3383	111-3392	111-3401	301-12932	418-19097	515-24128	521-24418	562-26559
BL$SHF	004472	G	11-181	11-184	11-188	11-192	12-217	17-406	17-409	17-413	17-417
			17-438	21-595	21-599	21-605	22-632	27-838	27-842	27-848	28-880
			#37-1236								
BOE	= 000400	G	#94-2565								
B.CAL	015364		#89-2269	108-3289	*108-3290	*108-3291	111-3410	*483-22548	484-22583	*488-22799	489-22829
B.GEN	015372		#89-2272	111-3393	111-3410	*483-22545	484-22576	*488-22796	488-22818		
CAL.CR	017042		#107-3224	483-22550	488-22801						
CHIP.S	014012		#89-2248	*205-8022	*205-8042	551-25998					
C$AU	= 000052		#6-13								
C$AUTO	= 000061		#6-13								
C$BRK	= 000022		#6-13								
C$BSEG	= 000004		#6-13								
C$BSUB	= 000002		#6-13								
C$CEFG	= 000045		#6-13								
C$CLCK	= 000062		#6-13								
C$CLEA	= 000012		#6-13								
C$CLOS	= 000035		#6-13								
C$CLP1	= 000006		#6-13								
C$CVEC	= 000036		#6-13								
C$DCLN	= 000044		#6-13								
C$DODU	= 000051		#6-13								
C$DRPT	= 000024		#6-13								
C$DU	= 000053		#6-13								
C$EDIT	= 000003		#6-13								
C$ERDF	= 000055		#6-13								
C$ERHR	= 000056		#6-13								
C$ERRO	= 000060		#6-13								
C$ERSF	= 000054		#6-13								
C$ERSO	= 000057		#6-13								
C$ESCA	= 000010		#6-13								
C$ESEG	= 000005		#6-13								
C$ESUB	= 000003		#6-13								
C$ETST	= 000001		#6-13								
C$EXIT	= 000032		#6-13								
C$GETB	= 000026		#6-13								
C$GETW	= 000027		#6-13								
C$GMAN	= 000043		#6-13								
C$GPHR	= 000042		#6-13								
C$GPLO	= 000030		#6-13								
C$GPRI	= 000040		#6-13								
C$INIT	= 000011		#6-13								
C$INLP	= 000020		#6-13								
C$MANI	= 000050		#6-13								
C$MEM	= 000031		#6-13								
C$MSG	= 000023		#6-13								

```
SYMBOL  VALUE              REFERENCES
C$OPEN  = 000034           #6-13
C$PNTB  = 000014           #6-13
C$PNTF  = 000017           #6-13
C$PNTS  = 000016           #6-13
C$PNTX  = 000015           #6-13
C$QIO   = 000377           #6-13
C$RDBU  = 000007           #6-13
C$REFG  = 000047           #6-13
C$RESE  = 000033           #6-13       #6-13
C$REVI  = 000003           #6-13        6-65
C$RFLA  = 000021           #6-13
C$RPT   = 000025           #6-13
C$SEFG  = 000046           #6-13
C$SPRI  = 000041           #6-13
C$SVEC  = 000037           #6-13
C$TPRI  = 000013           #6-13
DATA.L  = 013154           #98-2780
DAT.DM    016412           #101-2946   103-3054   348-15380  373-16711  373-16732  397-18014  398-18054  404-18369  405-18405
                            417-19006  446-20582  446-20619  514-24074  521-24425
DFPTBL    002360  G        #6-119
DIAGMC  = 000000            6-13        6-13
DIVMOD    004264           #33-1091     35-1171    36-1202
DRIVE.    015354           #89-2262    *205-8019  *205-8039   236-9552   236-9565
DROP.C    015402           #89-2276    *203-7936  *204-7972  *204-7976   204-7979
DT.1      015764           #93-2485     526-24707  526-24709   527-24731  529-24840
DUMPER    026302  G        #191-7313    196-7603   196-7622    199-7727   199-7744   209-8208   212-8370   216-8528   219-8672
                            220-8731    224-8930   224-8939    224-8948   229-9189   229-9198   230-9211   235-9495   235-9506
                            235-9515    236-9557   243-9901    243-9910   249-10238  250-10262  250-10282  253-10419  256-10575
                            256-10591   257-10611  261-10818   262-10837  262-10854  262-10871  262-10887  263-10925  263-10930
                            263-10944   264-10972  264-10977   268-11191  269-11210  269-11227  269-11244  269-11260  270-11298
                            270-11303   271-11321  271-11345   271-11350  271-11371  275-11588  276-11607  276-11624  276-11641
                            277-11661   277-11695  277-11700   278-11718  278-11742  278-11747  279-11772  282-11938  283-11958
                            283-11976   283-11992  284-12013   284-12030  289-12285  290-12354  292-12439  295-12603  300-12835
                            300-12852   300-12870  301-12906   301-12939  302-12985  306-13192  306-13208  307-13229  307-13262
                            308-13312   316-13716  316-13730   316-13743  316-13759  317-13779  318-13836  318-13856  319-13913
                            320-13959   324-14135  324-14157   325-14185  329-14370  329-14385  335-14698  336-14771  342-15048
                            348-15403   349-15427  349-15451   350-15475  350-15495  355-15776  361-16099  369-16529  374-16801
                            380-17130   383-17251  391-17675   393-17798  399-18128  400-18154  406-18465  411-18690  418-19077
                            424-19407   430-19713  430-19748   431-19790  435-20024  440-20290  441-20352  447-20682  453-20951
                            453-20976   454-20999  458-21212   459-21238  459-21276  460-21302  460-21317  463-21467  466-21590
                            469-21764   469-21799  472-21966   479-22306  484-22566  488-22808  494-23084  495-23114  496-23186
                            496-23215   502-23487  503-23518   508-23792  509-23823  515-24170  522-24481  528-24796  528-24811
                            529-24866   535-25185  536-25245   538-25354  542-25558  547-25797  547-25802  551-25986  557-26345
                            558-26383   558-26395
D1.TEM  = 013374           #98-2785    *104-3082  *355-15744  *360-16052   367-16403  *368-16457  *374-16759  *380-17092  *382-17209
                           *398-18077  *405-18428  *417-19028  *447-20649  *478-22274  *515-24129   515-24135  *520-24392   521-24422
D2.TEM  = 013376           #98-2786    *104-3083  *355-15745  *361-16057   367-16404  *368-16458  *374-16760  *380-17093  *382-17210
                           *398-18078  *406-18433  *417-19029  *447-20650  *478-22275  *515-24130   515-24136  *520-24393   521-24423
EF.CON  = 000036  G        #94-2548
EF.NEW  = 000035  G        #94-2549
EF.PWR  = 000034  G        #94-2550
EF.RES  = 000037  G        #94-2547
```

SYMBOL	VALUE		REFERENCES								
EF.STA	= 000040	G	#94-2546								
EIG.FM	= 010364		#95-2626								
ELV.FM	= 010466		#95-2629								
ERRBLK	002160	G	#6-96								
ERRMSG	002156	G	#6-96								
ERRNBR	002154	G	#6-96								
ERR1HR	002402	G	#6-161	93-2513	205-8011						
ERRTYP	002152	G	#6-96								
ERR.CH	017442		#110-3367	484-22555	488-22802						
EVL	= 000004	G	#94-2559								
ESEND	= 002100		#6-13								
ESLOAD	= 000035		#6-13	6-65							
E2.TEM	= 013400		#98-2787	*104-3084	*355-15746	*361-16058	367-16405	*368-16459	*374-16761	*380-17094	*382-17211
			*398-18079	*406-18434	*417-19030	*447-20651	*478-22276	*515-24131	515-24137	*520-24372	*520-24373
			521-24424								
FIND.C	017660		#112-3475	494-23072	496-23176						
FIRST.	020262		#120-3807	261-10811	268-11184	275-11581	323-14119	341-14976	354-15696	354-15718	381-17164
FIV.FM	= 010304		#95-2623	212-8376	236-9525	256-10581	257-10601	257-10617	261-10824	262-10843	262-10860
			262-10877	263-10897	263-10936	264-10954	264-10983	268-11197	269-11216	269-11233	269-11250
			270-11270	270-11309	271-11327	271-11356	275-11594	276-11613	276-11630	276-11647	277-11667
			277-11706	278-11724	278-11753	283-11949	283-11964	283-11982	284-12003	284-12020	292-12445
			300-12858	301-12912	302-12949	306-13198	306-13215	307-13235	307-13268	308-13318	316-13749
			317-13769	317-13785	324-14141	324-14163	325-14191	355-15782	375-16811	406-18471	435-20030
			440-20296	454-21005	460-21308	463-21473	497-23225	503-23524	509-23829		
FMT.1	= 006442		#95-2593	220-8741	528-24821	538-25364	542-25572	547-25815			
FMT.10	= 007116		#95-2602	335-14711							
FMT.11	= 007170		#95-2603	213-8385							
FMT.12	= 007224		#95-2604	324-14146	324-14168	325-14196					
FMT.13	= 007254		#95-2605	337-14807							
FMT.14	= 007330		#95-2606	393-17809							
FMT.15	= 007370		#95-2607	431-19764							
FMT.16	= 007436		#95-2608	225-8970	230-9229	236-9532	243-9925	289-12303	291-12376	292-12452	319-13922
FMT.17	= 007526		#95-2609	206-8073							
FMT.18	= 007562		#95-2610								
FMT.19	= 007632		#95-2611	484-22577	489-22823	529-24842					
FMT.2	= 006506		#95-2594	219-8690	236-9566	253-10435	342-15067	411-18709	436-20045	537-25263	
FMT.20	= 007702		#95-2612	484-22584	489-22830						
FMT.21	= 007754		#95-2613	494-23094	495-23125	496-23196	497-23231	502-23498	503-23530	508-23803	509-23835
FMT.22	= 010026		#95-2614	516-24184	522-24491						
FMT.23	= 010072		#95-2615	192-7383							
FMT.24	= 010142		#95-2616	192-7390	192-7397	192-7404	193-7415	193-7422	193-7429	193-7436	193-7443
			193-7450	193-7457	193-7464	194-7475	194-7483	194-7492	194-7499		
FMT.25	= 010166		#95-2617	246-10078							
FMT.26	= 010206		#95-2618	246-10088							
FMT.3	= 006574		#95-2595								
FMT.4	= 006620		#95-2596	295-12613							
FMT.5	= 006650		#95-2597	406-18478	430-19728	431-19771	431-19805	479-22325			
FMT.6	= 006752		#95-2598	448-20695							
FMT.7	= 007002		#95-2599	302-12997	350-15511	458-21221	459-21247	460-21289			
FMT.8	= 007030		#95-2600	551-26008	551-26019						
FMT.9	= 007064		#95-2601	308-13277	308-13323	361-16108	441-20363	551-25994			
FNC.1	= 012150		#97-2728	253-10421							

```
SYMBOL   VALUE           REFERENCES
FNC.10   = 012272        #97-2737   306-13194  306-13210  307-13231  307-13264  308-13314
FNC.11   = 012302        #97-2738   306-13211  316-13731  316-13745  316-13761  317-13781
FNC.12   = 012322        #97-2739   324-14137  324-14159  325-14187
FNC.13   = 012334        #97-2740   381-17143  440-20295  453-20954  453-20979  454-21000
FNC.14   = 012344        #97-2741   355-15777
FNC.15   = 012360        #97-2742   369-16530  391-17680  440-20300
FNC.16   = 012372        #97-2743   374-16803
FNC.17   = 012404        #97-2744   391-17678  393-17801  440-20294  441-20356
FNC.18   = 012414        #97-2745   418-19083  424-19413
FNC.19   = 012426        #97-2746   466-21591
FNC.2    = 012164        #97-2729   256-10577  257-10597  257-10613  300-12872  301-12908
FNC.21   = 012440        #98-2751   335-14704
FNC.22   = 012452        #98-2752   380-17131
FNC.23   = 012464        #98-2753   209-8210   235-9517   292-12441
FNC.3    = 012172        #97-2730   219-8676   220-8734   283-11945  283-11995  284-12016  284-12036  468-21734  472-21967
FNC.4    = 012200        #97-2731   261-10820  262-10873  263-10893  263-10932  264-10950  264-10979  473-21976
FNC.5    = 012216        #97-2732   224-8952   230-9215   268-11193  269-11246  270-11266  270-11305  271-11323  271-11352
                          272-11377 289-12289  290-12358  399-18130  400-18156  406-18466  418-19079  430-19715  430-19750
                          431-19792 472-21969
FNC.6    = 012226        #97-2733   212-8371   224-8951   230-9214   275-11590  276-11643  277-11663  277-11702  278-11720
                          278-11749 279-11774  289-12288  290-12357  335-14699  411-18692  424-19409  435-20028  472-21968
FNC.7    = 012234        #97-2734   283-11944  283-11994  284-12015  284-12035  529-24868
FNC.8    = 012244        #97-2735   295-12604  295-12607  300-12836  300-12853
FNC.9    = 012260        #97-2736
FORCE.     020032        #115-3603  494-23065  495-23163  502-23469  508-23770
FOR.EC     020210        #118-3748  535-25173  536-25216  537-25284  538-25330  546-25747
FOR.FM   = 010270        #95-2622   295-12608  300-12840  300-12876  329-14375  329-14390  337-14794  342-15053  393-17803
                          441-20357 453-20956  453-20981  460-21322  466-21595  469-21769  470-21808  479-22311  495-23119
                          502-23492 508-23797  528-24801  529-24871  542-25563
F$AU     = 000015        #6-13
F$AUTO   = 000020        #6-13
F$BGN    = 000040        #6-13      6-39       7-229      7-277      7-316      7-348
F$CLEA   = 000007        #6-13
F$DU     = 000016        #6-13
F$END    = 000041        #6-13      6-13       6-13       6-13       6-13       6-13       6-13       6-13       6-13
                          6-13      6-13       6-13       6-13       6-13       6-13       6-13       6-13       6-39
                          7-250     7-295      7-348
F$HARD   = 000004        #6-13      7-229      7-250
F$HW     = 000013        #6-13      6-119      6-139
F$INIT   = 000006        #6-13
F$JMP    = 000050        #6-13
F$MOD    = 000000        #6-13      6-39       7-348
F$MSG    = 000011        #6-13
F$PROT   = 000021        #6-13      7-316      7-322
F$PWR    = 000017        #6-13
F$RPT    = 000012        #6-13
F$SEG    = 000003        #6-13
F$SOFT   = 000005        #6-13      7-277      7-295
F$SRV    = 000010        #6-13
F$SUB    = 000002        #6-13
F$SW     = 000014        #6-13      6-150      6-165
F$TEST   = 000001        #6-13
```

```
SYMBOL  VALUE          REFERENCES
GD.BLK    020312       #121-3858   398-18041   405-18396   410-18653   411-18674   416-18989   423-19351   423-19368   429-19683
                        435-19990   435-20007   446-20607   452-20903   452-20925   469-21755   469-21790   470-21817   472-21947
                        472-21957   477-22237   477-22242   483-22540   488-22791   493-23038   494-23066   495-23164   501-23421
                        502-23470   507-23722   508-23775   513-24052   515-24154   520-24387   521-24461   527-24735   541-25522
                        542-25537   546-25750   546-25761
GOOD.B    014006       #88-2241    101-2948    121-3860    *205-8007   *369-16507
G$CNTO  = 000200       #6-13
G$DELM  = 000372       #6-13
G$DISP  = 000003       #6-13
G$EXCP  = 000400       #6-13
G$HILI  = 000002       #6-13
G$LOLI  = 000001       #6-13
G$NO    = 000000       #6-13
G$OFFS  = 000400       #6-13       7-242       7-243       7-244       7-245       7-246       7-247       7-248       7-288
                        7-289       7-290       7-291       7-292
G$OFSI  = 000376       #6-13       7-242       7-243       7-244       7-245       7-246       7-247       7-248       7-288
                        7-289       7-290       7-291       7-292
G$PRMA  = 000001       #6-13       7-242
G$PRMD  = 000002       #6-13       7-243       7-244       7-245       7-247
G$PRML  = 000000       #6-13       7-246       7-248       7-288       7-289       7-290       7-291       7-292
G$RADA  = 000140       #6-13
G$RADB  = 000000       #6-13
G$RADD  = 000040       #6-13       7-245
G$RADL  = 000120       #6-13       7-246       7-248       7-288       7-289       7-290       7-291       7-292
G$RADO  = 000020       #6-13       7-242       7-243       7-244       7-247
G$XFER  = 000004       #6-13
G$YES   = 000010       #6-13       7-242       7-243       7-244       7-245       7-246       7-247       7-248       7-288
                        7-289       7-290       7-291       7-292
HELP    = 000000       #6-4        6-8         6-30        6-48        6-67        6-105       6-121       6-152       #7-170
                        7-231       7-252       7-279       7-298       7-324       7-341       7-351
HOE     = 100000   G   #94-2572
HW.OR.    013402       #88-2234    *341-14991  342-15035
IBE     = 010000   G   #94-2569
IDU     = 000040   G   #94-2562
IER     = 020000   G   #94-2570
INTER   = 013114       #98-2779    196-7602    196-7621    199-7726    199-7743    219-8671    271-11370   279-11771   335-14697
                        369-16528   380-17129   391-17674   400-18153   440-20289
INTERV    002410   G   #6-164      93-2512     315-13690   556-26269
IO.BUF    014022       #89-2253    101-2949    113-3492    *113-3493   120-3810    121-3861    122-3911    *317-13795  317-13803
                       *320-13938   320-13945   360-16004   360-16028   366-16376   367-16429   389-17547   392-17712  *398-18039
                       *405-18395  *410-18652  *411-18670   411-18681   416-18945   422-19309  *423-19359   424-19389  *429-19677
                       *429-19678  *434-19980  *434-19981  *435-19998   435-20015  *439-20242   439-20255   440-20268  *440-20316
                        441-20323  *441-20334   441-20339  *445-20568  *445-20572  *445-20573  *445-20574  *452-20902  *452-20917
                        453-20943   453-20968   458-21199   463-21457   465-21576  *469-21752  *469-21774  *470-21816  *472-21946
                       *483-22511  *483-22534  *484-22597  *488-22787  *493-23037  *494-23059   494-23102  *495-23157   496-23207
                       *501-23420  *502-23463   502-23506  *507-23721  *508-23764   508-23811  *513-24043   515-24162  *520-24383
                        522-24473   535-25176   536-25230   537-25286   538-25332  *541-25521  *546-25749   550-25969  *556-26282
                        556-26290   558-26365
ISR     = 000100   G   #94-2563
IXE     = 004000   G   #94-2568
I$AU    = 000041       #6-13
I$AUTO  = 000041   .   #6-13
```

SYMBOL	VALUE		REFERENCES								
I$CLN	= 000041		#6-13								
I$DU	= 000041		#6-13								
I$HRD	= 000041		#7-229	#7-250							
I$INIT	= 000041		#6-13								
I$MOD	= 000041		#6-13	6-39	#6-39	7-348	#7-348				
I$MSG	= 000041		#6-13								
I$PROT	= 000040		#6-13	#7-316							
I$PTAB	= 000041		#6-13								
I$PWR	= 000041		#6-13								
I$RPT	= 000041		#6-13								
I$SFG	= 000041		#6-13								
I$SETU	= 000041		#6-13								
I$SFT	= 000041		#7-277	#7-295							
I$SRV	= 000041		#6-13								
I$SUB	= 000041		#6-13								
I$TST	= 000041		#6-13								
J$JMP	= 000167		#6-13								
LAST.B	020344		#122-3908	379-17042	379-17064	381-17182	459-21265				
LAU	004704		#44-188	44-206							
LAUTO	004660		#42-87	42-105							
LCLEAN	113536		#560-26478	561-26502							
LDU	004672		#43-136	43-154							
LD.LNG	021354		#133-4443	367-16398	515-24134	521-24421					
LIMIT	015400		#89-2275	*205-8013	*205-8015	212-8365	224-8919	229-9178	235-9480	242-9886	289-12278
			290-12347	292-12432	307-13257	308-13307	324-14130	324-14152	324-14176	335-14693	342-15043
			348-15398	349-15422	349-15446	350-15470	350-15490	406-18458	411-18685	418-19070	424-19402
			430-19708	430-19743	431-19785	447-20675	453-20946	453-20971	458-21207	459-21233	463-21462
			478-22295	484-22559	494-23077	494-23105	496-23181	496-23210	502-23480	502-23509	508-23785
			509-23818	515-24165	522-24476	536-25238	538-25347	551-25981			
LINIT	027650		#203-7927	206-8101							
LOAD.S	016034		#98-2795	104-3111							
LOE	= 040000	G	#94-2571								
LOT	= 000010	G	#94-2560								
LRPT	004646		#41-34	41-52							
LST.AR	014020		#89-2252	*205-8009	*205-8029	*205-8030	*206-8055	*206-8056	307-13242	308-13291	359-15983
			369-16513	369-16522	440-20279	440-20283	535-25200				
LST.BL	014014		#89-2250	122-3910	*205-8010	*205-8035	*205-8036	*205-8037	*206-8063	*206-8064	*206-8065
			317-13807	320-13952	381-17151	458-21185	459-21269	460-21288	463-21447	465-21577	535-25174
			536-25223	537-25287	537-25304	538-25327	550-25958	557-26298	558-26376		
LST.DU	015356		#89-2264	194-7487	*206-8069	*206-8071	212-8338				
L$ACP	002110	G	#6-65								
L$APT	002036	G	#6-65								
L$AU	004706	G	6-65	#44-206							
L$AUT	002070	G	#6-65								
L$AUTO	004662	G	6-65	#42-105							
L$CCP	002106	G	#6-65								
L$CLEA	113546	G	6-65	#560-26497							
L$CO	002032	G	#6-65								
L$DEPO	002011	G	#6-65								
L$DESC	002130	G	6-65	#6-87							
L$DESP	002076	G	#6-65								
L$DEVP	002060	G	#6-65								

SYMBOL	VALUE		REFERENCES
L$DISP	002164	G	6-65 #6-103
L$DLY	002116	G	#6-65 103-3031 103-3058 103-3071 209-8196 212-8355 216-8516 220-8711 256-10563 263-10904 270-11277 277-11674 282-11926 328-14354 334-14633 341-14982 341-15006 341-15019 354-15701 354-15722 355-15737 360-16010 360-16033 360-16045 367-16410 367-16433 368-16450 373-16716 373-16736 374-16752 379-17047 379-17068 380-17085 381-17186 382-17202 389-17553 392-17718 397-18019 398-18058 398-18070 405-18378 405-18409 405-18421 417-19010 417-19021 446-20587 446-20623 447-20642 478-22251 478-22267 514-24078 515-24141 521-24429 557-26316 557-26333
L$DTP	002040	G	#6-65
L$DTYP	002034	G	#6-65
L$DU	004674	G	6-65 #43-154
L$DUT	002072	G	#6-65
L$DVTY	002122	G	6-65 #6-80
L$EF	002052	G	#6-65
L$ENVI	002044	G	#6-65
L$ERRT	002152	G	6-65 #6-96
L$ETP	002102	G	#6-65
L$EXP1	002046	G	#6-65
L$EXP4	002064	G	#6-65
L$EXP5	002066	G	#6-65
L$HARD	002414	G	6-65 7-229 #7-229
L$HIME	002120	G	#6-65
L$HPCP	002016	G	#6-65
L$HPTP	002022	G	#6-65
L$HW	002360	G	6-65 6-119 #6-119
L$ICP	002104	G	#6-65
L$INIT	030714	G	6-65 #206-8101
L$LADP	002026	G	#6-65
L$LAST	= 113562	G	6-65 #561-26527
L$LOAD	002100	G	#6-65
L$LUN	002074	G	#6-65
L$MREV	002050	G	#6-65
L$NAME	002000	G	#6-65
L$PRIO	002042	G	#6-65
L$PROT	003444	G	6-65 #7-316
L$PRT	002112	G	#6-65
L$REPP	002062	G	#6-65
L$REV	002010	G	#6-65
L$RPT	004650	G	6-65 #41-52
L$SOFT	003122	G	6-65 7-277 #7-277
L$SPC	002056	G	#6-65
L$SPCP	002020	G	#6-65
L$SPTP	002024	G	#6-65
L$STA	002030	G	#6-65
L$SW	002400	G	6-65 6-150 #6-150
L$TEST	002114	G	#6-65
L$TIML	002014	G	#6-65
L$UNIT	002012	G	#6-65 203-7920 204-7942 204-7987
L10000	002376		6-119 #6-139
L10001	002412		6-150 #6-165
L10002	002510		7-229 #7-250
L10003	003160		7-277 #7-295

ML11 CREATED BY MACRO ON 30-MAR-82 AT 11:24 PAGE 9
SYMBOL CROSS REFERENCE CREF SEQ 0558

SYMBOL	VALUE	REFERENCES								
MB.DIS	= 005570	#95-2585	319-13914							
MEM.AR	= 013052	#98-2778								
ML.DUT	016032	#93-2509	103-3049	104-3120	169-6292	170-6317	173-6482	174-6507	178-6695	179-6723
		196-7586	199-7713	*205-8002	206-8078	209-8188	212-8348	216-8509	219-8661	219-8665
		219-8667	219-8683	219-8689	220-8702	220-8721	220-8726	220-8740	223-8898	229-9161
		234-9446	235-9468	240-9769	249-10227	250-10252	250-10272	253-10401	256-10556	261-10806
		263-10916	264-10963	268-11179	270-11289	271-11336	275-11576	277-11686	278-11733	282-11918
		289-12261	290-12330	291-12403	295-12587	299-12806	299-12813	299-12820	300-12884	301-12895
		301-12921	301-12928	302-12963	302-12967	302-12974	306-13181	307-13248	308-13298	315-13700
		317-13811	323-14114	328-14336	333-14606	340-14966	341-14997	347-15328	354-15688	354-15713
		359-15991	360-16022	366-16369	367-16422	373-16702	373-16728	379-17033	379-17059	381-17158
		381-17177	389-17539	392-17705	397-18006	398-18035	398-18047	404-18362	405-18390	405-18401
		410-18647	411-18663	416-18985	417-18998	422-19299	423-19364	423-19373	429-19668	434-19971
		435-20003	439-20247	440-20310	441-20329	445-20560	446-20599	446-20612	452-20897	452-20910
		458-21193	459-21260	462-21451	465-21571	468-21743	469-21785	472-21941	472-21952	477-22206
		483-22536	487-22762	493-23047	495-23149	501-23452	507-23753	513-24048	513-24057	515-24150
		520-24376	521-24457	526-24695	529-24857	535-25165	536-25208	537-25276	538-25320	538-25339
		541-25512	542-25533	546-25739	546-25757	550-25963	556-26278	557-26301		
ML.LUN	016030	#93-2508	196-7609	197-7632	199-7733	199-7750	*203-7935	*204-7941	204-7942	204-7945
		204-7981	*204-7985	*204-7986	204-7987	204-7990	206-8072	209-8223	213-8399	216-8541
		225-8989	230-9253	236-9573	243-9948	264-10995	272-11389	279-11786	292-12472	337-14819
		342-15082	356-15790	362-16124	369-16537	375-16815	400-18144	407-18501	412-18727	419-19120
		424-19437	432-19835	436-20060	441-20367	448-20721	479-22343	485-22612	489-22844	
ML.REG	015404	#89-2278	101-2947	101-2948	101-2949	101-2950	103-3048	103-3051	103-3052	103-3055
		103-3067	103-3068	104-3082	104-3083	104-3084	104-3119	104-3122	104-3123	115-3620
		115-3631	115-3634	115-3635	118-3749	118-3757	118-3758	120-3808	120-3809	120-3810
		120-3811	121-3859	121-3860	121-3861	121-3862	122-3909	122-3910	122-3911	122-3912
		135-4569	135-4571	135-4572	135-4574	137-4668	137-4670	137-4671	137-4674	137-4675
		138-4728	138-4730	138-4731	138-4733	140-4828	140-4830	140-4831	140-4834	140-4835
		141-4888	141-4890	141-4891	141-4893	143-4987	143-4989	143-4990	143-4993	143-4994
		144-5047	144-5049	144-5050	144-5052	146-5147	146-5149	146-5150	146-5153	146-5154
		147-5200	147-5206	147-5208	147-5209	147-5211	147-5212	149-5302	149-5308	149-5310
		149-5311	149-5314	149-5315	149-5320	150-5362	150-5368	150-5370	150-5371	150-5373
		150-5374	152-5464	152-5470	152-5472	152-5473	152-5476	152-5477	152-5482	153-5537
		*153-5538	153-5544	153-5546	153-5547	154-5553	*154-5554	154-5555	156-5649	*156-5650
		156-5656	156-5658	156-5659	156-5662	156-5663	*156-5668	156-5669	157-5724	157-5725
		157-5731	157-5733	158-5738	158-5740	158-5741	158-5742	158-5743	160-5837	160-5838
		160-5844	160-5846	160-5847	160-5850	160-5851	160-5856	160-5857	163-5990	166-6125
		167-6170	167-6176	167-6178	167-6179	167-6181	167-6182	169-6285	169-6286	169-6287
		169-6288	169-6291	169-6294	169-6295	169-6296	169-6302	169-6304	170-6309	170-6312
		170-6316	170-6319	170-6320	171-6362	171-6368	171-6370	171-6371	171-6573	171-6374
		173-6475	173-6476	173-6477	173-6478	173-6481	173-6484	173-6485	173-6486	173-6492
		173-6494	173-6495	174-6502	174-6506	174-6509	174-6510	175-6566	*175-6567	175-6573
		175-6575	175-6576	176-6582	*176-6583	176-6584	*178-6687	178-6688	178-6689	178-6690
		178-6691	178-6694	178-6697	178-6698	178-6699	178-6705	179-6711	179-6712	179-6715
		179-6716	*179-6721	179-6722	179-6725	179-6726	182-6869	182-6871	182-6872	182-6875
		182-6876	192-7387	192-7388	192-7394	192-7395	192-7401	192-7402	192-7408	192-7409
		193-7419	193-7420	193-7426	193-7427	193-7433	193-7434	193-7440	193-7441	193-7447
		193-7448	193-7454	193-7455	193-7461	193-7462	194-7472	194-7473	194-7480	194-7481
		194-7489	194-7490	194-7496	194-7497	196-7585	196-7588	196-7589	196-7590	196-7591
		196-7592	196-7593	196-7594	196-7595	196-7596	196-7598	196-7613	196-7615	196-7617
		199-7712	199-7715	199-7716	199-7717	199-7718	199-7719	199-7720	199-7721	199-7722

SYMBOL VALUE REFERENCES

199-7737 199-7739 *204-7963 206-8077 206-8080 206-8081 209-8187 209-8190 209-8191
209-8192 209-8193 209-8203 212-8347 212-8350 212-8351 212-8352 212-8362 213-8384
216-8508 216-8511 216-8512 216-8513 216-8523 219-8660 219-8663 219-8664 219-8665
219-8666 220-8706 220-8707 220-8708 220-8723 220-8724 220-8725 223-8897 223-8900
224-8905 225-8969 229-9160 229-9163 229-9164 230-9228 *234-9439 234-9445 234-9448
234-9449 234-9460 234-9461 234-9463 235-9470 235-9471 236-9531 *236-9551 236-9552
240-9768 240-9771 240-9772 243-9924 246-10050 246-10057 246-10060 246-10066 246-10076
249-10226 249-10229 249-10230 249-10231 249-10232 249-10233 250-10251 250-10254 250-10255
250-10256 250-10257 250-10271 250-10274 250-10275 250-10276 250-10277 252-10396 253-10403
253-10404 253-10407 253-10426 256-10555 256-10558 256-10559 256-10560 256-10570 256-10586
257-10606 261-10805 261-10808 261-10809 261-10812 261-10813 261-10828 262-10849 262-10866
262-10882 263-10911 263-10913 263-10915 263-10918 263-10919 263-10920 264-10960 264-10962
264-10965 264-10966 264-10967 268-11178 268-11181 268-11182 268-11185 268-11186 268-11201
269-11222 269-11239 269-11255 270-11284 270-11286 270-11288 270-11291 270-11292 270-11293
271-11333 271-11335 271-11338 271-11339 271-11340 271-11366 275-11575 275-11578 275-11579
275-11582 275-11583 276-11602 276-11619 276-11636 277-11656 277-11681 277-11683 277-11685
277-11688 277-11689 277-11690 278-11730 278-11732 278-11735 278-11736 278-11737 279-11767
282-11917 282-11920 282-11921 282-11922 282-11923 282-11933 283-11953 283-11971 283-11987
284-12008 284-12025 289-12260 289-12263 289-12264 289-12302 290-12329 290-12332 290-12333
291-12375 291-12402 291-12405 291-12406 291-12417 291-12418 292-12451 295-12586 295-12589
295-12590 295-12597 295-12598 299-12805 299-12808 299-12809 299-12810 299-12811 299-12812
300-12847 300-12864 300-12865 300-12883 301-12918 301-12919 301-12933 301-12934 302-12959
302-12960 302-12965 302-12966 306-13180 306-13183 306-13184 306-13185 306-13186 306-13187
306-13203 307-13224 307-13247 307-13250 307-13251 307-13252 307-13253 307-13254 308-13297
308-13300 308-13301 308-13302 308-13303 308-13304 315-13699 315-13702 315-13703 315-13704
315-13706 315-13707 316-13724 316-13725 316-13738 316-13754 317-13774 317-13810 317-13813
317-13814 318-13825 318-13829 318-13849 318-13851 319-13900 319-13901 319-13921 320-13948
323-14113 323-14116 323-14117 323-14120 323-14121 324-14127 324-14173 328-14335 328-14338
328-14339 328-14340 328-14347 328-14349 328-14351 328-14361 329-14380 333-14605 334-14612
334-14613 334-14615 334-14630 334-14640 340-14965 340-14968 340-14969 341-14975 341-14977
341-14978 341-14991 341-14992 341-14996 341-14999 341-15000 341-15001 341-15003 341-15013
341-15016 341-15026 347-15327 347-15330 347-15331 348-15381 348-15383 348-15384 348-15385
348-15386 354-15687 354-15690 354-15691 354-15692 354-15693 354-15694 354-15695 354-15697
354-15708 354-15712 354-15715 354-15716 354-15717 354-15719 355-15734 355-15744 355-15745
355-15746 359-15990 359-15993 359-15994 359-15995 359-15996 360-16001 360-16002 360-16003
360-16004 360-16005 360-16006 360-16017 360-16021 360-16024 360-16025 360-16026 360-16027
360-16028 360-16029 360-16030 360-16041 360-16042 360-16052 361-16057 361-16058 366-16368
366-16371 366-16372 366-16373 366-16374 366-16375 366-16376 367-16403 367-16404 367-16405
367-16406 367-16417 367-16421 367-16424 367-16425 367-16426 367-16427 367-16428 367-16429
367-16430 367-16442 367-16443 368-16457 368-16458 368-16459 373-16701 373-16704 373-16705
373-16708 373-16709 373-16710 373-16712 373-16723 373-16727 373-16730 373-16731 373-16733
373-16744 373-16745 374-16759 374-16760 374-16761 379-17032 379-17035 379-17036 379-17038
379-17039 379-17040 379-17041 379-17043 379-17054 379-17058 379-17061 379-17062 379-17063
379-17065 379-17077 380-17082 380-17092 380-17093 380-17094 381-17153 381-17154 381-17155
381-17157 381-17160 381-17161 381-17163 381-17165 381-17169 381-17176 381-17179 381-17180
381-17181 381-17183 382-17198 382-17199 382-17209 382-17210 382-17211 389-17538 389-17541
389-17542 389-17545 389-17546 389-17547 389-17549 389-17550 389-17560 390-17595 390-17603
390-17604 390-17609 390-17618 390-17619 390-17623 390-17631 390-17632 390-17637 391-17650
391-17651 391-17656 391-17657 391-17664 392-17704 392-17707 392-17708 392-17710 392-17711
392-17712 392-17714 392-17715 392-17725 392-17741 392-17754 393-17760 393-17762 393-17764
393-17766 393-17768 393-17774 393-17776 393-17782 393-17790 397-18005 397-18008 397-18009
397-18011 397-18012 397-18013 397-18015 397-18026 398-18034 398-18037 398-18038 398-18040

ML11 CREATED BY MACRO ON 30-MAR-82 AT 11:24 PAGE 11
SYMBOL CROSS REFERENCE CREF SEQ 0560

SYMBOL VALUE REFERENCES

398-18042 398-18043 398-18046 398-18049 398-18050 398-18055 398-18066 398-18067 398-18077
398-18078 398-18079 404-18361 404-18364 404-18365 404-18366 404-18367 404-18368 404-18370
405-18385 405-18389 405-18392 405-18393 405-18394 405-18397 405-18398 405-18400 405-18403
405-18404 405-18406 405-18417 405-18418 405-18428 406-18433 406-18434 410-18646 410-18649
410-18650 410-18651 410-18654 410-18655 411-18662 411-18665 411-18666 411-18675 411-18676
416-18984 416-18987 416-18988 416-18990 416-18991 416-18993 417-19000 417-19001 417-19007
417-19017 417-19018 417-19028 417-19029 417-19030 422-19298 422-19301 422-19302 423-19352
423-19353 423-19363 423-19366 423-19367 423-19369 423-19370 423-19372 423-19375 423-19376
429-19667 429-19670 429-19671 429-19684 429-19685 434-19970 434-19973 434-19974 435-19991
435-19992 435-20002 435-20005 435-20006 435-20008 435-20009 439-20246 439-20249 439-20250
439-20251 439-20252 439-20254 439-20255 439-20256 439-20257 439-20258 439-20259 439-20261
440-20267 440-20268 440-20269 440-20270 440-20271 440-20272 440-20274 440-20309 440-20312
440-20313 440-20314 440-20315 440-20317 440-20318 441-20323 441-20324 441-20325 441-20326
441-20328 441-20331 441-20332 441-20333 441-20335 441-20338 441-20339 441-20342 441-20343
441-20344 441-20345 441-20347 441-20361 445-20559 445-20562 445-20563 445-20575 445-20576
446-20581 446-20583 446-20594 446-20598 446-20601 446-20602 446-20606 446-20608 446-20609
446-20611 446-20614 446-20615 446-20620 447-20638 447-20639 447-20649 447-20650 447-20651
452-20896 452-20899 452-20900 452-20901 452-20904 452-20905 452-20909 452-20912 452-20913
452-20923 452-20924 452-20926 453-20935 453-20936 454-20994 458-21192 458-21195 458-21196
458-21197 458-21198 458-21199 458-21200 458-21201 458-21202 458-21204 459-21230 459-21259
459-21262 459-21263 459-21264 459-21266 459-21267 459-21271 460-21294 460-21296 460-21297
463-21450 463-21453 463-21454 463-21455 463-21456 463-21457 463-21458 463-21459 465-21570
465-21573 465-21574 465-21575 465-21576 465-21577 466-21582 466-21583 466-21585 468-21742
468-21745 469-21750 469-21751 469-21756 469-21757 469-21759 469-21775 469-21784 469-21787
469-21788 469-21789 469-21791 469-21792 469-21794 470-21818 470-21819 472-21940 472-21943
472-21944 472-21945 472-21948 472-21949 472-21951 472-21954 472-21955 472-21956 472-21958
472-21959 472-21961 477-22205 477-22208 477-22209 477-22210 477-22221 477-22222 477-22225
477-22226 477-22229 477-22230 477-22235 477-22238 477-22239 477-22241 477-22243 478-22263
478-22264 478-22274 478-22275 478-22276 483-22535 483-22538 483-22539 483-22541 483-22542
487-22761 488-22768 488-22769 488-22792 488-22793 493-23036 493-23039 493-23040 493-23046
493-23049 493-23050 494-23067 494-23068 494-23069 495-23148 495-23151 495-23152 496-23169
496-23170 496-23171 501-23419 501-23422 501-23423 501-23451 502-23458 502-23459 502-23471
502-23472 502-23473 502-23475 502-23477 507-23720 507-23723 507-23724 507-23752 507-23755
508-23760 508-23771 508-23773 508-23774 508-23776 508-23777 508-23778 508-23780 508-23782
513-24047 513-24050 513-24051 513-24053 513-24054 513-24056 513-24059 513-24060 514-24067
514-24068 514-24069 514-24075 514-24090 514-24105 515-24135 515-24136 515-24137 515-24138
515-24149 515-24152 515-24153 515-24155 515-24156 515-24157 520-24375 520-24378 520-24379
520-24388 520-24389 521-24422 521-24423 521-24424 521-24426 521-24441 521-24452 521-24456
521-24459 521-24460 521-24462 521-24463 521-24464 526-24694 526-24697 526-24698 526-24713
527-24718 527-24730 527-24733 527-24734 527-24736 527-24737 527-24738 527-24742 527-24749
527-24756 527-24763 528-24774 528-24785 528-24819 529-24856 529-24859 529-24860 529-24861
529-24876 535-25164 535-25167 535-25168 535-25174 535-25175 535-25176 535-25177 535-25178
535-25180 536-25207 536-25210 536-25211 536-25224 536-25227 536-25229 536-25230 536-25231
536-25232 536-25234 537-25275 537-25278 537-25279 537-25285 537-25286 537-25287 537-25288
537-25289 537-25293 537-25309 538-25319 538-25322 538-25323 538-25327 538-25331 538-25332
538-25333 538-25334 538-25338 538-25341 538-25342 538-25343 541-25511 541-25514 541-25515
541-25516 541-25517 541-25518 541-25519 541-25520 541-25523 541-25524 541-25528 542-25535
542-25536 542-25540 542-25543 542-25545 542-25546 542-25550 542-25568 546-25738 546-25741
546-25742 546-25748 546-25751 546-25752 546-25756 546-25759 546-25760 546-25768 546-25769
546-25771 546-25772 546-25775 546-25776 546-25778 546-25779 546-25783 547-25811 550-25962
550-25965 550-25966 550-25967 550-25968 550-25969 550-25970 550-25971 550-25972 550-25974
556-26277 556-26280 556-26281 557-26300 557-26303 557-26304 557-26305 557-26312 557-26329

SYMBOL		VALUE		REFERENCES									
				557-26340	558-26368	558-26370	560-26478						
MSGH1		002510		7-242	#7-259								
MSGH2		002556		7-243	#7-260								
MSGH3		002624		7-244	#7-261								
MSGH4		002670		7-245	#7-262								
MSGH5		002736		7-246	#7-263								
MSGH6		003004		7-247	#7-264								
MSGH7		003052		7-248	#7-265								
MSGS1		003160		7-288	#7-304								
MSGS2		003224		7-289	#7-305								
MSGS3		003270		7-290	#7-306								
MSGS4		003334		7-291	#7-307								
MSGS5		003400		7-292	#7-308								
MSDUMP		026312		191-7313	#192-7374								
NIB.SA		013374		#88-2232	98-2785	98-2786	98-2787	99-2822	99-2828	99-2834	99-2840	99-2846	
				99-2852	100-2866	100-2872	100-2882	100-2891	125-4066	125-4073	125-4084	126-4096	
				126-4108	126-4115	126-4126	126-4134	126-4146	127-4158	130-4296	130-4299	130-4307	
				130-4309	130-4311	130-4318	130-4324	130-4326	130-4332	130-4336	*133-4460	*133-4467	
				*133-4471	*134-4483	*134-4484	*134-4487	*134-4494	*134-4498	*134-4506	*134-4507	*134-4511	
				*134-4519	*134-4520	479-22316							
NIN.FM	=	010410		#95-2627									
NO.INI	=	005470		#95-2584	320-13960								
ONEFIL	=	000001		#2-4	4-1152	5-1153	6-34	6-167	7-193	7-364			
ONEPAS		002406	G	#6-163	93-2512	204-7977							
ONE.FM	=	010240		#95-2619	192-7379	196-7605	196-7624	199-7729	199-7746	216-8530	315-13694	316-13718	
				317-13816	318-13838	318-13844	318-13858	319-13915	320-13961	556-26273	557-26307	557-26324	
				557-26347	558-26385	558-26397							
OP.NUM		014002		#88-2238	*205-8006	205-8025	205-8031	205-8045	206-8057	253-10405	253-10433	308-13289	
				389-17535	391-17696								
OSAPTS	=	000001		#6-13	#6-46	6-65							
OSAU	=	000001		#6-13	#6-46	6-65							
OSBGNR	=	000001		#6-13	#6-46	6-65							
OSBGNS	=	000001		#6-13	#6-46	6-65							
OSDU	=	000001		#6-13	#6-46	6-65							
OSERRT	=	000001		#6-13	#6-46	6-65							
OSGNSW	=	000001		#6-13	#6-46	6-65							
OSPOIN	=	000001		#6-13	#6-46	6-46	6-65						
OSSETU	=	000001		#6-13	#6-46	6-65							
PAR.DI		014010		#89-2247	*205-8000	249-10222	468-21730						
PD.TEM		015342		#89-2256	*103-3067	104-3092	*360-16041	361-16064	*367-16442	368-16466	*373-16744	374-16767	
				*379-17077	380-17100	*382-17198	382-17217	*389-17560	389-17567	*392-17725	392-17731	*398-18066	
				399-18089	*405-18417	406-18440	*417-19017	417-19040	*447-20638	447-20657	478-22262	*478-22263	
				514-24089	*514-24090	521-24440	*521-24441						
PHR.1	=	011610		#97-2713	261-10822	262-10841	264-10952	268-11195	269-11214	271-11325	275-11592	276-11611	
				278-11722	283-11980	295-12606	300-12838	300-12855	307-13233	308-13316	324-14161	329-14373	
				329-14388	349-15456	350-15480	350-15500	454-21003	460-21320	463-21471	466-21593	494-23085	
				496-23187									
PHR.10	=	012006		#97-2722	369-16531	391-17681	440-20301						
PHR.11	=	012020		#97-2723	528-24812	535-25186							
PHR.12	=	012036		#97-2724	516-24177								
PHR.13	=	012060		#97-2725	522-24484								
PHR.14	=	012104		#97-2726	192-7378								

SYMBOL	VALUE		REFERENCES								
PHR.15	= 012136		#97-2727	246-10077	246-10087						
PHR.2	= 011626		#97-2714	256-10579	262-10858	263-10934	264-10981	269-11231	270-11307	271-11354	276-11628
			277-11704	273-11751	300-12874	301-12910	302-12947	306-13196	316-13747	324-14139	460-21306
PHR.3	= 011644		#97-2715	216-8529							
PHR.4	= 011676		#97-2716	209-8209	219-8678	224-8954	230-9217	236-9524	236-9558	289-12291	290-12360
			292-12444	342-15049	355-15780	375-16809	380-17133	399-18134	400-18160	430-19717	430-19752
			431-19794	435-20025	536-25246	551-25987					
PHR.5	= 011714		#97-2717	212-8374	257-10599	257-10615	262-10855	262-10875	263-10895	269-11228	269-11248
			270-11268	272-11779	276-11645	277-11665	279-11776	283-11947	283-11959	283-11962	284-12001
			284-12018	306-13213	307-13266	316-13763	317-13783	325-14189	349-15412	349-15432	458-21214
			495-23115	547-25804							
PHR.6	= 011726		#97-2718	262-10838	269-11211	276-11608	276-11625	283-11977	302-12990		
PHR.7	= 011740		#97-2719	302-12986							
PHR.8	= 011762		#97-2720	453-20955							
PHR.9	= 011774		#97-2721	453-20980							
PNT	= 001000	G	#94-2566								
PRI	= 002000	G	#94-2567								
PRI00	= 000000	G	#94-2558								
PRI01	= 000040	G	#94-2557								
PRI02	= 000100	G	#94-2556								
PRI03	= 000140	G	#94-2555								
PRI04	= 000200	G	#94-2554								
PRI05	= 000240	G	#94-2553								
PRI06	= 000300	G	#94-2552								
PRI07	= 000340	G	#94-2551								
PRSN	002400	G	#6-160	93-2512	245-10044						
PTBL.P	014000		#88-2236	*204-7947	204-7948	204-7950	204-7951	204-7953	*204-7992	205-7997	205-7999
			205-8001	205-8003	205-8016	206-8066					
PUP.BB	= 006060		#95-2589	557-26323							
PWR.OF	= 005112		#94-2575	317-13815	557-26306						
PWR.ON	= 005202		#94-2576	318-13843							
P.AAA	004716		#66-960	94-2573							
P.AAB	005012		#66-980	94-2574							
P.AAC	005112		#66-1002	94-2575							
P.AAD	005202		#67-1025	94-2576							
P.AAE	005272		#67-1044	94-2577							
P.AAF	005340		#67-1057	94-2578							
P.AAG	005414		#68-1076	95-2583							
P.AAH	005470		#68-1091	95-2584							
P.AAI	005570		#68-1113	95-2585							
P.AAJ	005652		#69-1134	95-2586							
P.AAK	005722		#69-1148	95-2587							
P.AAL	005764		#69-1160	95-2588							
P.AAM	006060		#70-1184	95-2589							
P.AAN	006150		#70-1203	95-2590							
P.AAO	006242		#70-1223	95-2591							
P.AAP	006334		#71-1247	95-2592							
P.AAQ	006442		#71-1271	95-2593							
P.AAR	006506		#71-1283	95-2594							
P.AAS	006574		#72-1305	95-2595							
P.AAT	006620		#72-1312	95-2596							
P.AAU	006650		#72-1320	95-2597							

SYMBOL CROSS REFERENCE

SYMBOL	VALUE	REFERENCES	
P.AAV	006752	#72-1342	95-2598
P.AAW	007002	#73-1354	95-2599
P.AAX	007030	#73-1362	95-2600
P.AAY	007064	#73-1372	95-2601
P.AAZ	007116	#73-1381	95-2602
P.ABA	007170	#73-1395	95-2603
P.ABB	007224	#74-1409	95-2604
P.ABC	007254	#74-1417	95-2605
P.ABD	007330	#74-1432	95-2606
P.ABE	007370	#74-1443	95-2607
P.ABF	007436	#74-1456	95-2608
P.ABG	007526	#75-1479	95-2609
P.ABH	007562	#75-1489	95-2610
P.ABI	007632	#75-1503	95-2611
P.ABJ	007702	#76-1521	95-2612
P.ABK	007754	#76-1535	95-2613
P.ABL	010026	#76-1549	95-2614
P.ABM	010072	#76-1561	95-2615
P ABN	010142	#77-1579	95-2616
P.ABO	010166	#77-1586	95-2617
P.ABP	010206	#77-1592	95-2618
P.ABQ	010240	#77-1601	95-2619
P.ABR	010246	#77-1603	95-2620
P.ABS	010256	#77-1606	95-2621
P.ABT	010270	#77-1610	95-2622
P.ABU	010304	#77-1614	95-2623
P.ABV	010322	#77-1619	95-2624
P.ABW	010342	#77-1625	95-2625
P.ABX	010364	#78-1635	95-2626
P.ABY	010410	#78-1642	95-2627
P.ABZ	010436	#78-1650	95-2628
P.ACA	010466	#78-1658	95-2629
P.ACB	010520	#78-1667	95-2630
P.ACC	010524	#78-1669	95-2631
P.ACD	010536	#78-1673	95-2632
P.ACE	010544	#78-1675	95-2633
P.ACF	010552	#78-1677	95-2634
P.ACG	010560	#78-1679	96-2639
P.ACH	010566	#78-1681	96-2640
P.ACI	010602	#79-1689	96-2641
P.ACJ	010616	#79-1693	96-2642
P.ACK	010630	#79-1697	96-2643
P.ACL	010640	#79-1700	96-2644
P.ACM	010650	#79-1703	96-2645
P.ACN	010660	#79-1706	96-2646
P.ACO	010664	#79-1708	96-2647
P.ACP	010676	#79-1712	96-2648
P.ACQ	010704	#79-1714	96-2649
P.ACR	010712	#79-1716	96-2650
P.ACS	010724	#79-1720	96-2651
P.ACT	010730	#79-1722	96-2652
P.ACU	010736	#79-1724	96-2653

SYMBOL	VALUE	REFERENCES	
P.ACV	010744	#79-1726	96-2654
P.ACW	010752	#79-1728	96-2655
P.ACX	010766	#79-1732	96-2656
P.ACY	010774	#79-1734	96-2657
P.ACZ	011002	#79-1736	96-2658
P.ADA	011016	#80-1744	96-2659
P.ADB	011024	#80-1746	96-2660
P.ADC	011032	#80-1748	96-2661
P.ADD	011046	#80-1752	96-2662
P.ADE	011054	#80-1754	96-2663
P.ADF	011070	#80-1758	96-2664
P.ADG	011076	#80-1760	96-2665
P.ADH	011102	#80-1762	96-2666
P.ADI	011110	#80-1764	96-2667
P.ADJ	011116	#80-1766	96-2668
P.ADK	011122	#80-1768	96-2669
P.ADL	011132	#80-1771	96-2670
P.ADM	011140	#80-1773	96-2671
P.ADN	011150	#80-1776	96-2672
P.ADO	011154	#80-1778	96-2673
P.ADP	011170	#80-1782	96-2674
P.ADQ	011176	#80-1784	96-2675
P.ADR	011206	#80-1787	96-2676
P.ADS	011214	#80-1789	96-2677
P.ADT	011226	#80-1793	96-2678
P.ADU	011240	#81-1801	96-2679
P.ADV	011250	#81-1804	96-2680
P.ADW	011260	#81-1807	96-2681
P.ADX	011270	#81-1810	96-2682
P.ADY	011276	#81-1812	96-2683
P.ADZ	011310	#81-1816	96-2684
P.AEA	011316	#81-1818	96-2685
P.AEB	011326	#81-1821	96-2686
P.AEC	011336	#81-1824	96-2687
P.AED	011344	#81-1826	96-2688
P.AEE	011352	#81-1828	96-2689
P.AEF	011362	#81-1831	96-2690
P.AEG	011374	#81-1835	97-2695
P.AEH	011400	#81-1837	97-2696
P.AEI	011416	#81-1842	97-2697
P.AEJ	011432	#81-1846	97-2698
P.AEK	011440	#81-1848	97-2699
P.AEL	011446	#81-1850	97-2700
P.AEM	011454	#82-1856	97-2701
P.AEN	011462	#82-1858	97-2702
P.AEO	011470	#82-1860	97-2703
P.AEP	011474	#82-1862	97-2704
P.AEQ	011502	#82-1864	97-2705
P.AER	011514	#82-1868	97-2706
P.AES	011524	#82-1871	97-2707
P.AET	011532	#82-1873	97-2708
P.AEU	011540	#82-1875	97-2709

SYMBOL	VALUE	REFERENCES	
P.AEV	011552	#82-1879	97-2710
P.AEW	011562	#82-1882	97-2711
P.AEX	011570	#82-1884	97-2712
P.AEY	011610	#82-1890	97-2713
P.AEZ	011626	#82-1895	97-2714
P.AFA	011644	#82-1900	97-2715
P.AFB	011676	#83-1913	97-2716
P.AFC	011714	#83-1918	97-2717
P.AFD	011726	#83-1922	97-2718
P.AFE	011740	#83-1926	97-2719
P.AFF	011762	#83-1932	97-2720
P.AFG	011774	#83-1936	97-2721
P.AFH	012006	#83-1940	97-2722
P.AFI	012020	#83-1944	97-2723
P.AFJ	012036	#83-1949	97-2724
P.AFK	012060	#83-1955	97-2725
P.AFL	012104	#83-1962	97-2726
P.AFM	012136	#84-1975	97-2727
P.AFN	012150	#84-1979	97-2728
P.AFO	012164	#84-1983	97-2729
P.AFP	012172	#84-1985	97-2730
P.AFQ	012200	#84-1987	97-2731
P.AFR	012216	#84-1992	97-2732
P.AFS	012226	#84-1995	97-2733
P.AFT	012234	#84-1997	97-2734
P.AFU	012244	#84-2000	97-2735
P.AFV	012260	#84-2004	97-2736
P.AFW	012272	#84-2008	97-2737
P.AFX	012302	#84-2011	97-2738
P.AFY	012322	#84-2017	97-2739
P.AFZ	012334	#85-2025	97-2740
P.AGA	012344	#85-2028	97-2741
P.AGB	012360	#85-2032	97-2742
P.AGC	012372	#85-2036	97-2743
P.AGD	012404	#85-2040	97-2744
P.AGE	012414	#85-2043	97-2745
P.AGF	012426	#85-2047	97-2746
P.AGG	012440	#85-2051	98-2751
P.AGH	012452	#85-2055	98-2752
P.AGI	012464	#85-2059	98-2753
P.AGJ	012500	#85-2063	98-2754
P.AGK	012510	#85-2066	98-2755
P.AGL	012516	#85-2068	98-2756
P.AGM	012524	#85-2070	98-2757
P.AGN	012532	#85-2072	98-2758
P.AGO	012540	#85-2074	98-2759
P.AGP	012546	#86-2080	98-2760
P.AGQ	012554	#86-2082	98-2761
P.AGR	012562	#86-2084	98-2762
P.AGS	012570	#86-2086	98-2763
P.AGT	012576	#86-2088	98-2764
P.AGU	012604	#86-2090	98-2765

SYMBOL	VALUE	REFERENCES
P.AGV	012612	#86-2092 98-2766
P.AGW	012620	#86-2094 98-2767
P.AGX	012626	#86-2096 98-2768
P.AGY	012634	#86-2098 98-2769
P.AGZ	012642	#86-2100 98-2770
P.AHA	012652	#86-2103 98-2771
P.AHB	012660	#86-2105 98-2772
P.AHC	012666	#86-2107 98-2773
P.AHD	012676	#86-2110 98-2774
P.AHE	012706	#86-2113 98-2775
P.AHF	012750	#86-2125 98-2776
P.AHG	013012	#87-2141 98-2777
P.AHH	013052	#87-2152 98-2778
P.AHI	013114	#87-2164 98-2779
P.AHJ	013154	#87-2175 98-2780
P.AHK	013214	#87-2186 98-2781
P.AHL	013250	#88-2200 98-2782
P.AHM	013302	#88-2209 98-2783
P.AHN	013330	#88-2217 98-2784
P.CAL	015366	#89-2270 108-3277 *108-3278 *108-3279 111-3406 *483-22549 484-22581 *488-22800 489-22827
P.CNT	015376	#89-2274 *212-8336 *212-8364 212-8365 *223-8891 *224-8918 224-8919 *229-9153 *229-9177 229-9178 *234-9438 *235-9479 235-9480 *240-9766 *242-9885 242-9886 *288-12252 *289-12277 289-12278 *290-12346 290-12347 *292-12431 292-12432 *307-13241 *307-13256 307-13257 *308-13288 *308-13306 308-13307 *323-14105 *324-14129 324-14130 *324-14151 324-14152 *324-14175 324-14176 *333-14604 *335-14692 335-14693 *340-14964 *342-15042 342-15043 *347-15323 *348-15397 348-15398 *349-15421 349-15422 *349-15445 349-15446 *350-15469 350-15470 *350-15489 350-15490 *404-18355 *406-18457 406-18458 *410-18641 *411-18684 411-18685 *415-18934 *418-19069 418-19070 *422-19297 *424-19401 424-19402 *429-19665 *430-19707 430-19708 *430-19742 430-19743 *431-19784 431-19785 *445-20557 *447-20674 447-20675 *452-20895 *453-20945 453-20946 *453-20970 453-20971 *458-21184 *458-21206 458-21207 *459-21232 459-21233 *463-21446 *463-21461 463-21462 *477-22200 *478-22294 478-22295 *483-22507 *484-22558 484-22559 *493-23035 *494-23076 494-23077 *494-23104 494-23105 *496-23180 496-23181 *496-23209 496-23210 *501-23418 *502-23479 502-23480 *502-23508 502-23509 *507-23719 *508-23784 508-23785 *509-23817 509-23818 *513-24037 *515-24164 515-24165 *520-24371 *522-24475 522-24476 *535-25162 *536-25237 536-25238 *538-25346 538-25347 *550-25957 *550-25976 551-25981
P.GEN	015374	#89-2273 111-3378 111-3406 *483-22546 484-22574 *488-22797 488-22816
RAS.IN	015346	#89-2259 *205-8021 *205-8041 369-16509 439-20243 439-20253
RD.CS1	021762	#137-4661 189-7216 242-9847
RD.DA	022316	#143-4980 189-7226 242-9861
RD.DAT	015352	#89-2261 *137-4676 137-4677 *140-4836 140-4837 *143-4995 143-4996 *146-5155 146-5156 *149-5316 149-5317 *152-5478 152-5479 *156-5664 156-5665 *160-5852 160-5853 *163-5990 163-5991 *166-6125 166-6126 *170-6312 170-6313 *174-6502 174-6503 *179-6717 179-6718 *182-6877 182-6878 225-8963 230-9222 236-9529 243-9918 289-12296 290-12365 292-12449 319-13919
RD.DS	025216	#182-6862 190-7270
RD.D1	024104	#169-6283 190-7275
RD.D2	024416	#173-6473 190-7280
RD.D3	024744	#178-6685 190-7285
RD.EE	023760	#166-6123 189-7256
RD.EL	023716	#163-5988 189-7261
RD.ER	022140	#140-4821 189-7221 242-9854
RD.E1	023074	#152-5462 189-7236 242-9875

```
SYMBOL   VALUE              REFERENCES
RD.E2      023314           #156-5647    189-7241    242-9882
RD.MR      022474           #146-5140    189-7231
RD.PA      022666           #149-5300    189-7246    242-9868
RD.PD      023556           #160-5835    189-7251
RD.REG     025712           #188-7189    224-8915    229-9174    235-9476    289-12274   290-12343   292-12428   319-13907
REGDMP     002404   G       #6-162       93-2512     192-7375
REG.IN     015360           #89-2266     169-6289    173-6479    178-6692   *291-12397  *291-12419  *292-12469
REG.1    = 012500           #98-2754     192-7389
REG.10   = 012570           #98-2763
REG.11   = 012576           #98-2764
REG.12   = 012604           #98-2765
REG.13   = 012612           #98-2766
REG.14   = 012620           #98-2767     194-7474    528-24814   529-24870
REG.15   = 012626           #98-2768     194-7482    535-25188   536-25247   538-25357
REG.16   = 012634           #98-2769
REG.17   = 012642           #98-2770     193-7421    209-8211
REG.18   = 012652           #98-2771     192-7396
REG.19   = 012660           #98-2772     192-7403
REG.2    = 012510           #98-2755     193-7428    300-12857
REG.20   = 012666           #98-2773     194-7491
REG.21   = 012676           #98-2774     194-7498
REG.3    = 012516           #98-2756     193-7435    284-12031
REG.4    = 012524           #98-2757     193-7449
REG.5    = 012532           #98-2758     193-7442    301-12940   302-12987
REG.6    = 012540           #98-2759     193-7414    219-8673    459-21241   460-21283   460-21303
REG.7    = 012546           #98-2760     193-7456    236-9559
REG.8    = 012554           #98-2761
REG.9    = 012562           #98-2762     193-7463
REM.TB     015664           #92-2448     108-3275    108-3281    108-3287    115-3622    115-3626    115-3632
RE2        004640            39-1349    #39-1379
RE3        004636            39-1357    #39-1378
RE4        004634            39-1366    #39-1377
RH.ADD     016022           #93-2505    *204-7950    204-7961
RH.ERR   = 013302           #98-2783     209-8207    212-8369
RH.TYP     016024           #93-2506    *204-7952
RH.VEC     016026           #93-2507    *204-7954
SC.SET   = 013214           #98-2781
SEV.FM   = 010342           #95-2625     391-17683   399-18136   411-18698
SFPTBL     002400   G       #6-150
SIX.FM   = 010322           #95-2624     219-8679    224-8955    230-9218    272-11382   279-11779   284-12037   289-12292   290-12361
                             302-12992   335-14705   337-14784   349-15414   349-15434   349-15458   350-15482   350-15502   400-18161
                             418-19084   424-19414   430-19720   430-19755   431-19797   473-21977
STACK      015034           #89-2255    *100-2885   *100-2886   *100-2898   *100-2899    107-3235    111-3377    429-19701   430-19726
                             430-19739   431-19769   431-19782   431-19803
STK.OF     015022           #89-2254     98-2798    *103-3044    104-3103   *104-3107
STRIPP     016444           #103-3038    107-3228    429-19689
SVCGBL   = 177777           #6-13       #6-22        6-65        6-65        6-65        6-65        6-65        6-65        6-65
                             6-65        6-65        6-65        6-65        6-65        6-65        6-65        6-65        6-65
                             6-65        6-65        6-65        6-65        6-65        6-65        6-65        6-65        6-65
                             6-65        6-65        6-65        6-65        6-65        6-65        6-65        6-65        6-65
                             6-65        6-65        6-65        6-65        6-65        6-65        6-65        6-65        6-65
                             6-65        6-65        6-65        6-65        6-65        6-65        6-65        6-65        6-65
```

```
SYMBOL   VALUE            REFERENCES
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-80
                          6-80      6-80      6-87      6-87      6-87      6-96      6-96      6-96      6-103
                          6-103     6-103     6-119     6-119     6-119     6-119     6-119     6-119     6-150
                          6-150     6-150     6-150     6-150     6-150     7-229     7-229     7-229     7-277
                          7-277     7-277     7-316     7-316     7-316
SVCINS   = 177777        #6-13     #6-19      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65      6-65
                          6-65      6-65      6-65      6-65      6-80      6-80      6-87      6-87      6-103
                          6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103
                          6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103
                          6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103
                          6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103
                          6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103
                          6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-103
                          6-103     6-103     6-103     6-103     6-103     6-103     6-103     6-119     6-150
                          7-229     7-242     7-242     7-242     7-242     7-243     7-243     7-243     7-243
                          7-243     7-244     7-244     7-244     7-244     7-244     7-245     7-245     7-245
                          7-245     7-245     7-246     7-246     7-246     7-247     7-247     7-247     7-247
                          7-247     7-248     7-248     7-248     7-250     7-277     7-288     7-288     7-288
                          7-289     7-289     7-289     7-290     7-290     7-290     7-291     7-291     7-291
                          7-292     7-292     7-292     7-295
SVCSUB   = 177777        #6-13     #6-21
SVCTAG   = 177777        #6-13     #6-23      6-139     6-139     6-139     6-165     6-165     6-165     7-250
                          7-250     7-250     7-295     7-295     7-295
SVCTST   = 177777        #6-13     #6-20
SYNC     = 012750        #98-2776   224-8938   229-9197   235-9505   263-10929  264-10976  270-11302  271-11349  277-11699
                          278-11746  329-14384  399-18127  406-18464  411-18689  430-19712  430-19747  431-19789  435-20023
                          453-20950  453-20975  454-20998  466-21589  469-21763  469-21798  479-22305  484-22565  488-22807
                          494-23083  495-23113  496-23185  496-23214  502-23486  503-23517  508-23791  509-23822  515-24169
                          522-24480  528-24810  529-24865  535-25184  538-25353  542-25557  547-25796
S$LSYM   = 010000        #6-13     #6-139    #6-165    #7-250    #7-295
TEN.FM   = 010436        #95-2628
THR.FM   = 010256        #95-2621   209-8212   220-8735   243-9914   250-10246  250-10266  250-10286  361-16103  369-16533
                          381-17138  383-17255  440-20303  447-20686  458-21216  459-21242  460-21284  468-21735  484-22570
                          488-22812  494-23088  496-23190  516-24178  522-24485  528-24815  535-25189  538-25358  547-25806
TIME.O   = 013330        #98-2784
TRBLE.   = 013250        #98-2782   472-21965
TST.LN     020376        #125-4047  355-15752  361-16078  368-16480  374-16781  380-17114  382-17231  418-19066  447-20671
                          478-22291
TWO.FM   = 010246        #95-2620   236-9560   251-10302  253-10422  316-13733  381-17144  536-25248  551-25989
T$ARGC   = 000002        #6-65      6-65     #6-65      6-65      6-65     #6-65      6-65      6-65     #6-65
                          6-65      6-65     #6-65      6-65      6-65     #6-65      6-65      6-65
T$CODE   = 004130        #7-242     7-242    #7-242     7-242    #7-242     7-242    #7-243     7-243    #7-243
```

```
SYMBOL   VALUE        REFERENCES
                       7-243    #7-243     7-243    #7-244     7-244    #7-244     7-244    #7-244     7-244
                      #7-245     7-245    #7-245     7-245    #7-245     7-245    #7-246     7-246    #7-246
                       7-246    #7-246     7-246    #7-247     7-247    #7-247     7-247    #7-247     7-247
                      #7-248     7-248    #7-248     7-248    #7-248     7-248    #7-288     7-288    #7-288
                       7-288    #7-288     7-288    #7-289     7-289    #7-289     7-289    #7-289     7-289
                      #7-290     7-290    #7-290     7-290    #7-290     7-290    #7-291     7-291    #7-291
                       7-291    #7-291     7-291    #7-292     7-292    #7-292     7-292    #7-292     7-292
T$ERRN = 000000       #6-13
T$EXCP = 000000       #7-242     7-242    #7-243     7-243    #7-244     7-244    #7-245     7-245    #7-247
                       7-247
T$FREE   113562   G    561-26522 561-26523 #561-26524
T$GMAN = 000000       #6-13
T$HILI = 000007       #7-242     7-242    #7-243     7-243    #7-244     7-244    #7-245     7-245    #7-247
                       7-247
T$LAST = 000000       #6-13
T$LOLI = 000000       #7-242     7-242    #7-243     7-243    #7-244     7-244    #7-245     7-245    #7-247
                       7-247
T$LSYM = 010000       #6-13      6-13      6-139     6-165     7-250     7-295
T$NEST = 177777       #6-13      6-39     #6-39      6-39      6-119    #6-119     6-119     6-139     6-139
                       6-139    #6-139     6-150    #6-150     6-150     6-165     6-165     6-165    #6-165
                       7-229    #7-229     7-229     7-250     7-250     7-250    #7-250     7-277    #7-277
                       7-277     7-295     7-295     7-295    #7-295     7-316    #7-316     7-316     7-322
                       7-322     7-322    #7-322     7-348     7-348     7-348    #7-348
T$NS0  = 000000       #6-39      7-348
T$NS1  = 000021       #6-119     6-139    #6-150     6-165    #7-229     7-250    #7-277     7-295    #7-316
                       7-322
T$PTHV = 000000   G    6-65     #561-26528
T$PTNU = 000000       #6-13
T$SAVL = 177777       #6-13
T$SEGL = 177777       #6-13
T$SUBN = 000000       #6-13
T$TAGL = 177777       #6-13
T$TAGN = 010005       #6-13      6-119     6-119    #6-119     6-150     6-150    #6-150     7-229     7-229
                      #7-229     7-277     7-277    #7-277     7-316     7-316    #7-316
T$TEMP = 000000       #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
                      #6-103     6-103     6-103    #6-103     6-103     6-103    #6-103     6-103     6-103
```

SYMBOL	VALUE		REFERENCES								
			#6-103	6-103	6-103	#6-103	6-103	6-103	#6-103	6-103	6-103
			#6-103	6-103	6-103	#6-103	6-103	6-103	#6-103	6-103	6-103
			#6-103	6-103	6-103	#6-103	#6-139	6-139	#6-165	6-165	#7-242
			7-242	#7-242	7-242	#7-242	7-242	#7-243	7-243	#7-243	7-243
			#7-243	7-243	#7-244	7-244	#7-244	7-244	#7-244	7-244	#7-245
			7-245	#7-245	7-245	#7-245	7-245	#7-246	7-246	#7-246	7-246
			#7-246	7-246	#7-247	7-247	#7-247	7-247	#7-247	7-247	#7-248
			7-248	#7-248	7-248	#7-248	7-248	#7-250	7-250	#7-288	7-288
			#7-288	7-288	#7-288	7-288	#7-289	7-289	#7-289	7-289	#7-289
			7-289	#7-290	7-290	#7-290	7-290	#7-290	7-290	#7-291	7-291
			#7-291	7-291	#7-291	7-291	#7-292	7-292	#7-292	7-292	#7-292
			7-292	#7-295	7-295	#7-322	7-322	#7-348	7-348		
T$TEST	= 000000		#6-13								
T$TSTM	= 177777		#6-13								
T$TSTS	= 000000		#6-13								
T$$HAR	= 010002		#7-229	7-229	7-250						
T$$HW	= 010000		#6-119	6-119	6-139						
T$$PRO	= 010004		#7-316								
T$$SOF	= 010003		#7-277	7-277	7-295						
T$$SW	= 010001		#6-150	6-150	6-165						
T.21	= 004716		#94-2573	315-13693							
T.61	= 005012		#94-2574	556-26272							
T1	031134	G	6-103	#210-8249							
T10	036266	G	6-103	#251-10325							
T11	036510	G	6-103	#254-10464							
T12	037066	G	6-103	#257-10645							
T13	040230	G	6-103	#265-11021							
T14	041462	G	6-103	#272-11411							
T15	042714	G	6-103	#279-11808							
T16	043556	G	6-103	#285-12070							
T17	045044	G	6-103	#293-12498							
T18	045260	G	6-103	#296-12647							
T19	046430	G	6-103	#303-13033							
T2	031460	G	6-103	#213-8421							
T20	047430	G	6-103	#309-13356							
T21	051176	G	6-103	#321-13995							
T22	051724	G	6-103	#325-14231							
T23	052262	G	6-103	#330-14424							
T24	053456	G	6-103	#338-14843							
T25	054270	G	6-103	#343-15108							
T26	055466	G	6-103	#351-15546							
T27	056244	G	6-103	#356-15813							
T28	057230	G	6-103	#362-16150							
T29	060342	G	6-103	#370-16564							
T3	031666	G	6-103	#217-8565							
T30	061136	G	6-103	#375-16838							
T31	062534	G	6-103	#383-17289							
T32	064272	G	6-103	#394-17844							
T33	065304	G	6-103	#400-18185							
T34	066266	G	6-103	#407-18528							
T35	066744	G	6-103	#412-18755							
T36	070030	G	6-103	#419-19142							

SYMBOL	VALUE		REFERENCES								
T37	070756	G	6-103	#425-19463							
T38	072032	G	6-103	#432-19857							
T39	072460	G	6-103	#436-20081							
T4	032336	G	6-103	#221-8773							
T40	073436	G	6-103	#442-20393							
T41	074516	G	6-103	#448-20743							
T42	075324	G	6-103	#454-21035							
T43	076254	G	6-103	#461-21352							
T44	076504	G	6-103	#464-21506							
T45	076670	G	6-103	#466-21618							
T46	077366	G	6-103	#470-21843							
T47	077660	G	6-103	#473-22004							
T48	100640	G	6-103	#480-22367							
T49	101376	G	6-103	#485-22634							
T5	033006	G	6-103	#225-9011							
T50	102014	G	6-103	#489-22865							
T51	103202	G	6-103	#497-23266							
T52	104020	G	6-103	#504-23572							
T53	104660	G	6-103	#510-23877							
T54	105722	G	6-103	#516-24223							
T55	106612	G	6-103	#523-24530							
T56	107742	G	6-103	#530-24905							
T57	111234	G	6-103	#539-25401							
T58	111640	G	6-103	#543-25605							
T59	112306	G	6-103	#548-25849							
T6	033476	G	6-103	#231-9279							
T60	112660	G	6-103	#552-26052							
T61	113522	G	6-103	#559-26426							
T7	034354	G	6-103	#237-9599							
T8	035370	G	6-103	#244-9974							
T9	035624	G	6-103	#247-10114							
UAM	= 000200	G	#94-2564								
UNS.ER	= 005272		#94-2577	318-13837							
VV.CLE	= 005414		#95-2583	316-13717							
VV.NOT	= 005340		#94-2578	316-13732							
VV.SET	= 005764		#95-2588	318-13857							
WC.ERR	= 005652		#95-2586	196-7604	196-7623						
WRD.1	= 010520		#95-2630	256-10580	261-10823	262-10839	262-10856	263-10935	264-10982	268-11196	269-11212
			269-11229	270-11308	271-11355	275-11593	276-11609	276-11626	277-11705	278-11752	283-11948
			283-11960	283-11978	306-13197	316-13748	324-14140				
WRD.10	= 010630		#96-2643	243-9911	335-14703	361-16100	383-17252	393-17799	406-18468	411-18695	418-19081
			424-19411	447-20683	453-20952	453-20977	463-21468	473-21975	479-22307	502-23490	508-23795
			528-24799	538-25355	542-25561	547-25803					
WRD.11	= 010640		#96-2644	256-10578	257-10598	257-10614	263-10933	264-10951	264-10980	270-11306	271-11324
			271-11353	277-11703	278-11721	278-11750	283-11946	300-12837	300-12854	300-12873	301-12909
			301-12942	302-12989	306-13195	306-13212	316-13746	316-13762	317-13782	324-14138	324-14160
			329-14372	329-14387	454-21002	460-21305	460-21319	463-21470	466-21592		
WRD.12	= 010650		#96-2645	212-8373	219-8677	224-8953	230-9216	235-9519	261-10821	262-10874	263-10894
			268-11194	269-11247	270-11267	272-11378	275-11591	276-11644	277-11664	279-11775	284-12000
			284-12017	289-12290	290-12359	292-12443	295-12605	307-13232	307-13265	308-13315	325-14188
			335-14702	355-15779	375-16808	391-17679	399-18132	400-18158	406-18467	411-18694	418-19080
			424-19410	430-19716	430-19751	431-19793	473-21974	496-23217	503-23520	509-23825	

SYMBOL CROSS REFERENCE

CREF

SYMBOL	VALUE	REFERENCES								
WRD.13	= 010660	#96-2646	219-8674	284-12032	380-17132					
WRD.14	= 010664	#96-2647	220-8732	253-10420	284-12033	369-16532	391-17682	440-20302	441-20353	459-21239
		459-21277	528-24797	529-24867	542-25559					
WRD.15	= 010676	#96-2648	300-12839	300-12875	301-12911	302-12948	302-12991			
WRD.16	= 010704	#96-2649	300-12856	307-13234						
WRD.17	= 010712	#96-2650	301-12941	302-12988	460-21304					
WRD.18	= 010724	#96-2651								
WRD.19	= 010730	#96-2652	256-10576	257-10596	257-10612	261-10819	262-10872	262-10888	263-10931	264-10949
		264-10978	268-11192	269-11245	270-11265	270-11304	271-11322	271-11351	271-11372	275-11589
		276-11642	277-11662	277-11701	278-11719	278-11748	279-11773	282-11939	283-11993	284-12014
		284-12034	300-12871	301-12907	306-13193	306-13209	307-13230	307-13263	308-13313	316-13744
		316-13760	317-13780	324-14136	324-14158	325-14186	342-15050	399-18129	400-18155	411-18691
		418-19078	424-19408	430-19714	430-19749	431-19791	435-20027	453-20953	453-20978	454-21001
WRD.2	= 010524	#95-2631	262-10842	262-10859	264-10953	269-11215	269-11232	271-11326	276-11612	276-11629
		278-11723	283-11963	283-11981						
WRD.20	= 010736	#96-2653	272-11380	279-11777	329-14389					
WRD.21	= 010744	#96-2654	329-14371	329-14374	329-14386					
WRD.22	= 010752	#96-2655	355-15781	375-16810	399-18135					
WRD.23	= 010766	#96-2656	399-18131	400-18157	411-18693	430-19719	430-19754	431-19796	435-20029	469-21768
		470-21807								
WRD.24	= 010774	#96-2657	349-15408	349-15428	349-15452	350-15476	350-15496	406-18470	411-18697	479-22309
WRD.25	= 011002	#96-2658	406-18469	411-18696						
WRD.26	= 011016	#96-2659	466-21594							
WRD.27	= 011024	#96-2660	458-21215	460-21307	460-21318	460-21321				
WRD.29	= 011032	#96-2661	458-21213							
WRD.3	= 010536	#95-2632	257-10600	262-10876	269-11249	276-11646	284-12002	306-13214	317-13768	324-14162
WRD.30	= 011046	#96-2662	463-21469	463-21472						
WRD.31	= 011054	#96-2663	459-21240	459-21278						
WRD.32	= 011070	#96-2664	349-15411	349-15431	349-15455	350-15479	350-15499			
WRD.33	= 011076	#96-2665	349-15409	349-15429	349-15453	350-15477	350-15497			
WRD.34	= 011102	#96-2666	349-15413	349-15433	349-15457	350-15481	350-15501			
WRD.35	= 011110	#96-2667	335-14700	342-15052	361-16102	551-25988				
WRD.36	= 011116	#96-2668	342-15051							
WRD.37	= 011122	#96-2669	219-8675	220-8733	243-9912	361-16101	391-17677	393-17800	468-21732	
WRD.38	= 011132	#96-2670	243-9913							
WRD.39	= 011140	#96-2671	393-17802	430-19718	430-19753	431-19795	435-20026			
WRD.4	= 010544	#95-2633	257-10616	263-10896	270-11269	277-11666	284-12019	307-13267	308-13317	317-13784
		325-14190								
WRD.40	= 011150	#96-2672	336-14774	337-14790						
WRD.41	= 011154	#96-2673	447-20685							
WRD.42	= 011170	#96-2674	336-14775	337-14791						
WRD.43	= 011176	#96-2675	262-10840	262-10857	269-11213	269-11230	276-11610	276-11627	283-11961	283-11979
WRD.44	= 011206	#96-2676								
WRD.45	= 011214	#96-2677	335-14701	355-15778						
WRD.46	= 011226	#96-2678	337-14781	337-14793	447-20684					
WRD.47	= 011240	#96-2679	337-14792							
WRD.48	= 011250	#96-2680	374-16802							
WRD.49	= 011260	#96-2681								
WRD.5	= 010552	#95-2634	249-10241	349-15454	350-15478	350-15498	484-22569	488-22811		
WRD.50	= 011270	#96-2682	383-17254	418-19082	424-19412	440-20293	441-20355			
WRD.51	= 011276	#96-2683	337-14780	383-17253						
WRD.52	= 011310	#96-2684	212-8372	224-8950	230-9213	235-9518	289-12287	290-12356	292-12442	528-24813

SYMBOL	VALUE	REFERENCES								
		529-24869	535-25187							
WRD.53	= 011316	#96-2685	250-10296							
WRD.54	= 011326	#96-2686	337-14782							
WRD.55	= 011336	#96-2687	337-14783							
WRD.56	= 011344	#96-2688	224-8949	230-9212	235-9516	289-12286	290-12355	292-12440	381-17142	391-17676
		440-20291								
WRD.57	= 011352	#96-2689								
WRD.58	= 011362	#96-2690								
WRD.59	= 011374	#97-2695	454-21004							
WRD.6	= 010560	#96-2639	250-10265	250-10285	349-15410	349-15430	469-21767	470-21806		
WRD.60	= 011400	#97-2696	440-20292	441-20354						
WRD.61	= 011416	#97-2697	272-11381	279-11778	495-23118	496-23220	503-23523	509-23828		
WRD.62	= 011432	#97-2698	212-8375							
WRD.63	= 011440	#97-2699								
WRD.64	= 011446	#97-2700	479-22310	484-22568	488-22810					
WRD.65	= 011454	#97-2701								
WRD.67	= 011462	#97-2702	496-23216	502-23491	503-23519	509-23824	528-24800	542-25562		
WRD.68	= 011470	#97-2703								
WRD.69	= 011474	#97-2704	528-24798	542-25560						
WRD.7	= 010566	#96-2640	249-10240	250-10264	250-10284	250-10297	468-21733	469-21766	469-21801	
WRD.70	= 011502	#97-2705								
WRD.71	= 011514	#97-2706	538-25356							
WRD.72	= 011524	#97-2707	547-25805							
WRD.73	= 011532	#97-2708	479-22308							
WRD.74	= 011540	#97-2709	494-23087	495-23117	496-23189	496-23219	503-23522	509-23827		
WRD.75	= 011552	#97-2710	494-23086	495-23116	496-23188	496-23218	503-23521	509-23826		
WRD.76	= 011562	#97-2711	502-23489	508-23794	515-24172	522-24483				
WRD.77	= 011570	#97-2712	508-23796							
WRD.8	= 010602	#96-2641	250-10283	469-21800	484-22567	488-22809				
WRD.9	= 010616	#96-2642	249-10239	250-10263	399-18133	400-18159	469-21765	502-23488	508-23793	515-24171
		522-24482								
WRT.CH	027156	#196-7584	320-13947	558-26367						
WRT.CS	021712	#135-4563	185-7017	240-9777	241-9809					
WRT.DA	022246	#141-4882	185-7027	241-9787	241-9817					
WRT.DS	025214	#180-6775	186-7071							
WRT.D1	024020	#167-6169	186-7076							
WRT.D2	024332	#171-6361	186-7081							
WRT.D3	024644	#175-6565	186-7086							
WRT.EE	023756	#164-6041	185-7057							
WRT.EL	023714	#161-5904	186-7066							
WRT.ER	022070	#138-4722	185-7022	240-9780	241-9813					
WRT.E1	023010	#150-5361	185-7037	241-9793	241-9825					
WRT.E2	023216	#153-5536	185-7042	241-9796						
WRT.MR	022424	#144-5041	185-7032							
WRT.PA	022602	#147-5199	185-7047	241-9790	241-9821					
WRT.PD	023450	#157-5723	185-7052	446-20605						
WRT.RE	025324	#184-6990	224-8910	229-9169	234-9454	289-12269	290-12338	291-12411		
WRT.TR	027426	#199-7711	317-13805	557-26296						
WT.DAT	015350	#89-2260	*137-4673	137-4677	*140-4833	140-4837	*143-4992	143-4996	*146-5152	146-5156
		*149-5313	149-5317	*152-5475	152-5479	*156-5661	156-5665	*160-5849	160-5853	*163-5989
		163-5991	*166-6124	166-6126	*170-6311	170-6313	*174-6501	174-6503	*179-6714	179-6718
		*182-6874	182-6878	225-8964	230-9223	243-9919	289-12297	291-12370		

SYMBOL	VALUE		REFERENCES								
W.C.SI	015344		#89-2257	*205-8020	*205-8040	439-20254	440-20267	441-20340			
W.ERR	= 005722		#95-2587	199-7728	199-7745						
XOR.LN	021072		#129-4275	399-18104	406-18454						
XSALWA	= 000000		#6-13								
XSFALS	= 000040		#6-13								
XSOFFS	= 000400		#6-13								
XSTRUE	= 000020		#6-13								
$END.L	113564	G	#561-26535								
$PATCH	003452	G	#7-338								
$SAVE2	004540	G	32-1022	37-1269	#39-1344	112-3476	125-4048	135-4563	137-4661	138-4722	140-4821
			141-4882	143-4980	144-5041	146-5140	147-5199	149-5300	150-5361	152-5462	153-5536
			156-5647	157-5723	160-5835	167-6169	169-6283	171-6361	173-6473	175-6565	178-6685
			182-6862	184-6990	188-7189	209-8183	252-10394	261-10802	268-11175	275-11572	328-14331
			439-20241	472-21937	550-25956	562-26559					
$SAVE3	004554	G	11-169	12-217	16-384	17-438	27-829	28-880	#39-1351	98-2796	110-3368
			203-7927	219-8657	245-10043	249-10221	295-12585	299-12804	306-13179	452-20894	458-21183
			541-25510	546-25737	562-26558						
$SAVE4	004572	G	#39-1359	115-3604	129-4276	216-8504	223-8889	562-26558			
$SAVE5	004612	G	33-1091	37-1269	#39-1368	103-3039	107-3224	212-8334	229-9151	234-9436	240-9764
			288-12248	315-13688	323-14104	333-14602	340-14961	347-15321	354-15681	359-15979	366-16361
			373-16698	379-17029	388-17527	397-18003	404-18353	410-18640	415-18932	422-19295	429-19663
			434-19969	445-20555	477-22198	483-22504	487-22757	493-23033	501-23417	507-23718	513-24035
			520-24369	526-24692	535-25160	556-26267	562-26558				
$T1	030724		#209-8183	210-8250							
$T10	035640		#249-10221	251-10326							
$T11	036302		#252-10394	254-10465							
$T12	036524		#256-10552	257-10646							
$T13	037102		#261-10802	265-11022							
$T14	040244		#268-11175	272-11412							
$T15	041476		#275-11572	279-11809							
$T16	042730		#282-11914	285-12071							
$T17	043572		#288-12248	293-12499							
$T18	045060		#295-12585	296-12648							
$T19	045274		#299-12804	303-13034							
$T2	031150		#212-8334	213-8422							
$T20	046644		#306-13179	309-13357							
$T21	047444		#315-13688	321-13996							
$T22	051212		#323-14104	325-14232							
$T23	051740		#328-14331	330-14425							
$T24	052276		#333-14602	338-14844							
$T25	053472		#340-14961	343-15109							
$T26	054304		#347-15321	351-15547							
$T27	055502		#354-15681	356-15814							
$T28	056260		#359-15979	362-16151							
$T29	057244		#366-16361	370-16565							
$T3	031474		#216-8504	217-8566							
$T30	060356		#373-16698	375-16839							
$T31	061152		#379-17029	383-17290							
$T32	062550		#388-17527	394-17845							
$T33	064306		#397-18003	400-18186							
$T34	065320		#404-18353	407-18529							
$T35	066302		#410-18640	412-18756							

SYMBOL CROSS REFERENCE

SYMBOL	VALUE	REFERENCES	
$T36	066760	#415-18932	419-19143
$T37	070044	#422-19295	425-19464
$T38	070772	#429-19663	432-19858
$T39	072046	#434-19969	436-20082
$T4	031702	#219-8657	221-8774
$T40	072474	#439-20241	442-20394
$T41	073452	#445-20555	448-20744
$T42	074532	#452-20894	454-21036
$T43	075340	#458-21183	461-21353
$T44	076270	#463-21445	464-21507
$T45	076520	#465-21570	466-21619
$T46	076704	#468-21729	470-21844
$T47	077402	#472-21937	473-22005
$T48	077674	#477-22198	480-22368
$T49	100654	#483-22504	485-22635
$T5	032352	#223-8889	225-9012
$T50	101412	#487-22757	489-22866
$T51	102030	#493-23033	497-23267
$T52	103216	#501-23417	504-23573
$T53	104034	#507-23718	510-23878
$T54	104674	#513-24035	516-24224
$T55	105736	#520-24369	523-24531
$T56	106626	#526-24692	530-24906
$T57	107756	#535-25160	539-25402
$T58	111250	#541-25510	543-25606
$T59	111654	#546-25737	548-25850
$T6	033022	#229-9151	231-9280
$T60	112322	#550-25956	552-26053
$T61	112674	#556-26267	559-26427
$T7	033512	#234-9436	237-9600
$T8	034370	#240-9764	244-9975
$T9	035404	#245-10043	247-10115