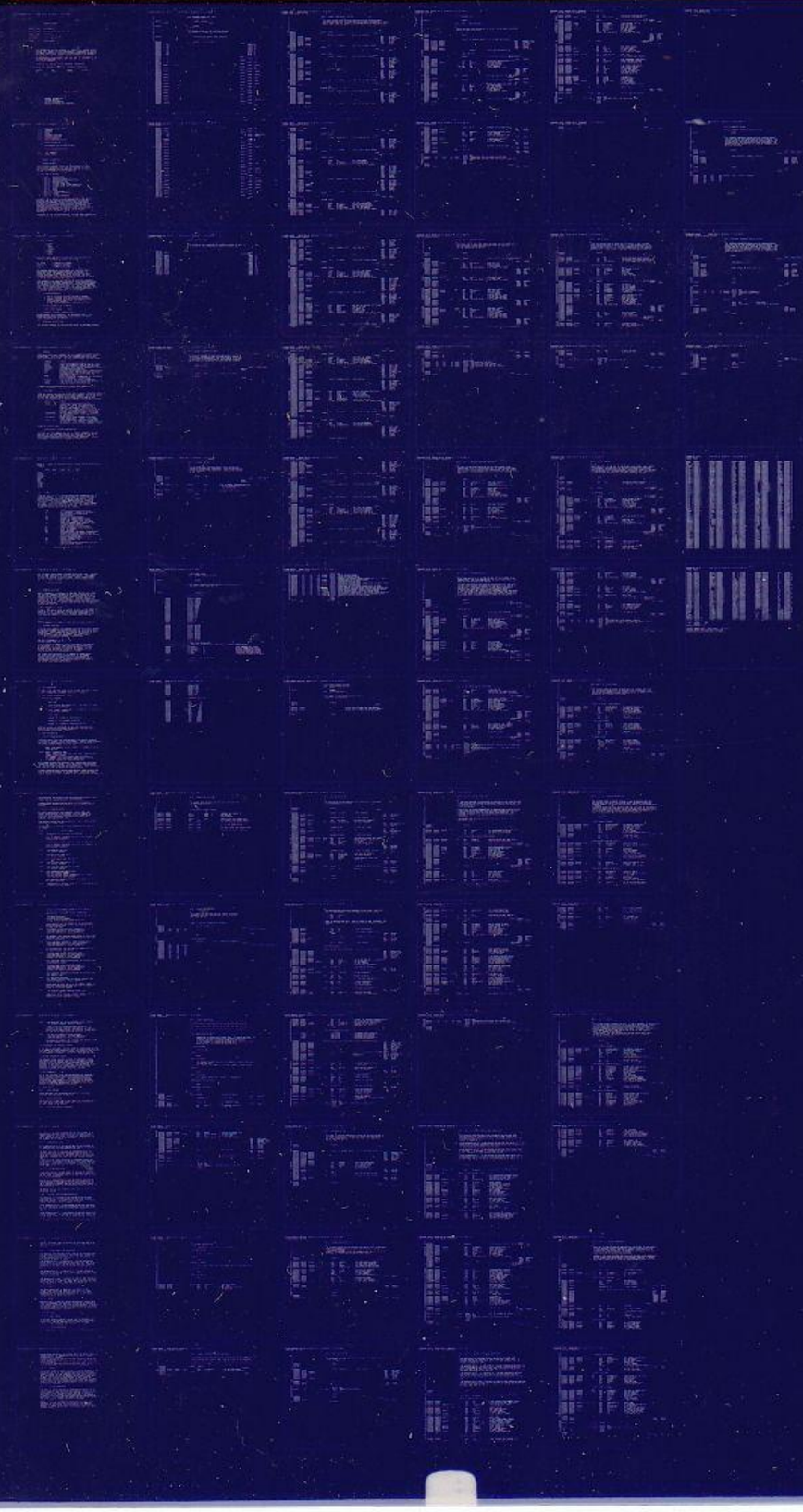
VTV31-K

VTV31-K VID INTF DIAG
CZVTVA0

AH-T061A-MC
FICHE 1 OF 1



.REM &

IDENTIFICATION

PRODUCT CODE: AC-T059A-MC

PRODUCT NAME: CZVTVA0 VTV31K VID INTF DIAG

PRODUCT DATE: 19-OCT-81

MAINTAINER: C.S.S., READING, U.K.

AUTHOR: DAVE HUNTER

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

THIS DIAGNOSTIC PROGRAM CHECKS THE FUNCTIONALITY OF THE VTV31-K VIDEO INTERFACE. TEN TESTS ARE PROVIDED TO VERIFY THE VTV31-K HARDWARE, AND IN ADDITION, ONE TEST OUTPUTS PATTERNS FOR SETTING UP A DISPLAY MONITOR.

THE TESTS ARE AS FOLLOWS:

TEST 1 - REGISTER NXM
TEST 2 - REGISTER ACCESS
TEST 3 - REGISTER ADDRESS UNIQUENESS
TEST 4 - REGISTER BIT SET
TEST 5 - PICTURE STORE DATA
TEST 6 - PICTURE STORE COLUMN ADDRESSING
TEST 7 - PICTURE STORE LINE ADDRESSING
TEST 8 - 8-DOT
TEST 9 - PRESET
TEST 10 - VIDEO ENABLE
TEST 11 - DISPLAY ADDRESSING
TEST 12 - OFFSET
TEST 13 - PATTERN GENERATOR

IN TESTS 1 TO 9, SPECIFIC VALUES ARE WRITTEN TO THE DEVICE REGISTERS AND THE CSR READ TO CONFIRM THAT THE REQUIRED OPERATION IS PERFORMED. TESTS 10 TO 13 ARE VISUAL ONLY. ALTHOUGH THEY WILL RUN WITHOUT A DISPLAY MONITOR, NO CHECKS ARE MADE BY THE PROGRAM. OBSERVATION BY THE OPERATOR IS THEREFORE REQUIRED TO FULLY CHECK THE VTV31K. IN TEST 13, THE OPERATOR CAN CONTROL THE RATE OF CHANGE OF THE DISPLAYED PATTERNS BY SELECTING MANUAL CONTROL IN THE STARTUP QUESTIONS.

IN SEVERAL OF THE TEST DESCRIPTIONS, PICTURE STORE CONTENTS ARE REFERRED TO BY DATA VALUE. THESE DATA VALUES CORRESPOND TO THE

FOLLOWING COLOURS:-

0 = BLACK
1 = RED
2 = GREEN
3 = YELLOW
4 = BLUE
5 = MAGENTA
6 = CYAN
7 = WHITE

EXECUTION TIMES VARY WITH THE CPU TYPE. THE FOLLOWING ARE TYPICAL TIMES OBSERVED ON A PDP-11/23 SYSTEM:

ALL TESTS	16 MINUTES 25 SECONDS
TESTS 1-9	14 MINUTES 40 SECONDS
TESTS 10-13	1 MINUTE 45 SECONDS
TEST 5	10 MINUTES 30 SECONDS

THE PROGRAM SUPPORTS UP TO 20 UNITS, ALL SELECTED TESTS BEING RUN ON ONE UNIT BEFORE PROCEDING TO THE NEXT. IF DURING ONE PASS, THE NUMBER OF ERRORS DETECTED ON A UNIT EXCEEDS A MAXIMUM VALUE (REQUESTED AT STARTUP), THE UNIT IS DROPPED FROM TESTING. UNIT DROPPING IS PREVENTED IF EITHER THE IDU OR LOE FLAGS HAVE BEEN SELECTED.

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

A. PDP-11 PROCESSOR WITH 16K OR MORE OF MEMORY.
B. CONSOLE TERMINAL WITH INTERFACE ADDRESS 777560.
C. XXDP+ SUPPORTED LOAD DEVICE (RX,RM,RK,TM ETC.) OR PAPER TAPE READER.
D. COLOUR DISPLAY MONITORS (1 FOR EACH VTV31-K).

1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USER'S MANUAL CHQUS

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

BEFORE RUNNING THIS DIAGNOSTIC, THE APPROPRIATE PDP-11 CPU, MEMORY AND PERIPHERAL DIAGNOSTICS SHOULD BE RUN TO VERIFY CORRECT OPERATION OF THE SYSTEM.

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 RUNTIME 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 QUESTION FOR EACH UNIT:

```
VTV31-K ADDRESS (O) 174000 ?
```

IN REPLY, YOU SHOULD ENTER AN ADDRESS IN OCTAL IN THE RANGE 160000 TO 177776. TO PREVENT THE QUESTIONS BEING OUTPUT FOR EACH UNIT, ALL ADDRESSES COULD BE ENTERED IN RESPONSE TO THE FIRST ADDRESS REQUEST AS IN THE FOLLOWING SAMPLE DIALOGUE:

```
# UNITS (D) ? 4<CR>

UNIT 0
VTV31-K ADDRESS (O) 174000 ? 174000,174010,174020,174030<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 QUESTIONS WILL THEN BE ASKED:

```
RUN TEST 13 MANUALLY? (L) N ?
AUTODROP ERROR COUNT (D) 5 ?
```

TEST 13 OUTPUTS A NUMBER OF DISPLAY PATTERNS TO THE MONITOR. IF YOU ANSWER 'Y' TO THE FIRST QUESTION, EACH PATTERN WILL BE SHOWN UNTIL YOU PRESS THE CARRIAGE RETURN KEY. IF YOU RESPOND TO THE QUESTION BY TYPING 'N' OR CARRIAGE RETURN, EACH PATTERN WILL LAST APPROXIMATELY 5 SECONDS.

THE SECOND QUESTION REFERS TO THE NUMBER OF ERRORS WHICH MAY OCCUR ON A UNIT WITHIN A GIVEN PASS. IF THE NUMBER EXCEEDS THAT WHICH IS ENTERED (OR 5 IF THE RESPONSE IS A CARRIAGE RETURN), THE UNIT IS DROPPED FROM TESTING UNTIL A NEW COMMAND IS ENTERED. DROPPING IS PREVENTED IF EITHER THE IDU OR LOE FLAGS HAVE BEEN SELECTED.

2.6 CLOCK QUESTIONS

IF THERE IS NO LINE TIME CLOCK ON THE SYSTEM, THE USER IS ASKED TO TYPE 2 CHARACTERS 6 SECONDS APART ON THE CONSOLE.

2.7 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

1. BOOT XXDP+
2. GIVE THE DATE AND ANSWER THE LSI AND 50HZ (IF THERE IS A CLOCK) QUESTIONS
3. TYPE 'R NAME', WHERE NAME IS THE NAME OF THE BIN OR BIC FILE FOR THIS PROGRAM
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.

EXTENDED ERROR MESSAGES ARE NOT USED IN THIS DIAGNOSTIC, ALL SUPPLEMENTARY INFORMATION BEING OUTPUT IN THE BASIC ERROR MESSAGES.

3.2 SPECIFIC ERROR MESSAGES

ALL FAULTS DETECTED BY THE DIAGNOSTIC CAUSE A ONE LINE GENERAL ERROR MESSAGE TO BE PRINTED. THE TEST NUMBER IN WHICH THE FAULT OCCURS IS CONTAINED IN THE HIGHER THREE DIGITS OF THE ERROR NUMBER. THE LOWER TWO DIGITS ARE NUMBERED FROM ZERO FOR EACH TEST.

EACH GENERAL MESSAGE MAY BE FOLLOWED BY A BASIC ERROR MESSAGE CONSISTING OF SEVERAL LINES.

THE FOLLOWING LIST PROVIDES A BRIEF DESCRIPTION OF EACH OF THE ERRORS.

ERROR	DESCRIPTION
00100	'ADDRESSING THE CSR CAUSES AN NXM TRAP' THIS IS PROBABLY CAUSED BY THE DEVICE CSR ADDRESS BEING ENTERED INCORRECTLY.
00101	'ADDRESSING THE DBUFF REGISTER CAUSES AN NXM TRAP' THIS IS PROBABLY CAUSED BY THE DEVICE CSR ADDRESS BEING ENTERED INCORRECTLY.
00200	'CSR READY BIT NOT SET' 'CSR CONTENTS NNNNNN'
00201	'CSR BITS 3 OR 6 NOT CLEAR' 'CSR CONTENTS NNNNNN' BITS OTHER THAN 3 AND 6 MAY BE SET TO 0 OR 1.
00202	'CSR FIELD BIT DOES NOT GET SET' 'CSR CONTENTS NNNNNN'
00203	'CSR FIELD BIT DOES NOT GET CLEARED' 'CSR CONTENTS NNNNNN'
00300	'CSR CHANGED BY WRITING TO DBUFF' 'OLD CSR CONTENTS NNNNNN' 'NEW CSR CONTENTS NNNNNN (IGNORE BITS 0,1,2 AND 4)' ' DATA WRITTEN TO DBUFF NNNNNN' THIS INDICATES ADDRESS INTERACTION PETWEEN THE CSR AND DBUFF REGISTERS.
00400	'COMBINATION OF CSR BITS 5 AND 8 - 15 COULD NOT BE SET.' 'ATTEMPTED SETTING NNNNNN'

```
            'ACTUAL SETTING NNNNNN (IGNORE BITS 0-4 AND 6-7)

            INDICATES BIT 'STICKING' OR BIT INTERACTION.

00401       'COMBINATION OF CSR BITS 0 - 2 COULD NOT BE SET VIA
             THE DBUFF REGISTER.'
            'ATTEMPTED SETTING NNNNNN'
            'ACTUAL SETTING NNNNNN (IGNORE BITS 3-15)'

            COULD INDICATE BIT SET ERRORS IN THE CSR OR DBUFF
            REGISTERS OR AN ERROR IN THE FIRST LOCATION OF THE
            PICTURE STORE RAM.

00500       'BACKGROUND COLOUR READ BACK INCORRECTLY'
            'LINE NNN (NNN OCTAL), DOT NNN (NNN OCTAL)'
            'EXPECTED CONTENTS N, ACTUAL CONTENTS N'

            THE ADDRESS LINES HAVE NOT BEEN TESTED AT THIS STAGE.
            THEREFORE, THE LINE AND DOT NUMBERS MAY NOT BE THOSE
            ACTUALLY WRITTEN TO.

00501       'DATA READ BACK INCORRECTLY FROM PICTURE STORE'
            'LINE NNN (NNN OCTAL), DOT NNN (NNN OCTAL)'
            'EXPECTED CONTENTS N, ACTUAL CONTENTS N'

            AS FOR ERROR 500, THE LINE AND DOT NUMBERS MAY NOT
            BE THOSE ACTUALLY WRITTEN TO.

00600       'PICTURE STORE COLUMN ADDRESSING ERROR'
            'LINE NNN (NNN OCTAL), DOT NNN (NNN OCTAL)'
            'EXPECTED CONTENTS N, ACTUAL CONTENTS N'
            'ADDRESS BITS BEING TESTED NNNNNNNNN'

            THIS INDICATES INTERACTION BETWEEN THE PICTURE STORE
            COLUMN ADDRESS LINES.

00700       'PICTURE STORE LINE ADDRESSING ERROR'
            'LINE NNN (NNN OCTAL), DOT NNN (NNN OCTAL)'
            'EXPECTED CONTENTS N, ACTUAL CONTENTS N'
            'ADDRESS BITS BEING TESTED NNNNNNNNN'

            THIS INDICATES INTERACTION BETWEEN THE PICTURE STORE
            LINE ADDRESS LINES.

00800       '8 - DOT WRITE FAILURE'
            'DOT ADDRESS WRITTEN NNN, FAILING DOT ADDRESS NNN'
            'EXPECTED CONTENTS N, ACTUAL CONTENTS N'

            HAVING MADE AN 8-DOT WRITE TO THE ADDRESS SHOWN, THE
            EXPECTED COLOUR WAS NOT READ BACK FROM THE ADDRESS
            GIVEN AS THE FAILING ADDRESS.

00900       'CSR READY NOT CLEAR DURING A PRESET'
            'CSR CONTENTS NNNNNN, EXPECTED COLOUR N'

            IMMEDIATELY AFTER ISSUING A PRESET COMMAND, THE CSR
            READY BIT DID NOT READ BACK AS ZERO.
```

00901 'CSR READY NOT SET 50 MILLISECONDS AFTER A PRESET'
'CSR CONTENTS NNNNNN, EXPECTED COLOUR'

AFTER A DELAY OF 50 MILLISECONDS FOLLOWING
A PRESET, THE CSR READY BIT WAS NOT SET. IF THE
PRESET DID ACTUALLY OCCUR, THE EXPECTED COLOUR SHOULD
BE SEEN ON THE MONITOR AND BE REFLECTED IN THE CSR.

00902 'DOT NOT SET CORRECTLY AFTER A PRESET'
'LINE NNN (NNN OCTAL), DOT NNN (NNN OCTAL)'
'EXPECTED CONTENTS N, ACTUAL CONTENTS N'

FOLLOWING THE PRESET, THE PICTURE STORE ADDRESS SHOWN
DID NOT CONTAIN THE PRESET COLOUR.

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

THE HARDWARE PTABLE CREATED AT START UP CONTAINS A SINGLE
WORD ENTRY FOR EACH DEVICE UNDER TEST. THIS WORD IS LOADED
WITH THE DEVICE CSR ADDRESS AS THE HARDWARE QUESTIONS ARE
ANSWERED. AT THE END OF EACH SUBPASS, THE CSR ADDRESS OF THE
NEXT UNIT TO BE TESTED IS LOADED INTO LOCATION VTVCSR. THE
ADDRESS+2 IS LOADED INTO LOCATION DBUFF. THE TEST CODE
REFERENCES THE DEVICE THROUGH THESE TWO LOCATIONS.

6.0 TEST SUMMARIES

ALL OF THE FOLLOWING TESTS ARE RUN WHENEVER THE PROGRAM IS
RUN (BY DEFAULT) UNLESS SPECIFICALLY EXCLUDED BY THE TEST
SWITCH (SECTION 2.2). TESTS 1 TO 9 REQUIRE NO OPERATOR
INTERVENTION, THE DIAGNOSTIC VERIFYING THAT THE REQUIRED
ACTION IS PERFORMED CORRECTLY. THE REMAINING TESTS MUST BE
VERIFIED BY OPERATOR OBSERVATION, ALTHOUGH THEY WILL RUN
WITHOUT A DISPLAY MONITOR ON THE SYSTEM.

6.1 LOGIC TESTS

TEST 1 - REGISTER NXM

CHECKS ARE MADE THAT ACCESSING THE DEVICE CSR AND DBUFF
REGISTERS DOES NOT CAUSE AN NXM TRAP.

TEST 2 - REGISTER ACCESS

THE DEVICE CSR IS READ AND A CHECK MADE THAT THE READY BIT IS
SET, THAT BITS 3 AND 6 ARE CLEAR, AND THAT THE FIELD BIT IS
SET AND CLEARED BY THE DEVICE.

TEST 3 - REGISTER ADDRESS UNIQUENESS

ALL BITS IN THE DBUFF REGISTER ARE SET AND CLEARED IN ALL COMBINATIONS WITH THE EXCEPTION OF BIT 14 (READ PICTURE STORE), WHICH IS LEFT CLEAR. FOR EACH BIT COMBINATION, THE CSR IS READ AND CHECKED TO BE UNCHANGED. CSR BITS 0,1,2 AND 4 ARE IGNORED IN THIS CHECK.

TEST 4 - REGISTER BIT SET

BITS 5 AND BITS 8 TO 15 IN THE CSR ARE SET AND CLEARED IN ALL COMBINATIONS. FOR EACH COMBINATION, THE CSR IS READ AND A CHECK MADE THAT THE RELEVANT BITS ARE SET.

THE DBUFF REGISTER IS THEN USED IN CONJUNCTION WITH THE FIRST PICTURE STORE ELEMENT TO ENSURE THAT CSR BITS 0 - 2 CAN BE SET. CSR BITS 8 - 15 AND DBUFF BITS 0 - 8 ARE CLEAR THROUGHOUT, ENSURING THAT ONLY THE FIRST PICTURE ELEMENT IS USED.
A 'WRITE PICTURE STORE' OPERATION IS PERFORMED USING THE COLOUR BLACK. THE ELEMENT IS THEN READ AND A CHECK MADE THAT CSR BITS 0 - 2 READ ZERO. THIS PROCESS IS REPEATED FOR EACH COLOUR CODE, VERIFYING THAT THE CORRECT COLOUR IS REFLECTED IN THE CSR.

TEST 5 - PICTURE STORE DATA

A ZERO DATA PATTERN (COLOUR BLACK) IS WRITTEN TO THE FIRST PICTURE STORE LOCATION AND CHECKED TO READ BACK CORRECTLY IN THE CSR. THIS IS REPEATED FOR ALL PICTURE LOCATIONS TO FORM AN ALL BLACK BACKGROUND.

DATA IN ALL 8 COMBINATIONS IS THEN WRITTEN TO EACH PICTURE STORE LOCATION IN TURN AND AFTER EACH WRITE OPERATION, THE CSR IS EXAMINED TO ENSURE THAT THE DATA READS BACK CORRECTLY. EACH LOCATION IS RESET TO THE BACKGROUND COLOUR AFTER THE ABOVE OPERATION.

THE ENTIRE TEST IS THEN REPEATED USING WHITE (DATA 7) AS THE BACKGROUND COLOUR.

TEST 6 - PICTURE STORE COLUMN ADDRESSING

THIS VERIFIES TH[illegible] WRITING TO ANY COLUMN OF THE PICTURE STORE DOES NOT AF[illegible]T THE CONTENTS OF ANY OTHER COLUMN. SINCE EACH LOCATI[illegible]AN HOLD A 3 BIT DATA VALUE, THE COLUMN ADDRESS LINES ARE [illegible]CKED IN GROUPS OF THREE.

COLUMN ADDRESS BITS 0 - 2 ARE CHECKED BY WRITING THE VALUES 0 - 7 IN CONSECUTIVE LOCATIONS THROUGHOUT THE PICTURE STORE AND THEN CHECKING THAT THE DATA READS BACK CORRECTLY VIA THE CSR.

COLUMN ADDRESS BITS 3 - 5 ARE CHECKED BY WRITING ZERO TO THE FIRST 8 LOCATIONS, 1 TO THE NEXT 8 LOCATIONS ETC. UNTIL THE PICTURE STORE IS FULL. THE DATA IS THEN VERIFIED VIA THE CSR.

COLUMN ADDRESS BITS 6 - 8 ARE CHECKED BY WRITING ZERO TO THE

FIRST 64 LOCATIONS, ONE TO THE NEXT 64 LOCATIONS AND SO ON, UNTIL THE PICTURE STORE IS FULL. THE DATA IS THEN VERIFIED VIA THE CSR.

TEST 7 - PICTURE STORE LINE ADDRESSING

THIS VERIFIES THAT WRITING TO ANY LINE OF THE PICTURE STORE DOES NOT AFFECT THE CONTENTS OF ANY OTHER LINE. SINCE EACH LOCATION CAN HOLD A 3 BIT DATA VALUE, THE LINE ADDRESS LINES ARE CHECKED IN GROUPS OF THREE.

LINE ADDRESS BITS 0 - 2 (CSR BITS 8 - 10) ARE CHECKED BY WRITING ZERO TO ALL LOCATIONS IN THE FIRST LINE, ONE TO ALL LOCATIONS IN THE NEXT LINE AND SO ON, UNTIL ALL LINES ARE WRITTEN TO. THE DATA IS THEN READ BACK AND VERIFIED VIA THE CSR.

LINE ADDRESS BITS 3 - 5 (CSR BITS 11 - 13) ARE CHECKED BY WRITING ZERO TO THE FIRST EIGHT LINES, ONE TO THE NEXT EIGHT LINES ETC. TO THE END OF THE PICTURE STORE. THE DATA IS THEN VERIFIED VIA THE CSR.

LINE ADDRESS BIT 6 AND 7 (CSR BITS 14 AND 15) ARE CHECKED BY WRITING ZERO TO THE FIRST 64 LINES, ONE TO THE NEXT 64 LINES, TWO TO THE NEXT 64 LINES, AND THREE TO THE LAST 64 LINES. THE PICTURE STORE DATA IS THEN READ BACK AND VERIFIED.

TEST 8 - 8 DOT

THIS TEST CHECKS THAT IF DBUFF BIT 15 (8-DOT) IS SET, PICTURE STORE WRITES ARE MADE TO EIGHT DOTS AT A TIME, REGARDLESS OF THE VALUES OF DBUFF BITS 0 - 2. FOR EACH COMBINATION OF THE LOWER DOT ADDRESS BITS, A NEW COLOUR IS USED AND THE 8-DOT WRITE VERIFIED.

TEST 9 - PRESET

DBUFF BITS 10 - 12 ARE SET TO THE COLOUR BLACK (ZERO) AND A PRESET IS PERFORMED. THE READY BIT IN THE CSR IS CHECKED TO CLEAR DURING THE PRESET, AND AFTER 50 MILLISECONDS, TO BE SET. THE PICTURE STORE IS THEN READ TO CONFIRM THAT ALL LOCATIONS HAVE BEEN CORRECTLY SET. THE PROCESS IS THEN REPEATED FOR EACH OF THE EIGHT SELECTABLE COLOURS.

6.2 VISUAL TESTS

TEST 10 - VIDEO ENABLE

THE PICTURE STORE IS PRESET TO BLUE FOR FIVE SECONDS. BIT 5 OF THE CSR (VIDEO ENABLE) IS THEN CLEARED FOR FIVE SECONDS, DURING WHICH TIME THE MONITOR DISPLAY SHOULD BE DARK. THE DISPLAY IS THEN REENABLED FOR A FURTHER FIVE SECONDS.

TEST 11 - DISPLAY ADDRESSING

THE PICTURE STORE IS PRESET TO BLUE. LINES OF MAGENTA ARE THEN

WRITTEN FROM THE TOP OF THE DISPLAY UNTIL THE WHOLE SCREEN HAS CHANGED COLOUR. THE TRANSITION SHOULD BE SMOOTH, TAKING APPROXIMATELY 7 SECONDS.
COLUMNS OF WHITE ARE THEN WRITTEN FROM THE LEFT SIDE OF THE DISPLAY UNTIL THE SCREEN IS ALL WHITE. AGAIN, THE TRANSITION SHOULD BE SMOOTH.
IF THE DISPLAY DOES NOT APPEAR AS DESCRIBED HERE, AN ADDRESSING FAULT ON THE OUTPUT SIDE OF THE DEVICE IS INDICATED.

TEST 12 - OFFSET

THE PICTURE STORE IS PRESET TO BLUE. A LINE OF MAGENTA IS THEN WRITTEN TO THE TOP OF THE DISPLAY AND THE OFFSET REGISTER CHANGED TO MOVE THE LINE DOWN ONE POSITION. THIS IS REPEATED UNTIL THE WHOLE DISPLAY IS MAGENTA, WHICH SHOULD TAKE ABOUT 7 SECONDS. THE DISPLAY IS THEN CHANGED TO WHITE FROM THE BOTTOM USING THE SAME PROCESS, UNTIL THE SCREEN IS ALL WHITE. IN BOTH CASES, THE LINE BETWEEN COLOURS SHOULD MOVE SMOOTHLY ACROSS THE SCREEN. ANY UNEVEN MOVEMENT INDICATES A PROBLEM IN THE OFFSET REGISTER.

TEST 13 - PATTERN GENERATOR

THIS TEST IS FOR SETTING UP AND CHECKING THE OUTPUT TO THE DISPLAY MONITOR. NINE PATTERNS ARE PROVIDED - EIGHT TO PRESET THE SCREEN TO EACH POSSIBLE COLOUR, AND ONE TO OUTPUT A CROSS HATCH OF 32 COLUMN, 16 LINE RECTANGLES. THE NUMBER OF RECTANGLES WHICH ARE DISPLAYED DEPENDS ON THE CONFIGURATION BEING USED - 12/16 ACROSS THE SCREEN FOR 384/512 DOT VERSIONS, AND 15/16 DOWN THE SCREEN FOR 60/50 HZ VERSIONS.

NORMALLY, THE NINE PATTERNS ARE DISPLAYED IN TURN, EACH REMAINING ON THE SCREEN FOR APPROXIMATELY FIVE SECONDS. HOWEVER, IF MANUAL SELECTION IS SELECTED AT START UP, EACH PATTERN IS KEPT DISPLAYED UNTIL THE RETURN KEY IS HIT ON THE OPERATOR CONSOLE.

&

```
                                .TITLE PROGRAM HEADER AND TABLES
                                .SBTTL  PROGRAM HEADER

                                        .ENABL  ABS,AMA
002000                          .       =       2000

                                        BGNMOD

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

                                        POINTER BGNSW,BGNSFT,BGNDU,BGNSETUP


                                        HEADER  CZVTV,A,0,660.,0
002000                                                                  L$NAME::
002000  103                                                                     .ASCII /C/
002001  132                                                                     .ASCII /Z/
002002  126                                                                     .ASCII /V/
002003  124                                                                     .ASCII /T/
002004  126                                                                     .ASCII /V/
002005  000                                                                     .BYTE   0
002006  000                                                                     .BYTE   0
002007  000                                                                     .BYTE   0
002010                                                                  L$REV::
002010  101                                                                     .ASCII  /A/
002011                                                                  L$DEPO::
002011  060                                                                     .ASCII  /0/
002012                                                                  L$UNIT::
002012  000001                                                                  .WORD   T$PTHV
002014                                                                  L$TIML::
002014  001224                                                                  .WORD   660.
002016                                                                  L$HPCP::
002016  012350                                                                  .WORD   L$HARD
002020                                                                  L$SPCP::
002020  012402                                                                  .WORD   L$SOFT
002022                                                                  L$HPTP::
002022  002160                                                                  .WORD   L$HW
002024                                                                  L$SPTP::
002024  002164                                                                  .WORD   L$SW
002026                                                                  L$LADP::
002026  013004                                                                  .WORD   L$LAST
002030                                                                  L$STA::
002030  000000                                                                  .WORD   0
002032                                                                  L$CO::
002032  000000                                                                  .WORD   0
002034                                                                  L$DTYP::
002034  000000                                                                  .WORD   0
002036                                                                  L$APT::
002036  000000                                                                  .WORD   0
002040                                                                  L$DTP::
002040  002124                                                                  .WORD   L$DISPATCH
002042                                                                  L$PRIO::
002042  000000                                                                  .WORD   0
002044                                                                  L$ENVI::
```

PROGRAM HEADER

```
002044  000000                          .WORD   0
002046                          L$EXP1::
002046  000000                          .WORD   0
002050                          L$MREV::
002050     003                          .BYTE   C$REVISION
002051     003                          .BYTE   C$EDIT
002052                          L$EF::
002052  000000                          .WORD   0
002054  000000                          .WORD   0
002056                          L$SPC::
002056  000000                          .WORD   0
002060                          L$DEVP::
002060  002210                          .WORD   L$DVTYP
002062                          L$REPP::
002062  000000                          .WORD   0
002064                          L$EXP4::
002064  000000                          .WORD   0
002066                          L$EXP5::
00206_  000000                          .WORD   0
002070                          L$AUT::
002070  000000                          .WORD   0
002072                          L$DUT::
002072  005562                          .WORD   L$DU
002074                          L$LUN::
002074  000000                          .WORD   0
002076                          L$DESP::
002076  002220                          .WORD   L$DESC
002100                          L$LOAD::
002100  104035                          EMT     E$LOAD
002102                          L$ETP::
002102  000000                          .WORD   0
002104                          L$ICP::
002104  004722                          .WORD   L$INIT
002106                          L$CCP::
002106  005510                          .WORD   L$CLEAN
002110                          L$ACP::
002110  005424                          .WORD   L$AUTO
002112                          L$PRT::
002112  004714                          .WORD   L$PROT
002114                          L$TEST::
002114  000000                          .WORD   0
002116                          L$DLY::
002116  000000                          .WORD   0
002120                          L$HIME::
002120  000000                          .WORD   0
72
```

```
      1                                 .SBTTL  DISPATCH TABLE
      2
      3                                 ;++
      4                                 ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
      5                                 ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
      6                                 ;--
      7
      8 002122                                  DISPATCH 13
        002122  000015                                                          .WORD   13
        002124                                                          L$DISPATCH::
        002124  005640                                                  .WORD   T1
        002126  006140                                                  .WORD   T2
        002130  006452                                                  .WORD   T3
        002132  006606                                                  .WORD   T4
        002134  007200                                                  .WORD   T5
        002136  007632                                                  .WORD   T6
        002140  010164                                                  .WORD   T7
        002142  010520                                                  .WORD   T8
        002144  010714                                                  .WORD   T9
        002146  011316                                                  .WORD   T10
        002150  011424                                                  .WORD   T11
        002152  011610                                                  .WORD   T12
        002154  012030                                                  .WORD   T13
      9
```

```
                                .SBTTL  DEFAULT HARDWARE P-TABLE
                                ;++
                                ; 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.
                                ;--

002156                          BGNHW   DFPTBL
002156  000001                                                 .WORD   L10000-L$HW/2
002160                                                      L$HW::
002160                                                      DFPTBL::

002160  174000                  .WORD   174000          ; VTV31-K CSR ADDRESS

002162                          ENDHW
002162                                                      L10000:
```

```
                                .SBTTL  SOFTWARE P-TABLE
                                ;++
                                ; THE SOFTWARE 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.
                                ;--

002162                                  BGNSW   SFPTBL
002162  000002                                                                 .WORD   L10001-L$SW/2
002164                                                                  L$SW::
002164                                                                  SFPTBL::

002164  000000                  MANVEN::      .WORD   0       ; TEST 13 MANUAL INTERVENTION FLAG
                                                              ; 0 FOR NO INTERVENTION (DEFAULT)
                                                              ; 1 FOR RUN MANUALLY

002166  000005                  MAXERR::      .WORD   5       ; AUTODROP ERROR COUNT

002170                                  ENDSW
002170                                                                  L10001:

002170                                  ENDMOD
```

```
     12                                 .TITLE GLOBAL AREAS
     13                                 .SBTTL  GLOBAL EQUATES SECTION
     41
     51
     52 002170                                  BGNMOD
     53
     54                                 ;++
     55                                 ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
     56                                 ; ARE USED IN MORE THAN ONE TEST.
     57                                 ;--
     58
     73
     74 002170                                  EQUALS
                                        ;
                                        ; BIT DIFINITIONS
                                        ;
                100000                  BIT15== 100000
                040000                  BIT14== 40000
                020000                  BIT13== 20000
                010000                  BIT12== 10000
                004000                  BIT11== 4000
                002000                  BIT10== 2000
                001000                  BIT09== 1000
                000400                  BIT08== 400
                000200                  BIT07== 200
                000100                  BIT06== 100
                000040                  BIT05== 40
                000020                  BIT04== 20
                000010                  BIT03== 10
                000004                  BIT02== 4
                000002                  BIT01== 2
                000001                  BIT00== 1
                                        ;
                001000                  BIT9==  BIT09
                000400                  BIT8==  BIT08
                000200                  BIT7==  BIT07
                000100                  BIT6==  BIT06
                000040                  BIT5==  BIT05
                000020                  BIT4==  BIT04
                000010                  BIT3==  BIT03
                000004                  BIT2==  BIT02
                000002                  BIT1==  BIT01
                000001                  BIT0==  BIT00
                                        ;
                                        ; EVENT FLAG DEFINITIONS
                                        ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                        ;
                000040                  EF.START==      32.                     ; START COMMAND WAS ISSUED
                000037                  EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
                000036                  EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
                000035                  EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
                000034                  EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
                                        ;
                                        ;
                                        ; PRIORITY LEVEL DEFINITIONS
                                        ;
                000340                  PRI07== 340
```

```
GLOBAL AREAS    MACRO V03.01 18-NOV-81 11:10:45 PAGE 8-1
GLOBAL EQUATES SECTION

                000300                  PRI06== 300
                000240                  PRI05== 240
                000200                  PRI04== 200
                000140                  PRI03== 140
                000100                  PRI02== 100
                000040                  PRI01== 40
                000000                  PRI00== 0
                                        ;
                                        ;OPERATOR FLAG BITS
                                        ;
                000004                  EVL==         4
                000010                  LOT==        10
                000020                  ADR==        20
                000040                  IDU==        40
                000100                  ISR==       100
                000200                  UAM==       200
                000400                  BOE==       400
                001000                  PNT==      1000
                002000                  PRI==      2000
                004000                  IXE==      4000
                010000                  IBE==     10000
                020000                  IER==     20000
                040000                  LOE==     40000
                100000                  HOE==    100000
```

GLOBAL DATA SECTION

```
                                .SBTTL  GLOBAL DATA SECTION

                                ;++
                                ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
                                ; IN MORE THAN ONE TEST.
                                ;--


                                ; UUT REGISTER ADDRESS STORAGE

002170  000000                  UUT::          .WORD   0       ; CURRENT UNIT UNDER TEST
002172  000000                  VTVCSR::       .WORD   0       ; CSR ADDRESS
002174  000000                  DBUFF::        .WORD   0       ; DBUFF ADDRESS

002176  000000                  ERRCNT::       .WORD   0       ; ERROR COUNT FOR CURRENT UNIT
002200  000000                  DOT::   .WORD   0              ; DOT POSITION FOR ERROR REPORTS

002202  000000                  NXMFLG::.WORD   0              ; SET IF NXM ERROR OCCURS

002204  000000                  ABITS:: .WORD   0              ; BIT MASK STORE FOR ADDRESS TESTS

002206  000000                  COUNT:: .WORD   0              ; COUNTER FOR WAIT25 SUBROUTINE
```

```
                                .SBTTL  GLOBAL TEXT SECTION

                                ;++
                                ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
                                ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
                                ; MORE THAN ONE TEST.
                                ;--
                                ;
                                ; NAMES OF DEVICES SUPPORTED BY PROGRAM
                                ;
002210                                  DEVTYP  <VTV31-K>
002210                                                                  L$DVTYP::
002210  126     124     126                                                     .ASCIZ  /VTV31-K/
002213  063     061     055
002216  113     000
                                                                                .EVEN

                                ; TEST DESCRIPTION
                                ;
002220                                  DESCRIPT        <VTV31-K INTERFACE DIAGNOSTIC>
002220                                                                  L$DESC::
002220  126     124     126                                                     .ASCIZ  /VTV31-K INTERFACE D
002223  063     061     055
002226  113     040     111
002231  116     124     105
002234  122     106     101
002237  103     105     040
002242  104     111     101
002245  107     116     117
002250  123     124     111
002253  103     000
                                                                                .EVEN
                                .EVEN
```

```
                                .SBTTL  GLOBAL SUBROUTINES
                                ;*****************************************************************
                                ;                                                               *
                                ; THE GLOBAL SUBROUTINES ARE CALLED BY MORE THAN ONE TEST.      *
                                ;                                                               *
                                ;*****************************************************************
                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;
                                ;       SUBROUTINE TO UPDATE UNIT ERROR COUNT. IF DURING ONE PASS, THE
                                ;       ERROR COUNT EXCEEDS MAXERR (THE AUTODROP COUNT REQUESTED AT
                                ;       STARTUP), THE UNIT IS DROPPED FROM THE TEST CYCLE.
                                ;       IF THE PROGRAM IS LOOPING ON AN ERROR OR IF THE FLAG IDU HAS
                                ;       BEEN SELECTED, THE SUBROUTINE DOES NOTHING.
                                ;
                                ; CALLING SEQUENCE:
                                ;
                                ;       JSR PC,CHKMAX
                                ;
                                ; INPUTS:
                                ;
                                ;       NONE.
                                ;
                                ; IMPLICIT INPUTS:
                                ;
                                ;       THE VARIABLE MAXERR CONTAINS THE NUMBER OF ERRORS AFTER WHICH
                                ;       TO DROP THE UNIT.
                                ;       UUT CONTAINS THE NUMBER OF THE UNIT CURRENTLY BEING TESTED.
                                ;
                                ; OUTPUTS:
                                ;
                                ;       NONE.
                                ;
                                ; IMPLICIT OUTPUTS:
                                ;
                                ;       THE SUBPASS ERROR COUNT (ERRCNT) IS INCREMENTED.
                                ;
                                ; SUBORDINATE ROUTINES USED:
                                ;
                                ;       NONE.
                                ;
                                ; FUNCTIONAL SIDE EFFECTS:
                                ;
                                ;       NONE.
                                ;
                                ;--
002256                          CHKMAX::INLOOP                  ; LOOPING ON ERROR?
002256  104420                                                                  TRAP    C$INLP
                                        BCOMPLETE 1$            ; IF YES, EXIT
002260                                                                          BCS     1$
002260  103432
                                        RFLAGS  R0              ; GET OPERATOR FLAGS
002262                                                                          TRAP    C$RFLA
002262  104421
002264  032700  000040                  BIT     #IDU,R0         ; IS DROPPING INHIBITED?
002270  001026                          BNE     1$              ; IF YES, EXIT
```

```
   002272  005267  177700                   INC     ERRCNT            ; UPDATE THE ERROR COUNT
   002276  026767  177674  177662           CMP     ERRCNT,MAXERR     ; TOO MANY ERRORS?
   002304  003420                           BLE     1$                ; IF NOT, JUMP
   002306                                   PRINTF  #NERRS,MAXERR,UUT ; 'TOO MANY ERRORS'
   002306  016746  177656                                                                    MOV     UUT,-(SP)
   002312  016746  177650                                                                    MOV     MAXERR,-(SP)
   002316  012746  002350                                                                    MOV     #NERRS,-(SP)
   002322  012746  000003                                                                    MOV     #3,-(SP)
   002326  010600                                                                            MOV     SP,R0
   002330  104417                                                                            TRAP    C$PNTF
   002332  062706  000010                                                                    ADD     #10,SP
   002336                                   DODU    UUT               ; DROP THE UNIT
   002336  016700  177626                                                                    MOV     UUT,R0
   002342  104451                                                                            TRAP    C$DODU

   002344                                   DOCLN                     ; END THE SUBPASS
   002344  104444                                                                            TRAP    C$DCLN

   002346  000207                   1$:     RTS     PC

                                            .NLIST  BEX
   002350     045     116     045   NERRS:  .ASCIZ  /%N%AMORE THAN%D3%A ERRORS ON UNIT%D2/
                                            .LIST   BEX
                                            .EVEN
```

```
                        ;++
                        ; FUNCTIONAL DESCRIPTION:
                        ;
                        ;       SUBROUTINE TO WAIT FOR 25 MILLISECONDS
                        ;
                        ; CALLING SEQUENCE:
                        ;
                        ;       JSR     PC,WAIT25
                        ;
                        ; INPUTS:
                        ;
                        ;       NONE.
                        ;
                        ; IMPLICIT INPUTS:
                        ;
                        ;       THE VARIABLE COUNT MUST HAVE BEEN SET UP BY ROUTINE SETCLK.
                        ;
                        ; OUTPUTS:
                        ;
                        ;       NONE.
                        ;
                        ; IMPLICIT OUTPUTS:
                        ;
                        ;       NONE.
                        ;
                        ;--
                        ;
002416  016700  177564  WAIT25::MOV      COUNT,R0        ; GET WAIT COUNT
002422  005300          1$:     DEC      R0              ; ALL DONE?
002424  001376                  BNE      1$              ; IF NOT, WAIT SOME MORE
002426  000207                  RTS      PC              ; ELSE RETURN
```

```
                                .SBTTL  GLOBAL INTERRUPT SERVICE ROUTINES
                                ;*****************************************************************
                                ;                                                               *
                                ; THESE ROUTINES ARE USED IN MORE THAN ONE PLACE IN THE PROGRAM. *
                                ;                                                               *
                                ;*****************************************************************
                                BGNSRV  NXMTRP
002430                                                                          NXMTRP::
002430
002430  012737  000001  002202          MOV     #1,@#NXMFLG     ; FLAG UNIT NXM ERROR
002436  012702  000001                  MOV     #1,R2           ; FLAG REGISTER NXM ERROR
002442                          ENDSRV
002442                                                                          L10002:
002442  000002                                                                          RTI
```

GLOBAL ERROR REPORT SECTION

```
                                .SBTTL  GLOBAL ERROR REPORT SECTION

                                ;++
                                ; THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
                                ; USED BY MORE THAN TEST TO OUTPUT ADDITIONAL ERROR INFORMATION.  PRINTB
                                ; (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
                                ;--


002444                          BGNMSG  ER200
002444                                                                  ER200::
                                        PRINTB  #E200B,R1       ; PRINT CSR CONTENTS
002444  010146                                                            MOV     R1,-(SP)
002446  012746  003636                                                     MOV     #E200B,-(SP)
002452  012746  000002                                                     MOV     #2,-(SP)
002456  010600                                                            MOV     SP,R0
002460  104414                                                            TRAP    C$PNTB
002462  062706  000006                                                     ADD     #6,SP
002466  004767  177564                JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
002472                          ENDMSG
002472                                                                  L10003:
002472  104423                                                            TRAP    C$MSG

002474                          BGNMSG  ER300
002474                                                                  ER300::
                                        PRINTB  #E300B,R1,R3    ; PRINT REGISTER CONTENTS
002474  010346                                                            MOV     R3,-(SP)
002476  010146                                                            MOV     R1,-(SP)
002500  012746  003663                                                     MOV     #E300B,-(SP)
002504  012746  000003                                                     MOV     #3,-(SP)
002510  010600                                                            MOV     SP,R0
002512  104414                                                            TRAP    C$PNTB
002514  062706  000010                                                     ADD     #10,SP
002520                                  PRINTB  #E300C,R2       ;
002520  010246                                                            MOV     R2,-(SP)
002522  012746  003776                                                     MOV     #E300C,-(SP)
002526  012746  000002                                                     MOV     #2,-(SP)
002532  010600                                                            MOV     SP,R0
002534  104414                                                            TRAP    C$PNTB
002536  062706  000006                                                     ADD     #6,SP
002542  004767  177510                JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
002546                          ENDMSG
002546                                                                  L10004:
002546  104423                                                            TRAP    C$MSG

002550                          BGNMSG  ER400
002550                                                                  ER400::
                                        PRINTB  #E400B,R1,R4    ; PRINT GOOD AND BAD
002550  010446                                                            MOV     R4,-(SP)
002552  010146                                                            MOV     R1,-(SP)
002554  012746  004036                                                     MOV     #E400B,-(SP)
002560  012746  000003                                                     MOV     #3,-(SP)
002564  010600                                                            MOV     SP,R0
002566  104414                                                            TRAP    C$PNTB
002570  062706  000010                                                     ADD     #10,SP
002574  004767  177456                JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
```

```
     40 002600                          ENDMSG
        002600                                                                  L10005:
        002600  104423                                                                    TRAP    C$MSG
     41
     42 002602                          BGNMSG  ER401
        002602                                                                  ER401::
     43 002602                                  PRINTB  #E401B,R3,R5    ; PRINT GOOD AND BAD
        002602  010546                                                                    MOV     R5,-(SP)
        002604  010346                                                                    MOV     R3,-(SP)
        002606  012746  004152                                                            MOV     #E401B,-(SP)
        002612  012746  000003                                                            MOV     #3,-(SP)
        002616  010600                                                                    MOV     SP,R0
        002620  104414                                                                    TRAP    C$PNTB
        002622  062706  000010                                                            ADD     #10,SP
     44 002626  004767  177424                  JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
     45 002632                          ENDMSG
        002632                                                                  L10006:
        002632  104423                                                                    TRAP    C$MSG
     46
     47 002634                          BGNMSG  ER500
        002634                                                                  ER500::
     48 002634  000303                          SWAB    R3              ; GET LINE NUMBER
     49 002636  042703  177400                  BIC     #177400,R3      ; CLEAR OTHER BITS
     50 002642                                  PRINTB  #LINDOT,R3,R3,R1,R1 ; LINE AND DOT NUMBER
        002642  010146                                                                    MOV     R1,-(SP)
        002644  010146                                                                    MOV     R1,-(SP)
        002646  010346                                                                    MOV     R3,-(SP)
        002650  010346                                                                    MOV     R3,-(SP)
        002652  012746  004503                                                            MOV     #LINDOT,-(SP)
        002656  012746  000005                                                            MOV     #5,-(SP)
        002662  010600                                                                    MOV     SP,R0
        002664  104414                                                                    TRAP    C$PNTB
        002666  062706  000014                                                            ADD     #14,SP
     51 002672                                  PRINTX  #GOOBAD,R2,R3   ; GOOD AND BAD
        002672  010346                                                                    MOV     R3,-(SP)
        002674  010246                                                                    MOV     R2,-(SP)
        002676  012746  004571                                                            MOV     #GOOBAD,-(SP)
        002702  012746  000003                                                            MOV     #3,-(SP)
        002706  010600                                                                    MOV     SP,R0
        002710  104415                                                                    TRAP    C$PNTX
        002712  062706  000010                                                            ADD     #10,SP
     52 002716  004767  177334                  JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
     53 002722                          ENDMSG
        002722                                                                  L10007:
        002722  104423                                                                    TRAP    C$MSG
     54
     55 002724                          BGNMSG  ER501
        002724                                                                  ER501::
     56 002724  000304                          SWAB    R4              ; GET THE LINE NUMBER
     57 002726  042704  177400                  BIC     #177400,R4      ; CLEAR THE OTHER BITS
     58 002732  010137  002200                  MOV     R1,@#DOT        ; GET THE DATA PATTERN
     59 002736  042737  177000  002200          BIC     #177000,@#DOT   ; GET THE DOT NUMBER
     60 002744                                  PRINTB  #LINDOT,R4,R4,DOT,DOT ; LINE AND DOT NUMBER
        002744  016746  177230                                                            MOV     DOT,-(SP)
        002750  016746  177224                                                            MOV     DOT,-(SP)
        002754  010446                                                                    MOV     R4,-(SP)
        002756  010446                                                                    MOV     R4,-(SP)
```

```
                002760  012746  004503                                                  MOV     #LINDOT,-(SP)
                002764  012746  000005                                                  MOV     #5,-(SP)
                002770  010600                                                          MOV     SP,R0
                002772  104414                                                          TRAP    C$PNTB
                002774  062706  000014                                                  ADD     #14,SP
            61  003000                          PRINTX  #GOOBAD,R2,R3   ; GOOD AND BAD
                003000  010346                                                          MOV     R3,-(SP)
                003002  010246                                                          MOV     R2,-(SP)
                003004  012746  004571                                                  MOV     #GOOBAD,-(SP)
                003010  012746  000003                                                  MOV     #3,-(SP)
                003014  010600                                                          MOV     SP,R0
                003016  104415                                                          TRAP    C$PNTX
                003020  062706  000010                                                  ADD     #10,SP
            62  003024  004767  177226                JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
            63  003030                      ENDMSG
                003030                                                          L10010:
                003030  104423                                                          TRAP    C$MSG
            64
            65  003032                      BGNMSG  ER600
                003032                                                          ER600::
            66  003032  000305                        SWAB    R5              ; GET THE LINE NUMBER
            67  003034  042705  177400                BIC     #177400,R5      ; CLEAR OTHER BITS
            68  003040  010137  002200                MOV     R1,@#DOT        ; GET THE DOT NUMBER
            69  003044  042737  177000  002200        BIC     #177000,@#DOT   ; CLEAR THE OTHER BITS
            70  003052                              PRINTB  #LINDOT,R5,R5,DOT,DOT ; LINE AND DOT NUMBER
                003052  016746  177122                                                  MOV     DOT,-(SP)
                003056  016746  177116                                                  MOV     DOT,-(SP)
                003062  010546                                                          MOV     R5,-(SP)
                003064  010546                                                          MOV     R5,-(SP)
                003066  012746  004503                                                  MOV     #LINDOT,-(SP)
                003072  012746  000005                                                  MOV     #5,-(SP)
                003076  010600                                                          MOV     SP,R0
                003100  104414                                                          TRAP    C$PNTB
                003102  062706  000014                                                  ADD     #14,SP
            71  003106                          PRINTX  #GOOBAD,R4,R3   ; GOOD AND BAD
                003106  010346                                                          MOV     R3,-(SP)
                003110  010446                                                          MOV     R4,-(SP)
                003112  012746  004571                                                  MOV     #GOOBAD,-(SP)
                003116  012746  000003                                                  MOV     #3,-(SP)
                003122  010600                                                          MOV     SP,R0
                003124  104415                                                          TRAP    C$PNTX
                003126  062706  000010                                                  ADD     #10,SP
            72  003132  016767  004754  177044        MOV     NCOLS,ABITS     ; GET NO. OF COLUMNS PER COLOUR
            73  003140  006367  177040                ASL     ABITS           ; SHIFT TO FORM
            74  003144  006367  177034                ASL     ABITS           ; BIT MASK OF
            75  003150  006367  177030                ASL     ABITS           ; ADDRESS LINES
            76  003154  166767  004732  177022        SUB     NCOLS,ABITS     ; BEING TESTED
            77  003162                              PRINTX  #EADD,ABITS     ; AND ADDRESS LINES BEING TESTED
                003162  016746  177016                                                  MOV     ABITS,-(SP)
                003166  012746  004652                                                  MOV     #EADD,-(SP)
                003172  012746  000002                                                  MOV     #2,-(SP)
                003176  010600                                                          MOV     SP,R0
                003200  104415                                                          TRAP    C$PNTX
                003202  062706  000006                                                  ADD     #6,SP
            78  003206  004767  177044                JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
            79  003212                      ENDMSG
                003212                                                          L10011:
```

```
003212  104423                                                                          TRAP    C$MSG

003214                            BGNMSG  ER700
003214                                                                          ER700::
003214  000305                            SWAB    R5              ; GET THE LINE NUMBER
003216  042705  177400                    BIC     #177400,R5      ; CLEAR OTHER BITS
003222  010137  002200                    MOV     R1,@#DOT        ; GET THE DOT NUMBER
003226  042737  177000  002200            BIC     #177000,@#DOT   ; CLEAR THE OTHER BITS
003234                                    PRINTB  #LINDOT,R5,R5,DOT,DOT ; LINE AND DOT NUMBER
003234  016746  176740                                                                  MOV     DOT,-(SP)
003240  016746  176734                                                                  MOV     DOT,-(SP)
003244  010546                                                                          MOV     R5,-(SP)
003246  010546                                                                          MOV     R5,-(SP)
003250  012746  004503                                                                  MOV     #LINDOT,-(SP)
003254  012746  000005                                                                  MOV     #5,-(SP)
003260  010600                                                                          MOV     SP,R0
003262  104414                                                                          TRAP    C$PNTB
003264  062706  000014                                                                  ADD     #14,SP
003270                                    PRINTX  #GOOBAD,R4,R3   ; GOOD AND BAD
003270  010346                                                                          MOV     R3,-(SP)
003272  010446                                                                          MOV     R4,-(SP)
003274  012746  004571                                                                  MOV     #GOOBAD,-(SP)
003300  012746  000003                                                                  MOV     #3,-(SP)
003304  010600                                                                          MOV     SP,R0
003306  104415                                                                          TRAP    C$PNTX
003310  062706  000010                                                                  ADD     #10,SP
003314  016767  005130  176662            MOV     NLINS,ABITS     ; GET NO. OF LINES PER COLOUR
003322  006367  176656                    ASL     ABITS           ; SHIFT TO FORM
003326  006367  176652                    ASL     ABITS           ; MASK OF ADDRESS
003332  006367  176646                    ASL     ABITS           ; LINES BEING TESTED
003336  166767  005106  176640            SUB     NLINS,ABITS     ;
003344  042767  177400  176632            BIC     #177400,ABITS   ; CLEAR THE TOP BYTE
003352                                    PRINTX  #EADD,ABITS     ; AND ADDRESS LINES BEING TESTED
003352  016746  176626                                                                  MOV     ABITS,-(SP)
003356  012746  004652                                                                  MOV     #EADD,-(SP)
003362  012746  000002                                                                  MOV     #2,-(SP)
003366  010600                                                                          MOV     SP,R0
003370  104415                                                                          TRAP    C$PNTX
003372  062706  000006                                                                  ADD     #6,SP
003376  004767  176654                    JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
003402                            ENDMSG
003402                                                                          L10012:
003402  104423                                                                          TRAP    C$MSG

003404                            BGNMSG  ER800
003404                                                                          ER800::
003404  010167  176570                    MOV     R1,DOT          ; GET THE DOT NUMBER
003410  042767  177000  176562            BIC     #177000,DOT     ; CLEAR THE OTHER BITS
003416                                    PRINTB  #E800B,R3,DOT   ; DOT ADDRESS,GOOD,BAD
003416  016746  176556                                                                  MOV     DOT,-(SP)
003422  010346                                                                          MOV     R3,-(SP)
003424  012746  004257                                                                  MOV     #E800B,-(SP)
003430  012746  000003                                                                  MOV     #3,-(SP)
003434  010600                                                                          MOV     SP,R0
003436  104414                                                                          TRAP    C$PNTB
003440  062706  000010                                                                  ADD     #10,SP
003444                                    PRINTB  #E800C,R3,R4    ;
```

```
                003444  010446                                                                  MOV     R4,-(SP)
                003446  010346                                                                  MOV     R3,-(SP)
                003450  012746  004344                                                          MOV     #E800C,-(SP)
                003454  012746  000003                                                          MOV     #3,-(SP)
                003460  010600                                                                  MOV     SP,R0
                003462  104414                                                                  TRAP    C$PNTB
                003464  062706  000010                                                          ADD     #10,SP
    103         003470  004767  176562                JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
    104         003474                      ENDMSG
                003474                                                                  L10013:
                003474  104423                                                                  TRAP    C$MSG
    105
    106         003476                      BGNMSG  ER900
                003476                                                                  ER900::
    107         003476                              PRINTB  #E900B,R3,R2    ; CSR CONTENTS, PRESET COLOUR
                003476  010246                                                                  MOV     R2,-(SP)
                003500  010346                                                                  MOV     R3,-(SP)
                003502  012746  004427                                                          MOV     #E900B,-(SP)
                003506  012746  000003                                                          MOV     #3,-(SP)
                003512  010600                                                                  MOV     SP,R0
                003514  104414                                                                  TRAP    C$PNTB
                003516  062706  000010                                                          ADD     #10,SP
    108         003522  004767  176530                JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
    109         003526                      ENDMSG
                003526                                                                  L10014:
                003526  104423                                                                  TRAP    C$MSG
    110
    111         003530                      BGNMSG  ER902
                003530                                                                  ER902::
    112         003530  000305                              SWAB    R5              ; GET THE LINE NUMBER
    113         003532  042705  177400                      BIC     #177400,R5      ; CLEAR THE OTHER BITS
    114         003536  010337  002200                      MOV     R3,@#DOT        ; GET THE DOT NUMBER
    115         003542  042737  177000  002200            BIC     #177000,@#DOT   ; CLEAR THE OTHER BITS
    116         003550                              PRINTB  #LINDOT,R5,R5,DOT,DOT ; LINE AND DOT NUMBER
                003550  016746  176424                                                          MOV     DOT,-(SP)
                003554  016746  176420                                                          MOV     DOT,-(SP)
                003560  010546                                                                  MOV     R5,-(SP)
                003562  010546                                                                  MOV     R5,-(SP)
                003564  012746  004503                                                          MOV     #LINDOT,-(SP)
                003570  012746  000005                                                          MOV     #5,-(SP)
                003574  010600                                                                  MOV     SP,R0
                003576  104414                                                                  TRAP    C$PNTB
                003600  062706  000014                                                          ADD     #14,SP
    117         003604                              PRINTX  #GOOBAD,R2,R4   ; GOOD AND BAD
                003604  010446                                                                  MOV     R4,-(SP)
                003606  010246                                                                  MOV     R2,-(SP)
                003610  012746  004571                                                          MOV     #GOOBAD,-(SP)
                003614  012746  000003                                                          MOV     #3,-(SP)
                003620  010600                                                                  MOV     SP,R0
                003622  104415                                                                  TRAP    C$PNTX
                003624  062706  000010                                                          ADD     #10,SP
    118         003630  004767  176422                JSR     PC,CHKMAX       ; CHECK FOR TOO MANY ERRORS
    119         003634                      ENDMSG
                003634                                                                  L10015:
                003634  104423                                                                  TRAP    C$MSG
    120
    121
```

```
    122                                             .NLIST  BEX
    123 003636      045     101     103  E200B:    .ASCIZ  /%ACSR CONTENTS %O6%N/
    124 003663      045     101     117  E300B:    .ASCII  /%AOLD CSR CONTENTS %O6/
    125 003711      045     116     045            .ASCIZ  /%N%ANEW CSR CONTENTS %O6%A (IGNORE BITS 0,1,2 AND 4)/
    126 003776      045     116     045  E300C:    .ASCIZ  /%N%ADATA WRITTEN TO DBUFF %O6%N/
    127 004036      045     101     101  E400B:    .ASCII  /%AATTEMPTED SETTING %O6/
    128 004065      045     116     045            .ASCIZ  /%N%AACTUAL SETTING %O6%A (IGNORE BITS 0-4 AND 6-7)%N/
    129 004152      045     101     101  E401B:    .ASCII  /%AATTEMPTED SETTING %O6/
    130 004201      045     116     045            .ASCIZ  /%N%AACTUAL SETTING %O6%A (IGNORE BITS 3-15)%N/
    131 004257      045     101     104  E800B:    .ASCIZ  /%ADOT ADDRESS WRITTEN %O1%A, FAILING DOT ADDRESS %O1/
    132 004344      045     116     045  E800C:    .ASCIZ  /%N%AEXPECTED CONTENTS %O1%A, ACTUAL CONTENTS %O1%N/
    133 004427      045     101     103  E900B:    .ASCIZ  /%ACSR CONTENTS %O6%A, EXPECTED COLOUR %O1%N/
    134 004503      045     101     114  LINDOT:   .ASCIZ  /%ALINE %D3%A (%O3%A OCTAL), DOT %D3%A (%O3%A OCTAL)%N/
    135 004571      045     101     105  GOOBAD:   .ASCIZ  /%AEXPECTED CONTENTS %O1%A, ACTUAL CONTENTS %O1%N/
    136 004652      045     101     101  EADD:     .ASCIZ  /%AADDRESS BITS BEING TESTED %B9%N/
    137                                             .LIST   BEX
    138                                             .EVEN
    139 004714                                      ENDMOD
```

```
                                .TITLE MISCELLANEOUS SECTIONS
                                .SBTTL  REPORT CODING SECTION

004714                                  BGNMOD
                                .SBTTL  PROTECTION TABLE

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

004714                                  BGNPROT
004714                                                                  L$PROT::

004714  000000                          0               ;OFFSET INTO P-TABLE FOR CSR ADDRESS
004716  177777                          -1              ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
004720  177777                          -1              ;OFFSET INTO P-TABLE FOR DRIVE NUMBER

004722                                  ENDPROT
```

```
                        .SBTTL  INITIALIZE SECTION

                        ;++
                        ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                        ; AT THE BEGINNING OF EACH PASS.
                        ;--

 004722                         BGNINIT
 004722                                                                  L$INIT::

                        READEF  #EF.START       ; START COMMAND?
 004722 012700 000040                                                   MOV     #EF.START,R0
 004726 104447                                                          TRAP    C$REFG
                        BNCOMPLETE CONT         ; IF NOT, BRANCH
 004730 103002                                                          BCC     CONT
                        JSR     PC,SETCLK       ; SET UP CLOCK COUNTER
                CONT:   READEF  #EF.CONTINUE    ; CONTINUE COMMAND?
 004736 012700 000036                                                   MOV     #EF.CONTINUE,R0
 004742 104447                                                          TRAP    C$REFG
                        BCOMPLETE END           ; IF YES, NO P TABLE
 004744 103441                                                          BCS     END
                        READEF #EF.NEW          ; NEW PASS?
 004746 012700 000035                                                   MOV     #EF.NEW,R0
 004752 104447                                                          TRAP    C$REFG
                        BNCOMPLETE NEXT         ; IF NOT, SKIP SETUP
 004754 103003                                                          BCC     NEXT
 004756 012767 177777 175204    SETUP:  MOV     #-1,UUT         ; INITIALISE LOGICAL UNIT NUMBER
 004764 005267 175200           NEXT:   INC     UUT             ; POINT TO NEXT LOGICAL UNIT
 004770 026767 175174 175014            CMP     UUT,L$UNIT      ; ALL DONE?
 004776 001421                          BEQ     ABORT           ; IF YES, END OF PASS
                                        GPHARD  UUT,R2          ; ELSE GET P TABLE
 005000 016700 175164                                                   MOV     UUT,R0
 005004 104442                                                          TRAP    C$GPHRD
 005006 010002                                                          MOV     R0,R2
                        BNCOMPLETE NEXT         ; IF NOT AVAILABLE, GET NEXT
 005010 103365                                                          BCC     NEXT
 005012 011267 175154                   MOV     @R2,VTVCSR      ; SAVE NEW CSR ADDRESS
 005016 016767 175150 175150            MOV     VTVCSR,DBUFF    ; AND DBUFF ADDRESS
 005024 062767 000002 175142            ADD     #2,DBUFF        ;
 005032 005067 175140                   CLR     ERRCNT          ; CLEAR SUBPASS ERROR COUNT
                                        EXIT    INIT            ; AND EXIT
 005036 104432                                                          TRAP    C$EXIT
 005040 000010                                                          .WORD   L10017-.

 005042                         ABORT:  DOCLN                   ; CLEAN UP AND ABORT THE PASS
 005042 104444                                                          TRAP    C$DCLN
                                EXIT    INIT            ; AND EXIT
 005044 104432                                                          TRAP    C$EXIT
 005046 000002                                                          .WORD   L10017-.

                        .EVEN

 005050                         END:    ENDINIT
 005050                                                                 L10017:
 005050 104411                                                          TRAP    C$INIT
```

```
                                        ;++
                                        ; THIS ROUTINE SETS UP A DELAY VARIABLE CALLED COUNT. THIS GIVES A
                                        ; DELAY OF APPROXIMATELY 25 MILLISECONDS IF USED AS FOLLOWS:-
                                        ;
                                        ;       MOV     COUNT,R0
                                        ;   1$: DEC     R0
                                        ;       BNE     1$
                                        ;
                                        ; COUNT IS DERIVED FROM THE L CLOCK IF THERE IS ONE. OTHERWISE, THE
                                        ; OPERATOR IS ASKED TO TYPE 2 CHARACTERS ON THE CONSOLE 6 SECONDS
                                        ; APART.
                                        ;--

005052  005004                  SETCLK: CLR     R4              ; CLEAR THE COUNTERS
005054  005005                          CLR     R5              ;
005056                                  GETPRI  R2              ; SAVE CURRENT PRIORITY IN R2
005056  104440                                                                  TRAP    C$GPRI
005060  010002                                                                  MOV     R0,R2
005062                                  CLOCK   L,R1            ; IS THERE AN L CLOCK?
005062  012700  000114                                                          MOV     #'L,R0
005066  104462                                                                  TRAP    C$CLCK
005070  010001                                                                  MOV     R0,R1
005072  103052                          BCC     NOCLOK          ; IF NOT, BRANCH
                                        ;
                                        ; USE THE L CLOCK
                                        ;
005074                          LCLOCK: SETVEC  4(R1),#KLINT,#340 ; SET UP THE CLOCK VECTOR
005074  012746  000340                                                          MOV     #340,-(SP)
005100  012746  005162                                                          MOV     #KLINT,-(SP)
005104  016146  000004                                                          MOV     4(R1),-(SP)
005110  012746  000003                                                          MOV     #3,-(SP)
005114  104437                                                                  TRAP    C$SVEC
005116  062706  000010                                                          ADD     #10,SP
005122  012703  000005                  MOV     #5,R3           ; IF 50 HZ, 100 MS = 5 INTERRUPTS
005126  026127  000006  000062          CMP     6(R1),#50.      ; 50 HZ CORRECT?
005134  001401                          BEQ     10$             ; IF YES, BRANCH
005136  005203                          INC     R3              ; ELSE ALLOW 6 INTERRUPTS

005140                          10$:    SETPRI  #0              ; NOW DROP THE PRIORITY
005140  012700  000000                                                          MOV     #0,R0
005144  104441                                                                  TRAP    C$SPRI
005146  012771  000100  000000          MOV     #100,@(R1)      ; ALLOW INTERRUPTS

005154  005204                  30$:    INC     R4              ; COUNT THE DELAY
005156  001376                          BNE     30$             ;
005160  000417                          BR      NOCLOK          ; IF TOO LONG, ASSUME NO CLOCK

005162  005303                  KLINT:  DEC     R3              ; 5 OR 6 INTERRUPTS?
005164  100401                          BMI     40$             ; IF YES, TIDY UP
005166  000002                          RTI                     ; ELSE KEEP COUNTING

005170                          40$:    SETPRI  R2              ; RESTORE THE PRIORITY
005170  010200                                                                  MOV     R2,R0
005172  104441                                                                  TRAP    C$SPRI
005174  022626                          CMP     (SP)+,(SP)+     ; TIDY UP THE STACK
005176  005071  000000                  CLR     @(R1)           ; DISABLE CLOCK INTERRUPTS
```

```
     43 005202  006204                        ASR     R4              ; DIVIDE THE 100 MILLISECOND COUNTER
     44 005204  006204                        ASR     R4              ; BY 4 TO GIVE 25 MILLISECONDS
     45 005206  042704  140000                BIC     #140000,R4      ; ENSURE TOP BITS ARE CLEAR
     46 005212  010467  174770                MOV     R4,COUNT        ; AND SAVE THE COUNTER
     47 005216  000207                        RTS     PC              ; RETURN
     48
     49                                ;
     50                                ; USE THE CONSOLE FOR TIMING
     51                                ;
     52          177560                        TKS=177560              ; KEYBOARD STATUS REGISTER
     53          177562                        TKB=177562              ; KEYBOARD DATA BUFFER
     54          177564                        TPS=177564              ; PRINTER STATUS REGISTER
     55          177566                        TPB=177566              ; PRINTER DATA BUFFER
     56
     57 005220                        NOCLOK: SETVEC  #60,#TTINT,#340 ; SET UP INTERRUPT VECTOR
        005220  012746  000340                                                  MOV     #340,-(SP)
        005224  012746  005326                                                  MOV     #TTINT,-(SP)
        005230  012746  000060                                                  MOV     #60,-(SP)
        005234  012746  000003                                                  MOV     #3,-(SP)
        005240  104437                                                          TRAP    C$SVEC
        005242  062706  000010                                                  ADD     #10,SP
     58 005246                                PRINTF  #TIMMSG         ; 'TYPE 2 CHARACTERS 6 SECONDS APART'
        005246  012746  005354                                                  MOV     #TIMMSG,-(SP)
        005252  012746  000001                                                  MOV     #1,-(SP)
        005256  010600                                                          MOV     SP,R0
        005260  104417                                                          TRAP    C$PNTF
        005262  062706  000004                                                  ADD     #4,SP
     59
     60 005266  105767  172266        10$:    TSTB    TKS             ; IS FIRST CHARACTER READY?
     61 005272  100375                        BPL     10$             ; IF NOT, WAIT
     62
     63 005274  016767  172262  172264        MOV     TKB,TPB         ; NOW ECHO THE CHARACTER
     64 005302                                SETPRI  #0              ; DROP THE PRIORITY
        005302  012700  000000                                                  MOV     #0,R0
        005306  104441                                                          TRAP    C$SPRI
     65 005310  012767  000100  172242        MOV     #100,TKS        ; ALLOW INTERRUPTS
     66
     67 005316  105205                20$:    INCB    R5              ; START COUNTING
     68 005320  001376                        BNE     20$             ;
     69 005322  005204                        INC     R4              ; UPDATE THE COUNTER
     70 005324  000774                        BR      20$             ;
     71
     72 005326                        TTINT:  SETPRI  R2              ; RESTORE THE PRIORITY
        005326  010200                                                          MOV     R2,R0
        005330  104441                                                          TRAP    C$SPRI
     73 005332  016767  172224  172226        MOV     TKB,TPB         ; ECHO THE CHARACTER
     74 005340  022626                        CMP     (SP)+,(SP)+     ; TIDY UP THE STACK
     75 005342  005067  172212                CLR     TKS             ; DISABLE INTERRUPTS
     76
     77 005346  010467  174634                MOV     R4,COUNT        ; SAVE THE COUNTER
     78                                                                ; 6 SECONDS/256 = 25 MILLISECONDS
     79 005352  000207                        RTS     PC              ; RETURN
     80
     81                                        .NLIST  BEX
     82 005354      045     116      045 TIMMSG: .ASCIZ  /%N%ATYPE 2 CHARACTERS 6 SECONDS APART >/
     83                                        .LIST   BEX
     84                                        .EVEN
```

AUTODROP SECTION

```
                                .SBTTL  AUTODROP SECTION
                                ;++
                                ; THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF
                                ; THE "ADR" FLAG WAS SET.  THE UNIT(S) UNDER TEST ARE CHECKED TO
                                ; SEE IF THEY WILL RESPOND.  THOSE THAT DON'T ARE IMMEDIATELY
                                ; DROPPED FROM TESTING.
                                ;--

005424                                  BGNAUTO
005424                                                                  L$AUTO::

                                SETVEC  #4,#NXMTRP,#PRI07 ; SET UP NXM TRAP VECTOR
005424  012746  000340                                                  MOV     #PRI07,-(SP)
005430  012746  002430                                                  MOV     #NXMTRP,-(SP)
005434  012746  000004                                                  MOV     #4,-(SP)
005440  012746  000003                                                  MOV     #3,-(SP)
005444  104437                                                          TRAP    C$SVEC
005446  062706  000010                                                  ADD     #10,SP
005452  005037  002202                CLR     @#NXMFLG        ; INITIALISE NXM FLAG
005456  005777  174510                TST     @VTVCSR         ; TEST THE CSR ADDRESS
005462  005737  002202                TST     @#NXMFLG        ; WAS THERE A TRAP?
005466  001404                        BEQ     10$             ; IF NOT, BRANCH
005470                                  DODU    UUT             ; ELSE DROP THE UNIT
005470  016700  174474                                                  MOV     UUT,R0
005474  104451                                                          TRAP    C$DODU
005476                                  DOCLN                   ; AND TIDY UP END OF PASS
005476  104444                                                          TRAP    C$DCLN
005500                          10$:    CLRVEC  #4              ; RESTORE THE DRS NXM TRAP
005500  012700  000004                                                  MOV     #4,R0
005504  104436                                                          TRAP    C$CVEC

005506                                  ENDAUTO
005506                                                          L10020:
005506  104461                                                          TRAP    C$AUTO
```

```
                                .SBTTL  CLEANUP CODING SECTION
                                ;++
                                ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
                                ; AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED OR IF CONTROL/C IS
                                ; TYPED DURING TESTING.
                                ; A PRESET IS PERFORMED TO SET THE SCREEN TO ALL BLACK, THE OFFSET
                                ; REGISTER IS CLEARED, AND THE ERROR COUNT ZEROED.
                                ;--

005510                                  BGNCLN
005510                                                                          L$CLEAN::

005510  005737  002202                  TST     @#NXMFLG        ; HAS NXM ERROR OCCURRED?
005514  001015                          BNE     2$              ; IF YES, SKIP CLEAN UP
005516  005077  174452                  CLR     @DBUFF          ; SET DBUFF COLOUR TO BLACK
005522  012777  000150  174442          MOV     #150,@VTVCSR    ; RESET OFFSET AND DO PRESET

005530  012701  177777                  MOV     #-1,R1          ; SET UP WAIT COUNTER
005534  105777  174432          1$:     TSTB    @VTVCSR         ; PRESET FINISHED?
005540  100403                          BMI     2$              ; IF YES, BRANCH
005542                                  BREAK                   ; ALLOW OPERATOR BREAK IN
005542  104422                                                                  TRAP    C$BRK
005544  005301                          DEC     R1              ; WAITED TOO LONG?
005546  001372                          BNE     1$              ; IF NOT, WAIT LONGER
005550  005067  174422          2$:     CLR     ERRCNT          ; CLEAR SUBPASS ERROR COUNT

005554                                  EXIT    CLN
005554  104432                                                                  TRAP    C$EXIT
005556  000002                                                                  .WORD   L10021-.

                                .EVEN

005560                                  ENDCLN
005560                                                                  L10021:
005560  104412                                                                  TRAP    C$CLEAN
```

```
                                .SBTTL  DROP UNIT SECTION
                                ;++
                                ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                ; TO NO LONGER BE TESTED.
                                ;--

 005562                                 BGNDU
 005562                                                                 L$DU::
                                PRINTF  #DROPD,R0       ; 'UNIT DROPPED'
 005562  010046                                                                 MOV     R0,-(SP)
 005564  012746  005610                                                         MOV     #DROPD,-(SP)
 005570  012746  000002                                                         MOV     #2,-(SP)
 005574  010600                                                                 MOV     SP,R0
 005576  104417                                                                 TRAP    C$PNTF
 005600  062706  000006                                                         ADD     #6,SP

                                EXIT    DU
 005604  000167                                                                 .WORD   J$JMP
 005606  000026                                                                 .WORD   L10022-2-.

                                .NLIST  BEX
 005610     045     116     045  DROPD: .ASCIZ  /%N%AUNIT%D2%A DROPPED/
                                .LIST   BEX

                                .EVEN

 005636                                 ENDDU
 005636                                                                 L10022:
 005636  104453                                                                 TRAP    C$DU
 005640                                 ENDMOD
```

```
      2                                 .TITLE HARDWARE TESTS
     10
     35
     36 005640                                BGNMOD
     37
     38                                .SBTTL  TEST 1: REGISTER NXM
     39
     40                                ;++
     41                                ; CHECKS ARE MADE THAT ACCESSING THE DEVICE CSR AND DBUFF REGISTERS
     42                                ; DOES NOT CAUSE AN NXM TRAP.
     43                                ;--
     44
     45 005640                                BGNTST
        005640                                                                                  T1::
     46
     47 005640                                SETVEC  #4,#NXMTRP,#PRI07 ; SET UP NXM TRAP VECTOR
        005640  012746  000340                                                          MOV     #PRI07,-(SP)
        005644  012746  002430                                                          MOV     #NXMTRP,-(SP)
        005650  012746  000004                                                          MOV     #4,-(SP)
        005654  012746  000003                                                          MOV     #3,-(SP)
        005660  104437                                                                  TRAP    C$SVEC
        005662  062706  000010                                                          ADD     #10,SP
     48 005666  005037  002202                        CLR     @#NXMFLG        ; INITIALISE NXM ERROR FLAG
     49                                ;
     50                                ;       CHECK THE CSR
     51                                ;
     52 005672                                BGNSUB                          ;************************ BEGIN SUBTEST
        005672                                                                                  T1.1:
        005672  104402                                                                  TRAP    C$BSUB
     53 005674  005002                        CLR     R2              ; INITIALISE CSR NXM ERROR FLAG
     54 005676  005777  174270                TST     @VTVCSR         ; TEST THE ADDRESS
     55 005702  005702                        TST     R2              ; WAS THERE A TRAP?
     56 005704  001404                        BEQ     10$             ; IF NOT, BRANCH
     57 005706                                ERRDF   100,E100        ; ELSE REPORT THE ERROR
        005706  104455                                                                  TRAP    C$ERDF
        005710  000144                                                                  .WORD   100
        005712  006006                                                                  .WORD   E100
        005714  000000                                                                  .WORD   0
     58 005716                          10$:  ENDSUB                          ;************************ END SUBTEST
        005716                                                                                  L10024:
        005716  104403                                                                  TRAP    C$ESUB
     59 005720                                BGNSUB                          ;************************ BEGIN SUBTEST
        005720                                                                                  T1.2:
        005720  104402                                                                  TRAP    C$BSUB
     60 005722  005002                        CLR     R2              ; INITIALISE DBUFF NXM ERROR FLAG
     61 005724  005777  174244                TST     @DBUFF          ; TEST THE ADDRESS
     62 005730  005702                        TST     R2              ; WAS THERE A TRAP?
     63 005732  001404                        BEQ     20$             ; IF NOT, BRANCH
     64 005734                                ERRDF   101,E101        ; ELSE REPORT THE ERROR
        005734  104455                                                                  TRAP    C$ERDF
        005736  000145                                                                  .WORD   101
        005740  006054                                                                  .WORD   E101
        005742  000000                                                                  .WORD   0
     65 005744                          20$:  ENDSUB                          ;************************ END SUBTEST
        005744                                                                                  L10025:
        005744  104403                                                                  TRAP    C$ESUB
     66
```

```
     67 005746                                  CLRVEC  #4              ; RESTORE THE DRS NXM TRAP
        005746  012700  000004                                                      MOV     #4,R0
        005752  104436                                                              TRAP    C$CVEC
     68 005754  005737  002202                  TST     @#NXMFLG        ; DID EITHER REGISTER TRAP?
     69 005760  001410                          BEQ     30$             ; IF NOT, BRANCH
     70 005762                                  RFLAGS  R0              ; CHECK THE OPERATOR FLAGS
        005762  104421                                                              TRAP    C$RFLA
     71 005764  032700  000040                  BIT     #IDU,R0         ; IS DROPPING INHIBITED?
     72 005770  001004                          BNE     30$             ; IF YES, BRANCH
     73 005772                                  DODU    UUT             ; ELSE DROP THE UNIT
        005772  016700  174172                                                      MOV     UUT,R0
        005776  104451                                                              TRAP    C$DODU
     74 006000                                  DOCLN                   ; AND TIDY UP END OF PASS
        006000  104444                                                              TRAP    C$DCLN
     75 006002                          30$.    EXIT TST                ;
        006002  104432                                                              TRAP    C$EXIT
        006004  000132                                                              .WORD   L10023-.
     76
     77                                         .NLIST  BEX
     78 006006     101     104     104  E100:  .ASCIZ  /ADDRESSING THE CSR CAUSES AN NXM TRAP/
     79 006054     101     104     104  E101:  .ASCIZ  /ADDRESSING THE DBUFF REGISTER CAUSES AN NXM TRAP/
     80                                         .LIST   BEX
     81                                         .EVEN
     82 006136                                  ENDTST
        006136                                                                  L10023:
        006136  104401                                                              TRAP    C$ETST
```

```
                                .SBTTL  TEST 2: REGISTER ACCESS

                                ;++
                                ; THE DEVICE CSR IS READ AND CHECKS ARE MADE THAT THE READY BIT IS
                                ; SET, THAT BITS 3 AND 6 ARE ZERO, AND THAT THE FIELD BIT (4) IS
                                ; SET AND CLEARED BY THE DEVICE.
                                ;--


                                        BGNTST
                                                                                T2::

                                        BGNSUB                  ;********************** BEGIN SUBTEST
                                                                                T2.1:
 006140  104402                                                                TRAP    C$BSUB
 006142  017701  174024         MOV     @VTVCSR,R1      ; READ THE CSR
 006146  105701                 TSTB    R1              ; READY BIT SET?
 006150  100404                 BMI     1$              ; IF YES, BRANCH
 006152                         ERRHRD  200,E200,ER200  ; ELSE REPORT THE ERROR
 006152  104456                                                                TRAP    C$ERHRD
 006154  000310                                                                .WORD   200
 006156  006266                                                                .WORD   E200
 006160  002444                                                                .WORD   ER200

 006162  032701  000110         1$:     BIT     #110,R1         ; CHECK BITS 3 AND 6
 006166  001404                 BEQ     2$              ; IF CLEAR, JUMP
 006170                         ERRHRD  201,E201,ER200  ; ELSE REPORT THE ERROR
 006170  104456                                                                TRAP    C$ERHRD
 006172  000311                                                                .WORD   201
 006174  006314                                                                .WORD   E201
 006176  002444                                                                .WORD   ER200

 006200  005002         2$:     CLR     R2              ; ZERO WAIT COUNTER
 006202  017701  173764 3$:     MOV     @VTVCSR,R1      ; READ THE CSR
 006206  032701  000020         BIT     #20,R1          ; FIELD BIT SET?
 006212  001006                 BNE     4$              ; IF YES, BRANCH
 006214  005302                 DEC     R2              ; ELSE WAITED TOO LONG?
 006216  001371                 BNE     3$              ; IF NOT, KEEP WAITING
 006220                         ERRHRD  202,E202,ER200  ; ELSE REPORT THE ERROR
 006220  104456                                                                TRAP    C$ERHRD
 006222  000312                                                                .WORD   202
 006224  006346                                                                .WORD   E202
 006226  002444                                                                .WORD   ER200

 006230  005002         4$:     CLR     R2              ; ZERO WAIT COUNTER
 006232  017701  173734 5$:     MOV     @VTVCSR,R1      ; READ THE CSR
 006236  032701  000020         BIT     #20,R1          ; FIELD BIT CLEAR?
 006242  001406                 BEQ     6$              ; IF YES, BRANCH
 006244  005302                 DEC     R2              ; ELSE WAITED TOO LONG?
 006246  001371                 BNE     5$              ; IF NOT, KEEP WAITING
 006250                         ERRHRD  203,E203,ER200  ; ELSE REPORT THE ERROR
 006250  104456                                                                TRAP    C$ERHRD
 006252  000313                                                                .WORD   203
 006254  006405                                                                .WORD   E203
 006256  002444                                                                .WORD   ER200

 006260                 6$:     ENDSUB                  ;********************** END SUBTEST
```

```
HARDWARE TESTS  MACRO V03.01 18-NOV-81 11:10:45 PAGE 35-1
TEST 2: REGISTER ACCESS

        006260                                                                  L10027:
        006260  104403                                                                  TRAP    C$ESUB
     39 006262                                  EXIT    TST
        006262  104432                                                                  TRAP    C$EXIT
        006264  000164                                                                  .WORD   L10026-.
     40
     41                                         .NLIST  BEX
     42 006266      103     123     122  E200:  .ASCIZ  /CSR READY BIT NOT SET/
     43 006314      103     123     122  E201:  .ASCIZ  /CSR BITS 3 OR 6 NOT CLEAR/
     44 006346      103     123     122  E202:  .ASCIZ  /CSR FIELD BIT DOES NOT GET SET/
     45 006405      103     123     122  E203:  .ASCIZ  /CSR FIELD BIT DOES NOT GET CLEARED/
     46                                         .LIST   BEX
     47                                         .EVEN
     48 006450                                  ENDTST
        006450                                                                  L10026:
        006450  104401                                                                  TRAP    C$ETST
     49
```

```
                                .SBTTL  TEST 3: REGISTER ADDRESS UNIQUENESS
                                ;++
                                ; ALL BITS IN THE DBUFF REGISTER ARE SET AND CLEARED IN ALL
                                ; COMBINATIONS WITH THE EXCEPTION OF BIT 14 (READ PICTURE STORE),
                                ; WHICH IS LEFT CLEAR. FOR EACH BIT COMBINATION, THE CSR IS READ
                                ; AND CHECKED TO BE UNCHANGED. BITS 0,1,2 AND 4 ARE IGNORED IN THIS
                                ; CHECK.
                                ;--


                                        BGNTST
006452                                                                          T3::
006452
006452  017701  173514                1$:     MOV     @VTVCSR,R1      ; SAVE CSR CONTENTS
006456  042701  000027                        BIC     #27,R1          ; CLEAR BIT 4
006462  005002                        2$:     CLR     R2              ; INITIALISE DBUFF TEST DATA

006464  010277  173504                        MOV     R2,@DBUFF       ; WRITE DBUFF TEST DATA
006470  017703  173476                        MOV     @VTVCSR,R3      ; READ CSR
006474  042703  000027                        BIC     #27,R3          ; CLEAR BIT 4
006500  020301                        3$:     CMP     R3,R1           ; CSR UNCHANGED?
006502  001405                                BEQ     2$              ; IF YES, NEXT PATTERN
006504                                        ERRHRD  300,E300,ER300  ; ELSE REPORT THE ERROR
006504  104456                                                                TRAP    C$ERHRD
006506  000454                                                                .WORD   300
006510  006544                                                                .WORD   E300
006512  002474                                                                .WORD   ER300
006514  010301                                MOV     R3,R1           ; SAVE NEW CSR CONTENTS

006516  005202                        4$:     INC     R2              ; NEXT DATA PATTERN
006520  020227  040000                        CMP     R2,#40000       ; BIT 14 SET?
006524  001002                                BNE     5$              ; IF NOT, JUMP
006526  006302                                ASL     R2              ; SKIP SOME PATTERNS
006530  000755                                BR      1$              ; NEXT PATTERN

006532  020227  140000                5$:     CMP     R2,#140000      ; ALL DONE?
006536  001352                                BNE     1$              ; IF NOT  NEXT PATTERN
006540                                        EXIT    TST
006540  104432                                                                TRAP    C$EXIT
006542  000042                                                                .WORD   L10030-.

                                        .NLIST  BEX
006544      103     123     122 E300:   .ASCIZ  /CSR CHANGED BY WRITING TO DBUFF/
                                        .LIST   BEX
                                        .EVEN
006604                                  ENDTST
006604                                                                  L10030:
006604  104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL  TEST 4: REGISTER BIT SET
                                ;++
                                ; BITS 5 AND BITS 8 TO 15 IN THE CSR ARE SET AND CLEARED IN ALL
                                ; COMBINATIONS. FOR EACH COMBINATION, THE CSR IS READ AND A CHECK
                                ; MADE THAT THE RELEVANT BITS ARE SET.
                                ;
                                ; THE DBUFF REGISTER IS THEN USED IN CONJUNCTION WITH THE FIRST
                                ; PICTURE STORE ELEMENT TO ENSURE THAT CSR BITS 0 - 2 CAN BE SET.
                                ; CSR BITS 8 - 15 AND DBUFF BITS 0 - 8 ARE CLEAR THROUGHOUT,
                                ; ENSURING THAT ONLY THE FIRST PICTURE ELEMENT IS USED.
                                ; A 'WRITE PICTURE STORE' OPERATION IS PERFORMED USING THE COLOUR
                                ; BLACK. THE ELEMENT IS THEN READ AND A CHECK MADE THAT CSR BITS
                                ; 0 - 2 READ ZERO. THIS PROCESS IS REPEATED FOR EACH COLOUR CODE,
                                ; VERIFYING THAT THE CORRECT COLOUR IS REFLECTED IN THE CSR.
                                ;--

006606                                  BGNTST
006606                                                                  T4::
006606                                  BGNSUB                  ;********************** BEGIN SUBTEST
006606                                                                  T4.1:
006606  104402                                                                  TRAP    C$BSUB

                                ;
                                ; DIRECT BIT SET
                                ;
006610  005001                  T4A:    CLR     R1              ; INITIALISE TEST PATTERN

006612                                  BGNSEG                  ;---------------------- BEGIN SEGMENT
006612  104404                                                                  TRAP    C$BSEG
006614  010177  173352                  MOV     R1,@VTVCSR      ; WRITE TEST PATTERN TO CSR
006620  017703  173346                  MOV     @VTVCSR,R3      ; READ THE CSR
006624  010304                          MOV     R3,R4           ; SAVE THE CONTENTS
006626  042703  000337                  BIC     #337,R3         ; CLEAR UNWANTED BITS
006632  020301                          CMP     R3,R1           ; CORRECT BITS WRITTEN?
006634  001404                          BEQ     1$              ; IF YES, BRANCH
006636                                  ERRHRD  400,E400,ER400  ; ELSE REPORT THE ERROR
006636  104456                                                                  TRAP    C$ERHRD
006640  000620                                                                  .WORD   400
006642  006774                                                                  .WORD   E400
006644  002550                                                                  .WORD   ER400
006646                          1$:     ENDSEG                  ;---------------------- END SEGMENT
006646                                                                  10000$:
006646  104405                                                                  TRAP    C$ESEG

006650  062701  000400                  ADD     #400,R1         ; NEXT PATTERN
006654  103356                          BCC     T4A             ; IF MORE, BRANCH
006656  020127  000040                  CMP     R1,#40          ; ELSE SECOND TIME THROUGH?
006662  001403                          BEQ     2$              ; IF YES, BRANCH
006664  012701  000040                  MOV     #40,R1          ; ELSE SET BIT 5
006670  000750                          BR      T4A             ; AND REPEAT PATTERNS
006672                          2$:     ENDSUB                  ;********************** END SUBTEST
006672                                                                  L10032:
006672  104403                                                                  TRAP    C$ESUB

006674                                  BGNSUB                  ;********************** BEGIN SUBTEST
006674                                                                  T4.2:
006674  104402                                                                  TRAP    C$BSUB
```

```
                                        ;
                                        ; BIT SET VIA DBUFF AND PICTURE STORE
                                        ;
006676  005077  173270                  CLR     @VTVCSR         ; CLEAR THE CSR
006702  005001                          CLR     R1              ; INITIALISE TEST DBUFF PATTERN
006704  005003                          CLR     R3              ; INITIALISE EXPECTED CSR PATTERN

006706                          1$:     BGNSEG                  ;---------------------- BEGIN SEGMENT
006706  104404                                                                          TRAP    C$BSEG
006710  010177  173260                  MOV     R1,@DBUFF       ; WRITE DBUFF PATTERN
006714  012777  040000  173252          MOV     #40000,@DBUFF   ; LATCH PATTERN INTO THE CSR
006722  017704  173244                  MOV     @VTVCSR,R4      ; READ THE CSR
006726  010405                          MOV     R4,R5           ; SAVE THE CONTENTS
006730  042704  177770                  BIC     #177770,R4      ; CLEAR UNWANTED BITS
006734  020403                          CMP     R4,R3           ; EXPECTED PATTERN?
006736  001404                          BEQ     2$              ; IF YES, BRANCH
006740                                  ERRHRD  401,E401,ER401  ; ELSE REPORT THE ERROR
006740  104456                                                                          TRAP    C$ERHRD
006742  000621                                                                          .WORD   401
006744  007067                                                                          .WORD   E401
006746  002602                                                                          .WORD   ER401
006750                          2$:     ENDSEG                  ;---------------------- END SEGMENT
006750                                                                              10000$:
006750  104405                                                                      TRAP    C$ESEG

006752  005203                          INC     R3              ; NEXT EXPECTED CSR PATTERN
006754  062701  002000                  ADD     #2000,R1        ; NEXT DBUFF WRITE PATTERN
006760  020127  020000                  CMP     R1,#20000       ; ALL PATTERNS DONE?
006764  001350                          BNE     1$              ; IF NOT, DO NEXT

006766                                  EXIT TST                ; ELSE END OF TEST
006766  104432                                                                      TRAP    C$EXIT
006770  000206                                                                      .WORD   L10031-.
006772                                  ENDSUB                  ;********************** END SUBTEST
006772                                                                          L10033:
006772  104403                                                                  TRAP    C$ESUB

                                        .NLIST  BEX
006774    103     117     115   E400:   .ASCII  /COMBINATION OF CSR BIT 5 AND BITS 8 - 15 COULD NOT /
007057    102     105     040           .ASCIZ  /BE SET./
007067    103     117     115   E401:   .ASCII  /COMBINATION OF CSR BITS 0 - 2 COULD NOT BE SET VIA /
007152    124     110     105           .ASCIZ  /THE DBUFF REGISTER./
                                        .LIST   BEX
                                        .EVEN

007176                                  ENDTST
007176                                                                      L10031:
007176  104401                                                              TRAP    C$ETST
```

```
HARDWARE TESTS  MACRO V03.01 18-NOV-81 11:10:45 PAGE 38
TEST 5: PICTURE STORE DATA

                                .SBTTL  TEST 5: PICTURE STORE DATA

                                ;++
                                ; A ZERO DATA PATTERN (COLOUR BLACK) IS WRITTEN TO THE FIRST PICTURE
                                ; STORE LOCATION AND CHECKED TO READ BACK CORRECTLY IN THE CSR. THIS
                                ; IS REPEATED FOR ALL OF THE PICTURE STORE LOCATIONS TO FORM AN ALL
                                ; BLACK BACKGROUND.
                                ;
                                ; DATA IN ALL 8 COLOUR COMBINATIONS IS THEN WRITTEN TO EACH PICTURE
                                ; STORE LOCATION IN TURN AND AFTER EACH WRITE OPERATION, THE CSR IS
                                ; EXAMINED TO ENSURE THAT THE DATA READS BACK CORRECTLY. EACH
                                ; LOCATION IS RESET TO THE BACKGROUND COLOUR AFTER THE ABOVE
                                ; OPERATION.
                                ;
                                ; THE ENTIRE TEST IS THEN REPEATED USING WHITE (DATA 7) AS THE
                                ; BACKGROUND COLOUR.
                                ;--

007200                                  BGNTST
007200                                                                  T5::
                                
007200  012777  000050  172764          MOV     #50,@VTVCSR     ; SET VIDEO ENABLE AND ZERO OFFSET
007206  005067  000266                  CLR     BKGND           ; SET BACKGROUND TO BLACK
007212  005004                          CLR     R4              ; AND EXPECTED BACKGROUND
                                ;
                                ; SET PICTURE STORE TO BACKGROUND COLOUR
                                ;
007214  016701  000260          T5A:    MOV     BKGND,R1        ; INITIALISE DOT NO. AND COLOUR
007220  012705  040000                  MOV     #40000,R5       ; READ PICTURE STORE MASK

007224                          1$:     BGNSEG                  ;---------------------- BEGIN SEGMENT
007224  104404                                                                  TRAP    C$BSEG
007226  010177  172742                  MOV     R1,@DBUFF       ; WRITE BACKGROUND TO DBUFF
007232  010577  172736                  MOV     R5,@DBUFF       ; LATCH COLOUR INTO CSR
007236  017702  172730                  MOV     @VTVCSR,R2      ; READ THE CSR
007242  010203                          MOV     R2,R3           ; SAVE THE CONTENTS
007244  042702  177770                  BIC     #177770,R2      ; GET THE COLOUR
007250  020204                          CMP     R2,R4           ; BACKGROUND COLOUR?
007252  001404                          BEQ     2$              ; IF YES, BRANCH
007254                                  ERRHRD  500,E500,ER500  ; ELSE REPORT THE ERROR
007254  104456                                                                  TRAP    C$ERHRD
007256  000764                                                                  .WORD   500
007260  007502                                                                  .WORD   E500
007262  002634                                                                  .WORD   ER500
007264                          2$:     ENDSEG                  ;---------------------- END SEGMENT
007264                                                                  10000$:
007264  104405                                                                  TRAP    C$ESEG

007266  005205                          INC     R5              ; NEXT DOT TO READ
007270  005201                          INC     R1              ; AND TO WRITE
007272  032701  000777                  BIT     #777,R1         ; ALL DOTS DONE?
007276  001352                          BNE     1$              ; IF NOT, DO NEXT

007300  062777  000400  172664          ADD     #400,@VTVCSR    ; NEXT LINE
007306  103342                          BCC     T5A             ; IF MORE LINES, GO BACK
                                ;
                                ; SET ALL OTHER BIT COMBINATIONS
```

```
                                                ;
007310  012777  000040  172654                MOV     #40,@VTVCSR     ; SET VIDEO ENABLE ONLY
007316  005002                                CLR     R2              ; FIRST EXPECTED COLOUR

007320  005001                        3$:     CLR     R1              ; INITIALISE DOT NO. AND DATA
007322  012705  040000                        MOV     #40000,R5       ; READ PICTURE STORE MASK

007326                                4$:     BGNSEG                  ;---------------------- BEGIN SEGMENT
007326  104404                                                                                  TRAP    C$BSEG

007330  010177  172640                        MOV     R1,@DBUFF       ; WRITE COLOUR TO DBUFF
007334  010577  172634                        MOV     R5,@DBUFF       ; LATCH COLOUR INTO CSR
007340  017703  172626                        MOV     @VTVCSR,R3      ; READ THE CSR
007344  010304                                MOV     R3,R4           ; SAVE THE CONTENTS
007346  042703  177770                        BIC     #177770,R3      ; GET THE COLOUR
007352  020302                                CMP     R3,R2           ; EXPECTED VALUE?
007354  001404                                BEQ     5$              ; IF YES, BRANCH
007356                                        ERRHRD  501,E501,ER501  ; ELSE REPORT THE ERROR
007356  104456                                                                                  TRAP    C$ERHRD
007360  000765                                                                                  .WORD   501
007362  007552                                                                                  .WORD   E501
007364  002724                                                                                  .WORD   ER501
007366                                5$:     ENDSEG                  ;---------------------- END SEGMENT
007366                                                                                10001$:
007366  104405                                                                                  TRAP    C$ESEG

007370  005202                                INC     R2              ; NEXT EXPECTED COLOUR
007372  062701  002000                        ADD     #2000,R1        ; NEXT DBUFF WRITE COLOUR
007376  032701  020000                        BIT     #20000,R1       ; ALL COLOURS DONE?
007402  001751                                BEQ     4$              ; IF NOT, DO NEXT

007404  005002                                CLR     R2              ; ELSE RESET EXPECTED COLOUR
007406  042701  020000                        BIC     #20000,R1       ; CLEAR COLOUR OVERFLOW BIT
007412  056701  000062                        BIS     BKGND,R1        ; SET BACKGROUND COLOUR
007416  010177  172552                        MOV     R1,@DBUFF       ; WRITE BACKGROUND COLOUR
007422  042701  016000                        BIC     #16000,R1       ; CLEAR COLOUR BITS

007426  005205                                INC     R5              ; NEXT READ DOT POSITION
007430  005201                                INC     R1              ; AND WRITE DOT POSITION
007432  032701  001000                        BIT     #1000,R1        ; ALL DOTS DONE?
007436  001733                                BEQ     4$              ; IF NOT, DO NEXT

007440  062777  000400  172524                ADD     #400,@VTVCSR    ; ELSE NEXT LINE
007446  103324                                BCC     3$              ; IF MORE, GO BACK

007450  005767  000024                        TST     BKGND           ; WHITE BACKGROUND DONE?
007454  001007                                BNE     6$              ; IF YES, BRANCH
007456  012767  016000  000014                MOV     #16000,BKGND    ; ELSE FLAG WHITE BACKGROUND
007464  012704  000007                        MOV     #7,R4           ; AND CSR COLOUR
007470  000167  177520                        JMP     T5A             ; AND REPEAT THE TEST

007474                                6$:     EXIT    TST             ; END THE TEST
007474  104432                                                                          TRAP    C$EXIT
007476  000132                                                                          .WORD   L10034-.

007500  000000                        BKGND:  .WORD   0               ; BACKGROUND COLOUR TO WRITE
```

```
     98                                             .NLIST  BEX
     99 007502      102      101      103  E500:    .ASCIZ  /BACKGROUND COLOUR READ BACK INCORRECTLY/
    100 007552      104      101      124  E501:    .ASCIZ  /DATA READ BACK INCORRECTLY FROM PICTURE STORE/
    101                                             .LIST   BEX
    102                                             .EVEN
    103 007630                                      ENDTST
        007630                                                                          L10034:
        007630  104401                                                                          TRAP    C$ETST
```

```
                                .SBTTL  TEST 6: PICTURE STORE COLUMN ADDRESSING
                                ;++
                                ; THIS VERIFIES THAT WRITING TO ANY COLUMN OF THE PICTURE STORE
                                ; DOES NOT AFFECT THE CONTENTS OF ANY OTHER COLUMN. SINCE EACH
                                ; LOCATION CAN HOLD A 3 BIT DATA VALUE, THE COLUMN ADDRESS LINES ARE
                                ; CHECKED IN GROUPS OF 3.
                                ;
                                ; COLUMN ADDRESS BITS 0 - 2 ARE CHECKED BY WRITING THE VALUES 0 - 7 IN
                                ; CONSECUTIVE LOCATIONS THROUGHOUT THE PICTURE STORE AND CHECKING THAT
                                ; THE DATA READS BACK CORRECTLY VIA THE CSR.
                                ;
                                ; COLUMN ADDRESS BITS 3 - 5 ARE CHECKED BY WRITING ZERO TO THE FIRST
                                ; 8 LOCATIONS, 1 TO THE NEXT 8 LOCATIONS ETC. UNTIL THE PICTURE STORE
                                ; IS FULL. THE DATA IS THEN VERIFIED VIA THE CSR.
                                ;
                                ; COLUMN ADDRESS BITS 6 - 8 ARE CHECKED BY WRITING ZERO TO THE FIRST
                                ; 64 LOCATIONS, ONE TO THE NEXT 64 LOCATIONS AND SO ON, UNTIL THE
                                ; PICTURE STORE IS FULL. THE DATA IS THEN VERIFIED VIA THE CSR.
                                ;
                                ;--

007632                                  BGNTST
007632                                                                  T6::

007632  012767  000001  000252          MOV     #1,NCOLS        ; INITIALISE 1 COLUMN PER COLOUR

                                ; WRITE TO THE PICTURE STORE
                                ;
007640  005002                  T6A:    CLR     R2              ; DOT COUNTER FOR CURRENT COLOUR
007642  005001                          CLR     R1              ; INITIALISE DOT NO. AND COLOUR
007644  012777  000050  172320          MOV     #50,@VTVCSR     ; INITIALISE LINE NUMBER

007652  010177  172316          1$:     MOV     R1,@DBUFF       ; WRITE COLOUR
007656  005202                          INC     R2              ; INCREMENT DOT COUNTER
007660  020267  000226                  CMP     R2,NCOLS        ; READY FOR NEXT COLOUR?
007664  103410                          BLO     2$              ; IF NOT, BRANCH
007666  005002                          CLR     R2              ; ELSE CLEAR DOT COUNTER
007670  062701  002000                  ADD     #2000,R1        ; NEXT COLOUR
007674  032701  020000                  BIT     #20000,R1       ; ALL COLOURS USED?
007700  001402                          BEQ     2$              ; IF NOT, BRANCH
007702  042701  020000                  BIC     #20000,R1       ; ELSE CLEAR COLOUR BITS

007706  005201                  2$:     INC     R1              ; NEXT DOT POSITION
007710  032701  000777                  BIT     #777,R1         ; ALL DOTS DONE?
007714  001356                          BNE     1$              ; IF NOT, NEXT DOT
007716  162701  001000                  SUB     #1000,R1        ; ELSE CLEAR DOT NUMBER

007722  062777  000400  172242          ADD     #400,@VTVCSR    ; NEXT LINE
007730  103350                          BCC     1$              ; IF MORE LINES, BRANCH

                                ; READ BACK THE DATA
                                ;
007732  005002                          CLR     R2              ; DOT COUNTER FOR CURRENT COLOUR
007734  012701  040000                  MOV     #40000,R1       ; DOT 0, READ PICTURE STORE
007740  005004                          CLR     R4              ; INITIALISE EXPECTED COLOUR
007742  012777  000040  172222          MOV     #40,@VTVCSR     ; INITIALISE LINE NUMBER
```

```
     57
     58 007750                                  T6B:    BGNSEG                          ;---------------------- BEGIN SEGMENT
        007750  104404                                                                                          TRAP    C$BSEG
     59 007752  010177  172216                          MOV     R1,@DBUFF               ; LATCH COLOUR INTO CSR
     60 007756  017703  172210                          MOV     @VTVCSR,R3              ; READ THE CSR
     61 007762  010305                                  MOV     R3,R5                   ; SAVE THE CONTENTS
     62 007764  042703  177770                          BIC     #177770,R3              ; GET THE COLOUR
     63 007770  020304                                  CMP     R3,R4                   ; AS EXPECTED?
     64 007772  001404                                  BEQ     1$                      ; IF YES, BRANCH
     65 007774                                          ERRHRD  600,E600,ER600          ; ELSE REPORT THE ERROR
        007774  104456                                                                                          TRAP    C$ERHRD
        007776  001130                                                                                          .WORD   600
        010000  010114                                                                                          .WORD   E600
        010002  003032                                                                                          .WORD   ER600
     66 010004                                  1$:     ENDSEG                          ;---------------------- END SEGMENT
        010004                                                                                          10000$:
        010004  104405                                                                                          TRAP    C$ESEG
     67
     68 010006  005202                                  INC     R2                      ; INCREMENT DOT COUNTER
     69 010010  020267  000076                          CMP     R2,NCOLS                ; READY FOR NEXT COLOUR?
     70 010014  103406                                  BLO     2$                      ; IF NOT, BRANCH
     71 010016  005002                                  CLR     R2                      ; ELSE CLEAR DOT COUNTER
     72 010020  005204                                  INC     R4                      ; EXPECT NEXT COLOUR
     73 010022  032704  000010                          BIT     #10,R4                  ; ALL COLOURS USED?
     74 010026  001401                                  BEQ     2$                      ; IF NOT, BRANCH
     75 010030  005004                                  CLR     R4                      ; ELSE NEXT COLOUR
     76
     77 010032  005201                          2$:     INC     R1                      ; NEXT DOT PATTERN
     78 010034  032701  000777                          BIT     #777,R1                 ; ALL DOTS DONE?
     79 010040  001343                                  BNE     T6B                     ; IF NOT, NEXT DOT
     80 010042  162701  001000                          SUB     #1000,R1                ; ELSE CLEAR DOT NUMBER
     81
     82 010046  062777  000400  172116                  ADD     #400,@VTVCSR            ; NEXT LINE
     83 010054  103335                                  BCC     T6B                     ; IF MORE LINES, BRANCH
     84
     85 010056  006367  000030                          ASL     NCOLS                   ; FLAG NEXT 3
     86 010062  006367  000024                          ASL     NCOLS                   ; ADDRESS LINES
     87 010066  006367  000020                          ASL     NCOLS                   ; TO BE TESTED
     88 010072  042767  177400  000012                  BIC     #177400,NCOLS           ; STOP IF TOP BYTE USED
     89 010100  001402                                  BEQ     3$                      ; IF FINISHED, BRANCH
     90 010102  000167  177532                          JMP     T6A                     ; ELSE NEXT ADDRESS LINES
     91
     92 010106                                  3$:     EXIT    TST                     ; END OF TEST
        010106  104432                                                                                          TRAP    C$EXIT
        010110  000052                                                                                          .WORD   L10035-.
     93
     94 010112  000000                          NCOLS:  .WORD   0                       ; NO. OF COLUMNS OF CURRENT COLOUR
     95                                                 .NLIST  BEX
     96 010114     120     111     103          E600:   .ASCIZ  /PICTURE STORE COLUMN ADDRESSING ERROR/
     97                                                 .LIST   BEX
     98                                                 .EVEN
     99 010162                                          ENDTST
        010162                                                                                          L10035:
        010162  104401                                                                                          TRAP    C$ETST
```

```
                                .SBTTL  TEST 7: PICTURE STORE LINE ADDRESSING
                                ;++
                                ; THIS VERIFIES THAT WRITING TO ANY LINE OF THE PICTURE STORE
                                ; DOES NOT AFFECT THE CONTENTS OF ANY OTHER LINE. SINCE EACH
                                ; LOCATION CAN HOLD A 3 BIT DATA VALUE, THE LINE ADDRESS LINES ARE
                                ; CHECKED IN GROUPS OF THREE.
                                ;
                                ; LINE ADDRESS BITS 0 - 2 (CSR BITS 8 - 10) ARE CHECKED BY WRITING ZERO
                                ; TO ALL LOCATIONS IN THE FIRST LINE, ONE TO ALL LOCATIONS IN THE
                                ; NEXT LINE AND SO ON, UNTIL ALL LINES ARE WRITTEN TO. THE DATA IS THEN
                                ; READ BACK AND VERIFIED VIA THE CSR.
                                ;
                                ; LINE ADDRESS BITS 3 - 5 (CSR BITS 11 - 13) ARE CHECKED BY WRITING
                                ; ZERO TO THE FIRST EIGHT LINES, ONE TO THE NEXT EIGHT LINES ETC.,
                                ; TO THE END OF THE PICTURE STORE. THE DATA IS THEN VERIFIED VIA THE
                                ; CSR.
                                ;
                                ; LINE ADDRESS BITS 6 AND 7 (CSR BITS 14 AND 15) ARE CHECKED BY WRITING
                                ; ZERO TO THE FIRST 64 LINES, ONE TO THE NEXT 64 LINES, TWO TO THE NEXT
                                ; 64 LINES, AND THREE TO THE LAST 64 LINES. THE PICTURE STORE DATA IS
                                ; THEN READ BACK AND VERIFIED.
                                ;--

 010164                                 BGNTST
 010164                                                                  T7::
 010164  012767  000001  000256         MOV     #1,NLINS        ; INITIALISE ONE LINE PER COLOUR
                                ;
                                ; WRITE TO THE PICTURE STORE
                                ;
 010172  005002                 T7A:    CLR     R2              ; LINE COUNTER FOR CURRENT COLOUR
 010174  005001                         CLR     R1              ; INITIALISE DOT NO. AND COLOUR
 010176  012777  000050  171766         MOV     #50,@VTVCSR     ; INITIALISE LINE NUMBER

 010204  010177  171764         1$:     MOV     R1,@DBUFF       ; WRITE COLOUR
 010210  005201                         INC     R1              ; NEXT DOT POSITION
 010212  032701  000777                 BIT     #777,R1         ; ALL DOTS DONE?
 010216  001372                         BNE     1$              ; IF NOT, NEXT DOT
 010220  162701  001000                 SUB     #1000,R1        ; ELSE CLEAR DOT NUMBER

 010224  062777  000400  171740         ADD     #400,@VTVCSR    ; NEXT LINE
 010232  103415                         BCS     2$              ; IF ALL LINES DONE, BRANCH
 010234  005202                         INC     R2              ; ELSE INCREMENT LINE COUNTER
 010236  020267  000206                 CMP     R2,NLINS        ; READY FOR NEXT COLOUR?
 010242  103760                         BLO     1$              ; IF NOT, DO NEXT LINE
 010244  005002                         CLR     R2              ; ELSE CLEAR LINE COUNTER
 010246  062701  002000                 ADD     #2000,R1        ; NEXT COLOUR
 010252  032701  020000                 BIT     #20000,R1       ; ALL COLOURS USED?
 010256  001752                         BEQ     1$              ; IF NOT, BRANCH
 010260  042701  020000                 BIC     #20000,R1       ; ELSE CLEAR COLOUR BITS
 010264  000747                         BR      1$              ; AND DO NEXT LINE
                                ;
                                ; READ BACK THE DATA
                                ;
 010266  005002                 2$:     CLR     R2              ; LINE COUNTER FOR CURRENT COLOUR
 010270  012701  040000                 MOV     #40000,R1       ; DOT 0, READ PICTURE STORE
```

```
     57 010274  005004                          CLR     R4                      ; INITIALISE EXPECTED COLOUR
     58 010276  012777  000040  171666          MOV     #40,@VTVCSR             ; INITIALISE LINE NUMBER
     59
     60 010304                          T7B:    BGNSEG                          ;---------------------- BEGIN SEGMENT
        010304  104404                                                                  TRAP    C$BSEG
     61 010306  010177  171662                  MOV     R1,@DBUFF               ; LATCH COLOUR INTO CSR
     62 010312  017703  171654                  MOV     @VTVCSR,R3              ; READ THE CSR
     63 010316  010305                          MOV     R3,R5                   ; SAVE THE CONTENTS
     64 010320  042703  177770                  BIC     #177770,R3              ; GET THE COLOUR
     65 010324  020304                          CMP     R3,R4                   ; AS EXPECTED?
     66 010326  001404                          BEQ     1$                      ; IF YES, BRANCH
     67 010330                                  ERRHRD  700,E700,ER700          ; ELSE REPORT THE ERROR
        010330  104456                                                                  TRAP    C$ERHRD
        010332  001274                                                                  .WORD   700
        010334  010452                                                                  .WORD   E700
        010336  003214                                                                  .WORD   ER700
     68 010340                          1$:     ENDSEG                          ;---------------------- END SEGMENT
        010340                                                                      10000$:
        010340  104405                                                                  TRAP    C$ESEG
     69
     70 010342  005201                          INC     R1                      ; NEXT DATA PATTERN
     71 010344  032701  000777                  BIT     #777,R1                 ; ALL DOTS DONE?
     72 010350  001355                          BNE     T7B                     ; IF NOT, NEXT DOT
     73 010352  162701  001000                  SUB     #1000,R1                ; ELSE CLEAR DOT NUMBER
     74
     75 010356  062777  000400  171606          ADD     #400,@VTVCSR            ; NEXT LINE
     76 010364  103413                          BCS     2$                      ; IF ALL LINES DONE, BRANCH
     77
     78 010366  005202                          INC     R2                      ; ELSE INCREMENT LINE COUNTER
     79 010370  020267  000054                  CMP     R2,NLINS                ; READY FOR NEXT COLOUR?
     80 010374  103743                          BLO     T7B                     ; IF NOT, READ NEXT LINE
     81
     82 010376  005002                          CLR     R2                      ; ELSE CLEAR LINE COUNTER
     83 010400  005204                          INC     R4                      ; EXPECT NEXT COLOUR
     84 010402  032704  000010                  BIT     #10,R4                  ; ALL COLOURS USED?
     85 010406  001736                          BEQ     T7B                     ; IF NOT, DO NEXT LINE
     86 010410  005004                          CLR     R4                      ; ELSE RESET COLOUR
     87 010412  000734                          BR      T7B                     ; AND READ NEXT LINE
     88
     89 010414  006367  000030          2$:     ASL     NLINS                   ; FLAG NEXT 3
     90 010420  006367  000024                  ASL     NLINS                   ; ADDRESS LINES
     91 010424  006367  000020                  ASL     NLINS                   ; TO BE TESTED
     92 010430  026727  000014  001000          CMP     NLINS,#1000             ; ALL FINISHED?
     93 010436  001402                          BEQ     3$                      ; IF YES, BRANCH
     94 010440  000167  177526                  JMP     T7A                     ; ELSE NEXT ADDRESS LINES
     95
     96 010444                          3$:     EXIT    TST                     ; END OF TEST
        010444  104432                                                                  TRAP    C$EXIT
        010446  000050                                                                  .WORD   L10036-.
     97
     98 010450  000000                  NLINS:  .WORD   0                       ; NO. OF LINES PER COLOUR
     99
    100                                         .NLIST  BEX
    101 010452     120     111     103  E700:   .ASCIZ  /PICTURE STORE LINE ADDRESSING ERROR/
    102                                         .LIST   BEX
    103                                         .EVEN
    104 010516                                  ENDTST
```

```
        010516
        010516  104401                                          L10036:
                                                                        TRAP    C$ETST
```

```
                                .SBTTL  TEST 8: 8-DOT

                                ;++
                                ; THIS TEST CHECKS THAT IF DBUFF BIT 15 (8-DOT) IS SET, PICTURE
                                ; STORE WRITES ARE MADE TO EIGHT DOTS AT A TIME REGARDLESS OF THE
                                ; VALUES OF DBUFF BITS 0 - 2. FOR EACH COMBINATION OF THE LOWER DOT
                                ; ADDRESS BITS, A NEW COLOUR IS USED AND THE 8-DOT WRITE VERIFIED.
                                ;--

                                        BGNTST
                                                                                        T8::

                                        CLR     R2              ; INITIALISE DBUFF WRITE COLOUR
                                        CLR     R3              ; INITIALISE CSR COLOUR AND DOT NO.
                                        MOV     #50,@VTVCSR     ; SET VIDEO ENABLE AND ZERO OFFSET

                                ;
                                ; INITIALISE FIRST 8 DOTS
                                ;
                                        MOV     #16000,R1       ; SET COLOUR TO WHITE
                                1$:     MOV     R1,@DBUFF       ; WRITE DOT
                                        INC     R1              ; NEXT DOT
                                        BIT     #7,R1           ; 8 DOTS DONE?
                                        BNE     1$              ; IF NOT, DO MORE
                                ;
                                ; DO 8 DOT TEST
                                ;
                                T8A:    BGNSEG                  ;---------------------- BEGIN SEGMENT
                                                                                        TRAP    C$BSEG
                                        MOV     #100000,R1      ; SET 8-DOT
                                        BIS     R2,R1           ; SET THE COLOUR
                                        BIS     R3,R1           ; AND DOT NUMBER
                                        MOV     R1,@DBUFF       ; WRITE TO DBUFF
                                        BIS     #40000,R1       ; SET READ PICTURE STORE
                                        BIC     #100000,R1      ; CLEAR 8-DOT

                                T8B:    MOV     R1,@DBUFF       ; LATCH COLOUR INTO CSR
                                        MOV     @VTVCSR,R4      ; READ THE CSR
                                        MOV     R4,R5           ; SAVE THE CONTENTS
                                        BIC     #177770,R4      ; GET THE COLOUR
                                        CMP     R4,R3           ; EXPECTED COLOUR
                                        BEQ     1$              ; IF YES, BRANCH
                                        ERRHRD  800,E800,ER800  ; ELSE REPORT THE ERROR
                                                                                        TRAP    C$ERHRD
                                                                                        .WORD   800
                                                                                        .WORD   E800
                                                                                        .WORD   ER800
                                1$:     INC     R1              ; NEXT DOT ADDRESS
                                        BIT     #10,R1          ; 8 DOTS READ?
                                        BEQ     T8B             ; IF NOT, READ NEXT
                                        ENDSEG                  ;---------------------- END SEGMENT
                                                                                10000$:
                                                                                        TRAP    C$ESEG

                                        ADD     #2000,R2        ; ELSE NEXT COLOUR
                                        INC     R3              ; NEXT READ COLOUR,DOT NO.
                                        BIT     #10,R3          ; ALL COLOURS DONE?
```

```
HARDWARE TESTS  MACRO V03.01 18-NOV-81 11:10:45 PAGE 41-1
TEST 8: 8-DOT

     50 010656  001735                          BEQ     T8A             ; IF NOT, DO NEXT
     51 010660                                  EXIT    TST             ; ELSE END OF TEST
        010660  104432                                                                      TRAP    C$EXIT
        010662  000030                                                                      .WORD   L10037-.
     52
     53                                         .NLIST  BEX
     54 010664     070     040     055  E800:   .ASCIZ  /8 - DOT WRITE FAILURE/
     55                                         .LIST   BEX
     56                                         .EVEN
     57 010712                                  ENDTST
        010712                                                                  L10037:
        010712  104401                                                                      TRAP    C$ETST
```

```
                                        .SBTTL  TEST 9: PRESET
                                        ;++
                                        ; DBUFF BITS 10 - 12 ARE SET TO THE COLOUR BLACK (ZERO) AND A PRESET
                                        ; IS PERFORMED. THE READY BIT IN THE CSR IS CHECKED TO CLEAR AND
                                        ; THEN TO BE SET AFTER 50 MILLISECONDS. THE PICTURE STORE IS THEN READ
                                        ; TO CONFIRM THAT ALL LOCATIONS HAVE BEEN CORRECTLY SET. THE PROCESS
                                        ; IS THEN REPEATED FOR EACH OF THE EIGHT SELECTABLE COLOURS.
                                        ;--

 010714                                         BGNTST
 010714                                                                                  T9::

 010714  005001                                 CLR     R1              ; INITIALISE PRESET COLOUR
 010716  005002                                 CLR     R2              ; AND EXPECTED CSR COLOUR

 010720                                 T9A:    BGNSEG                  ;---------------------- BEGIN SEGMENT
 010720  104404                                                                          TRAP    C$BSEG
                                        ;
                                        ; DO THE PRESET
                                        ;
 010722                                         BGNSEG                  ;---------------------- BEGIN SEGMENT
 010722  104404                                                                          TRAP    C$BSEG
 010724  010177  171244                         MOV     R1,@DBUFF       ; WRITE COLOUR TO DBUFF
 010730  012777  000150  171234                 MOV     #150,@VTVCSR    ; PERFORM THE PRESET
 010736  017703  171230                         MOV     @VTVCSR,R3      ; READ THE CSR
 010742  105703                                 TSTB    R3              ; READY BIT CLEAR?
 010744  100004                                 BPL     1$              ; IF YES, BRANCH
 010746                                         ERRHRD  900,E900,ER900  ; ELSE REPORT THE ERROR
 010746  104456                                                                  TRAP    C$ERHRD
 010750  001604                                                                  .WORD   900
 010752  011122                                                                  .WORD   E900
 010754  003476                                                                  .WORD   ER900

 010756  004767  171434                 1$:     JSR     PC,WAIT25       ; WAIT FOR 50 MILLISECONDS
 010762  004767  171430                         JSR     PC,WAIT25       ;
 010766  017703  171200                         MOV     @VTVCSR,R3      ; READ THE CSR
 010772  105703                                 TSTB    R3              ; READY SET?
 010774  100404                                 BMI     2$              ; IF YES, BRANCH
 010776                                         ERRHRD  901,E901,ER900  ; ELSE REPORT THE ERROR
 010776  104456                                                                  TRAP    C$ERHRD
 011000  001605                                                                  .WORD   901
 011002  011166                                                                  .WORD   E901
 011004  003476                                                                  .WORD   ER900

 011006                                 2$:     ENDSEG                  ;---------------------- END SEGMENT
 011006                                                                                  10001$:
 011006  104405                                                                         TRAP    C$ESEG
                                        ;
                                        ; CHECK THE PICTURE STORE
                                        ;
 011010  012703  040000                         MOV     #40000,R3       ; FIRST DOT POSITION
 011014  012777  000040  171150                 MOV     #40,@VTVCSR     ; FIRST LINE

 011022  010377  171146                 T9B:    MOV     R3,@DBUFF       ; LATCH COLOUR INTO CSR
 011026  017704  171140                         MOV     @VTVCSR,R4      ; READ THE CSR
 011032  010405                                 MOV     R4,R5           ; SAVE THE CONTENTS
```

```
     45 011034  042704  177770                BIC     #177770,R4      ; GET THE COLOUR
     46 011040  020402                        CMP     R4,R2           ; EXPECTED COLOUR?
     47 011042  001404                        BEQ     1$              ; IF YES, BRANCH
     48 011044                                ERRHRD  902,E902,ER902  ; ELSE REPORT THE ERROR
        011044  104456                                                          TRAP    C$ERHRD
        011046  001606                                                          .WORD   902
        011050  011247                                                          .WORD   E902
        011052  003530                                                          .WORD   ER902
     49
     50 011054  005203                1$:     INC     R3              ; NEXT DOT
     51 011056  032703  001000                BIT     #1000,R3        ; ALL DOTS DONE?
     52 011062  001757                        BEQ     T9B             ; IF NOT, DO NEXT
     53 011064  042703  001000                BIC     #1000,R3        ; ELSE CLEAR THE DOT NUMBER
     54
     55 011070  062777  000400  171074        ADD     #400,@VTVCSR    ; NEXT LINE
     56 011076  103351                        BCC     T9B             ; IF NOT, DO NEXT
     57 011100                                ENDSEG                  ;---------------------- END SEGMENT
        011100                                                                  10000$:
        011100  104405                                                          TRAP    C$ESEG
     58
     59 011102  005202                        INC     R2              ; NEXT EXPECTED COLOUR
     60 011104  062701  002000                ADD     #2000,R1        ; NEXT PRESET COLOUR
     61 011110  032701  020000                BIT     #20000,R1       ; ALL COLOURS DONE?
     62 011114  001701                        BEQ     T9A             ; IF NOT, DO NEXT
     63
     64 011116                                EXIT    TST             ; ELSE END OF TEST
        011116  104432                                                          TRAP    C$EXIT
        011120  000174                                                          .WORD   L10040-.
     65
     66                                       .NLIST  BEX
     67 011122     103     123     122  E900:  .ASCIZ  /CSR READY NOT CLEAR DURING A PRESET/
     68 011166     103     123     122  E901:  .ASCIZ  /CSR READY NOT SET 50 MILLISECONDS AFTER A PRESET/
     69 011247     104     117     124  E902:  .ASCIZ  /DOT NOT SET CORRECTLY AFTER A PRESET/
     70                                       .LIST   BEX
     71                                       .EVEN
     72 011314                                ENDTST
        011314                                                          L10040:
        011314  104401                                                          TRAP    C$ETST
```

```
                                        .SBTTL  TEST 10: VIDEO ENABLE

                                        ;++
                                        ; THE PICTURE STORE IS PRESET TO BLUE FOR FIVE SECONDS. BIT 5 OF
                                        ; THE CSR (VIDEO ENABLE) IS THEN CLEARED FOR FIVE SECONDS, DURING
                                        ; WHICH TIME THE MONITOR DISPLAY SHOULD BE DARK. THE DISPLAY IS THEN
                                        ; REENABLED FOR A FURTHER FIVE SECONDS.
                                        ;--

011316                                          BGNTST
011316                                                                                  T10::
                                        ;
                                        ; PRESET TO WHITE, VIDEO ENABLED
                                        ;
011316  012777  010000  170650          1$:     MOV     #10000,@DBUFF   ; SET PRESET COLOUR BLUE
011324  012777  000140  170640                  MOV     #140,@VTVCSR    ; PERFORM THE PRESET
011332  012701  000310                          MOV     #200.,R1        ; WAIT FOR 5 SECONDS
011336  004767  171054                          JSR     PC,WAIT25       ; 25 MILLISECOND DELAY
011342                                          BREAK                   ; ALLOW OPERATOR BREAK-IN
011342  104422                                                                  TRAP    C$BRK
011344  005301                                  DEC     R1              ; TIME UP?
011346  001373                                  BNE     1$              ; IF NOT, WAIT LONGER
                                        ;
                                        ; CLEAR VIDEO ENABLE
                                        ;
011350  005077  170616                          CLR     @VTVCSR         ; CLEAR VIDEO ENABLE
011354  012701  000310                          MOV     #200.,R1        ; WAIT FOR 5 SECONDS
011360  004767  171032                  2$:     JSR     PC,WAIT25       ; 25 MILLISECOND DELAY
011364                                          BREAK                   ; ALLOW OPERATOR BREAK-IN
011364  104422                                                                  TRAP    C$BRK
011366  005301                                  DEC     R1              ; TIME UP?
011370  001373                                  BNE     2$              ; IF NOT, WAIT LONGER
                                        ;
                                        ; REENABLE VIDEO
                                        ;
011372  012777  000040  170572                  MOV     #40,@VTVCSR     ; SET VIDEO ENABLE
011400  012701  000310                          MOV     #200.,R1        ; WAIT FOR 5 SECONDS
011404  004767  171006                  3$:     JSR     PC,WAIT25       ; 5 MILLISECOND DELAY
011410                                          BREAK                   ; ALLOW OPERATOR BREAK-IN
011410  104422                                                                  TRAP    C$BRK
011412  005301                                  DEC     R1              ; TIME UP?
011414  001373                                  BNE     3$              ; IF NOT, WAIT LONGER

011416                                          EXIT    TST             ; END OF TEST
011416  104432                                                                  TRAP    C$EXIT
011420  000002                                                                  .WORD   L10041-.
011422                                          ENDTST
011422                                                                  L10041:
011422  104401                                                                  TRAP    C$ETST
```

TEST 11: DISPLAY ADDRESSING

```
                                .SBTTL  TEST 11: DISPLAY ADDRESSING

                                ;++
                                ; THE PICTURE STORE IS PRESET TO BLUE. LINES OF MAGENTA ARE THEN
                                ; WRITTEN FROM THE TOP OF THE DISPLAY UNTIL THE WHOLE SCREEN HAS
                                ; CHANGED COLOUR. THE TRANSITION SHOULD BE SMOOTH, TAKING APPROXIMATELY
                                ; 7 SECONDS.
                                ; COLUMNS OF WHITE ARE THEN WRITTEN FROM THE LEFT SIDE OF THE DISPLAY
                                ; UNTIL THE SCREEN IS ALL WHITE. AGAIN THE TRANSITION SHOULD BE SMOOTH.
                                ; IF THE DISPLAY DOES NOT APPEAR AS DESCRIBED HERE, AN ADDRESSING
                                ; FAULT ON THE OUTPUT SIDE OF THE DEVICE IS INDICATED.
                                ;--

011424                                  BGNTST
011424                                                                          T11::
                                ;
                                ; PRESET SCREEN TO BLUE
                                ;
011424  012777  010000  170542          MOV     #10000,@DBUFF   ; SET PRESET COLOUR BLUE
011432  012777  000150  170532          MOV     #150,@VTVCSR    ; PERFORM THE PRESET
011440  012701  177777                  MOV     #-1,R1          ; SET UP WAIT COUNTER
011444  105777  170522          1$:     TSTB    @VTVCSR         ; READY SET?
011450  100403                          BMI     2$              ; IF YES, BRANCH
011452                                  BREAK                   ; ELSE ALLOW OPERATOR BREAK IN
011452  104422                                                                  TRAP    C$BRK
011454  005301                          DEC     R1              ; WAITED TOO LONG?
011456  001372                          BNE     1$              ; IF NOT, WAIT LONGER
                                ;
                                ; MOVE MAGENTA DOWN THE DISPLAY
                                ;
011460  012777  000040  170504  2$:     MOV     #40,@VTVCSR     ; START AT LINE 0
011466  012701  012000                  MOV     #12000,R1       ; SET WRITE COLOUR TO MAGENTA

011472  010177  170476          3$:     MOV     R1,@DBUFF       ; WRITE DOT
011476  005201                          INC     R1              ; NEXT DOT ADDRESS
011500  032701  000777                  BIT     #777,R1         ; LINE FINISHED?
011504  001372                          BNE     3$              ; IF NOT, DO NEXT DOT

011506  004767  170704                  JSR     PC,WAIT25       ; WAIT FOR 25 MILLISECONDS
011512                                  BREAK                   ; ALLOW OPERATOR BREAK IN
011512  104422                                                                  TRAP    C$BRK

011514  042701  001000                  BIC     #1000,R1        ; RESET TO START OF LINE
011520  062777  000400  170444          ADD     #400,@VTVCSR    ; AND NEXT LINE
011526  103361                          BCC     3$              ; IF MORE LINES, GO BACK
                                ;
                                ; MOVE WHITE ACROSS THE DISPLAY
                                ;
011530  012777  000040  170434          MOV     #40,@VTVCSR     ; START AT LINE 0
011536  012701  016000                  MOV     #16000,R1       ; SET WRITE COLOUR TO WHITE

011542  010177  170426          5$:     MOV     R1,@DBUFF       ; WRITE DOT
011546  062777  000400  170416          ADD     #400,@VTVCSR    ; NEXT LINE ADDRESS
011554  103372                          BCC     5$              ; IF MORE LINES, GO BACK

011556  016700  170424                  MOV     COUNT,R0        ; GET 25 MILLISECOND COUNT
011562  006200                          ASR     R0              ; MAKE IT 12 MILLISECONDS
```

```
     55 011564  005300                  6$:     DEC     R0              ; WAIT 12 MILLISECONDS
     56 011566  001376                          BNE     6$              ;
     57 011570                                  BREAK                   ; ALLOW OPERATOR BREAK IN
        011570  104422                                                                  TRAP    C$BRK
     58
     59 011572  005201                          INC     R1              ; NEXT COLUMN
     60 011574  032701  000777               BIT     #777,R1         ; ALL COLUMNS DONE?
     61 011600  001360                          BNE     5$              ; IF NOT, DO NEXT
     62
     63 011602                                  EXIT TST                ; ELSE END OF TEST
        011602  104432                                                                  TRAP    C$EXIT
        011604  000002                                                                  .WORD   L10042-.
     64 011606                                  ENDTST
        011606                                                          L10042:
        011606  104401                                                                  TRAP    C$ETST
```

```
                                        .SBTTL  TEST 12: OFFSET
                                        ;++
                                        ; THE PICTURE STORE IS PRESET TO BLUE. A LINE OF MAGENTA IS THEN WRITTEN
                                        ; TO THE TOP OF THE DISPLAY AND THE OFFSET REGISTER CHANGED TO MOVE
                                        ; THE LINE DOWN ONE POSITION. THIS IS REPEATED UNTIL THE WHOLE
                                        ; DISPLAY IS MAGENTA, WHICH SHOULD TAKE ABOUT 7 SECONDS. THE DISPLAY IS
                                        ; THEN CHANGED TO WHITE FROM THE BOTTOM USING THE SAME PROCESS,
                                        ; UNTIL THE SCREEN IS ALL WHITE.
                                        ; IN BOTH CASES, THE LINE BETWEEN THE COLOURS SHOULD MOVE SMOOTHLY
                                        ; ACROSS THE SCREEN. ANY UNEVEN MOVEMENT INDICATES  A PROBLEM
                                        ; IN THE OFFSET REGISTER.
                                        ;--

 011610                                         BGNTST
 011610                                                                                 T12::
                                        ;
                                        ; PRESET SCREEN TO BLUE
                                        ;
 011610  012777  010000  170356                 MOV     #10000,@DBUFF   ; SET PRESET COLOUR BLUE
 011616  012777  000140  170346                 MOV     #140,@VTVCSR    ; PERFORM THE PRESET
 011624  012701  177777                         MOV     #-1,R1          ; SET UP WAIT COUNTER
 011630  105777  170336         1$:     TSTB    @VTVCSR         ; READY SET?
 011634  100403                         BMI     2$              ; IF YES, BRANCH
 011636                                 BREAK                   ; ELSE ALLOW OPERATOR BREAK IN
 011636  104422                                                                 TRAP    C$BRK
 011640  005301                         DEC     R1              ; WAITED TOO LONG?
 011642  001372                         BNE     1$              ; IF NOT, WAIT LONGER
                                        ;
                                        ; MOVE MAGENTA DOWN THE DISPLAY
                                        ;
 011644  012701  177450         2$:     MOV     #177450,R1      ; OFFSET LAST LINE TO TOP
 011650  010177  170316         3$:     MOV     R1,@VTVCSR      ; WRITE THE OFFSET
 011654  012702  112000                 MOV     #112000,R2      ; SET FIRST 8 DOTS MAGENTA
 011660  010277  170310         5$:     MOV     R2,@DBUFF       ; WRITE 8 DOTS
 011664  062702  000010                 ADD     #10,R2          ; NEXT DOT
 011670  032702  000777                 BIT     #777,R2         ; LINE WRITTEN?
 011674  001371                         BNE     5$              ; IF NOT, DO MORE DOTS
 011676  162702  001000                 SUB     #1000,R2        ; ELSE CLEAR DOT NUMBER
 011702  004767  170510                 JSR     PC,WAIT25       ; WAIT FOR 25 MILLISECONDS
 011706                                 BREAK                   ; ALLOW OPERATOR BREAK-IN
 011706  104422                                                                 TRAP    C$BRK

 011710  162701  000400                 SUB     #400,R1         ; REDUCE THE OFFSET
 011714  103355                         BCC     3$              ; IF MORE LINES, BRANCH
                                        ;
                                        ; MOVE WHITE UP THE DISPLAY
                                        ;
 011716  012701  000450                 MOV     #450,R1         ; OFFSET FIRST LINE TO BOTTOM
 011722  012703  000040                 MOV     #40,R3          ; FIRST WRITE LINE
 011726  032777  000020  170236 7$:     BIT     #20,@VTVCSR     ; FIELD BIT CLEAR?
 011734  001374                         BNE     7$              ; IF NOT, WAIT
 011736  010177  170230                 MOV     R1,@VTVCSR      ; WRITE THE OFFSET
 011742  010377  170224                 MOV     R3,@VTVCSR      ; SET UP TO WRITE LINE
 011746  012702  116000                 MOV     #116000,R2      ; SET FIRST 8 DOTS WHITE
 011752  010277  170216         8$:     MOV     R2,@DBUFF       ; WRITE 8 DOTS
 011756  062702  000010                 ADD     #10,R2          ; NEXT DOTS
```

```
     55 011762  032702  001000                BIT     #1000,R2        ; LINE WRITTEN?
     56 011766  001771                        BEQ     8$              ; IF NOT, DO MORE DOTS
     57 011770  162702  001000                SUB     #1000,R2        ; ELSE CLEAR THE DOT NUMBER
     58 011774  004767  170416                JSR     PC,WAIT25       ; WAIT FOR 25 MILLISECONDS
     59 012000                                BREAK                   ; ALLOW OPERATOR BREAK-IN
        012000  104422                                                                  TRAP    C$BRK
     60
     61 012002  062703  000400                ADD     #400,R3         ; NEXT WRITE LINE
     62 012006  062701  000400                ADD     #400,R1         ; INCREASE THE OFFSET
     63 012012  103345                        BCC     7$              ; IF MORE LINES, BRANCH
     64 012014  012777  000010  170150        MOV     #10,@VTVCSR     ; CLEAR THE OFFSET REGISTER
     65
     66 012022                                EXIT    TST
        012022  104432                                                                  TRAP    C$EXIT
        012024  000002                                                                  .WORD   L10043-.
     67 012026                                ENDTST
        012026                                                                  L10043:
        012026  104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL  TEST 13: PATTERN GENERATOR
                                ;++
                                ; THIS TEST IS FOR SETTING UP AND CHECKING THE OUTPUT TO THE DISPLAY
                                ; MONITOR. NINE PATTERNS ARE PROVIDED - EIGHT TO PRESET THE SCREEN
                                ; TO EACH POSSIBLE COLOUR, AND ONE TO OUTPUT A CROSS HATCH OF 32
                                ; COLUMN BY 16 LINE RECTANGLES. THE NUMBER OF RECTANGLES DISPLAYED
                                ; WILL DEPEND ON THE DEVICE CONFIGURATION.
                                ;
                                ; NORMALLY, THE NINE PATTERNS ARE DISPLAYED IN TURN, EACH REMAINING
                                ; ON THE SCREEN FOR APPROXIMATELY FIVE SECONDS. HOWEVER, IF MANUAL
                                ; SELECTION IS SELECTED AT START UP, EACH PATTERN IS KEPT DISPLAYED
                                ; UNTIL THE RETURN KEY IS HIT ON THE OPERATOR CONSOLE.
                                ;--

012030                                  BGNTST
012030                                                                          T13::

012030  005001                  CLR     R1              ; FIRST PRESET COLOUR

012032  010177  170136  T13A:   MOV     R1,@DBUFF       ; SET THE PRESET COLOUR
012036  012777  000150  170126  MOV     #150,@VTVCSR    ; DO THE PRESET
                                ;
                                ; WAIT FOR NEXT PATTERN
                                ;
012044  005767  170114  2$:     TST     MANVEN          ; RUN MANUALLY?
012050  001413                  BEQ     3$              ; IF NOT, BRANCH
012052                          MANUAL                  ; MANUAL ALLOWED?
012052  104450                                                          TRAP    C$MANI
012054                          BNCOMPLETE 3$           ; IF NOT, BRANCH
012054  103011                                                          BCC     3$
012056                          GMANID  READY,FLAG,A,377,0,1,YES ; WAIT FOR <CR>
012056  104443                                                          TRAP    C$GMAN
012060  000406                                                          BR      10000$
012062  012302                                                          .WORD   FLAG
012064  000152                                                          .WORD   T$CODE
012066  012304                                                          .WORD   READY
012070  000377                                                          .WORD   377
012072  000000                                                          .WORD   T$LOLIM
012074  000001                                                          .WORD   T$HILIM
012076                                                          10000$:
012076  000407                  BR      5$              ; CONTINUE

012100  012704  000310  3$:     MOV     #200.,R4        ; WAIT FOR 5 SECONDS
012104  004767  170306  4$:     JSR     PC,WAIT25       ;
012110                          BREAK                   ; ALLOW OPERATOR BREAK-IN
012110  104422                                                          TRAP    C$BRK
012112  005304                  DEC     R4              ; TIME UP?
012114  001373                  BNE     4$              ; IF NOT, WAIT LONGER
                                ;
                                ; NEXT PATTERN
                                ;
012116  032701  020000  5$:     BIT     #20000,R1       ; ALL PATTERNS DONE?
012122  001065                  BNE     12$             ; IF YES, END TEST
012124  062701  002000          ADD     #2000,R1        ; ELSE NEXT COLOUR
012130  032701  020000          BIT     #20000,R1       ; ALL PRESETS DONE?
012134  001736                  BEQ     T13A            ; IF NOT, NEXT COLOUR
```

TEST 13: PATTERN GENERATOR

```
                                        ;
                                        ; GENERATE CROSS HATCH
                                        ;
012136  005077  170032                  CLR     @DBUFF          ; SET PRESET COLOUR BLACK
012142  012777  000140  170022          MOV     #140,@VTVCSR    ; DO THE PRESET
012150  012701  177777                  MOV     #-1,R1          ; SET UP WAIT COUNTER
012154  105777  170012          6$:     TSTB    @VTVCSR         ; READY SET?
012160  100403                          BMI     7$              ; IF YES, BRANCH
012162                                  BREAK                   ; ELSE ALLOW OPERATOR BREAK IN
012162  104422                                                                  TRAP    C$BRK
012164  005301                          DEC     R1              ; WAITED TOO LONG?
012166  001372                          BNE     6$              ; IF NOT, WAIT LONGER
                                        ;
                                        ; WRITE FULL LINE
                                        ;
012170  012702  116000          7$:     MOV     #116000,R2      ; INITIALISE DOT NO.,WHITE
012174  010277  167774          8$:     MOV     R2,@DBUFF       ; WRITE 8 DOTS
012200  062702  000010                  ADD     #10,R2          ; NEXT DOTS
012204  032702  001000                  BIT     #1000,R2        ; LINE WRITTEN?
012210  001771                          BEQ     8$              ; IF NOT, DO MORE DOTS
012212  000414                          BR      11$             ; ELSE DO NEXT LINE
                                        ;
                                        ; WRITE PARTIAL LINE
                                        ;
012214  012702  016000          9$:     MOV     #16000,R2       ; SET DOT 0, COLOUR WHITE
012220  010277  167750          10$:    MOV     R2,@DBUFF       ; WRITE FIRST DOT
012224  062702  000037                  ADD     #31.,R2         ; END OF SQUARE
012230  010277  167740                  MOV     R2,@DBUFF       ; WRITE PENULTIMATE DOT
012234  005202                          INC     R2              ; NEXT DOT
012236  032702  001000                  BIT     #1000,R2        ; LINE FINISHED?
012242  001766                          BEQ     10$             ; IF NOT, GO BACK

012244  062777  000400  167720  11$:    ADD     #400,@VTVCSR    ; NEXT LINE
012252  103674                          BCS     2$              ; IF FINISHED, GO AND WAIT
012254  017703  167712                  MOV     @VTVCSR,R3      ; ELSE READ CSR
012260  042703  170377                  BIC     #170377,R3      ; GET LINE NO. LOW BITS
012264  001741                          BEQ     7$              ; IF ZERO, DO FULL LINE
012266  020327  007400                  CMP     R3,#7400        ; LAST LINE OF SQUARE?
012272  001736                          BEQ     7$              ; IF YES, DO FULL LINE
012274  000747                          BR      9$              ; ELSE PARTIAL LINE

012276                          12$:    EXIT    TST
012276  104432                                                                  TRAP    C$EXIT
012300  000044                                                                  .WORD   L10044-.

012302  000000                  FLAG:   .WORD   0

                                        .NLIST  BEX
012304      120     122     105 READY:  .ASCIZ  /PRESS 'RETURN' FOR NEXT PATTERN/
                                        .LIST   BEX
                                        .EVEN
012344                                  ENDTST
012344                                                          L10044:
012344  104401                                                                  TRAP    C$ETST
```

```
      1 012346                                          ENDMOD
      2
```

```
      1                                 .TITLE PARAMETER CODING
     12
     13                                 .SBTTL  HARDWARE PARAMETER CODING SECTION
     42
     43 012346                                  BGNMOD
     44
     45                                 ;++
     46                                 ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
     47                                 ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
     48                                 ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
     49                                 ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
     50                                 ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
     51                                 ; WITH THE OPERATOR.
     52                                 ;--
     53
     54 012346                                  BGNHRD
        012346  000004                                                          .WORD L10045-L$HARD/2
        012350                                                          L$HARD::
     55
     65 012350                                  GPRMA GETADR,0,0,160000,177776,YES
        012350  000031                                                          .WORD   T$CODE
        012352  012360                                                          .WORD   GETADR
        012354  160000                                                          .WORD   T$LOLIM
        012356  177776                                                          .WORD   T$HILIM
     66
     67
     68 012360                                  ENDHRD
                                                                                .EVEN
        012360                                                          L10045:
     69
     76
     77 012360      126     124     126  GETADR: .ASCIZ  /VTV31-K ADDRESS/
        012363      063     061     055
        012366      113     040     101
        012371      104     104     122
        012374      105     123     123
        012377      000
```

```
                                .SBTTL  SOFTWARE PARAMETER CODING SECTION

                                ;++
                                ; 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.
                                ;--

012400                                  BGNSFT
012400  000010                                                          .WORD L10046-L$SOFT/2
012402                                                          L$SOFT::
                                GPRML MANINT,0,1,YES
012402  000130                                                          .WORD   T$CODE
012404  012422                                                          .WORD   MANINT
012406  000001                                                          .WORD   1
                                GPRMD GETMAX,2,D,377,0,255.,YES
012410  001052                                                          .WORD   T$CODE
012412  012450                                                          .WORD   GETMAX
012414  000377                                                          .WORD   377
012416  000000                                                          .WORD   T$LOLIM
012420  000377                                                          .WORD   T$HILIM

                                .EVEN

012422                                  ENDSFT
                                                                .EVEN
012422                                                          L10046:


                                .NLIST  BEX
012422  122 125 116     MANINT: .ASCIZ  /RUN TEST 13 MANUALLY?/
012450  101 125 124     GETMAX: .ASCIZ  /AUTODROP ERROR COUNT/
                                .LIST   BEX
                                .EVEN

012476                  $PATCH::

                                .BLKW   50.             ; PATCH AREA
                                .BLKB   400-<.&377>     ; LASTAD SHIFT FOR LSI BUG
013000                          LASTAD
                                                                .EVEN
013000  013012                                                  .WORD T$FREE
013002  000003                                                  .WORD T$SIZE
013004                  L$LAST::
013004                          ENDMOD
```

```
      1
      2
     15
     16 013004                          BGNSETUP        1
     17 013004                          BGNPTAB
        013004  000000                                                    .WORD   0
        013006  000001                                                    .WORD   L10051-./2-1
        013010                                                  L10047:
     18 013010  174000                  .WORD   174000
     19 013012                          ENDPTAB
        013012                                                  L10051:
     20 013012                          ENDSETUP
     21         000001              .END
```

PARAMETER CODING MACRO V03.01 18-NOV-81 11:10:45 PAGE 52-1
SYMBOL TABLE

```
ABITS   002204 G    C$ESEG= 000005    E$LOAD= 000035    G$NO  = 000000    L$DTP   002040 G
ABORT   005042      C$ESUB= 000003    E100    006006    G$OFFS= 000400    L$DTYP  002034 G
ADR   = 000020 G    C$ETST= 000001    E101    006054    G$OFSI= 000376    L$DU    005562 G
ASSEMB= 000010      C$EXIT= 000032    E200    006266    G$PRMA= 000001    L$DUT   002072 G
BIT0  = 000001 G    C$GETB= 000026    E200B   003636    G$PRMD= 000002    L$DVTY  002210 G
BIT00 = 000001 G    C$GETW= 000027    E201    006314    G$PRML= 000000    L$EF    002052 G
BIT01 = 000002 G    C$GMAN= 000043    E202    006346    G$RADA= 000140    L$ENVI  002044 G
BIT02 = 000004 G    C$GPHR= 000042    E203    006405    G$RADB= 000000    L$ETP   002102 G
BIT03 = 000010 G    C$GPLO= 000030    E300    006544    G$RADD= 000040    L$EXP1  002046 G
BIT04 = 000020 G    C$GPRI= 000040    E300B   003663    G$RADL= 000120    L$EXP4  002064 G
BIT05 = 000040 G    C$INIT= 000011    E300C   003776    G$RADO= 000020    L$EXP5  002066 G
BIT06 = 000100 G    C$INLP= 000020    E400    006774    G$XFER= 000004    L$HARD  012350 G
BIT07 = 000200 G    C$MANI= 000050    E400B   004036    G$YES = 000010    L$HIME  002120 G
BIT08 = 000400 G    C$MEM = 000031    E401    007067    HELP  = 000000    L$HPCP  002016 G
BIT09 = 001000 G    C$MSG = 000023    E401B   004152    HOE   = 100000 G  L$HPTP  002022 G
BIT1  = 000002 G    C$OPEN= 000034    E500    007502    IBE   = 010000 G  L$HW    002160 G
BIT10 = 002000 G    C$PNTB= 000014    E501    007552    IDU   = 000040 G  L$ICP   002104 G
BIT11 = 004000 G    C$PNTF= 000017    E600    010114    IER   = 020000 G  L$INIT  004722 G
BIT12 = 010000 G    C$PNTS= 000016    E700    010452    ISR   = 000100 G  L$LADP  002026 G
BIT13 = 020000 G    C$PNTX= 000015    E800    010664    IXE   = 004000 G  L$LAST  013004 G
BIT14 = 040000 G    C$QIO = 000377    E800B   004257    I$AU  = 000041    L$LOAD  002100 G
BIT15 = 100000 G    C$RDBU= 000007    E800C   004344    I$AUTO= 000041    L$LUN   002074 G
BIT2  = 000004 G    C$REFG= 000047    E900    011122    I$CLN = 000041    L$MREV  002050 G
BIT3  = 000010 G    C$RESE= 000033    E900B   004427    I$DU  = 000041    L$NAME  002000 G
BIT4  = 000020 G    C$REVI= 000003    E901    011166    I$HRD = 000041    L$PRIO  002042 G
BIT5  = 000040 G    C$RFLA= 000021    E902    011247    I$INIT= 000041    L$PROT  004714 G
BIT6  = 000100 G    C$RPT = 000025    FLAG    012302    I$MOD = 000041    L$PRT   002112 G
BIT7  = 000200 G    C$SEFG= 000046    F$AU  = 000015    I$MSG = 000041    L$REPP  002062 G
BIT8  = 000400 G    C$SPRI= 000041    F$AUTO= 000020    I$PROT= 000040    L$REV   002010 G
BIT9  = 001000 G    C$SVEC= 000037    F$BGN = 000040    I$PTAB= 000041    L$SOFT  012402 G
BKGND   007500      C$TPRI= 000013    F$CLEA= 000007    I$PWR = 000041    L$SPC   002056 G
BOE   = 000400 G    DBUFF   002174 G  F$DU  = 000016    I$RPT = 000041    L$SPCP  002020 G
CHKMAX  002256 G    DFPTBL  002160 G  F$END = 000041    I$SEG = 000041    L$SPTP  002024 G
CONT    004736      DIAGMC= 000000    F$HARD= 000004    I$SETU= 000041    L$STA   002030 G
COUNT   002206 G    DOT     002200 G  F$HW  = 000013    I$SFT = 000041    L$SW    002164 G
C$AU  = 000052      DROPD   005610    F$INIT= 000006    I$SRV = 000041    L$TEST  002114 G
C$AUTO= 000061      EADD    004652    F$JMP = 000050    I$SUB = 000041    L$TIML  002014 G
C$BRK = 000022      EF.CON= 000036 G  F$MOD = 000000    I$TST = 000041    L$UNIT  002012 G
C$BSEG= 000004      EF.NEW= 000035 G  F$MSG = 000011    J$JMP = 000167    L10000  002162
C$BSUB= 000002      EF.PWR= 000034 G  F$PROT= 000021    KLINT   005162    L10001  002170
C$CEFG= 000045      EF.RES= 000037 G  F$PWR = 000017    LCLOCK  005074    L10002  002442
C$CLCK= 000062      EF.STA= 000040 G  F$RPT = 000012    LINDOT  004503    L10003  002472
C$CLEA= 000012      END     005050    F$SEG = 000003    LOE   = 040000 G  L10004  002546
C$CLOS= 000035      ERRCNT  002176 G  F$SOFT= 000005    LOT   = 000010 G  L10005  002600
C$CLP1= 000006      ER200   002444 G  F$SRV = 000010    L$ACP   002110 G  L10006  002632
C$CVEC= 000036      ER300   002474 G  F$SUB = 000002    L$APT   002036 G  L10007  002722
C$DCLN= 000044      ER400   002550 G  F$SW  = 000014    L$AUT   002070 G  L10010  003030
C$DODU= 000051      ER401   002602 G  F$TEST= 000001    L$AUTO  005424 G  L10011  003212
C$DRPT= 000024      ER500   002634 G  GETADR  012360    L$CCP   002106 G  L10012  003402
C$DU  = 000053      ER501   002724 G  GETMAX  012450    L$CLEA  005510 G  L10013  003474
C$EDIT= 000003      ER600   003032 G  GOOBAD  004571    L$CO    002032 G  L10014  003526
C$ERDF= 000055      ER700   003214 G  G$CNTO= 000200    L$DEPO  002011 G  L10015  003634
C$ERHR= 000056      ER800   003404 G  G$DELM= 000372    L$DESC  002220 G  L10017  005050
C$ERRO= 000060      ER900   003476 G  G$DISP= 000003    L$DESP  002076 G  L10020  005506
C$ERSF= 000054      ER902   003530 G  G$EXCP= 000400    L$DEVP  002060 G  L10021  005560
C$ERSO= 000057      EVL   = 000004 G  G$HILI= 000002    L$DISP  002124 G  L10022  005636
C$ESCA= 000010      E$END = 002100    G$LOLI= 000001    L$DLY   002116 G  L10023  006136
```

SYMBOL TABLE

```
L10024  005716          ONEFIL= 000001          TKB    = 177562          T$SUBN= 000000          T13A    012032
L10025  005744          O$APTS= 000000          TKS    = 177560          T$TAGL= 177777          T2      006140 G
L10026  006450          O$AU  = 000000          TPB    = 177566          T$TAGN= 010052          T2.1    006140
L10027  006260          O$BGNR= 000000          TPS    = 177564          T$TEMP= 000000          T3      006452 G
L10030  006604          O$BGNS= 000001          TTINT    005326          T$TEST= 000015          T4      006606 G
L10031  007176          O$DU  = 000001          T$ARGC= 000002          T$TSTM= 177777          T4A     006612
L10032  006672          O$ERRT= 000000          T$CODE= 001052          T$TSTS= 000001          T4.1    006606
L10033  006772          O$GNSW= 000001          T$ERRN= 001606          T$$AUT= 010020          T4.2    006674
L10034  007630          O$POIN= 000001          T$EXCP= 000000          T$$CLE= 010021          T5      007200 G
L10035  010162          O$SETU= 000001          T$FLAG= 000040          T$$DAT= 010051          T5A     007214
L10036  010516          PNT   = 001000 G        T$FREE= 013012          T$$DU = 010022          T6      007632 G
L10037  010712          PRI   = 002000 G        T$GMAN= 000000          T$$HAR= 010045          T6A     007640
L10040  011314          PRI00 = 000000 G        T$HILI= 000377          T$$HW = 010000          T6B     007750
L10041  011422          PRI01 = 000040 G        T$LAST= 000001          T$$INI= 010017          T7      010164 G
L10042  011606          PRI02 = 000100 G        T$LOLI= 000000          T$$MSG= 010015          T7A     010172
L10043  012026          PRI03 = 000140 G        T$LSYM= 010000          T$$PC = 000001          T7B     010304
L10044  012344          PRI04 = 000200 G        T$LTNO= 000015          T$$PRO= 010016          T8      010520 G
L10045  012360          PRI05 = 000240 G        T$NEST= 177777          T$$PTA= 010050          T8A     010552
L10046  012422          PRI06 = 000300 G        T$NS0 = 000000          T$$SEG= 010000          T8B     010600
L10047  013010          PRI07 = 000340 G        T$NS1 = 000005          T$$SOF= 010046          T9      010714 G
L10051  013012          READY   012304          T$NS2 = 000003          T$$SRV= 010002          T9A     010720
MANINT  012422          SETCLK  005052          T$NS3 = 000003          T$$SUB= 010033          T9B     011022
MANVEN  002164 G        SETUP   004756          T$PCNT= 000000          T$$SW = 010001          UAM   = 000200 G
MAXERR  002166 G        SFPTBL  002164 G        T$PTAB= 010050          T$$TES= 010044          UUT     002170 G
NCOLS   010112          SVCGBL= 000000          T$PTHV= 000001          T1      005640 G        VTVCSR  002172 G
NERRS   002350          SVCINS= 000001          T$PTNU= 000001          T1.1    005672          WAIT25  002416 G
NEXT    004764          SVCSUB= 000001          T$SAVL= 177777          T1.2    005720          X$ALWA= 000000
NLINS   010450          SVCTAG= 000001          T$SEGL= 177777          T10     011316 G        X$FALS= 000040
NOCLOK  005220          SVCTST= 000001          T$SEK0= 010000          T11     011424 G        X$OFFS= 000400
NXMFLG  002202 G        S$LSYM= 010000          T$SEK1= 010001          T12     011610 G        X$TRUE= 000020
NXMTRP  002430 G        TIMMSG  005354          T$SIZE= 000003          T13     012030 G        $PATCH  012476 G

. ABS.  013012      000
        000000      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  20500 WORDS  ( 81 PAGES)
DYNAMIC MEMORY AVAILABLE FOR  70 PAGES
ZVTVA0.BIN,ZVTVA0.LST=RT.MAC/M,ZVTVA0.SRC
```