

TK25

TK25 FRT END FUNC #3
CZTKGROCOPYRIGHT (c) 1984
AH-T280A-MC
FICHE 01 OF 02

JUL 1984

digital

Made In USA

TK25

TK25 FRT END FUNC #3
CZTKGAO

COPYRIGHT (c) 1984
AH-T780A-MC
FICHE 02 OF 02

JUL 1984
digital
Made In USA



.REM\

IDENTIFICATION

PRODUCT ID: AC-T779A MC
PRODUCT TITLE: CZTKGA TK25 FRT END FUNC #3
PRODUCT DATE: MARCH, 1984
DEPARTMENT: TAPE DIAGNOSTIC ENGINEERING
AUTHOR: DICE SYSTEMS, INC.

COPYRIGHT (C) 1984 BY
DIGITAL EQUIPMENT CORPORATION,
WESTBORO, MASSACHUSETTS.
ALL RIGHTS RESERVED.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

TABLE OF CONTENTS

1.0	ABSTRACT
2.0	REQUIREMENTS
2.1	HARDWARE REQUIREMENTS
2.2	SOFTWARE REQUIREMENTS
2.3	PREREQUISITES
3.0	OPERATING INSTRUCTIONS OPERATOR COMMANDS
3.1	OPERATOR COMMANDS
3.2	HARDWARE PARAMETERS
3.3	SOFTWARE PARAMETERS
4.0	OPERATING INSTRUCTIONS - SAMPLE PRINTOUTS
4.1	SUCCESSFUL RUN EXAMPLES
4.2	ERROR MESSAGES
5.0	PROGRAM RUN TIMES
5.1	RUN TIME - CZTKG
6.0	TEST DESCRIPTIONS - CZTKG
6.1	TEST 1 - SPACE RECORDS TEST
6.2	TEST 2 - REREADS TEST
6.3	TEST 3 - WRITE DATA RETRY TEST
6.4	TEST 4 - WRITE/READ TAPE MARK

1.0 ABSTRACT

THIS IS A PDP-11/LSI RESIDENT DIAGNOSTIC WHICH CHECKS THE FUNCTIONALITY OF AN TK25 MAGTAPE SUBSYSTEM WHILE CONNECTED TO A PDP-11 SYSTEM (Q-BUS OR UNIBUS). THE PROGRAM HAS BEEN DIVIDED INTO FOUR MAJOR PIECES: CZTKE, CZTKF, CZTKG, CZTKH. SUCCESSFUL RUN EXAMPLES, AND TEST DESCRIPTIONS HAVE BEEN PROVIDED FOR EACH PROGRAM.

THE PROGRAMS PROVIDE ERROR MESSAGES WHICH IDENTIFY FAILING FUNCTIONS, AND AID IN DEVICE REPAIR. REFERENCE THE FOLLOWING DIGITAL EQUIPMENT DOCUMENTS:

1. CIQPMAD XXDP. PROGRAMMER'S MANUAL; DOCUMENT NUMBER AC-S296A-AC;
DATE: 14 JULY 1980.

1.1 REVISION HISTORY
NEW RELEASE APRIL 1984

2.0 REQUIREMENTS

2.1 HARDWARE REQUIREMENTS

PDP-11 FAMILY PROCESSOR WITH 32K WORDS OF MEMORY
TK25 MAGTAPE SUBSYSTEM (DRIVE AND CONTROLLER)
CAUTION:DIAGNOSTIC REQUIRES 32K WORDS OF MEMORY
(28K USEABLE I.E. 4K FOR I/O PAGE)

2.1.1 OPTIONAL HARDWARE

FOUR TK25 CONTROLLERS PER PDP-11. ONE
DRIVE PER CONTROLLER

2.2 SOFTWARE REQUIREMENTS

PDP-11 DIAGNOSTIC SUPERVISOR (CIQPMAD VERSION 34 OR LATER)
PDP-11 DIAGNOSTIC LOADER/MONITOR (XXDP.)

2.3 PREREQUISITES

FUNCTIONAL PDP-11/LSI FAMILY CENTRAL PROCESSOR AND MEMORY
FUNCTIONAL CONSOLE TERMINAL
FUNCTIONAL STANDALONE DIAGNOSTIC SUPERVISOR

3.0 OPERATING INSTRUCTIONS OPERATOR COMMANDS

3.1 OPERATOR COMMANDS

THE TK25 DIAGNOSTICS ARE PDP-11 DIAGNOSTIC SUPERVISOR COMPATIBLE PROGRAMS.
ALL LOADING AND RUN TIME INSTRUCTIONS CAN BE REFERENCED IN THE PDP-11
PROGRAMMER'S MANUAL "CIQPMAO XXDP" PROGRAMMER'S MANUAL NUMBER AC-S296A-AC.

BOOT THE DIAGNOSTIC XXDP. MEDIA (OPERATOR RESPONSES ARE UNDERLINED)

CHMOLEO XXDP. DL MONITOR
BOOTED VIA UNIT 0
28K NON UNIBUS SYSTEM

ENTER DATE <DD-MMM-YY>: 29-JAN-82

RESTART ADDRESS: 152010 -----
THIS IS XXDP. TYPE "H" OR "H/L" FOR HELP.

.R CZTKGA

- -----
CZTKGA.BIC

DRS-EO
CZTKG-A-0
CZTKGA TK-25 FRT END FUNC #3 UNIT IS TK25
RSTRT ADR 147642
DR>START/FLAG:PNT:MOE

THE ABOVE COMMAND WILL START THE DIAGNOSTIC. THE COMMAND HAS TWO
SWITCHES ON WHICH ARE "PRINT EACH TEST NBR. AS EXECUTED" AND "HALT ON
ERROR".

3.2 HARDWARE PARAMETERS

AFTER INITIAL STARTING OF THE PROGRAM (START COMMAND TO THE DIAGNOSTIC SUPERVISOR), THE PROGRAM WILL ISSUE THE "CHANGE HW?" QUESTION TO ASK IF THE HARDWARE PARAMETERS ARE TO BE CHANGED (BY THE OPERATOR).

ON A "N" (NO) RESPONSE TO THE QUESTION, THE PROGRAM WILL USE IT'S DEFAULT HARDWARE PARAMETER VALUES. IT WILL DEFAULT TO ONE UNIT SELECTED (UNIT 0), THE DEFAULT TSBA/TSDB WILL BE 172522 AND THE INTERRUPT VECTOR WILL BE 224.

ON A "Y" (YES) RESPONSE TO THE QUESTION, THE FOLLOWING QUESTIONS WILL THEN BE ASKED TO ALLOW THE OPERATOR TO SELECT THE UNITS TO BE TESTED. A VALUE, IF PRESENT, LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT VALUE THAT WILL BE TAKEN ONLY IF A CARRIAGE RETURN IS TYPED AS A RESPONSE. A "(D)" IN A QUESTION INDICATES THAT A DECIMAL NUMBER IS REQUIRED AS A RESPONSE. AN "(O)" INDICATES AN OCTAL NUMBER IS BEING SOLICITED. AN "(L)" THAT A LOGICAL RESPONSE IS TO BE MADE: "Y" FOR YES, "N" FOR NO.

UNITS (D) ? < ENTER THE NUMBER OF CONTROLLERS
PRESENT TO BE TESTED >

UNIT 0

DEVICE ADDRESS (O) 172522 ? <ENTER THE ADDRESS OF THE
TSBA/TSDB REGISTER>

VECTOR (O) 224 ? <ENTER ADDRESS OF INTERRUPT
VECTOR>

THE ADDRESS AND VECTOR QUESTIONS WILL BE ASKED FOR EACH OF THE NUMBER OF UNITS (CONTROLLERS) SPECIFIED IN THE " UNITS ?" QUESTION. LOGICAL UNIT NUMBERS ARE ASSIGNED IN ORDER BEGINNING AT 0. UP TO EIGHT UNITS CAN BE SELECTED FOR TESTING.

3.3 SOFTWARE PARAMETERS

THE FOLLOWING QUESTIONS ARE ASKED ON A START, RESTART, OR CONTINUE. THEY ALLOW FLEXIBILITY IN THE WAY THE PROGRAM BEHAVES.

CHANGE SW (L) ? < TYPE "Y" TO CAUSE THE FOLLOWING
QUESTIONS TO BE ASKED.>

INHIBIT ITERATIONS (L) N ? < TYPE "Y" TO PREVENT MULTIPLE
ITERATIONS OF CERTAIN TESTS.
THIS CAUSES EACH TEST PASS TO
RUN AS QUICKLY AS POSSIBLE.
ONLY QUICK-RUNNING LOGIC
TESTS USE MULTIPLE ITERATIONS.>

ENABLE CONTROLLER RAM DUMP ON ERROR (L) N? < TYPE "Y" TO DUMP
SELECTED RAM CONTENTS IN THE
CONTROLLER MODULE.>

4.0 OPERATING INSTRUCTIONS - SAMPLE PRINTOUTS

4.1 SUCCESSFUL RUN EXAMPLES

4.1.1 SUCCESSFUL RUN EXAMPLE - CZTKG

TST: 001 SPACE RECORDS TEST
TST: 002 REREADS TEST
TST: 003 WRITE DATA RETRY TEST
TST: 004 WRITE TAPE MARK TEST
CZTKG EOP 1
0 TOTAL ERRS

NOTE: PROGRAM NOW STARTS OVER AGAIN AT TEST 1

4.2 OPERATING INSTRUCTIONS - SAMPLE ERROR MESSAGES

ERROR MESSAGE EXAMPLE 1

TST: 001 SPACE RECORDS TEST
CZTKG HRD ERR 00120 ON UNIT 00 TST 001 SUB 003 PC: 025214
TAPE NOT AT BOT AFTER REWIND COMMAND

EXPD: 002022 RECV: 002020 XOR: 000002

ERROR MESSAGE EXAMPLE 2

CZTKG HRD ERR 00122 ON UNIT 00 TST 001 SUB 003 PC: 025332
TSSR NOT CORRECT AFTER POSITION (SPACE) COMMAND

TSSR=100306
TSSR BITS SET: SC, SSR, OFL
TERMINATION CODE = FUNCTION REJECT
*****CHECK CABLES BETWEEN CONTROLLER AND TRANSPORT*****
PACKET ADDRESS =030240
PACKET WORD #0 =140410
PACKET WORD #1 =000001
PACKET WORD #2 =000000
PACKET WORD #3 =000000

MESSAGE BUFFER ADDRESS =030130
MESSAGE BUFFER CONTENTS:
MESSAGE BUFFER HEADER =101021
DATA FIELD LENGTH =000012
RESIDUAL BYTE COUNTER =000000
XSTAT0 CONTENTS =000012
XSTAT1 CONTENTS =000000
XSTAT2 CONTENTS =001000
XSTAT3 CONTENTS =000040

5.0 PROGRAM RUN TIMES

THE AVERAGE RUN TIMES OF THE PROGRAMS ARE LISTED BELOW. THESE FIGURES ARE TO BE USED AS A GUIDE. THE TIMING WAS DONE ON A PDP-11/23 (LSI) PROCESSOR WITH A LA-120 CONSOLE.

THE PROGRAMS RUN IN NON-ITERATIVE MODE. EACH TEST IS RUN ONCE, WITH NO ITERATIONS. THEREFOR, THE DEFAULT MODE (NORMALLY ITERATIVE) AND THE NON ITERATIVE MODE TIMES ARE IDENTICAL.

5.1 RUN TIMES - CZTKG

TEST NUMBER	N/I SECS.	DEF SECS.
1	65	65
2	130	130
3	120	120
4	35	35

THE TIMES REQUIRED TO RUN TESTS 1 THROUGH 4 IN ONE COMMAND:

Q.V.	5 MINS 50 SECONDS
DEFAULT	5 MINS 50 SECONDS

6.0 TEST DESCRIPTIONS - CZTKG

6.1 TEST 1 - SPACE RECORDS TEST

* NOTE: THIS TEST MUST HAVE A GOOD MAGTAPE IN THE DRIVE *
* ANY TAPE ERRORS WILL BE DISPLAYED AS TAPE STATUS ALERT *

THIS TEST VERIFIES THAT THE SPACE RECORDS FORWARD AND SPACE RECORDS REVERSE POSITION COMMANDS OPERATE PROPERLY WHEN SPACING OVER NORMAL DATA RECORDS. OPERATION WHEN SPACING OVER TAPE MARKS IS VERIFIED IN A SUBSEQUENT TEST. THE BASIC WRITE DATA TEST SHOULD HAVE BEEN RUN SUCCESSFULLY FOR THIS TEST TO PRODUCE MEANINGFUL RESULTS. THIS TEST CONSISTS OF A SERIES OF SUBTESTS. IN EACH OF THE SUBTESTS, THE TAPE IS ENTIRELY WRITTEN WITH RECORDS OF VARYING SIZES AND DATA CONTENT; THE FIRST 4 BYTES OF EACH RECORD INDICATE THAT RECORD'S RELATIVE POSITION ON TAPE. AFTER EACH SPACING OPERATION, THE TAPE POSITION IS VERIFIED BY READING THE NEXT OR PREVIOUS RECORD AND COMPARING THE POSITION DATA WITH THE EXPECTED RESULT.

6.1.1 TEST 1, SUBTEST 1: -

THIS SUBTEST VERIFIES THAT A SPACE RECORDS FORWARD COMMAND WITH THE CLEAR VOLUME CHECK (CVC) BIT CLEAR IS REJECTED IF THE VOLUME CHECK FLAG (VCK) IS SET.

6.1.2 TEST 1, SUBTEST 2: -

THIS SUBTEST VERIFIES THAT A SPACE RECORDS REVERSE COMMAND WITH THE CLEAR VOLUME CHECK (CVC) BIT CLEAR IS REJECTED IF
THE VOLUME CHECK (VCK) FLAG IS
SET.

6.1.3 TEST 1, SUBTEST 3: -

THIS SUBTEST VERIFIES THAT SPACE RECORDS FORWARD CAN SPACE ONE RECORD OFF BOT AND CAUSE BOT STATUS TO BE CLEARED.

6.1.4 TEST 1, SUBTEST 4: -

THIS SUBTEST VERIFIES THAT SPACE RECORDS REVERSE CAN SPACE BACK OVER THE FIRST RECORD ON TAPE.

6.1.5 TEST 1, SUBTEST 5: -

THIS SUBTEST VERIFIES THAT SPACE RECORDS FORWARD CAN SPACE A MULTIPLE NUMBER OF RECORDS (2 THROUGH 64K, OR THE MAXIMUM NUMBER OF RECORDS WRITTEN ON THE TAPE, WHICHEVER IS LESS.).

6.1.6 TEST 1, SUBTEST 6: -

THIS SUBTEST VERIFIES THAT SPACE RECORDS REVERSE CAN SPACE A MULTIPLE NUMBER OF RECORDS (2 THROUGH 64K, OR THE MAXIMUM NUMBER OF RECORDS WRITTEN ON THE TAPE, WHICH EVER IS LESS).

6.1.7 TEST 1, SUBTEST 7: -

THIS SUBTEST VERIFIES THAT SPACE RECORDS REVERSE ISSUED WHILE TAPE IS AT EOT RESULTS IN FUNCTION REJECT TERMINATION WITH THE NONEXECUTABLE FUNCTION (NEF) ERROR BIT SET.

6.1.8 TEST 1, SUBTEST 8: -

THIS SUBTEST VERIFIES THAT A SPACE RECORDS REVERSE COMMAND THAT CAUSES THE TAPE TO RUN INTO BOT (WITH THE TAPE NOT INITIALLY AT BOT) CAUSES A TAPE STATUS ALERT TERMINATION AND SETS THE REVERSE INTO BOT (RIB) STATUS BIT.

6.2 TEST 2 - REREADS TEST

THIS TEST VERIFIES THAT THE REREAD PREVIOUS AND REREAD NEXT COMMANDS OPERATE PROPERLY. VARIOUS COMBINATIONS OF ODD AND EVEN DATA BUFFER BOUNDRIES, RECORD SIZES (UP TO 64K BYTES IF MEMORY SPACE IS AVAILIABLE), AND BYTE-SWAP (SWP) AND OPPOSITE (OPP) CONRTOL ARE USED. ALSO TESTED ARE PROPER TERMINATIONS ON EXCEPTIONAL OR ERROR CONDITIONS: RECORD LENGTH LONG, RECORD LENGTH SHORT, READ REVERSE AT BOT, ILLEGAL DATA BUFFER ADDRESSES, AND DATA BUFFERS IN NONEXISTENT MEMORY.

6.2.1 TEST 2, SUBTEST 1: -

THIS SUBTEST VERIFIES THAT THE REREAD PREVIOUS COMMAND WITH OPP=0 AND SWB=0 OPERATES PROPERLY. THE TAPE IS FIRST REWOUND AND THEN WRITTEN WITH A SERIES OF T/PE RECORDS VARYING IN LENGTH AND DATA CONTENT. THE TAPE IS THEN REWOUND AGAIN. FOR EACH RECORD THE TAPE IS SPACED FORWARD ONE RECORD AND THE REREAD PREVIOUS COMMAND IS ISSUED. RESULTS (STATUS, DATA, ETC.) ARE VERIFIED. THE BYTE COUNT ON EACH REREAD PREVIOUS COMMAND IS SET TO THE LENGTH OF THE EXPECTED RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD OCCUR.

6.2.2 TEST 2, SUBTEST 2: -

THIS SUBTEST VERIFIES THAT THE REREAD PREVIOUS COMMAND WITH OPP=0 AND SWB=1 OPERATES PROPERLY. THE TEST SEQUENCE IS THE SAME AS THAT USED IN SUBTEST 1, BUT IT IS VERIFIED THAT DATA STORED BY THE COMMAND CONTAINS SWAPPED BYTES

6.2.3 TEST 2 SUBTEST 3: -

THIS SUBTEST VERIFIES THAT THE REREAD PREVIOUS COMMAND WITH OPP=1 (READ REVERSE, SPACE FORWARD) AND SWB=0 OPERATES PROPERLY. THE TAPE IS FIRST REWOUND, AND THEN WRITTEN WITH A SERIES OF TEST RECORDS OF VARYING LENGTH AND DATA CONTENT; THE FIRST FOUR BYTES OF EACH RECORD CONTAIN ITS RECORD NUMBER (INDICATING POSITION ON TAPE). THE TAPE IS THEN REWOUND AGAIN. FOR EACH TEST RECORD THE FOLLOWING SEQUENCE IS EXECUTED:

1. THE REREAD PREVIOUS COMMAND WITH OPP=1 IS ISSUED AND THE RESULTS ARE CHECKED.
2. A READ FORWARD COMMAND IS THEN ISSUED AND THE DATA IS CHECKED TO VERIFY THAT THE TAPE WAS POSITIONED PROPERLY AFTER THE REREAD PREVIOUS COMMAND (E.G. THE TAPE SHOULD HAVE BEEN LEFT POSITIONED AT THE START OF THE TEST RECORD.). THE READ FORWARD COMMAND LEAVES THE TAPE POSITIONED PROPERLY AT THE START OF THE NEXT RECORD.

THE BYTE COUNT ON EACH REREAD PREVIOUS COMMAND IS SET TO THE LENGTH OF THE EXPECTED RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD OCCUR.

6.2.4 TEST 2, SUBTEST 4:

THIS SUBTEST VERIFIES THAT THE REREAD PREVIOUS COMMAND WITH OPP=1 AND SWB=1 OPERATES PROPERLY. THE TEST SEQUENCE IS THE SAME AS THAT USED IN SUBTEST 3, BUT IT IS VERIFIED THAT DATA STORED BY THE COMMAND CONTAINS SWAPPED BYTES.

6.2.5 TEST 2, SUBTEST 5: -

THIS SUBTEST VERIFIES THAT A REREAD PREVIOUS COMMAND READING A RECORD LONGER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH LONG (RLL) BIT SET. RESULTS ARE VERIFIED FOR BOTH STATES OF OPP (0 AND 1).

6.2.6 TEST 2, SUBTEST 6: -

THIS SUBTEST VERIFIES THAT A REREAD PREVIOUS COMMAND READING A RECORD SHORTER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH SHORT (RLS) BIT SET. IT IS VERIFIED THAT THE RESIDUAL BYTE COUNTER (RBPCR) IN THE MESSAGE BUFFER CONTAINS THE APPROPRIATE NONZERO VALUE (E.G THE DIFFERENCE BETWEEN THE ORIGINAL BYTE COUNT AND THE ACTUAL RECORD LENGTH). RESULTS ARE VERIFIED FOR BOTH STATES OF OPP (0 AND 1).

6.2.7 TEST 2, SUBTEST 7: -

THIS SUBTEST VERIFIES THAT THE REREAD NEXT COMMAND WITH OPP=0 AND SWB=0 OPERATES PROPERLY. THE TAPE IS FIRST REWOUND AND THEN WRITTEN WITH A SERIES OF TEST RECORDS OF VARYING LENGTH AND DATA CONTENT. THE TAPE IS THEN REWOUND AGAIN. FOR EACH TEST RECORD THE TAPE IS SPACED FORWARD ONE RECORD AND A REREAD NEXT COMMAND IS ISSUED. RESULTS (STATUS, DATA, ETC.) ARE VERIFIED. THE BYTE COUNT ON EACH REREAD NEXT COMMAND IS SET TO THE LENGTH OF THE EXPECTED RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD OCCUR.

6.2.8 TEST 2, SUBTEST 8:

THIS SUBTEST VERIFIES THAT THE REREAD NEXT COMMAND WITH OPP=0 AND SWB=1 OPERATES PROPERLY. THE TEST SEQUENCE IS THE SAME AS THAT USED IN SUBTEST 1, BUT IT IS VERIFIED THAT DATA STORED BY THE COMMAND CONTAINS SWAPPED BYTES.

6.2.9 TEST 2, SUBTEST 9:

THIS SUBTEST VERIFIES THAT THE REREAD NEXT COMMAND WITH OPP=1 (READ FORWARD, SPACE REVERSE) AND SWB=0 OPERATES PROPERLY. THE TAPE IS FIRST REWOUND AND THEN WRITTEN WITH A SERIES OF TAPE RECORDS VARYING IN LENGTH AND DATA CONTENT. THE FIRST FOUR BYTES OF EACH RECORD CONTAIN ITS RECORD NUMBER (INDICATING POSITION ON TAPE). THE TAPE IS THEN REWOUND AGAIN. FOR EACH TEST RECORD THE FOLLOWING SEQUENCE IS EXECUTED:

1. THE REREAD NEXT COMMAND WITH OPP=1 IS ISSUED AND THE RESULT IS CHECKED.
2. A READ FORWARD COMMAND IS THEN ISSUED AND THE DATA IS CHECKED TO VERIFY THAT THE TAPE WAS POSITIONED PROPERLY AFTER THE REREAD NEXT COMMAND (E.G. THE TAPE SHOULD HAVE BEEN LEFT POSITIONED AT THE START OF THE TEST RECORD). THE READ FORWARD COMMAND LEAVES THE TAPE POSITIONED PROPERLY AT THE START OF THE NEXT TEST RECORD.

THE BYTE COUNT ON EACH REREAD NEXT COMMAND IS SET TO THE LENGTH OF THE EXPECTED RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD OCCUR.

6.2.10 TEST 2, SUBTEST 10: -

THIS SUBTEST VERIFIES THAT THE REREAD NEXT COMMAND WITH OPP=1 AND SWB=1 OPERATES PROPERLY. THE TEST SEQUENCE IS THE SAME AS THAT USED IN SUBTEST 3, BUT IT IS VERIFIED THAT DATA STORED BY THE COMMAND CONTAINS SWAPPED BYTES.

6.2.11 TEST 2, SUBTEST 11: -

THIS SUBTEST VERIFIES THAT A REREAD NEXT COMMAND READING A RECORD LONGER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH LONG (RL) BIT SET. RESULTS ARE VERIFIED FOR BOTH STATES OF OPP (1 AND 0).

6.2.12 TEST 2, SUBTEST 12: -

THIS SUBTEST VERIFIES THAT A REREAD NEXT COMMAND READING A RECORD SHORTER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH SHORT (RLS) BIT SET. IT IS VERIFIED THAT THE RESIDUAL BYTE COUNTER IN THE MESSAGE BUFFER CONTAINS THE PROPER NONZERO MESSAGE (E.G. THE DIFFERENCE BETWEEN THE ORIGINAL BYTE COUNT AND THE ACTUAL RECORD LENGTH). RESULTS ARE VERIFIED FOR BOTH STATES OF OPP (0 AND 1).

6.2.13 TEST 2, SUBTEST 13. -

THIS SUBTEST VERIFIES THAT A DATA BUFFER ADDRESS REFERENCING NONEXISTANT MEMORY RECOVERABLE ERROR TERMINATION (TC=4 OR 5) WITH NXM=1 AND THAT THE TAPE IS ULTIMATELY POSITIONED PROPERLY. ALL COMBINATIONS OF REREAD PREVIOUS/NEXT AND OPP=0/1 ARE TESTED.

6.2.14 TEST 2, SUBTEST 14: -

THIS SUBTEST VERIFIES THAT THE REREAD PREVIOUS WITH OPP=0 (SPACE REVERSE, READ FORWARD) AND REREAD PREVIOUS WITH OPP=1 (READ REVERSE SPACE FORWARD) ISSUED WHEN THE TAPE IS POSITIONED AT BOT CAUSES FUNCTION REJECT TERMINATION WITH THE NONEXECUTABLE FUNCTION (NEF) ERROR BIT SET.

6.2.15 TEST 2, SUBTEST 15: -

THIS SUBTEST VERIFIES THAT THE REREAD PREVIOUS WITH OPP=1 (SPACE REVERSE, READ FORWARD) AND REREAD PREVIOUS WITH OPP=0 (READ REVERSE, SPACE FORWARD) ISSUED WHEN THE TAPE POSITIONED JUST BEFORE THE FIRST RECORD ON TAPE (BUT NOT AT BOT) CAUSES TAPE STATUS ALERT TERMINATION WITH THE REVERSE INTO BOT (RIB) STATUS BIT SET.

6.3 TEST 3 - WRITE DATA RETRY TEST

* NOTE: THIS TAPE MUST HAVE A GOOD MAGTAPE IN THE DRIVE *
* ANY TAPE ERRORS WILL BE DISPLAYED AS TAPE STATUS ALERT. *

THIS TEST VERIFIES PROPER OPERATION OF THE WRITE DATA RETRY COMMAND (SPACE REVERSE, ERASE, WRITE DATA). THE TEST CONSISTS OF THE FOLLOWING FIVE SUBTESTS.

6.3.1 TEST 3, SUBTEST 1: -

THIS SUBTEST VERIFIES THAT A WRITE DATA RETRY COMMAND ISSUED WHILE THE TAPE IS POSITIONED AT BOT CAUSES FUNCTION REJECT TERMINATION WITH THE NON EXECUTABLE FUNCTION (NEF) ERROR BIT SET.

6.3.2 TEST 3, SUBTEST 2: -

THIS SUBTEST VERIFIES THAT A WRITE DATA RETRY COMMAND ISSUED WHILE THE TAPE IS POSITIONED BEFORE THE FIRST RECORD ON TAPE (BUT NOT AT BOT) CAUSES TAPE STATUS ALERT TERMINATION, WITH THE REVERSE INTO BOT (RIB) STATUS ERROR BIT SET.

6.3.3 TEST 3, SUBTEST 3: -

THIS SUBTEST VERIFIES THAT A WRITE DATA RETRY COMMAND WITH SWB=0 TERMINATES PROPERLY AND WRITES CORRECT DATA ON TAPE (THE WRITTEN RECORD IS READ AND CHECKED). VARIOUS BYTE COUNTS AND DATA PATTERNS ARE USED.

6.3.4 TEST 3, SUBTEST 4: -

THIS SUBTEST VERIFIES THAT A WRITE DATA RETRY COMMAND WITH SWB=1 TERMINATES PROPERLY AND WRITES CORRECT DATA ON TAPE (THE WRITTEN RECORD IS READ AND CHECKED). VARIOUS BYTE COUNTS AND DATA PATTERNS ARE USED.

6.3.5 TEST 3, SUBTEST 5: -

THIS SUBTEST VERIFIES THAT A WRITE DATA RETRY COMMAND IS PERFORMING THE ERASE PART OF THE OPERATION BY COMPLETING THE FOLLOWING STEPS:

1. THE TAPE IS REWOUND AND A SERIES OF RECORDS ARE WRITTEN WITH THE NORMAL WRITE DATA COMMAND. THIS SHOULD RESULT IN RECORDS

SEPERATED BY THE STANDARD INTERRECORD GAP.

2. A PROGRAM TIMING VALUE IS CALIBRATED BY REWINDING THE TAPE AND THEN CONTINUING THE NUMBER OF CYCLES THROUGH A PROGRAMMED LOOP REQUIRED TO SPACE OVER THE SERIES OF RECORDS WRITTEN IN THE PREVIOUS STEP.
3. THE TAPE IS AGAIN REWOUND AND THE SAME SERIES OF RECORDS WRITTEN AGAIN, THIS TIME USING THE WRITE DATA RETRY COMMAND. THIS SHOULD RESULT IN RECORDS SEPERATED BY A A LONG INTERRECORD GAP.
4. THE TAPE IS AGAIN REWOUND, THE SPACING COMMAND ISSUED, AND THE NUMBER OF TIMING LOOP CYCLES COUNTED TO COMPLETE THE OPERATION.
5. THE TWO LOOPS ARE COMPARED, CHECKING TO SEE THAT THEY DIFFER BY A SIGNIFICANT AMOUNT.

6.4 TEST 4 - WRITE/READ TAPE MARK

 * NOTE; THIS TEST MUST HAVE A GOOD MAGTAPE IN THE DRIVE *
 * ANY TAPE ERRORS WILL BE DISPLAYED AS A TAPE STATUS ALERT *

THIS TEST VERIFIES THAT THE WRITE TAPE MARK COMMAND OPERATES PROPERLY. IT IS VERIFIED THAT THE TAPE MARK IS WRITTEN ONTO TAPE BY CHECKING THAT THE READ AND SPACE RECORDS COMMANDS DETECT THE TAPE MARK. IN ADDITION, SINCE WRITE TAPE MARK IS THE FIRST SUBCOMMAND UNDER THE FORMAT COMMAND BEING TESTED, IT IS VERIFIED THAT THE CLEAR VOLUME CHECK (CVC) BIT OPERATES PROPERLY AND THAT FORMAT COMMANDS WITH ILLEGAL MODE CODES ARE REJECTED.

6.4.1 TEST 4, SUBTEST 1: -

THIS SUBTEST VERIFIES THAT A FORMAT COMMAND (WITH ANY LEGAL MODE CODE) WITH THE CLEAR VOLUME CHECK (CVC) BIT CLEAR IS REJECTED IF THE VOLUME CHECK (VCK) FLAG IS SET. ALL VALID MODE CODES ARE CHECKED.

6.4.2 TEST 4, SUBTEST 2: -

THIS SUBTEST VERIFIES THAT A FORMAT COMMAND WITH AN ILLEGAL MODE CODE CAUSES FUNCTION REJECT TERMINATION WITH THE ILLEGAL COMMAND (ILC) ERROR BIT SET. ALL ILLEGAL MODE CODES ARE CHECKED.

6.4.3 TEST 4, SUBTEST 3: -

THIS SUBTEST VERIFIES THAT WRITE TAPE MARK COMMANDS OPERATE PROPERLY, AND THAT READ COMMANDS SUBSEQUENTLY ISSUED TO DETECT THE WRITTEN TAPE MARKS TERMINATE WITH TAPE STATUS ALERT AND THE TAPE MARK DETECTED (TMK) STATUS BIT SET. THE FOLLOWING SEQUENCE IS EXECUTED:

1. THE CONTROLLER IS INITIALIZED AND THE TAPE REWOUND. THIS SETS THE VOLUME CHECK (VCK) STATUS BIT.
2. A WRITE TAPE MARK COMMAND, WITH CVC=1, IS ISSUED AND PROPER TERMINATION AND STATUS IS VERIFIED (I.E. VCK=0, AND TMK=1).
3. SEVERAL MORE WRITE TAPE MARK COMMANDS, THESE WITH CVC=0, ARE ISSUED AND PROPER TERMINATION (NORMAL) AND STATUS (TMK) VERIFIED.
4. A READ REVERSE COMMAND IS ISSUED AND PROPER TERMINATION (TAPE STATUS ALERT) AND STATUS (TMK) VERIFIED. IT IS ALSO VERIFIED THAT NO DATA IS TRANSFERRED INTO MEMORY.
5. A SPACE RECORDS REVERSE COMMAND IS ISSUED AND PROPER TERMINATION

(TAPE STATUS ALERT) AND STATUS (TMK) VERIFIED.

6. THE TAPE IS REWOUND AND A READ FORWARD COMMAND IS ISSUED AND PROPER TERMINATION (TAPE STATUS ALERT) AND STATUS (TMK) VERIFIED. IT IS ALSO VERIFIED THAT NO DATA IS TRANSFERRED INTO MEMORY.
7. A SPACE RECORDS REVERSE COMMAND THAT CONTAINS A RECORD COUNT GREATER THAN 1 IS ISSUED, AND IT IS VERIFIED THAT TAPE STATUS ALERT TERMINATION OCCURED, TMK=1 AND THAT RBPCR (RESIDUAL BYTE/RECORD COUNTER) CONTAINS THE PROPER NONZERO VALUE. THIS OPERATION VERIFIES THAT DETECTION OF THE TAPE MARK CAUSE THE SPACE RECORDS OPERATION TO BE PREMATURELY TERMINATED. THIS SHOULD LEAVE THE DEVICE POSITIONED JUST BEFORE THE FIRST RECORD ON THE TAPE.
8. TAPE POSITION IS VERIFIED BY ISSUING ANOTHER SPACE RECORDS REVERSE COMMAND AND VERIFYING THAT TAPE STATUS ALERT TERMINATION OCCURS, WITH THE REVERSE INTO BOT (RIB) ERROR STATUS BIT SET.
9. A SPACE RECORDS FORWARD COMMAND THAT CONTAINS A RECORD GREATER THAN 1 IS ISSUED AND IT IS VERIFIED THAT TAPE STATUS ALERT TERMINATION OCCURED, TMK=1, AND THAT RBPCR (RESIDUAL BYTE/RECORD COUNTER) CONTAINS THE PROPER NONZERO VALUE. THIS OPERATION VERIFIES THAT DETECTION OF THE TAPE MARK CAUSES THE SPACE RECORDS OPERATIONS TO PREMATURELY TERMINATE.

```

752      .SBTTL  PROGRAM HEADER
758      .MCALL  SVC
759 000000      SVC          ; INITIALIZE SUPERVISOR MACROS
760      .ENABLE LC
761      .NLIST  BEX,CND
767 000000      .ENABL  AMA,ABS
768      .      2000
769 002000      BGNMOD  TUV2A
      002000      TUV2A::
770
771      ;**
772      ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
773      ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
774      ;--
775
776
777 002000      POINTER BGNSW,BGNSFT,BGNAU,BGNDU,BGNRPT,BGNSETUP
778 002000      HEADER  CZTKG,A,0,655,,0
      002000      L$NAME::          ;DIAGNOSTIC NAME
      002000          103          .ASCII /C/
      002001          132          .ASCII /Z/
      002002          124          .ASCII /T/
      002003          113          .ASCII /K/
      002004          107          .ASCII /G/
      002005          000          .BYTE  0
      002006          000          .BYTE  0
      002007          000          .BYTE  0
      002010      L$REV::          ;REVISION LEVEL
      002010          101          .ASCII  /A/
      002011      L$DEPO::          ;0
      002011          060          .ASCII  /O/
      002012      L$UNIT::          ;NUMBER OF UNITS
      002012 000001          .WORD   T$PTHV
      002014      L$TIML::          ;LONGEST TEST TIME
      002014 001217          .WORD   655.
      002016      L$HPCP::          ;POINTER TO H.W. QUES.
      002016 065460          .WORD   L$HARD
      002020      L$SPCP::          ;POINTER TO S.W. QUES.
      002020 065620          .WORD   L$SOFT
      002022      L$HPTP::          ;PTR. TO DEF. H.W. PTABLE
      002022 002124          .WORD   L$HW
      002024      L$SPTP::          ;PTR. TO S.W. PTABLE
      002024 002134          .WORD   L$SW
      002026      L$LADP::          ;DIAG. END ADDRESS
      002026 066026          .WORD   L$LAST
      002030      L$STA::          ;RESERVED FOR APT STATS
      002030 000000          .WORD   0
      002032      L$CO::          ;
      002032 000000          .WORD   0
      002034      L$DTYP::          ;DIAGNOSTIC TYPE
      002034 000000          .WORD   0
      002036      L$APT::          ;APT EXPANSION
      002036 000000          .WORD   0
      002040      L$DTP::          ;PTR. TO DISPATCH TABLE
      002040 066012          .WORD   L$DISPATCH
      002042      L$PRIO::          ;DIAGNOSTIC RUN PRIORITY
      002042 000000          .WORD   0

```

002044		L\$ENVI::		;FLAGS DESCRIBE HOW IT WAS SETUP
002044	000000		.WORD 0	
002046		L\$EXP1::		;EXPANSION WORD
002046	000000		.WORD 0	
002050		L\$MREV::		;SVC REV AND EDIT #
002050	003		.BYTE C\$REVISION	
002051	003		.BYTE C\$EDIT	
002052		L\$EF::		;DIAG. EVENT FLAGS
002052	000000		.WORD 0	
002054	000000		.WORD 0	
002056		L\$SPC::		
002056	000000		.WORD 0	
002060		L\$DEVP::		; POINTER TO DEVICE TYPE LIST
002060	003334		.WORD L\$DVTYP	
002062		L\$REPP::		;PTR. TO REPORT CODE
002062	023066		.WORD L\$RPT	
002064		L\$EXP4::		
002064	000000		.WORD 0	
002066		L\$EXP5::		
002066	000000		.WORD 0	
002070		L\$AUT::		;PTR. TO ADD UNIT CODE
002070	022560		.WORD L\$AU	
002072		L\$DUT::		;PTR. TO DROP UNIT CODE
002072	022656		.WORD L\$DU	
002074		L\$LUN::		;LUN FOR EXERCISERS TO FILL
002074	000000		.WORD 0	
002076		L\$DESP::		;POINTER TO DIAG. DESCRIPTION
002076	003342		.WORD L\$DESC	
002100		L\$LOAD::		;GENERATE SPECIAL AUTOLOAD EMT
002100	104035		EMT E\$LOAD	
002102		L\$ETP::		;POINTER TO ERR_TBL
002102	000000		.WORD 0	
002104		L\$ICP::		;PTR. TO INIT CODE
002104	022000		.WORD L\$INIT	
002106		L\$CCP::		;PTR. TO CLEAN UP CODE
002106	023040		.WORD L\$CLEAN	
002110		L\$ACP::		;PTR. TO AUTO CODE
002110	022764		.WORD L\$AUTO	
002112		L\$PRT::		;PTR. TO PROTECT TABLE
002112	021770		.WORD L\$PROT	
002114		L\$TEST::		;TEST NUMBER
002114	000000		.WORD 0	
002116		L\$DLY::		;DELAY COUNT
002116	000000		.WORD 0	
002120		L\$HIME::		;PTR. TO HIGH MEM
002120	000000		.WORD 0	

```

780          .SBTTL  DEFAULT HARDWARE P-TABLE
781
782          ;**
783          ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
784          ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
785          ; IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
786          ;--
787          BGNHW    DFPTBL      ;DEFAULT HARD P TABLE
          .WORD    L10000-L$HW/2
          L$HW::
          DFPTBL::
788
789          .WORD    172522      ; 2ND (OF 2) REGISTERS.
790          .WORD    224        ; INTERRUPT VECTOR
791          .WORD    PRI05      ; INTERRUPT PRIORITY.
792          ENDSHW
          L10000:
          002122    000003
          002124
          002124
          002124
          002124
          002124    172522
          002126    000224
          002130    000240
          002132
          002132

```

```

794          .SBTTL  SOFTWARE P-TABLE
795
796          ;**
797          ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
798          ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
799          ;--
800 002132      BGNSW  SFPTBL
      002132 000004      .WORD  L10001-L$SW/2
      002134
      002134
801
802 002134 000000      TRANSTST::      .WORD  0      ;ENABLE RAM DUMP IF =1
803 002136 000000      NOITS::      .WORD  0      ; INHIBIT ITERATION OPTION.
804                                     ; ... 0 = ITERATE.
805                                     ; ...NZ = INHIBIT ITERATE.
806 002140 000031      LERRMAX::      .WORD  25.      ; LOCAL (PER TEST) ERROR LIMIT
807 002142 000310      GERRMAX::      .WORD  200.      ; GLOBAL (PER UNIT) ERROR LIMIT
808 002144      ENDSW
      002144
809      L10001:

```

812
819
824
830
831
832
833
834
835
836
837
838
839
843 002144

.SBTTL GLOBAL EQUATES SECTION

.SBTTL GLOBAL EQUATES SECTION

; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
; ARE USED IN MORE THAN ONE TEST.

EQUALS ; GET STANDARD EQUATES.

;
; 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
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      MOE== 100000

```

844

845 002144

```

KT11      ;DEFINE MEMORY MANAGEMENT REGISTERS
.SBTTL    MEMORY MANAGEMENT DEFINITIONS
;*KT11    VECTOR ADDRESS
MMVEC=    250
;*KT11    STATUS REGISTER ADDRESSES
SR0=      177572
SR1=      177574
SR2=      177576
SR3=      172516
;IF NB
;*USER "I" PAGE DESCRIPTOR REGISTERS
UIPDR0=   177600
UIPDR1=   177602
UIPDR2=   177604
UIPDR3=   177606
UIPDR4=   177610
UIPDR5=   177612
UIPDR6=   177614
UIPDR7=   177616
;IF NB
;*USER "D" PAGE DESCRIPTOR REGISTERS
UDPDR0=   177620
UDPDR1=   177622
UDPDR2=   177624
UDPDR3=   177626
UDPDR4=   177630
UDPDR5=   177632
UDPDR6=   177634
UDPDR7=   177636
.FNDC
;*USER "I" PAGE ADDRESS REGISTERS

```

```
UIPAR0= 177640
UIPAR1= 177642
UIPAR2= 177644
UIPAR3= 177646
UIPAR4= 177650
UIPAR5= 177652
UIPAR6= 177654
UIPAR7= 177656
  .IF NB
  ;*USER "D" PAGE ADDRESS REGISTERS
  UDPAR0= 177660
  UDPAR1= 177662
  UDPAR2= 177664
  UDPAR3= 177666
  UDPAR4= 177670
  UDPAR5= 177672
  UDPAR6= 177674
  UDPAR7= 177676
  .ENDC
  .ENDC
  .IF NB
  ;*SUPERVISOR "I" PAGE DESCRIPTOR REGISTERS
  SIPDR0= 172200
  SIPDR1= 172202
  SIPDR2= 172204
  SIPDR3= 172206
  SIPDR4= 172210
  SIPDR5= 172212
  SIPDR6= 172214
  SIPDR7= 172216
  .IF NB
  ;*SUPERVISOR "D" PAGE DESCRIPTOR REGISTERS
  SDPDR0= 172220
  SDPDR1= 172222
  SDPDR2= 172224
  SDPDR3= 172226
  SDPDR4= 172230
  SDPDR5= 172232
  SDPDR6= 172234
  SDPDR7= 172236
  .ENDC
  ;*SUPERVISOR "I" PAGE ADDRESS REGISTERS
  SIPAR0= 172240
  SIPAR1= 172242
  SIPAR2= 172244
  SIPAR3= 172246
  SIPAR4= 172250
  SIPAR5= 172252
  SIPAR6= 172254
  SIPAR7= 172256
  .IF NB
  ;*SUPERVISOR "D" PAGE ADDRESS REGISTERS
  SDPAR0= 172260
  SDPAR1= 172262
  SDPAR2= 172264
  SDPAR3= 172266
  SDPAR4= 172270
```

```

SDPAR5= 172272
SDPAR6= 172274
SDPAR7= 172276
.ENDC
.ENDC
; *KERNEL "I" PAGE DESCRIPTOR REGISTERS
172300 KIPDR0= 172300
172302 KIPDR1= 172302
172304 KIPDR2= 172304
172306 KIPDR3= 172306
172310 KIPDR4= 172310
172312 KIPDR5= 172312
172314 KIPDR6= 172314
172316 KIPDR7= 172316
; IF NB
; *KERNEL "D" PAGE
DESCRIPTOR REGISTERS
KDPDR0= 172320
KDPDR1= 172322
KDPDR2= 172324
KDPDR3= 172326
KDPDR4= 172330
KDPDR5= 172332
KDPDR6= 172334
KDPDR7= 172336
.ENDC
; *KERNEL "I" PAGE ADDRESS REGISTERS
172340 KIPAR0= 172340
172342 KIPAR1= 172342
172344 KIPAR2= 172344
172346 KIPAR3= 172346
172350 KIPAR4= 172350
172352 KIPAR5= 172352
172354 KIPAR6= 172354
172356 KIPAR7= 172356
; IF NB
; *KERNEL "D" PAGE ADDRESS REGISTERS
KDPAR0= 172360
KDPAR1= 172362
KDPAR2= 172364
KDPAR3= 172366
KDPAR4= 172370
KDPAR5= 172372
KDPAR6= 172374
KDPAR7= 172376
.ENDC

```

```

850          .SBTTL  TK-25 REGISTER AND PACKET DEFINITIONS
851
852          ;
853          ; SOME GENERAL EQUATES.
854          ;
855
856          000004      ERRVEC==      4          ; POINTER TO ERROR VECTOR FOR BUS TIME OUT.
857          000060      TTIVEC==      60         ; INTERRUPT VECTOR FOR CONSOLE INPUT
858          177560      TTICSR==      177560     ; BUS ADDRESS OF CONSOLE INPUT
859          177562      TTIBFR==      177562     ; CONSOLE INPUT DATA BUFFER
860
861          ;*
862          ;BIT DEFINITIONS FOR TSSR REGISTER
863          ;-
864
865          100000      SC=      BIT15          ;SPECIAL CONDITION
866          040000      BIE=      BIT14          ;BUS INTERFACE ERROR
867          020000      SCE=      BIT13          ;SANITY CHECK ERROR
868          010000      RMR=      BIT12          ;MODIFICATION REFUSED
869          004000      NXM=      BIT11          ;NONEXISTANT MEMORY ERROR
870          002000      NBA=      BIT10          ;NEED BUFFER ADDRESS
871          001400      MIADDR= BIT9:BIT8        ;EXTENDED ADDRESS BITS
872          000200      SSR=      BIT7           ;SUB SYSTEM READY
873          000100      OFL=      BIT6           ;OFF LINE BIT
874          000060      FATERR= BIT4:BIT5        ;FATAL TERMINATION ERROR CODES
875          000016      TERCLS= BIT3:BIT2:BIT1    ;TERMINATION CODES
876
877          ;*
878          ;
879          ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 0
880          ;(XST0)
881          ;
882          ;-
883
884          100000      XSOTMK= BIT15          ;TAPE MARK DETECTED
885          040000      XSORLS= BIT14          ;RECORD LENGTH SHORT
886          020000      XSOLET= BIT13          ;LOGICAL END OF TAPE
887          010000      XSORLL= BIT12          ;RECORD LENGTH LONG
888          004000      XSOWLE= BIT11          ;WRITE LOCK ERROR
889          002000      XSONEF= BIT10          ;NON EXECUTABLE FUNCTION
890          001000      XSOILC= BIT9           ;ILLEGAL COMMAND
891          000400      XSOILA= BIT8           ;ILLEGAL ADDRESS
892          000200      XSOMOT= BIT7           ;TAPE IN MOTION
893          000100      XSOONL= BIT6           ;TRANSPORT ON LINE
894          000040      XSOIE=  BIT5           ;INTERRUPT ENABLE
895          000020      XSOVCK= BIT4           ;VOLUME CHECK BIT
896          000010      XSOPED= BIT3           ;PHASE ENCODED DRIVE
897          000004      XSOWLK= BIT2           ;WRITE LOCKED
898          000002      XS0BOT= BIT1           ;BEGINNING OF TAPE
899          000001      XS0EOT= BIT0           ;END OF TAPE
900
901          ;*
902          ;
903          ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 1
904          ;(XST1)
905          ;
906          ;

```

```

907      100000      X1.DLT  = BIT15      ;DATA LATE
908      040000      X1.SPARE= BIT14      ;NOT USED
909      020000      X1.COR  = BIT13      ;CORRECTABLE DATA ERROR
910      017375      X1.MBZ  = BIT12·BIT11·BIT10·BIT9·BIT7·BIT6·BIT5·BIT4·BIT3·BIT2·BIT0 ;ALWAYS 0
911      000400      X1.RBP  = BIT8       ;READ BUS PARITY ERROR
912      000002      X1.UNC  = BIT1       ;UNCORRECTABLE DATA OR HARD ERROR
913
914      ;*
915      ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 2
916      ;(XST2)
917      ;-
918      100000      X2.OPM   = BIT15      ;OPERATION IN PROGRESS (TAPE MOVING)
919      040000      X2.RCE   = BIT14      ;RAM CHECKSUM ERROR
920      035400      X2.SPARE= BIT13·BIT12·BIT11·BIT9·BIT8 ;NOT USED BY TK-25 (ALWAYS=0)
921      002000      X2.WCF   = BIT10      ;WRITE CLOCK FAILURE (FIFO NOT EMPTIED BY TRANSPORT)
922      000200      X2.EXTF  = BIT7       ;IF WRITE CHAR CMD THEN = EXTENDED FEATURES ENABLED
923      000100      X2.BUFE  = BIT6       ;IF WRITE CHAR CMD THEN = BUFFERING ENABLED
924      000077      X2.REV   = 000077    ;IF WRITE CHAR CMD THEN = MICROCODE REVISION LEVEL
925      000007      X2.UNIT  = BIT2·BIT1·BIT0 ;IF GET STATUS THEN = CURRENTLY SELECTED UNIT NO.
926
927      ;*
928      ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 3
929      ;(XST3)
930      ;-
931      177400      X3.MDE   = 177400    ;MICRO-DIAGNOSTIC ERROR CODE
932      000200      X3.SPARE= BIT7       ;NOT USED BY TK 25
933      000100      X3.OPI   = BIT6       ;OPERATION INCOMPLETE
934      000040      X3.REV   = BIT5       ;REVERSE
935      000020      X3.TRF   = BIT4       ;TRANSPORT RESPONSE FAILURE
936      000010      X3.DCK   = BIT3       ;DENSITY CHECK
937      000006      X3.MBZ  = BIT2·BIT1   ;NOT USED ALWAYS 0
938      000001      X3.RIB  = BIT0       ;REVERSE INTO BOT
939
940      ;*
941      ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 4
942      ;(XST4)
943      ;-
944      100000      X4.HSP   = BIT15      ;HIGH SPEED
945      040000      X4.RCE   = BIT14      ;RETRY COUNT EXCEEDED
946      020000      X4.TSM   = BIT13      ;TRANSPORT SPECIAL MODE
947      017400      X4.MBZ  = BIT12·BIT11·BIT10·BIT9·BIT8 ;NOT USED ALWAYS 0
948      000377      X4.WRC   = 000377    ;WRITE RETRY COUNT FIELD
949
950      ;*
951      ;
952      ;TSSR TERMINATION CODES (BIT 0-2)
953      ;
954      ;
955      ;
956
957      000006      TSREJ= 3·2      ;COMMAND REJECTED
958      000006      UNREC= 6      ;UNRECOVERABLE ERROR
959
960      ;*
961      ;
962      ;DEVICE REGISTER OFFSETS
963      ;

```

```

964      ; -
965
966      177776      TSBA== -2
967      177776      TSBAL== -2
968      177776      TSDB== -2      ;TSDB/TSBA REGISTER
969      177776      TSDBL== 2      ;TSDB/TSBA REGISTER
970      177777      TSBAM== -1
971      177777      TSDBM== 1      ;TSDB/TSBA REGISTER HIGH BYTE
972      000000      TSSR== 0      ;TSSR REGISTER
973      000001      TSSRM== 1      ;TSSR REGISTER HIGH BYTE
974
975      ;*
976      ; TSDB ADDRESS BIT DEFINITIONS
977      ; -
978      000003      A1716 = BIT1+BIT0      ;ADDRESS BITS 17:16 ARE IN 1:0
979
980      ;*
981      ; COMMAND DEFINITIONS
982      ; -
983      000017      P.GETSTAT = 17      ;GET STATUS
984      000013      P.INIT = 13      ;INITIALIZE
985      000012      P.CONTROL = 12      ;CONTROL COMMANDS
986      000011      P.FORMAT = 11      ;FORMAT
987      000010      P.POSITION = 10      ;POSITION
988      000006      P.WRTSUB = 6      ;SUBSYSTEM WRITE
989      000005      P.WRITE = 5      ;WRITE
990      000004      P.WRTCHAR = 4      ;WRITE CHARACTERISTICS
991      000001      P.READ = 1      ;READ
992
993      ;*
994      ; COMMAND PACKET HEADER WORD BIT DEFINITIONS
995      ; -
996      100000      P.ACK = BIT15      ;BUFFER AVAIL FOR CONTROLLER
997      040000      P.CVC = BIT14      ;CLEAR VOLUME CHECK
998      020000      P.OPP = BIT13      ;REVERSE SEQUENCE OF DATA BITS
999      010000      P.SWB = BIT12      ;SWAP BYTES IN MEMORY
1000      007400      P.MODE = BIT11:BIT10:BIT9:BIT8      ;EXTENDED COMMAND MODE FIELD
1001      000200      P.IE = BIT7      ;INTERRUPT ENABLE
1002      000140      P.FMT= BIT6:BITS      ;PACKET HEADER TYPE (ALWAYS=0)
1003      000037      P.CMD = 37      ;MAJOR COMMAND FIELD
1004
1005      ;*
1006      ; CONTROL COMMAND MODE CODES
1007      ; -
1007      000000      PC.RELEASE = 0*256.      ;RELEASE BUFFER
1008      000400      PC.REWIND = 1*256.      ;REWIND
1009      001000      PC.NOOP = 2*256.      ;NO-OP
1010      002000      PC.IEREW = 4*256.      ;REWIND IMMEDIATE INTERRUPT
1011      002400      PC.ERASE = 5*256.      ;SECURITY ERASE
1012
1013      ;*
1014      ; CONTROLLER RAM DEFINITIONS
1015      ;
1016      000167      RMCHBEG = 167      ;CHARACTERISTICS IO DATA BEGIN RAM ADDRESS
1017      000200      RMCHEND = 200      ;CHARACTERISTICS IO DATA END RAM ADDRESS
1018      000020      RMPKTBEG= 20      ;COMMAND PACKET BEGIN RAM ADDRESS
1019      000027      RMPKTEND= 27      ;COMMAND PACKET END RAM ADDRESS
1020      000104      RMMSGBEG= 104      ;MESSAGE BUFFER BEGIN RAM ADDRESS

```

```

1021      000117      RMMMSGEND= 117      ;MESSAGE BUFFER END RAM ADDRESS
1022      ;*
1023      ;
1024      ;REGISTER DEFINITIONS IN THE MESSAGE BUFFER
1025      ;
1026      ;-
1027
1028      000006      XST0== 6      ;EXTENDED STATUS REGISTER 0 (WORD 4)
1029      000010      XST1== 8.      ;EXTENDED STATUS REGISTER 1 (WORD 5)
1030      000012      XST2== 10.      ;EXTENDED STATUS REGISTER 2 (WORD 6)
1031      000014      XST3== 12.      ;EXTENDED STATUS REGISTER 3 (WORD 7)
1032      000016      XST4== 14.      ;EXTENDED STATUS REGISTER 4 (WORD 8)
1033
1034      ;*
1035      ;
1036      ;
1037      ;OFFSETS TO WORD LOCATIONS IN PACKET DEFINITIONS
1038      ;
1039      ;-
1040
1041      000002      PKLOW = 2      ;LOW ORDER CHARACTERISTIC DATA POINTER
1042      000004      PKHI = 4      ;HIGH ORDER CHARACTERISTIC DATA POINTER
1043      000006      PKBCNT = 6      ;NUMBER OF BYTES IN DATA PACKET
1044
1045      000010      EXBCNT=10      ;NUMBER OF BYTES IN EXTENDED DATA PACKET
1046
1047      ;*
1048      ;DATA PACKET OFFSETS FOR WRITE SUBSYSTEM COMMAND
1049      ;-
1050      000000      BSELO = 0      ;BYTE 0
1051      000001      BSEL1 = 1      ;BYTE 1
1052      000002      SEL2 = 2      ;WORD 2
1053      000004      SELDATA = 4      ;WORD 3
1054
1055      ;*
1056      ;BSELO SELECT CODES FOR WRITE SUBSYSTEM COMMAND
1057      ;-
1058      000000      PW.NOP = 0      ;NO-OP
1059      000001      PW.RDRAM = 1      ;READ RAM
1060      000002      PW.WTRAM = 2      ;WRITE RAM
1061      000003      PW.RFIFO = 3      ;READ FIFO
1062      000004      PW.WFIFO = 4      ;WRITE FIFO
1063      000005      PW.RDSTAT = 5      ;READ STATUS
1064      000006      PW.WCTL = 6      ;WRITE TAPE CONTROL
1065      000007      PW.WFMT = 7      ;WRITE TAPE FORMAT
1066      000010      PW.WMISC = 10      ;WRITE MISCELLANEOUS
1067      000011      PW.WNPR = 11      ;WRITE NPR CONTROL
1068      000020      PW.D22 = 20      ;DO MICROTEST 22
1069      000021      PW.D11 = 21      ;DO MICROTEST 11
1070      000022      PW.D13 = 22      ;DO MICROTEST 13
1071      000023      PW.NO1311 = 23      ;DISABLE MICROTEST 11 AND 13
1072      000024      PW.RDEXT = 24      ;READ EXT. TAPE STATUS (NOT SUPPORTED BY ALL TRANSP
RTS
1073
1074      ;*
1075      ;BSEL1 CODES FOR WRITE TAPE CONTROL
1076      ;-
1077      000200      WC.IFAD = BIT7      ;IFAD FORMATTER ADDRESS

```

```

1078      000100      WC.IOTAD      * BIT6      ;ITAD0 - TRANSPORT ADDRESS BIT 0
1079      000040      WC.I1TAD      * BIT5      ;ITAD1 - TRANSPORT ADDRESS BIT 1
1080      000020      WC.I5RESV      * BIT4      ;IRESV5 - RESERVED #5
1081      000010      WC.IREW       * BIT3      ;IREW - REWIND
1082      000004      WC.IRWU       * BIT2      ;IRWU - REWIND AND UNLOAD
1083      000002      WC.IFEN       * BIT1      ;IFEN - FORMATTER ENABLE
1084      000001      WC.IGO        * BIT0      ;GO
1085
1086      ;*
1087      ;BSEL1 CODES FOR WRITE FORMAT
1088      ;-
1089      000200      WF.IHISP       * BIT7      ;IHISP - HIGH SPEED
1090      000100      WF.IWRT       * BIT6      ;IWRT - WRITE
1091      000040      WF.IREV       * BIT5      ;IREV - REVERSE
1092      000020      WF.IWFM       * BIT4      ;IWFM - WRITE FILE MARK
1093      000010      WF.IEDIT      * BIT3      ;IEDIT - EDIT
1094      000004      WF.IERASE      * BIT2      ;IERASE - ERASE
1095      000002      WF.I3RESV      * BIT1      ;IRESV3 - RESERVED #3
1096      000001      WF.I4RESV      * BIT0      ;IRESV4 - RESERVED #4
1097
1098
1099      ;*
1100      ;BSEL1 CODES FOR WRITE MISCELLANEOUS SUBCOMMAND
1101      ;-
1102      000200      MS.EXT         * BIT7      ;INVERT SENSE OF EXTENDED FEATURES SWITCH
1103      000020      MS.RSFIFO      * BIT4      ;RESET FIFO AND INPUT PARITY ERRORR
1104      000010      MS.RSTAPE      * BIT3      ;RESET TAPE STATUS IN 2 FLIP-FLOPS
1105      000006      MS.ATTN        * BIT2!BIT1 ;ATTENTION TRIGGER FIELD
1106      000001      MS.RSD        * BIT0      ;RESET TIMER A,B THEN DELAY TIMES IN SEL2
1107
1108      ;*
1109      ; MS.ATTN SUBCODES
1110      ;-
1110      000000      MSA.NOP = 0*2      ;NO-OP (NOTHING TRIGGERED)
1111      000002      MSA.VOL = 1*2      ;SIMULATE ON-LINE/OFF-LINE TRANSITION
1112      000004      MSA.NRAM= 2*2      ;FORCE NON-FATAL RAM ERROR (FORCES ERRCODE 54)
1113      000006      MSA.FRAME= 3*2     ;FORCE FATAL RAM ERROR (CAUSES SCE TO SET)
1114
1115      ;*
1116      ; WRITE SUBSYSTEM WRITE NPR BSEL1 BIT DEFINITIONS
1117      ;-
1117      000200      NP.IR         * BIT7      ;INTERRUPT REQUEST (0-1 TRANSITION)
1118      000100      NP.OUT        * BIT6      ;TAPE DATA DIRECTION OUT (0= IN)
1119      000040      NP.LOOP       * BIT5      ;ENABLE TRANSPORT LOOPBACK
1120      000020      NP.WRP        * BIT4      ;WRITE CORRECT PARITY (SET=0 TO WRITE WRONG)
1121
1122      ;*
1123      ; READ STATUS MESSAGE BUFFER BIT DEFINITIONS
1124      ;-
1125      000200      S2.DIM         * BIT7      ;WORD #9 BYTE 2 DATA IN MISS
1126      000100      S2.ILW        * BIT6      ;
1127      000040      S2.OUTRDY      * BIT5      ;
1128      000020      S2.INRDY      * BIT4      ;
1129      000010      S2.ATIMR      * BIT3      ;
1130      000004      S2.BTIMR      * BIT2      ;
1131      000003      S2.UNDEF       * BIT1!BIT0 ;(UNDEFINED)
1132      100000      S1.PARIN      * BIT15      ;WORD #8 BYTE 1 PARIN H
1133      040000      S1.I2RESV      * BIT14      ;
1134      020000      S1.I1RESV      * BIT13      ;

```

1135	010000	S1.IEOT	= BIT12	:	IEOT L
1136	004000	S1.IIDENT	= BIT11	:	IIDENT H
1137	002000	S1.ICER	= BIT10	:	ICER H
1138	001000	S1.IFMK	= BIT9	:	IFMK H
1139	000400	S1.IHER	= BIT8	:	IHER H
1140	000200	SO.ISPEED	= BIT7	:	WORD #8 BYTE 0 ISPEED H
1141	000100	SO.IRDY	= BIT6	:	IRDY L
1142	000040	SO.IONL	= BIT5	:	IONL L
1143	000020	SO.ILDP	= BIT4	:	ILDP L
1144	000010	SO.IDBY	= BIT3	:	IDBY L
1145	000004	SO.IRWD	= BIT2	:	IRWD L
1146	000002	SO.IFBY	= BIT1	:	IFBY L
1147	000001	SO.IFPT	= BIT0	:	IFPT L
1148		:		:	
1149		:	SPECIAL KEYBOARD STUFF FOR MOVER PROGRAM	:	
1150	177560	TKS	=177560	:	;KEYBOARD STATUS REGISTER
1151	177562	TKB	=177562	:	;KEYBOARD DATA REGISTER
1152	177564	TPS	=177564	:	;CONSOLE PRINTER STATUS REGISTER
1153	177566	TPB	=177566	:	;CONSOLE PRINTER DATA REGISTER
1154	007776	HIMEM	=007776	:	;HIGH MEMORY MASK VALUE
1155		:	CONTROLLER DEFINITIONS	:	
1156		:		:	
1157		:		:	
1158	174400	CSR	=174400	:	;STATUS AND CONTROL REGISTER
1159	174402	BAR	=174402	:	;DL ADDRESS REGISTER
1160	174404	DAR	=174404	:	;PLATTER ADDRESS
1161	174406	MPR	=174406	:	;MULTIPURPOSE REGISTER
1162		:		:	
1163		:		:	
1164		:		:	
1165		:		:	
1166		:		:	
1167		:	CONTROLLER COMMANDS	:	
1168		:		:	
1169		:		:	
1170	000004	DLGETS	=4	:	;GET STATUS COMMAND
1171	000006	SEEK	=6	:	;SEEK TRACK AND HEAD SELECT
1172	000010	DLRDHD	=10	:	;READ SECTOR HEADER
1173	000014	READ	=14	:	;READ COMMAND
1174	000016	DLRDNH	=16	:	;READ SECTOR NO HEADER CHECK
1175		:		:	
1176		:		:	
1177		:		:	
1178		:		:	
1179		:		:	
1180		:		:	
1181	000001	READY	=1	:	;DRIVE READY BIT IN STATUS REG.
1182	000013	DLSR	=13	:	;STATUS AND RESET
1183	177730	DLERR	=177730	:	;MASK FOR COVER OPEN
1184	000006	DLUN	=6	:	;HEADS UNLOADED
1185	000177	DLCYL	=000177	:	;MASK FOR CYLINDER ADDRESS
1186	100200	DLDNER	=100200	:	;DONE SET OR ERROR SET BITS
1187		:		:	
1188		:		:	
1189		:		:	
1190		:		:	
1191	177560	TTICSR	= 177560	:	;KEYBOARD INPUT STATUS

J3

CZTKGA TK 25 FRT END FUNC #3 MACRO M1200 20-APR 84 08:13 PAGE 27-6
TK-25 REGISTER AND PACKET DEFINITIONS

SEQ 35

1192
1193
1194
1195

177562
177564
177566

TTIBFR = 177562
TTOCSR = 177564
TTOBFR = 177566

;KEYBOARD DATA REGISTER
;CONSOLE PRINTER STATUS REGISTER
;CONSOLE PRINTER DATA REGISTER

```
1197             .SBTTL  SPECIAL MACROS AND OPDEFS.
1198
1199
1200             ;*
1201             ;SAVE GENERAL REGS 1 TO 5
1202             ;-
1203
1204             .MACRO  SAVREG
1205             JSR     R5,REGSAV
1206             .ENDM
1207
1208             ;*
1209             ; MACRO TO FORCE AN ERROR
1210             ;-
1211             .MACRO  FORCERROR          TAG,NOTSSR
1212             .NLIST
1213             .IIF NDF LISTALL, .NLIST
1214             .LIST
1215             .IF B NOTSSR
1216                 MOV     TSSR(R5),R1          ;READ TSSR
1217             .ENDC
1218                 MOV     FORCER,FORCER        ;IS FORCER SET? (LEAVE C BIT ALONE)
1219                 BNE     TAG                   ;BR IF YES
1220             .NLIST
1221             .IIF NDF LISTALL, .LIST
1222             .LIST
1223             .ENDM
1224
1225             ;*
1226             ; MACRO TO FORCE AN EXIT TO AVOID SECTION ITERATIONS
1227             ; WILL EXIT TO A LABEL IF FORCER IS NEGATIVE
1228             ; SO TO FORCE ERRORS AND EXIT ON 1 ERROR SET
1229             ; FORCER TO 177777
1230             ; TO FORCE ERRORS AND ITERATIONS SET FORCER TO 1.
1231             ;-
1232             .MACRO  FORCEEXIT          TAG
1233             .NLIST
1234             .IIF NDF LISTALL, .NLIST
1235             .LIST
1236                 MOV     FORCER,FORCER        ;IS FORCER NEGATIVE?
1237                 BMI     TAG                   ;BR IF YES
1238             .NLIST
1239             .IIF NDF LISTALL, .LIST
1240             .LIST
1241             .ENDM
1242             ;*
1243             ; MACRO TO INCREMENT ERROR COUNTS
1244             ;-
1245             .MACRO  NEXT.ERRNO
1246             .NLIST
1247             ;:::IIF NDF LISTALL, .NLIST
1248                 ERRNO=ERRNO+1
1249             ;:::IIF NDF LISTALL, .LIST
1250             .LIST
1251             .ENDM
1252
1253             ;*
```

```

1254      ;MACRO TO PERFORM XOR
1255      ; -
1256
1257      .MACRO XOR      A,B
1258      MOV      A,-(SP)
1259      BIC      B,(SP)
1260      BIC      A,B
1261      BIS      (SP)+,B
1262      .ENDM
1263
1264      000000      EQU=0      ; INITIALIZE ERROR NUMBER
1265      .SBTTL FORCER - FORCE ERROR FLAG
1266
1267      ;
1268      ; THE FOLLOWING LOCATIONS MAY BE PATCHED BY THE USER
1269      ; TO OBTAIN THE RESULTS DESCRIBED FOR EACH.
1270      ;
1271
1272 002144 000000 FORCER::      0      ; FORCE TYPE ALL HARD ERRORS (THE ONES CALLED
1273      ; - BY THE MACRO "IFERROR"). AN ERROR NEED NOT -
1274      ; - EXIST, JUST ASSUME AND TYPE THE MESSAGE.
1275
1276
1277

```

.SBTTL GLOBAL DATA SECTION

```
1279
1280
1281      ;**
1282      ;THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
1283      ;IN MORE THAN ONE TEST.
1284      ;--
1285
1286      ;
1287      ;THE FOLLOWING DATA ARE SET FOR EACH UNIT AT INIT TIME.
1288      ;SINGLE UNIT DEFAULTS (LISTED) ARE IN THE DEFAULT P-TABLE.
1289      ;
1290 002146 000000 EPRTSW::      .WORD 0      ;PRINT SWITCH
1291 002150 000000 UNITN::      .WORD 0      ;UNIT # UNDER TEST.
1292 002152 000000 QVP::       .WORD 0      ;QUICK VERIFY FLAG.
1293 002154 000000 CSRADR::    .WORD 0      ;ADDRESS OF CSR FOR CURRENT DEVICE
1294 002156 000224 IVEC::      .WORD 224    ;INTERRUPT VECTOR
1295 002160 000200 IPRI::      .WORD PRI04  ;INTERRUPT PRIORITY.
1296 002162 000000 TSTCNT::    .WORD 0      ;NUMBER OF TESTS RUN IN THIS PASS
1297 002164 000000 LOOPCNT::   .WORD 0      ;REMAINING ITERATION COUNT FOR TEST
1298 002166 000000 DEVCNT::    .WORD 0      ;NUMBER OF DEVICE UNDER TEST
1299 002170 000000 FATFLG::    .WORD 0      ;SET IF FATAL ERROR IS DETECTED IN TEST
1300 002172 000000 INTRECV::   .WORD 0      ;SET IF TAPE INTERRUPT WAS RECEIVED
1301 002174 000000 BENBSW::    .WORD 0      ;BUFFER ENABLE SWITCH SW 0=OFF;1=ON
1302 002176 000000 EXPD::      .WORD 0      ;EXPECTED RAM DATA FOR PRAMPKT ROUTINE
1303 002200 000000 RECV::      .WORD 0      ;RECEIVED RAM DATA FOR PRAMPKT ROUTINE
1304 002202 000000 ERRHI::     .WORD 0      ;HIGH ADDRESS MEMORY ERROR
1305 002204 000000 ERRLO::     .WORD 0      ;LOW ADDRESS MEMORY ERROR
1306 002206 000000 RAMDATA::   .BLKW 16.    ;DATA READ FROM RAM PACKET OR MESSAGE BUF AREA
1307 002246 000000 RAMSIZ::    .WORD 0      ;RAM DATA SIZE FOR PRAMPKT ROUTINE
1308 002250 000000 RCVHIADD::  .WORD 0      ;RECEIVED BUFFER HIGH ADDRESS
1309 002252 000000 RCVLOADD::  .WORD 0      ;RECEIVED BUFFER LOW ADDRESS
1310 002254 000000 COUNT::     .WORD 0      ;TEST COUNT PATTERN
1311 002256 000000 DATA::      .WORD 0      ;TEST DATA
1312 002260 000000 TSTFLAG::   .WORD 0      ;TEST FLAG WORD
1313 002262 000000 TSTPTR::    .WORD 0      ;TSTBLK POINTER
1314 002264 000000 PRMNO::     .WORD 0      ;PRINT ROUTINE TEMP
1315 002266 000000 EXPMSG::     .BLKB 100.    ;EXPECTED MESSAGE BUFFER DATA
1316 002432 000000 RECMSG::     .BLKB 100.    ;RECEIVED MESSAGE BUFFER DATA
1317 002576 000000 TMPBFR::    .BLKB 80.     ;TEMPORARY STORAGE FOR PRINT
1318 002716 000000 MESBFA::     .WORD 0      ;STORES ADDRESS OF MESSAGE BUFFER FOR ERR PRT
```

```

1320          .SBTTL  TSTBLK  - TEST DATA TABLE
1321
1322          ;*
1323          ;
1324          ; THIS TABLE CONTAINS TEST DATA USED IN SEVERAL TESTS
1325          ;
1326          ; IN SEQUENCE THE DATA IS:
1327          ;
1328          ;     ALL ZEROS
1329          ;     ALL ONES
1330          ;     WALKING ONES
1331          ;     WALKING ZEROS
1332          ;     ALTERNATING ONES AND ZEROS
1333          ;
1334          ; -
1335
1336 002720
1337 002720 000000
1338 002722 177777
1339 002724 000001
1340 002726 000002
1341 002730 000004
1342 002732 000010
1343 002734 000020
1344 002736 000040
1345 002740 000100
1346 002742 000200
1347 002744 000400
1348 002746 001000
1349 002750 002000
1350 002752 004000
1351 002754 010000
1352 002756 020000
1353 002760 040000
1354 002762 100000
1355 002764 177776
1356 002766 177775
1357 002770 177773
1358 002772 177767
1359 002774 177757
1360 002776 177737
1361 003000 177677
1362 003002 177577
1363 003004 177377
1364 003006 177777
1365 003010 175777
1366 003012 173777
1367 003014 167777
1368 003016 157777
1369 003020 137777
1370 003022 077777
1371 003024 125252
1372 003026 052525
1373          003030

          TSTBLK::
              .WORD  0                      ; ALL ZEROS
              .WORD  177777                ; ALL ONES
              .WORD  BIT0                   ; DATA FOR WALKING ONES
              .WORD  BIT1
              .WORD  BIT2
              .WORD  BIT3
              .WORD  BIT4
              .WORD  BIT5
              .WORD  BIT6
              .WORD  BIT7
              .WORD  BIT8
              .WORD  BIT9
              .WORD  BIT10
              .WORD  BIT11
              .WORD  BIT12
              .WORD  BIT13
              .WORD  BIT14
              .WORD  BIT15
              .WORD  +CBIT0                  ; DATA FOR WALKING ZEROS
              .WORD  +CBIT1
              .WORD  +CBIT2
              .WORD  +CBIT3
              .WORD  +CBIT4
              .WORD  +CBIT5
              .WORD  +CBIT6
              .WORD  +CBIT7
              .WORD  +CBIT8
              .WORD  +CBIT9
              .WORD  +CBIT10
              .WORD  +CBIT11
              .WORD  +CBIT12
              .WORD  +CBIT13
              .WORD  +CBIT14
              .WORD  +CBIT15
              .WORD  125252                  ; ALTERNATING ONES, ZEROS
              .WORD  052525                  ; ALTERNATING ONES, ZERO OPPOSITE FROM ABOVE

          TBLEND==.

```

```

1375          .SBTTL GLOBAL ENVIRONMENT STORAGE
1376
1377          ; STORAGE FOR DEVICE REGISTERS
1378
1379 003030 000000 100000 000000 DUMMY: 0,100000,0,0          ; DUMMY DEVICE REGISTERS...
1380 003040 000000 000000 000000      0,0,0,0,0,0,0,0,0,0 ; ...FOR MULTI-UNIT CHECKOUT.
1381
1382
1383
1384 003060 000000          DUFLG::      .WORD 0          ; "DROPPED UNIT" FLAG.
1385                                ; INHIBITS CODE IN "CLEAN-UP".
1386 003062 000000          NODEV::      .WORD 0          ; FLAG TO SAY NO DEVICE.
1387
1388 003064 000000          TEMP1::      .WORD 0          ; SOME TEMP LOCATIONS.
1389 003066 000000          TEMP2::      .WORD 0
1390 003070 000000          XXCOMM::     .WORD 0          ; XXDP, COMM BLOCK POINTER.
1391 003072 000000          FREE::       .WORD 0          ; 1ST FREE MEMORY ADDRESS...
1392 003074 000000          FRESIZ::     .WORD 0          ; ...AND SIZE (IN WORDS).
1393 003076 000000          FREEHI::     .WORD 0          ; LAST WORD IN FREE SPACE
1394 003100 000000          KTFLG::      .WORD 0          ; KT11, MEM AVAIL FLAG -
1395                                ; - .WORD 0 = <24K OR NO KT
1396                                ; - NZ = >24K AND KT.
1397 003102 000000          KTENABLE::     .WORD 0          ; SET BY TEST ROUTINES TO FLAG >28K UNDER TEST
1398 003104 002000          PST32W::      .WORD 2000      ; 32W BLOCK ADDRESS FOR 32K START
1399 003106 000000          SIFLAG::      .WORD 0
1400 003110 000000          BADDAT::      .WORD 0          ;
1401 003112 000000          GDDAT::       .WORD 0          ; ACTUAL DATA
1402 003114 000000          LOOPFL::      .WORD 0          ; EXPECTED DATA
1403 003116          CTAB::               .WORD 0          ; CONFIGURATION TABLES.
1404 003116 000000          CTABM::        .WORD 0          ; CONFIG WORK.
1405 003120 000000          .WORD 0
1406 003122 000000          .WORD 0
1407 003124 000000          .WORD 0
1408 003126 177777          .WORD 0
1409 003130          .WORD -1          ; END OF MEM TABLE.
1410          CTABE::
1411          ; ERROR STATISTICS TABLE (1 WCHJ PER UNIT), 64 UNITS MAX:
1412          ;
1413          ; 0          *          UNIT NOT TESTED
1414          ; 100000    *          UNIT ONLINE, NO ERRORS
1415          ; 10XXXX    *          UNIT ONLINE, ENCOUNTERED XXXX ERRORS
1416          ; 160000    *          UNIT DROPPED, NON-EXISTENT DEVICE REGISTER
1417          ; 160001    *          UNIT DROPPED, NOT IDLE AT START
1418          ; 14XXXX    *          UNIT DROPPED, ENCOUNTERED XXXX ERRORS
1419 003130          ERTABL:                .BLKW 64.
1420 003330 000000          ERTABE:          .WORD 0
1421
1422 003332 000000          SKIPT: .WORD 0          ; 1=SKIP SUBTEST 0=NO SKIP OF SUBTEST

```

```

1424                                     .SBTTL  GLOBAL TEXT MESSAGES
1425                                     ;**
1426                                     ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
1427                                     ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
1428                                     ; MORE THAN ONE TEST.
1429                                     ; -
1430
1431                                     ;*
1432                                     ; NAMES OF DEVICES SUPPORTED
1433                                     ; -
1434
1435                                     DEVTYP  <TK-25>
1436                                     L%DVTYP:
1437 003334                                     .ASCIZ  /TK-25/
1438 003334                                     .EVEN
1439                                     ;
1440                                     ; TEST DESCRIPTION
1441                                     ; -
1442 003342                                     DESCRIPT <CZTKGA TK-25 FRT END FUNC #3>
1443 003342                                     L%DESC:
1444 003342                                     .ASCIZ  /CZTKGA TK-25 FRT END FUNC #3/
1445                                     .EVEN
1446
1447                                     ;*
1448                                     ; BIT TO ASCII CONVERSION FOR TSSR REGISTER
1449                                     ; -
1450 003400 003440 003443 003447 TSSRBIT:
1451 003420 003501 003505 003511 .WORD  1,2,3,4,5,6,7,8,
1452 003440 123 103 000 16: .ASCIZ  'SC'
1453 003443 102 111 105 24: .ASCIZ  'BIE'
1454 003447 123 103 105 32: .ASCIZ  'SCE'
1455 003453 122 115 122 40: .ASCIZ  'RMR'
1456 003457 116 130 115 56: .ASCIZ  'NXM'
1457 003463 116 102 101 64: .ASCIZ  'NBA'
1458 003467 102 111 124 72: .ASCIZ  'BIT9'
1459 003474 102 111 124 80: .ASCIZ  'BIT8'
1460 003501 123 123 122 96: .ASCIZ  'SSR'
1461 003505 117 106 114 104: .ASCIZ  'OFL'
1462 003511 102 111 124 112: .ASCIZ  'BIT5'
1463 003516 102 111 124 120: .ASCIZ  'BIT4'
1464 003523 102 111 124 128: .ASCIZ  'BIT3'
1465 003530 102 111 124 136: .ASCIZ  'BIT2'
1466 003535 102 111 124 144: .ASCIZ  'BIT1'
1467 003542 102 111 124 160: .ASCIZ  'BIT0'
1468                                     .EVEN
1469 003550 124 123 123 SFIERR: .ASCIZ  'TSSR ERROR AFTER SOFT INIT'
1470 003603 124 123 123 SFHERR: .ASCIZ  'TSSR ERROR AFTER BUS RESET'
1471 003636 040 040 116 NXR: .ASCIZ  / NON-EXISTANT DEVICE REGISTER/
1472 003675 045 101 040 NXR: .ASCIZ  /#A ADDRESS: #06/
1473 003716 045 101 040 TSSX: .ASCIZ  /#A TSBA,TSSR EXP'D: #06#A,#06#N/
1474 003756 045 101 040 TSSX: .ASCIZ  /#A TSBA,TSSR REC'D: #06#A,#06/
1475 004015 045 116 045 FUSI: .ASCIZ  /#N#A/
1476 004021 040 040 125 USI: .ASCIZ  / UNEXPECTED INTERRUPT/

```

```

1475 004050      040      040      111 NSI:      .ASCIZ / INTERRUPT EXPECTED, NOT RECEIVED/
1476 004113      045      116      045 FNOINTR: .ASCII /#N#A/
1477 004117      040      040      116 NOINTR: .ASCIZ / NO INTERRUPT WAS GENERATED/
1478 004154      040      040      111 IFALT:  .ASCIZ / INTERRUPT FAULT/
1479 004176      045      101      040 INTX:   .ASCIZ /#A CPU PC: #06#A TSBA: #06/
1480 004233      040      040      042 NOINIT: .ASCIZ / "BUS INIT" DIDN'T INITIALIZE CONTROLLER/
1481 004305      040      040      042 NSINIT: .ASCIZ / "SOFT-INIT" DIDN'T INITIALIZE THE DPU/
1482 004355      040      040      042 BRINIT: .ASCIZ / "BUS-RESET" DIDN'T INITIALIZE THE DPU/
1483
1484 004425      000              NUL:      .ASCIZ //
1485 004426      045      116      000 NULCR:  .ASCIZ /#N/
1486 004431      045      101      040 EXPGOT: .ASCIZ /#A EXP'D: #06#A, REC'D: #06/
1487 004465      045      116      045 EXPGT2: .ASCIZ /#N#A EXP'D: #06#A, #06#N#A REC'D: #0#A, #06/
1488 004541      045      101      040 DUAD12: .ASCIZ /#A REG(W) WRITTEN TO: #06#A REG(R) READ; EXP'D: #06#A, REC'D: #06/
1489 004643      122      101      115 PKTRAM: .ASCIZ 'RAM Contents Do Not Match Packet Sent'
1490 004711      040      040      103 SCME:   .ASCIZ / CONFIG DOESN'T MATCH MFG. MASTER/
1491 004754      127      122      111 WRTMSG: .ASCIZ 'WRITE CHARACTERISTICS Failed'
1492 005011      124      123      123 WRTERR: .ASCIZ 'TSSR Incorrect After WRITE Command, More Bits Set Than SSR'
1493 005104      124      123      123 RDERR:  .ASCIZ 'TSSR Incorrect After READ Command, More Bits Set Than SSR'
1494
1495
1496
1497

```

```

1499
1500
1501
1502
1503
1504
1505
1506
1507 005176
      005176
1508 005176
      005176 013746 003062
      005202 012746 003675
      005206 012746 000002
      005212 010600
      005214 104415
      005216 062706 000006
1509 005222 004737 005230
1510 005226
      005226
      005226 104423
1511
1512
1513
1514
1515
1516
1517 005230 005727
1518 005232 000000
1519 005234 001402
1520 005236 004777 177770
1521 005242
      005242 012746 004426
      005246 012746 000001
      005252 010600
      005254 104415
      005256 062706 000004
1522 005262 000207

      .SBTTL GLOBAL ERROR REPORT SECTION

      ;
      ; THE GLOBAL ERROR REPORT SECTION CONTAINS THE PRINTB AND PRINTX
      ; CALLS THAT ARE USED IN MORE THAN ONE TEST.
      ; ASCII TEXT STRINGS ARE FOUND IN THE GLOBAL TEXT SECTION.
      ; --

      BGNMSG NXRERR ;NON-EXISTANT DEVICE REGISTER.
NXRERR:
      PRINTX #NXRX,NODEV ;NODEV = NEXM ADDRESS.
      MOV NODEV,-(SP)
      MOV #NXRX,-(SP)
      MOV #2,-(SP)
      MOV SP,R0
      TRAP C#PNTX
      ADD #6,SP
      JSR PC,EXTEND ; PRINT EXTENSION IF REQUIRED.
      ENDMMSG
L10002:
      TRAP C#MSG

      ;
      ; THIS ROUTINE APPENDS A UNIQUE EXTENSION (IF REQUIRED)
      ; TO ANY OF THE ABOVE ERROR SIGNATURES.
      ;
      EXTEND: TST (PC)+
      EXTA: 0 ; 0 = NO EXTENSION.
      BEQ 1$
      JSR PC,SEXTA ; APPEND EXTENSION TEXT.
1$: PRINTX #NULCR ; PRINT A BLANK LINE
      MOV #NULCR,-(SP)
      MOV #1,-(SP)
      MOV SP,R0
      TRAP C#PNTX
      ADD #4,SP
      RTS PC

```

```

1525 .SBTTL PRITSSR - PRINT TSSR CONTENTS
1526
1527 ;*
1528 ;
1529 ;ROUTINE TO DISPLAY THE CONTENTS, AND BIT DEFINITIONS, OF
1530 ;THE TSSR REGISTER. THIS ROUTINE IS NORMALLY CALLED ONLY
1531 ;BY A MESSAGE PRINTING ROUTINE
1532 ;
1533 ;INPUTS:
1534 ;
1535 ;      R1      CONTENTS OF TSSR
1536 ;
1537 ;SUBORDINATE ROUTINES:
1538 ;
1539 ;      CHKAMB  CHECK FOR AMBIGUOUS CONTENTS
1540 ;
1541 ;-
1542
1543 PRITSSR:
1544     SAVREG                                ;SAVE GENERAL REGISTERS
1545     MOV      R1,R4                        ;SAVE THE TSSR CONTENTS
1546     PRINTB   #TSSRFOR,R4                 ;PRINT THE CONTENTS OF TSSR
1547     MOV      R4,-(SP)
1548     MOV      #TSSRFOR,-(SP)
1549     MOV      #2,-(SP)
1550     MOV      SP,R0
1551     TRAP     C#PNTB
1552     ADD      #6,SP
1553     MOV      R4,R0                        ;GET TSSR BACK FOR CHKAMB
1554     JSR      PC,CHKAMB                    ;ARE CONTENTS AMBIGUOUS ?
1555     BCS      5$                          ;BRANCH IF NOT
1556     PRINTX   #AMBTSSR                     ;SHOW CONTENTS ARE AMBIGUOUS
1557     MOV      #AMBTSSR,-(SP)
1558     MOV      #1,-(SP)
1559     MOV      SP,R0
1560     TRAP     C#PNTX
1561     ADD      #4,SP
1562     MOV      R4,R3
1563     BIC      #HIADDR!FATERR!TERCLS,R3    ;CONTENTS OF TSSR
1564     BEQ      20$                          ;CLEAR ALL MULTIPLE BIT FIELDS
1565     MOV      #TMPBFR,R2                    ;NO BITS ARE SET
1566     MOV      #TSSRBIT,R1                 ;TEMPORARY ASCII BUFFER
1567     TST      R3                          ;ASCII EQUIVALENT OF BITS
1568     BEQ      15$                          ;REMAINING BITS TO CONVERT
1569     CLC                                           ;BRANCH WHEN ALL ARE DONE
1570     ROL      R3                          ;CLEAR CARRY FOR SHIFT
1571     BCC      13$                          ;SHIFT NEXT BIT TO CARRY
1572     MOV      (R1),R0                      ;BRANCH IF BIT NOT SET
1573     MOVB     (R0),-(R2)                   ;POINTER TO BIT DEFINITION
1574     BNE      11$                          ;MOVE ASCII TO BUFFER
1575     MOVB     #' , -(R2)                   ;MOVE ALL BITS
1576     TST      (R1)                         ;INSERT A COMMA TO TERMINATE
1577     BR       10$                          ;POINT TO NEXT DESCRIPTION
1578     CLRB     -(R2)                       ;GET THE REMAINING BITS
1579     PRINTX   #TSSDEF,#TMPBFR              ;TERMINATE THE LINE
1580     MOV      #TMPBFR,-(SP)               ;PRINT THE BIT DEFINITIONS
1581     MOV      #TSSDEF,-(SP)

```

005430	012746	000002		MOV	#2,-(SP)	
005434	010600			MOV	SP,R0	
005436	104415			TRAP	C:PNTX	
005440	062706	000006		ADD	#6,SP	
1569						
1570	005444	010403	20:	MOV	R4,R3	;GET THE TSSR CONTENTS
1571	005446	042703		BIC	#1CTERCLS,R3	;CLEAR ALL BUT TERMINATION
1572	005452	016303		MOV	TCOCOD(R3),R3	;GET THE TERMINATION CODE MEANING
1573	005456			PRINTX	#TCOASC,R3	;PRINT THE TERMINATION CODE
	005456	010346		MOV	R3,-(SP)	
	005460	012746	006204	MOV	#TCOASC,-(SP)	
	005464	012746	000002	MOV	#2,-(SP)	
	005470	010600		MOV	SP,R0	
	005472	104415		TRAP	C:PNTX	
	005474	062706	000006	ADD	#6,SP	
1574	005500	010403		MOV	R4,R3	;TSSR CONTENTS AGAIN
1575	005502	042703	177717	BIC	#1CFATERR,R3	;CLEAR ALL BUT FATAL TERMINATION
1576	005506	001421		BEQ	25:	;DON'T PRINT IF ZERO
1577	005510	006203		ASR	R3	
1578	005512	006203		ASR	R3	
1579	005514	006203		ASR	R3	
1580	005516	016303	006744	MOV	TSFCOD(R3),R3	;ALINE TERMINATION CODE FOR INDEX
1581	005522			PRINTX	#TFCASC,R3	;GET THE FATAL TERMINATION CODE
	005522	010346		MOV	R3,-(SP)	;PRINT THE FATAL TERMINATION CODE
	005524	012746	006245	MOV	#TFCASC,-(SP)	
	005530	012746	000002	MOV	#2,-(SP)	
	005534	010600		MOV	SP,R0	
	005536	104415		TRAP	C:PNTX	
	005540	062706	000006	ADD	#6,SP	
1582	005544	012737	000031 002170	MOV	#25,FATFLG	;DROP THIS UNIT AFTER ERROR MESSAGE
1583	005552	010403		MOV	R4,R3	;GET TSSR CONTENTS
1584	005554	042703	176377	BIC	#1CHIADDR,R3	;CLEAR ALL BUT EXTENDED ADDRESS
1585	005560	001411		BEQ	30:	;DON'T PRINT IF ZERO
1586	005562			PRINTX	#TEXASC,R3	;PRINT THE EXTENDED ADDRESS BITS
	005562	010346		MOV	R3,-(SP)	
	005564	012746	006143	MOV	#TEXASC,-(SP)	
	005570	012746	000002	MOV	#2,-(SP)	
	005574	010600		MOV	SP,R0	
	005576	104415		TRAP	C:PNTX	
	005600	062706	000006	ADD	#6,SP	
1587	005604	022704	100210	30:	CMP	#100210,R4
1588	005610	001003		BNE	31:	;CHECK FOR MEDIA ERROR
1589	005612	012737	006032 002146	MOV	#EPRT3,EPRTSW	;BR, IF PROBABLY NOT TAPE ERROR
1590	005620	005737	002146	31:	TST	EPRTSW
1591	005624	001003		BNE	310:	; "PROBABLY MEDIA RELETED ERROR - BAD TAPE"
1592	005626	012737	005672 002146	MOV	#EPRT1,EPRTSW	;CHECK FOR THE SWITCH EMPTY
1593	005634	013737	002146 005644	310:	MOV	EPRTSW,32:2
1594	005642			32:	PRINTB	#EPRT1
	005642	012746	005672	MOV	#EPRT1,-(SP)	;SET SWITCH TO DEFAULT
	005646	012746	000001	MOV	#1,-(SP)	;PUT REAL SWITCHABLE MESSAGE IN PLACE
	005652	010600		MOV	SP,R0	;PRINT THE ERROR MESSAGE
	005654	104414		TRAP	C:PNTB	
	005656	062706	000004	ADD	#4,SP	
1595	005662	012737	005672 002146	MOV	#EPRT1,EPRTSW	;RESET TO NORMAL ERROR POINTER
1596	005670	000207		RTS	PC	;RETURN TO CALLER
1597						
1598	005672	045	116	045	EPRT1: .ASCIZ	'NMA *****CHECK CABLES BETWEEN CONTROLLER AND TRANSPORT*****S'

Address	Offset	Value	Field	Encoding	Description
1599	005771	045	116	045	EPRT2: .ASCIZ 'ENSA *****CHECK TRANSPORT*****S'
1600	006032	045	116	045	EPRT3: .ASCIZ 'ENSA *****POSSIBLE MEDIA RELATED ERROR - BAD TAPE*****S'
1601	006123	045	116	045	TSSRFOR: .ASCIZ 'ENSA TSSR = 06'
1602	006143	045	116	045	TEXASC: .ASCIZ 'ENSA Extended Address Bits = 06'
1603	006204	045	116	045	TCOASC: .ASCIZ 'ENSA Termination Class Code = T'
1604	006245	045	116	045	TFCASC: .ASCIZ 'ENSA Fatal Termination Class Code = T'
1605	006314	045	116	045	TSSDEF: .ASCIZ 'ENSA TSSR Bits Set: T'
1606	006343	045	116	045	AMBTSSR: .ASCIZ 'ENSA TSSR Contents Are Ambiguous'
1607					.EVEN
1608	006404	006424	006447	006475	TCOCOD: .WORD 1\$,2\$,3\$,4\$,5\$,6\$,7\$,8\$
1609	006424	116	157	162	1\$: .ASCIZ 'Normal Termination'
1610	006447	124	145	162	2\$: .ASCIZ 'Termination Condition'
1611	006475	124	141	160	3\$: .ASCIZ 'Tape Status Alert'
1612	006517	106	165	156	4\$: .ASCIZ 'Function Reject'
1613	006537	122	145	143	5\$: .ASCIZ 'Recoverable Error - Tape Position One Record Down'
1614	006621	122	145	143	6\$: .ASCIZ 'Recoverable Error - Tape Was Not Moved'
1615	006670	125	156	162	7\$: .ASCIZ 'Unrecoverable Error'
1616	006714	106	141	164	8\$: .ASCIZ 'Fatal Controller Error'
1617					.EVEN
1618					
1619	006744	006754	007010	007021	TSFCOD: .WORD 1\$,2\$,3\$,4\$
1620	006754	111	156	164	1\$: .ASCIZ 'Internal Diagnostic Failure'
1621	007010	122	145	163	2\$: .ASCIZ 'Reserved'
1622	007021	102	165	163	3\$: .ASCIZ 'Bus Interface or Sanity Check Error'
1623	007065	122	145	163	4\$: .ASCIZ 'Reserved'
1624					.EVEN

```

1626 .SBTTL PRIPKT - PRINT THE ADDRESS/CONTENTS OF COMMAND PACKET
1627
1628 ;*
1629 ;THIS ROUTINE PRINTS THE ADDRESS AND CONTENTS OF A COMMAND PACKET.
1630 ;THIS ROUTINE IS NORMALLY ONLY CALLED FROM A PRINT ROUTINE.
1631 ;
1632 ;INPUT:
1633 ;
1634 ; R0 NUMBER OF WORDS IN PACKET
1635 ; R3 HIGH ORDER COMMAND PACKET ADDRESS
1636 ; R4 ADDRESS OF COMMAND PACKET
1637 ;
1638 ; NOTE: R3 IS IGNORED IF THE KTENABLE FLAG IS CLEAR.
1639 ;
1640
1641 007076 PRIPKT::
1642 007076 SAVREG ;SAVE THE REGISTERS
1643 007102 010005 MOV R0,R5 ;SAVE NO. OF WORDS IN PACKET
1644 007104 005737 003102 TST KTENABLE ;ABOVE 28K UNDER TEST?
1645 007110 001001 BNE 10$ ;BR IF YES
1646 007112 005003 CLR R3 ;SET HIGH ORDER ADDRESS TO 0
1647 007114 010301 10$: MOV R3,R1 ;COPY HIGH ORDER ADDRESS
1648 007116 010400 MOV R4,R0 ;GET LOWER ADDRESS
1649 007120 006100 ROL R0 ;SHIFT BIT 15 INTO C BIT
1650 007122 006101 ROL R1 ;AND INTO HIGH ORDER.
1651 007124 PRINTB #PKTADD,R1,R4 ;PRINT PACKET ADDRESS
      007124 010446 MOV R4,-(SP)
      007126 010146 MOV R1,-(SP)
      007130 012746 007302 MOV #PKTADD,-(SP)
      007134 012746 000003 MOV #3,-(SP)
      007140 010600 MOV SP,R0
      007142 104414 TRAP C$PNTB
      007144 062706 000010 ADD #10,SP
1652 007150 010300 15$: MOV R3,R0 ;GET HIGH ORDER ADDRESS
1653 007152 001404 BEQ 20$ ;BR IF NOT ABOVE 28K.
1654 007154 010401 MOV R4,R1 ;GET LOW ORDER ADDRESS
1655 007156 004737 020304 JSR PC,SETMAP ;SETUP PAR6 MAPPING FOR 18 BIT ADDRESS
1656 007162 010004 MOV R0,R4 ;GET RETURNED PAR6 ADDRESS BIAS
1657 007164 005001 20$: CLR R1 ;SAVE WORD NUMBER
1658 007166 012402 25$: MOV (R4)+,R2 ;GET PACKET CONTENTS
1659 007170 PRINTB #PKTFRM,R1,R2 ;PRINT THE DATA
      007170 010246 MOV R2,-(SP)
      007172 010146 MOV R1,-(SP)
      007174 012746 007244 MOV #PKTFRM,(SP)
      007200 012746 000003 MOV #3,-(SP)
      007204 010600 MOV SP,R0
      007206 104414 TRAP C$PNTB
      007210 062706 000010 ADD #10,SP
1660 007214 005201 INC R1 ;NEXT WORD NUMBER
1661 007216 020105 CMP R1,R5 ;DONE ALL PACKET WORDS?
1662 007220 002762 BLT 25$ ;LOOP TILL ALL DONE
1663 007222 PRINTB #PKTNEW ;JUST A COUPLE NEW LINES
      007222 012746 007337 MOV #PKTNEW,(SP)
      007226 012746 000001 MOV #1,-(SP)
      007232 010600 MOV SP,R0
      007234 104414 TRAP C$PNTB
      007236 062706 000004 ADD #4,SP

```

1664	007242	000207			RTS	PC	,RETURN
1665							
1666	007244	045	116	045	PKTFRM:	.ASCIZ	'#N#A Packet Word #D1#A = #06'
1667	007302	045	116	045	PKTADD:	.ASCIZ	'#N#A Packet Address = #01#05'
1668							
1669	007337	045	116	045	PKTNEW:	.ASCIZ	'#N#N#A '
1670						.EVEN	
1671							

```

1673                                     .SBTTL PRIBXOR - PRINT EXPD, RECV AND XOR BYTE
1674
1675                                     ;*
1676                                     ;
1677                                     ;PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE DATA BYTE
1678                                     ;THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
1679                                     ;
1680                                     ;INPUTS:
1681                                     ;
1682                                     ;      R1      RECEIVED DATA
1683                                     ;      R2      EXPECTED DATA
1684                                     ;
1685                                     ;OUTPUT:
1686                                     ;
1687                                     ;      R0      XOR OF EXPECTED/RECEIVED DATA
1688                                     ;
1689                                     ;-
1690
1691 007350 PRIBXOR::
1692 007350 SAVREG                                ;SAVE THE REGISTERS
1693 007354 010203 MOV      R2,R3                ;EXPECTED DATA
1694 007356 XOR      R1,R3                ;FORM THE EXCLUSIVE OR
1695 007366 012700 177400 MOV      #C<377>,R0    ;BYTE MASK
1696 007372 040001 BIC      R0,R1                ;SAVE LOW BYTE RECV
1697 007374 040002 BIC      R0,R2                ;SAVE LOW BYTE EXPD
1698 007376 040003 BIC      R0,R3                ;SAVE LOW BYTE XOR
1699 007400 PRINTB  #XORBFOR,R2,R1,R3 ;PRINT THE MESSAGE
1700 007400 010346 MOV      R3,-(SP)
1701 007402 010146 MOV      R1,-(SP)
1702 007404 010246 MOV      R2,-(SP)
1703 007406 012746 007432 MOV      #XORBFOR,-(SP)
1704 007412 012746 000004 MOV      #4,-(SP)
1705 007416 010600 MOV      SP,R0
1706 007420 104414 TRAP     C:PNTB
1707 007422 062706 000012 ADD      #12,SP
1708 007426 010300 MOV      R3,R0                ;R0 HAS XOR ON RETURN
1709 007430 000207 RTS      PC                  ;RETURN TO CALLER
1710
1711 007432 045 116 045 XORBFOR: .ASCIZ 'NNA EXPD: 03NA RECV: 03NA XOR: 03'
1712 .EVEN

```

```

1707 .SBTTL PRI XOR - PRINT EXPD, RECV AND XOR
1708
1709 ;*
1710 ;
1711 ;PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE TWO
1712 ;THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
1713 ;
1714 ;INPUTS:
1715 ;
1716 ;      R1      RECEIVED DATA
1717 ;      R2      EXPECTED DATA
1718 ;
1719 ;OUTPUT:
1720 ;
1721 ;      R0      XOR OF EXPECTED/RECEIVED DATA
1722 ;
1723 ;-
1724
1725 007500 PRI XOR::
1726 007500 SAVREG                      ;SAVE THE REGISTERS
1727 007504 010203 MOV      R2,R3      ;EXPECTED DATA
1728 007506 XOR      R1,R3      ;FORM THE EXCLUSIVE OR
1729 007516 PRINTB   #XORFOR,R2,R1,R3 ;PRINT THE MESSAGE
1730 007516 010346 MOV      R3,-(SP)
1731 007520 010146 MOV      R1,-(SP)
1732 007522 010246 MOV      R2,-(SP)
1733 007524 012746 007550 MOV      #XORFOR,-(SP)
1734 007530 012746 000004 MOV      #4,-(SP)
1735 007534 010600 MOV      SP,R0
1736 007536 104414 TRAP     C#PNTB
1737 007540 062706 000012 ADD      #12,SP
1738 007544 010300 MOV      R3,R0      ;R0 HAS XOR ON RETURN
1739 007546 000207 RTS      PC        ;RETURN TO CALLER
1740
1741 007550 045 116 045 XORFOR: .ASCIZ '##N##A EXPD: #06##A RECV: #06##A XOR: #06'
1742 .EVEN

```

```

1736 .SBTTL PRIQU - PRINT BIT NUMBERS AS ASCII EQUIVALENT
1737
1738 ;*
1739 ;
1740 ;ROUTINE TO CONVERT BIT VALUES TO ASCII AND PRINT THE STRING
1741 ;THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
1742 ;
1743 ;INPUTS:
1744 ;
1745 ;      R0      OCTAL VALUE TO CONVERT
1746 ;      R1      TABLE OF POINTERS TO ASCII EQUIVALENT
1747 ;
1748 ;-
1749
1750 PRIQU:
1751     SAVREG
1752     RTS      PC      ;SAVE THE REGISTERS
                       ;RETURN TO CALLER
1753
1754
1755
1756
1757 .SBTTL PRIRAM - PRINT RAM ADDRESS
1758
1759 ;*
1760 ;
1761 ;PRINT CONTROLLER RAM ADDRESS.
1762 ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
1763 ;
1764 ;INPUTS:
1765 ;
1766 ;      R4      RAM ADDRESS
1767 ;
1768 ;-
1769 PRIRAM:
1770     SAVREG
1771     PRINTB  #RAMFOR,R4      ;SAVE R1 R5 UNTIL NEXT RETURN
                               ;PRINT RAM ADDRESS IN ERROR
1772     MOV     R4,-(SP)
1773     MOV     #RAMFOR,-(SP)
1774     MOV     #2,-(SP)
1775     MOV     SP,R0
1776     TRAP    C:PNTB
1777     ADD     #6,SP
1778     RTS     PC      ;RETURN
1779
1780
1781
1782
1783
1784
1785
1786
1773 007654      045      116      045 RAMFOR: .ASCIZ 'N/A CONTROLLER RAM ADDRESS - #06'
1774      .EVEN
1775
1776
1777 .SBTTL PRIADD - PRINT MEMORY ERROR ADDRESS
1778
1779 ;*
1780 ;
1781 ;PRINT MEMORY ADDRESS
1782 ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
1783 ;
1784 ; IMPLICIT INPUTS
1785 ;
1786 ;      ERRHI   - HIGH ORDER ADDRESS
1787 ;      ERRLO   - LOW ORDER ADDRESS

```

```

1787
1788
1789 007716
1790 007716
1791 007722 013700 002202
1792 007726 013701 002204
1793 007732 010102
1794 007734 006101
1795 007736 006100
1796 007740
    007740 010246
    007742 010046
    007744 012746 007766
    007750 012746 000003
    007754 010600
    007756 104414
    007760 062706 000010
1797 007764 000207
1798
1799 007766 045 116 045 PRIA0: .ASCIZ 'NMA MEMORY ERROR ADDRESS = 0105'
1800 .EVEN
1801
1802
1803 .SBTTL PRITADD PRINT MEMORY TEST ADDRESS
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815 010032
1816 010032
1817 010036 013700 002202
1818 010042 013701 002204
1819 010046 010102
1820 010050 006101
1821 010052 006100
1822 010054
    010054 010246
    010056 010046
    010060 012746 010102
    010064 012746 000003
    010070 010600
    010072 104414
    010074 062706 000010
1823 010100 000207
1824
1825 010102 045 116 045 PRITO: .ASCIZ 'NMA MEMORY TEST ADDRESS = 0105'
1826 .EVEN
1827
1828
1829

```

```

;
; -
PRIADD:
    SAVREG
    MOV ERRHI,R0 ;SAVE R1-R5 UNTIL NEXT RETURN
    MOV ERRLO,R1 ;GET HIGH ADDRESS
    MOV R1,R2 ;GET LOW ADDRESS
    ROL R1 ;COPY LOW ADDRESS
    ROL R0 ;SHIFT BIT 15 TO C BIT
    PRINTB @PRIA0,R0,R2 ;SHIFT INTO HIGH ORDER
    MOV R2,-(SP) ;PRINT MEMORY ADDRESS IN ERROR
    MOV R0,-(SP)
    MOV @PRIA0,-(SP)
    MOV #3,-(SP)
    MOV SP,R0
    TRAP C:PNTB
    ADD #10,SP
    RTS PC ;RETURN

;
; -
PRITADD:
    SAVREG
    MOV ERRHI,R0 ;SAVE R1-R5 UNTIL NEXT RETURN
    MOV ERRLO,R1 ;GET HIGH ADDRESS
    MOV R1,R2 ;GET LOW ADDRESS
    ROL R1 ;COPY LOW ADDRESS
    ROL R0 ;SHIFT BIT 15 TO C BIT
    PRINTB @PRITO,R0,R2 ;SHIFT INTO HIGH ORDER
    MOV R2,-(SP) ;PRINT MEMORY ADDRESS IN ERROR
    MOV R0,-(SP)
    MOV @PRITO,-(SP)
    MOV #3,(SP)
    MOV SP,R0
    TRAP C:PNTB
    ADD #10,SP
    RTS PC ;RETURN

```

```

1831                                     .SBTTL SPACE SPACE RECORDS (FORWARD AND REVERSE) COMMAND
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866 010144
1867 010144
1868 010150 012737 000764 010340
1869 010156 012737 140010 010330
1870 010164 005703
1871 010166 100403
1872 010170 010337 010332
1873 010174 000407
1874 010176 042703 100000
1875 010202 010337 010332
1876 010206 052737 000400 010330
1877 010214 012704 010330
1878 010220 010465 177776
1879 010224 004737 017134
1880 010230 103420
1881 010232
    010232 012727 000250
    010236 000000
    010240 013727 002116
    010244 000000
    010246 005367 177772
    010252 001375

ROUTINE TO ISSUE A SPACE RECORDS
COMMAND (FORWARD OR REVERSE)

INPUT:
    R3      NUMBER OF RECORDS TO BE SPACED OVER
            BIT15 CONTROLS DIRECTION
            BIT15 = 0 IS FORWARD
            BIT15 = 1 IS REVERSE
    R5      FIRST DEVICE UNIBUS ADDRESS

REQUIRES A WRITE CHARACTERISTICS DONE PREVIOUSLY

OUTPUT:
    CARRY   SET - SPACE RECORDS COMMAND OK
            CLR - SPACE RECORDS FAILED

    R0      THE CONTENTS OF R4 IS MOVED TO R0

IMPLICIT OUTPUT:
    TAPE HAS BEEN MOVED

SIDE EFFECTS:
    -

SPACE::
    SAVREG                                ;SAVE THE GENERAL REGISTERS
    MOV     #500,.SDELAY                  ;SET UP DELAY
    MOV     #140010,R0                    ;SET UP COMMAND, SPACE FORWARD
    TST     R3                            ;CHECK FOR DIRECTION
    BMI     5:                             ;BR, IF REVERSE INDICATED
    MOV     R3,R0                          ;LOAD UP NUMBER OF RECORDS TO SPACE
    BR      10:                             ;GO DO COMMAND
    BIC     #BIT15,R3                      ;CLEAR DIRECTION BIT
    MOV     R3,R0                          ;LOAD UP NUMBER OF RECORDS TO SPACE
    BIS     #BIT8,R0                       ;SET REVERSE BIT IN COMMAND PACKET
    10:     MOV     #R0,R4                 ;SET UP R4 WITH PACKET ADDRESS
    MOV     R4,TSDB(R5)                   ;SEND OUT COMMAND
    JSR     PC,WAITF                       ;WAIT FOR SSR
    BCS     20:                             ;BR, IF SSR IS SET AND OK
    DELAY   250                           ;DELAY ABOUT .25 SECONDS
    MOV     #250,(PC)
    .WORD   0
    MOV     L#DLX,(PC)
    .WORD   0
    DEC     -6(PC)
    BNE     -4

```

C5

CZIKGA TK 25 FBT END FUNC 03 MACRO M1200 20-APR 84 08:13 PAGE 40 1
 SPACE RECORDS (FORWARD AND REVERSE) COMMAND

SEQ 54

010254	005367	177756		DEC	-22(PC)	
010260	001367			BNE	.-20	
1882 010262	005337	010340		DEC	SDELAY	;BUMP DELAY COUNTER DOWN
1883 010266	001356			BNE	15:	;BR, IF MORE DELAY
1884 010270	000411			BR	60:	;BR IF TROUBLE CARRY = CLEAR
1885 010272	016501	0C0000	20:	MOV	TSSR(R5),R1	;READ TSSR
1886 010276	012702	000200		MOV	#SSR,R2	;SET UP EXPECTED
1887 010302	020201		25:	CMP	R2,R1	;ARE THEY OK
1888 010304	001401			BEQ	40:	;BR, IF EQUAL = OK
1889 010306	000402			BR	60:	;TROUBLE EXIT
1890 010310	000261		40:	SEC		;SET CARRY NO TROUBLE
1891 010312	000401			BR	70:	;EXIT
1892 010314	000241		60:	CLC		;CARRY CLEAR = ERROR
1893 010316			70:			
1894 010316	010400			MOV	R4,R0	;PASS PACKET ADDRESS
1895 010320	000207			RTS	PC	;RETURN

1897			:		
1898			:		
1899			:		
1900			:	PACKET FOR SPACE COMMAND	
1901			:		
1903 010322			:	.BLKB 10-<. TUV2A&7>	
1905			:		
1906			:	COMMAND WORD	
1907 010330 000000			:	801: .WORD	
1908			:	NUMBER OF RECORDS TO BE SPACED OVER WORD	
1909 010332 000000			:	901: .WORD	
1910 010334 000000			:	.WORD	
1911 010336 000000			:	.WORD	
1912 010340 000000			:	SDELAY: .WORD 0	:DELAY COUNTER
1913			:	.EVEN	

```

1915 .SBTTL WRCHMR - WRITE CHARACTERISTICS COMMAND
1916
1917 ;*
1918 ;
1919 ;ROUTINE TO ISSUE A WRITE CHARACTERISTICS
1920 ;COMMAND SO THAT OTHER COMMANDS WILL BE ACCEPTED
1921 ;
1922 ;INPUT:
1923 ;
1924 ; R4 ADDRESS OF PACKET FROM TEST
1925 ; R5 FIRST DEVICE UNIBUS ADDRESS
1926 ; REQUIRES A CALL TO SOFINIT BE DONE PREVIOUSLY
1927 ;
1928 ;OUTPUT:
1929 ;
1930 ; R0 TSSR CONTENTS
1931 ; CARRY SET - WRITE CHARACTERISTICS COMMAND OK
1932 ; CLR - WRITE CHARACTERISTICS FAILED
1933 ;
1934 ;IMPLICIT OUTPUT:
1935 ;
1936 ; MESSAGE BUFFER AND OTHER BUFFERS ALL SET UP
1937 ; SOFTWARE SWITCHES SET AS FOLLOWS:
1938 ; BENBSW = BUFFER ENABLE SWITCH ON OR OFF
1939 ;
1940 ;
1941 ;SIDE EFFECTS:
1942 ;
1943 ;
1944 ;
1945 ;
1946 010342 WRCHMR::
1947 010342 SAVREG
1948 010346 005037 002174 CLR BENBSW ;SAVE THE GENERAL REGISTERS
1949 010352 010465 177776 10$: MOV R4,TSD8(R5) ;CLEAR BUFFER ENABLE SWITCH
1950 010356 004737 017252 JSR PC,CHKTSSR ;SEND OUT COMMAND
1951 010362 103401 BCS 20$ ;WAIT FOR SSR
1952 010364 000423 BR 60$ ;BR, IF SSR IS SET AND OK
1953 010366 016501 000000 20$: MOV TSSR(R5),R1 ;BR IF TROUBLE CARRY = CLEAR
1954 010372 012702 000200 MOV #SSR,R2 ;READ TSSR
1955 010376 032701 000100 BIT #OFL,R1 ;SET UP EXPECTED
1956 010402 001402 BEQ 25$ ;WAS OFF LINE SET IN TSSR
1957 010404 052702 000100 BIS #OFL,R2 ;BR, IF NO OFL SET
1958 010410 020201 25$: CMP R2,R1 ;MAKE THEM LOOK ALIKE
1959 010412 001401 BEQ 40$ ;ARE THEY OK
1960 010414 000407 BR 60$ ;BR, IF EQUAL = OK
1961 010416 062704 000010 40$: ADD #8,,R4 ;TROUBLE EXIT
1962 010422 011403 MOV (R4),R3 ;POINT TO WRT CHARA DATA PACKET
1963 010424 010337 002716 MOV R3,MESBFA ;GET ADDRESS OF MESSAGE BUFFER
1964 010430 000261 SEC ;STORE FOR PRINT ROUTINES
1965 010432 000401 BR 70$ ;SET CARRY NO TROUBLE
1966 010434 000241 60$: CLC ;EXIT
1967 010436 016500 000000 70$: MOV TSSR(R5),R0 ;CARRY CLEAR = ERROR
1968 010442 000207 RTS PC ;RETURN TSSR CONTENTS
1969 ;
1970 ;

```

```

1972 .SBTTL REWIND - POSITION TAPE (REWIND) COMMAND
1973
1974 ;*
1975 ;THIS ROUTINE WILL REWIND THE SELECTED TAPE.
1976 ;
1977 ; CAUTION: THE ROUTINE DOES NOT WAIT FOR BOT
1978 ; TO ARRIVE. ALSO THE CALLER MUST CHECK FOR
1979 ; SSR TO SET IN THE TSSR
1980 ;
1981 ;
1982 ;CALLING SEQUENCE:
1983 ;
1984 ; DO A SOFT INIT
1985 ; DO A WRITE CHARACTERISTICS
1986 ; JSR PC,REWIND
1987 ;
1988 ;INPUT:
1989 ;
1990 ; R5 FIRST DEVICE UNIBUS ADDRESS
1991 ;
1992 ;
1993 ;OUTPUT
1994 ;
1995 ; R0 THE CONTENTS OF R4 IS PASSED TO R0
1996 ;
1997 REWIND::
1998 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
1999 MOV #RWPACK,R4 ;GET PACKET ADDRESS
2000 MOV R4,TSD8(R5) ;SEND PACKET ADDRESS TO EXECUTE
2001 MOV #360.,R3 ;ENOUGH TIME FOR 2400' REEL TO REWIND
2002 10$: JSR PC,WAITF ;WAIT FOR SSR TO SET
2003 BCS 20$ ;LEAVE WHEN SSR IS SET
2004 DELAY 250. ;WAIT FOR .25 SECONDS
      MOV #250.,(PC)+
      .WORD 0
      MOV L$DLY,(PC)+
      .WORD 0
      DEC -6(PC)
      BNE .-4
      DEC -22(PC)
      BNE .-20
2005 DEC R3 ;BUMP COUNTER DOWN
2006 BNE 10$ ;KEEP GOING
2007 CLC ;CLEAR CARRY TO SET ERROR
2008 20$: MOV R4,R0 ;PASS THE PACKET ADDRESS
2009 RTS PC ;RETURN
2011 .BLKB 10-<.-TUV2A&7>
2013 RWPACK:
2014 .WORD 102010 ;POSTION COMMAND (REWIND)
2015 .WORD 0 ;NOT USED

```

```

2017 .SBTTL CKRAM - COMPARE RAM TO I/O PACKET
2018
2019 ;*
2020 ;
2021 ;ROUTINE TO READ THE FIRST 8 BYTES FROM RAM
2022 ;MEMORY AND COMPARE THIS DATA TO A COMMAND PACKET.
2023 ;
2024 ;INPUT:
2025 ;
2026 ; R4 ADDRESS OF THE COMMAND PACKET
2027 ; R5 FIRST DEVICE UNIBUS ADDRESS
2028 ;
2029 ;OUTPUT:
2030 ;
2031 ; CARRY SET - RAM MATCHES PACKET
2032 ; CLR - RAM DOES NOT MATCH PACKET
2033 ;
2034 ;IMPLICIT OUTPUT:
2035 ;
2036 ; THE TABLE RAMDATA IS FILLED WITH THE
2037 ; DATA HELD IN RAM.
2038 ; RAMSIZ IS SET TO 8. FOR PRAMPKT ROUTINE
2039 ;
2040 ;SIDE EFFECTS:
2041 ;
2042 ;
2043 ;-
2044
2045 CKRAM::
2046 SAVREG ;SAVE THE GENERAL REGISTERS
2047 MOV #RAMDATA,R1 ;ADDRESS TO SAVE THE RAM DATA
2048 MOV #RMPKTBEG,R2 ;BYTE ADDRESS OF FIRST RAM DATA
2049 CLR R3 ;CLEAR THE ERROR FLAG
2050 JSR PC,CHKTSSR ;WAIT FOR SSR
2051 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
2052 MOVB R2,TSDBH(R5) ;SELECT NEXT RAM ADDRESS
2053 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
2054 MOVB TSBAL(R5),(R1) ;READ THE RAM DATA
2055 CMPB (R1)+,(R4)+ ;COMPARE TO EXPECTED
2056 BEQ 20$ ;BRANCH IF OK
2057 INC R3 ;SET ERROR FLAG
2058 INC R2 ;ADDRESS OF NEXT RAM LOCATION
2059 CMP R2,#RMPKTEND ;REACHED END YET ?
2060 BLE 10$ ;BRANCH TILL ALL READ
2061 TST R3 ;WAS AN ERROR FOUND ?
2062 BEQ 30$ ;BRANCH IF NOT
2063 CLC ;CLEAR CARRY TO SHOW ERROR
2064 BR 50$ ;AND EXIT
2065 SEC ;SHOW GOOD COMPARE
2066 MOV #8.,RAMSIZ ;SETUP RAMSIZ FOR PRAMPKT ROUTINE
2067 RTS PC ;RETURN
2068

```

2045	010544						
2046	010544						
2047	010550	012701	002206				
2048	010554	012702	000020				
2049	010560	005003					
2050	010562	004737	017252				
2051	010566	004737	017252	10\$:			
2052	010572	110265	177777				
2053	010576	004737	017252				
2054	010602	116511	177776				
2055	010606	122124					
2056	010610	001401					
2057	010612	005203					
2058	010614	005202		20\$:			
2059	010616	020227	000027				
2060	010622	003761					
2061	010624	005703					
2062	010626	001402					
2063	010630	000241					
2064	010632	000401					
2065	010634	000261		30\$:			
2066	010636	012737	000010 002246	50\$:			
2067	010644	000207					
2068							

```

2070 .SBTTL RAMER - READ AND DISPLAY SELECTED RAM
2071 ;*
2072 ;
2073 ;ROUTINE TO READ THE SELECTED RAM LOCATIONS
2074 ;
2075 ;INPUT:
2076 ;
2077 ; R5 FIRST DEVICE UNIBUS ADDRESS
2078 ; CONSOLE WILL ALSO BE PRINTED TO
2079 ;
2080 ;IMPLICIT OUTPUT:
2081 ;
2082 ; THE TABLE RAMDATA IS FILLED WITH THE
2083 ; DATA HELD IN RAM.
2084 ;
2085 ;SIDE EFFECTS:
2086 ;
2087 ;
2088 ;-
2089
2090 RAMER::
2091 SAVREG ;SAVE THE GENERAL REGISTERS
2092 MOV RAMR5H,R5 ;RESET R5 TO FIRST DEVICE REGISTER
2093 MOV @RAMDATA,R1 ;ADDRESS TO SAVE THE RAM DATA
2094 MOV RAMHLD,R2 ;BYTE ADDRESS OF THE FIRST RAM DATA
2095 MOV RAMSIZ,R3 ;SET THE SIZE OF THE READ UP
2096 10$: JSR PC,CHKTSSR ;WAIT FOR THE SSR TO SET
2097 MOV R2,TSDBH(R5) ;SELECT NEXT RAM ADDRESS
2098 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
2099 MOV TSBAL(R5),(R1)+ ;READ THE RAM DATA
2100 20$: ADD #1,R2 ;ADDRESS OF THE NEXT RAM LOCATION
2101 SOB R3,10$ ;NUMBER OF LOCATIONS COUNTER
2102 MOV RAMSIZ,R4 ;GET THE RAM SIZE
2103 MOV RAMHLD,R2 ;GET THE STARTING RAM ADDRESS
2104 ADD R2,R4 ;CALCULATE THE END ADDRESS
2105 SUB #1,R4 ;CORRECT VALUE OF PRINTOUT
2106 PRINTX @RAMIOP,R2,R4 ;RAM ADDRESS = 10 - 17, ETC.
2107 MOV R4,-(SP)
2108 MOV R2,-(SP)
2109 MOV @RAMIOP,-(SP)
2110 MOV #3,-(SP)
2111 MOV SP,R0
2112 TRAP C:PNTX
2113 ADD #10,SP
2114 MOV @RAMDATA,R1 ;ADDRESS OF WHERE RAM DATA IS
2115 MOV RAMSIZ,R3 ;THE SIZE OF THE RAM FIELD READ
2116 30$: CLR R4 ;NO EXTRA DATA LEFT OVER
2117 MOV (R1)+,R4 ;PICK UP BYTE OF RAM DATA
2118 BIC #177400,R4 ;GET RID OF SIGN EXTEND
2119 PRINTX @RAMPD,R4 ;"010 211 111 222 377 000 123 134 ETC."
2120 MOV R4,(SP)
2121 MOV @RAMPD,-(SP)
2122 MOV #2,-(SP)
2123 MOV SP,R0
2124 TRAP C:PNTX
2125 ADD #6,SP
2126 SOB R3,30$ ;LOOP UNTIL ALL PRINTED

```

CZTKGA TK 25 FRT END FUNC #3 MACRO M1200 20-APR-84 08:13 PAGE 45-1
 RAMER - READ AND DISPLAY SELECTED RAM

SEQ 60

```

2114 011026 000207          504:   RTS      PC          ;RETURN
2115
2116 011030 000000          RAMHLD: .WORD    0          ;RAM ADDR HOLDER 1ST ADDRESS
2117 011032 000000          RAMR5H: .WORD    0          ;HOLDS R5 FOR LATER
2118 011034      045      116      045 RAMIOP: .ASCIZ  'N#A Ram Address (Octal) * #03#A - #03#N'
2119 011105      045      101      040 RAMPD: .ASCIZ  'A #03#A '
2120
2121

```

```

2123 .SBTTL CKRAM2 - COMPARE RAM TO I/O CHARACTERISTICS DATA
2124 ;*
2125 ;
2126 ;ROUTINE TO READ THE FIRST 8 OR 10 BYTES FROM RAM
2127 ;MEMORY AND COMPARE THIS DATA TO A CHARACTERISTICS DATA BLOCK.
2128 ;
2129 ;INPUT:
2130 ;
2131 ; R4 ADDRESS OF THE CHARACTERISTICS DATA
2132 ; R5 FIRST DEVICE UNIBUS ADDRESS
2133 ;
2134 ;OUTPUT:
2135 ;
2136 ; CARRY SET - RAM MATCHES PACKET
2137 ; CLR - RAM DOES NOT MATCH PACKET
2138 ;
2139 ;IMPLICIT OUTPUT:
2140 ;
2141 ; THE TABLE RAMDATA IS FILLED WITH THE
2142 ; DATA HELD IN RAM.
2143 ; RAMSIZ IS SET TO 8. OR 10. FOR PRAMPKT ROUTINE
2144 ;
2145 ;SIDE EFFECTS:
2146 ;
2147 ;
2148 ;-
2149
2150 011120 CKRAM2::
2151 011120 SAVREG ;SAVE THE GENERAL REGISTER
2152 011124 012701 002206 MOV #RAMDATA,R1 ;ADDRESS TO SAVE THE RAM DATA
2153 011130 012702 000167 MOV #RMCHBEG,R2 ;BYTE ADDRESS OF FIRST RAM DATA
2154 011134 005003 CLR R3 ;CLEAR THE ERROR FLAG
2155 011136 004737 017252 JSR PC,CHKTSSR ;WAIT FOR SSR
2156 011142 004737 017252 10$: JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
2157 011146 110265 177777 MOVB R2,TSDBH(R5) ;SELECT NEXT RAM ADDRESS
2158 011152 004737 017252 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
2159 011156 116511 177776 MOVB TSBAL(R5),(R1) ;READ THE RAM DATA
2160 011162 122124 CMPB (R1),.(R4) ;COMPARE TO EXPECTED
2161 011164 001401 BEQ 20$ ;BRANCH IF OK
2162 011166 005203 INC R3 ;SET ERROR FLAG
2163 011170 005202 20$: INC R2 ;ADDRESS OF NEXT RAM LOCATION
2164 011172 012737 000010 002246 MOV #8.,RAMSIZ ;ASSUME NORMAL NOT SET
2165 011200 020227 000176 CMP R2, #RMCHEND-2 ;REACHED END YET ?
2166 011204 003756 BLE 10$ ;BRANCH TILL ALL READ
2167 011206 005703 27$: TST R3 ;WAS AN ERROR FOUND ?
2168 011210 001402 BEQ 30$ ;BRANCH IF NOT
2169 011212 000241 CLC ;CLEAR CARRY TO SHOW ERROR
2170 011214 000401 BR 50$ ;AND EXIT
2171 011216 000261 30$: SEC ;SHOW GOOD COMPARE
2172 011220 000207 50$: RTS PC ;RETURN

```

```

2174 .SBTTL CKMSG - COMPARE WRITE CHAR. MESSAGE BUFFERS
2175 ;*
2176 ;
2177 ;ROUTINE TO COMPARE A WRITE CHARACTERISTICS EXPD AND RECV
2178 ;BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
2179 ;ERROR PRINT ROUTINES.
2180 ;
2181 ;INPUT:
2182 ;
2183 ; R0 RECV MESSAGE BUFFER HIGH ORDER ADDRESS
2184 ; R1 RECV MESSAGE BUFFER LOW ORDER ADDRESS
2185 ; R2 EXPD MESSAGE BUFFER ADDRESS
2186 ;OUTPUT:
2187 ;
2188 ; CARRY SET - MESSAGE BUFFERS MATCH
2189 ; CLR -MESSAGE BUFFERS DON'T MATCH
2190 ;
2191 ;IMPLICIT OUTPUT:
2192 ;
2193 ; EXPMSG BUFFER IS SET TO EXPD DATA
2194 ; RECVMSG BUFFER IS SET TO RECV DATA
2195 ; RCVHIADD SET TO HIGH ORDER ADDRESS OF RECV
2196 ; RCVLOAD SET TO LOW ORDER ADDRESS OF RECV
2197 ;
2198 ;-
2199 CKMSG::
2200 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2201 MOV R0,RCVHIADD ;SAVE RECV HIGH ADDRESS
2202 MOV R1,RCVLOAD ;SAVE RECV LOW ADDRESS
2203 TST KTENABLE ;TESTING ABOVE 28K?
2204 BEQ 10$ ;BR IF NO
2205 JSR PC,SETMAP ;RETURN ADDRESS BIASED TO PAR6 IN R0
2206 MOV R0,R1 ;GET RETURNED ADDRESS BIASED TO PAR6
2207 10$: CLR R4 ;WORD IN BUFFER
2208 CLR R3 ;CLEAR ERROR SEEN FLAG
2209 MOV R2,R5 ;GET EXPD BUFFER ADDRESS
2210 15$: MOV (R2),EXPMSG(R4) ;SAVE EXPD FOR ERROR REPORT
2211 MOV (R1),RECVMSG(R4) ;SAVE RECV FOR ERROR REPORT
2212 CMP (R2), (R1) ;EXPD EQUAL RECV?
2213 BEQ 25$ ;BR IF YES
2214 INC R3 ;SET ERROR SEEN FLAG
2215 25$: ADD #2,R4 ;POINT TO NEXT WORD ADDRESS
2216 CMP R4,#14 ;DONE FIRST 7 WORDS?
2217 BLE 15$ ;BR IF NO
2218 BIT #X2.EXTF,XST2(R5) ;IS EXTENDED FEATURES SET IN EXPD?
2219 BEQ 50$ ;BR IF NO
2220 CMP R4,#16 ;DONE EXTENDED FEATURES WORD?
2221 BLE 15$ ;BR IF NO
2222 50$: TST R3 ;ANY ERRORS SEEN?
2223 BEQ 55$ ;BR IF NO
2224 CLC ;SET FAILURE
2225 BR 60$ ;
2226 55$: SEC ;SET SUCCESS
2227 60$: RTS PC ;RETURN
2228

```

```

2230 .SBTTL CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS
2231 ;*
2232 ;
2233 ;ROUTINE TO COMPARE AN EXPECTED AND RECEIVED MESSAGE
2234 ;BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
2235 ;ERROR PRINT ROUTINES.
2236 ;
2237 ;INPUT:
2238 ;
2239 ;      R0      RECV MESSAGE BUFFER HIGH ORDER ADDRESS
2240 ;      R1      RECV MESSAGE BUFFER LOW ORDER ADDRESS
2241 ;      R2      EXPD MESSAGE BUFFER ADDRESS
2242 ;      R3      NUMBER OF BYTES TO COMPARE
2243 ;
2244 ;OUTPUT:
2245 ;
2246 ;      CARRY   SET - MESSAGE BUFFERS MATCH
2247 ;             CLR - MESSAGE BUFFERS DON'T MATCH
2248 ;
2249 ;IMPLICIT OUTPUT:
2250 ;
2251 ;      EXPMSG   BUFFER IS SET TO EXPD DATA
2252 ;      RECVMSG  BUFFER IS SET TO RECV DATA
2253 ;      RCVHIADD SET TO HIGH ORDER ADDRESS OF RECV
2254 ;      RCVLOADD SET TO LOW ORDER ADDRESS OF RECV
2255 ;
2256 ;-
2257 CKMSG2::
2258 SAVREG                                ;SAVE R1-R5 UNTIL NEXT RETURN
2259 CMP      R3,#RECVMSG-EXPMSG,#800    ;IS COUNT ABOVE MAX ALLOWED?
2260 BLE      5$                          ;BR IF NO
2261 MOV      #RECVMSG-EXPMSG,R3,#800
2262 PRINTF   #DEBUGMSG,800
2263          MOV      #DEBUGMSG,-(SP)
2264          MOV      #1,-(SP)
2265          MOV      SP,R0
2266          TRAP    C#PRINTF
2267          ADD      #4,SP
2268 5$:      MOV      R0,RCVHIADD          ;SAVE RECV HIGH ADDRESS
2269          MOV      R1,RCVLOAD          ;SAVE RECV LOW ADDRESS
2270          TST      KTENABLE            ;TESTING ABOVE 28K?
2271          BEQ      10$                 ;BR IF NO
2272          JSR      PC,SETMAP           ;RETURN ADDRESS BIASED TO PAR6 IN R0
2273          MOV      R0,R1              ;GET RETURNED ADDRESS BIASED TO PAR6
2274 10$:      CLR      R4                 ;WORD IN BUFFER
2275          CLR      R5                 ;CLEAR ERROR SEEN FLAG
2276 15$:      MOVB     (R2),EXPMSG(R4)    ;SAVE EXPD FOR ERROR REPORT
2277          MOVB     (R1),RECVMSG(R4)   ;SAVE RECV FOR ERROR REPORT
2278          CMPB     (R2)+,(R1)+        ;EXPD EQUAL RECV?
2279          BEQ      25$                 ;BR IF YES
2280          INC      R5                 ;SET ERROR SEEN FLAG
2281 25$:      ADD      #1,R4              ;POINT TO NEXT BYTE
2282          CMP      R4,R3              ;DONE ALL BYTES?
2283          BGE      50$                 ;BR IF YES
2284          BR       15$                 ;DO NEXT BYTE
2285 50$:      TST      R5                 ;ANY ERRORS SEEN?
2286          BEQ      55$                 ;BR IF NO

```

M5

CZTKGA TK 25 FRT END FUNC #3 MACRO M1200 20 APR 84 08:13 PAGE 48 1
CKMSG2 COMPARE EXPD RECV MESSAGE BUFFERS

SEQ 64

```

2282 011464 000241      CLC                      ;SET FAILURE
2283 011466 000401      BR      60$              ;
2284 011470 000261      55$: SEC                  ;SET SUCCESS
2285 011472 000207      60$: RTS                  ;RETURN
2286
2287 011474      120      122      117  DEBUGMSG:      .ASCIIZ 'PROGRAM INTERNAL ERROR  CKMSG2 MESSAGE BUFFER EXCEEDED-' ;000
2288 011564      045      116      045  FERCM: .ASCIIZ /MSGA ***/
2289 011575      040      040      124  ERCM: .ASCIIZ / TSSR ERROR CODE REC'D = /
2290 011630      056      056      056  SIMSG: .ASCIIZ /... AFTER DOING SOFT INIT/
2291 011663      124      105      123  TINERR: .ASCIIZ /TEST: .../
2292                                .EVEN

```

```

2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310 011676      BGNMSG  SFMSG
      011676      SFMSG::
2311 011676 004737 005264      JSR    PC,PRITSSP      ;PRINT CONTENTS OF TSSR REGISTER
2312 011702 004737 020170      JSR    PC,CKDROP      ;DROP UNIT, IF ALLOWED
2313 011706      ENDMMSG
      011706      L10003:
      011706 104423      TRAP    C$MSG
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326 011710      BGNMSG  PKTSSR
      011710      PKTSSR::
2327 011710 004737 005264      JSR    PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
2328 011714 012700 000004      MOV    #4,R0          ;NO. OF WORDS IN PACKET
2329 011720 004737 007076      JSR    PC,PRIPKT      ;PRINT THE CONTENTS OF COMMAND PACKET
2330 011724 013700 002716      MOV    MESBFA,R0      ;ADDRESS OF MESSAGE BUFFER
2331 011730 005001          CLR    R1                ;ASSUME NO HIGH MEMORY
2332 011732 004737 014072      JSR    PC,PRMESS      ;PRINT THE MESSAGE BUFFER ALSO
2333 011736      ENDMMSG
      011736      L10004:
      011736 104423      TRAP    C$MSG
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344

```

```

2345
2346 011740          BGNMSG  PKTGETS
      011740          PKTGETS:
2347 011740 004737 005264      JSR      PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
2348 011744 012700 000002      MOV      #2,R0          ;NO. OF WORDS IN GET STATUS PACKET
2349 011750 004737 007076      JSR      PC,PRIPKT      ;PRINT THE CONTENTS OF COMMAND PACKET
2350 011754          ENDMMSG
      011754          L10005:
      011754 104423          TRAP      CMSG
2351
2352
2353          ;
2354          ;PRINT TSSR ERRORS FOR INITIALIZATION TESTS
2355          ;
2356          ;INPUTS:
2357          ;
2358          ;      R1      TSSR CONTENTS
2359          ;      R4      ADDRESS OF COMMAND PACKET
2360          ;
2361          ;
2362 011756          BGNMSG  SFFMSG
      011756          SFFMSG:
2363 011756 004737 005264      JSR      PC,PRITSSR      ;PRINT CONTENTS OF TSSR REGISTER
2364 011762          ENDMMSG
      011762          L10006:
      011762 104423          TRAP      CMSG
2365
2366          .SBTTL  PKTMES  - PRINT TSSR AND MESSAGE BUFFER
2367          ;
2368          ;
2369          ;PRINT ROUTINE TO PRINT THE CONTENTS OF TSSR AND MESSAGE
2370          ;BUFFER FOR ERROR REPORTS
2371          ;
2372          ;INPUTS:
2373          ;
2374          ;      R1      CONTENTS OF TSSR
2375          ;      R2      LOW ORDER MESSAGE BUFFER
2376          ;      R3      HIGH ORDER MESSAGE BUFFER ADDRESS
2377          ;      NOTE: R3 IS IGNORED IF KTENABLE FLAG IS CLEAR
2378          ;
2379          ;
2380 011764          BGNMSG  PKTMES
      011764          PKTMES:
2381 011764 004737 005264      JSR      PC,PRITSSR      ;PRINT CONTENTS OF TSSR
2382 011770 010200          MOV      R2,R0          ;LOW ORDER ADDRESS
2383 011772 010301          MOV      R3,R1          ;HIGH ORDER ADDRESS
2384 011774 004737 014072      JSR      PC,PRMESS      ;PRINT THE MESSAGE BUFFER
2385 012000          ENDMMSG
      012000          L10007:
      012000 104423          TRAP      CMSG
2386

```

```

2388 .SBTTL ADSSR - PRINT TEST ADDRESS AND TSSR
2389 ;
2390 ;PRINT ROUTINE TO PRINT THE CONTENTS OF
2391 ;TSSR AND A MEMORY TEST ADDRESS
2392 ;
2393 ;INPUTS:
2394 ;
2395 ; R5 FIRST DEVICE UNIBUS ADDRESS
2396 ; ERRHI HIGH ORDER MEMORY TEST ADDRESS
2397 ; ERRLO LOW ORDER MEMORY TEST ADDRESS
2398 ;
2399 ;
2400 012002 BGNMSG ADSSR
012002 ADSSR::
2401 012002 004737 010032 JSR PC,PRITADD ;PRINT MEMORY TEST ADDRESS
2402 012006 016501 000000 MOV TSSR(R5),R1 ;GET CURRENT TSSR
2403 012012 004737 005264 JSR PC,PRITSSR ;PRINT THE CONTENTS OF TSSR REGISTER
2404 012016 ENDMMSG
012016 L10010:
012016 104423 TRAP C#MSG

2405 .SBTTL MSGEXP - PRINT WRITE CHAR. EXPD-RCV MESSAGE BUFFERS
2406 ;
2407 ;PRINT ROUTINE TO PRINT WRITE CHARACTERISTIC MESSAGE BUFFER
2408 ;
2409 ;IMPLICIT INPUTS:
2410 ;
2411 ; EXPMSG - EXPECTED MESSAGE BUFFER
2412 ; RECMMSG - RECEIVED MESSAGE BUFFER
2413 ; RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2414 ; RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2415 ;
2416 ;
2417 ;
2418 ;
2419 012020 BGNMSG MSGEXP
012020 MSGEXP::
2420 012020 012700 000007 MOV #7,R0 ;ASSUME NO EXT FEATURES
2421 012024 004737 015436 JSR PC,PRMSGEXP ;PRINT EXPD/RCV MESSAGE BUFFERS
2422 012030 ENDMMSG
012030 L10011:
012030 104423 TRAP C#MSG

2423
2424

```

```

2426                                     .SBTTL FIFEXP - PRINT FIFO EXP/RECV DATA
2427                                     ;*
2428                                     ;
2429                                     ;PRINT ROUTINE TO PRINT FIFO EXP/RECV DATA
2430                                     ;
2431                                     ;      R1      - BYTE COUNT
2432                                     ;
2433                                     ;IMPLICIT INPUTS:
2434                                     ;
2435                                     ;      EXPMSG  - EXPECTED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY)
2436                                     ;      RECHMSG  - RECEIVED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY)
2437                                     ;
2438 012032      BGNMSG FIFEXP
2439 012032      FIFEXP::
2439 012032      PRINTX #FIF1MSG,R1      ;PRINT BYTES TRANSFERRED.
2439 012032      MOV      R1,-(SP)
2439 012034      MOV      #FIF1MSG,-(SP)
2439 012040      MOV      #2,-(SP)
2439 012044      MOV      SP,R0
2439 012046      TRAP     C:PNTX
2439 012050      ADD      #6,SP
2440 012054      PRINTX #FIF2MSG      ;PRINT HEADER MSG
2440 012054      MOV      #FIF2MSG,-(SP)
2440 012060      MOV      #1,-(SP)
2440 012064      MOV      SP,R0
2440 012066      TRAP     C:PNTX
2440 012070      ADD      #4,SP
2441 012074      MOV      R1,R0      ;GET BYTE COUNT
2442 012076      JSR      PC,PRBYTEXP ;PRINT FIFO BYTES IN ERROR
2443 012102      ENDMSG
2443 012102      L10012:
2443 012102      TRAP     C:MSG
2444 012104      045      116      045 FIF1MSG: .ASCIZ '###A NUMBER OF BYTES TRANSFERRED * #02'
2445 012153      045      116      045 FIF2MSG: .ASCIZ '###A FIFO DATA BYTES IN ERROR:'
2446                                     .EVEN
2447

```

```

2449          .SBTTL MSGSTAT - PRINT STATUS HEADER AND MESSAGE BUFFERS
2450          ;*
2451          ;
2452          ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV
2453          ;
2454          ;
2455          ;IMPLICIT INPUTS:
2456          ;
2457          ;      EXPMSG - EXPECTED MESSAGE BUFFER
2458          ;      RECMMSG - RECEIVED MESSAGE BUFFER
2459          ;      RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2460          ;      RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2461          ;
2462          ;-
2462          BGNMSG MSGSTAT
2463          MSGSTAT:
2463          012212 012701 012254      MOV      #STATCOD,R1      ;ASCII ADDRESS TABLE
2464          012216 012100      10$: MOV      (R1)+,R0      ;DONE ALL MSG LINES?
2465          012220 001410      BEQ      20$      ;BR IF YES
2466          012222      PRINTX R0      ;PRINT STATUS BIT NAMES
2466          012222 010046      MOV      R0,-(SP)
2466          012224 012746 000001      MOV      #1,-(SP)
2466          012230 010600      MOV      SP,R0
2466          012232 104415      TRAP     C$PNTX
2466          012234 062706 000004      ADD      #4,SP
2467          012240 000766      BR       10$      ;DO ANOTHER MSG LINE
2468          012242 012700 000012      20$: MOV      #10,R0      ;NUMBER OF WORDS IN A READ STATUS BUFFER
2469          012246 004737 015436      JSR      PC,PRMSGEXP      ;PRINT EXPD/RECV MESSAGE BUFFERS
2470          012252      ENDMSG
2470          012252      L10013:
2470          012252 104423      TRAP     C$MSG
2471
2472          012254 012272 012334 012425 STATCOD: .WORD 1$,2$,3$,4$,5$,6$,0
2473          012272 045 116 045 1$: .ASCIZ 'ANSA Tape Bus Signals in Word #8:'
2474
2475
2476          012334 045 116 045 2$: .ASCIZ 'ANSA PARERR<15> IEOT <12> IFMK <9> IRDY<6> IRWD<2>'
2477          012425 045 116 045 3$: .ASCIZ 'ANSA IRESV2<14> IIDENT<11> IHER <8> IONL<5> IFBY<1>'
2478          012516 045 116 045 4$: .ASCIZ 'ANSA IRESV1<13> ICER <10> ISPEED<7> ILDP<4> IFPT<0>'
2479          012607 045 116 045 5$: .ASCIZ 'ANSA Tape Bus Signals in Word #9:'
2480          012651 045 116 045 6$: .ASCIZ 'ANSA DATMIS<7> ILW<6> OUTRDY<5> INRDY<4>'
2481          .EVEN
2482
2483
2484
2485          .SBTTL MSGLOOP - PRINT LOOPBACK HEADER AND MESSAGE BUFFERS
2486          ;*
2487          ;
2488          ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV
2489          ;
2490          ;
2491          ;IMPLICIT INPUTS:
2492          ;
2493          ;      EXPMSG - EXPECTED MESSAGE BUFFER
2494          ;      RECMMSG - RECEIVED MESSAGE BUFFER
2495          ;      RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2496          ;      RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2497          ;
2497          ;-
2497          BGNMSG MSGLOOP

```

```

012726 MSGLOOP:
2498 012726 012701 012770      MOV    #LOOPCOD,R1      ;ASCII ADDRESS TABLE
2499 012732 012100      10$:  MOV    (R1)+,R0      ;DONE ALL MSG LINES?
2500 012734 001410      BEQ     20$      ;BR IF YES
2501 012736      PRINTX  R0      ;PRINT STATUS BIT NAMES
      MOV    R0,-(SP)
      MOV    #1,-(SP)
      MOV    SP,R0
      TRAP   C:PNTX
      ADD    #4,SP
2502 012754 000766      BR      10$      ;DO ANOTHER MSG LINE
2503 012756 012700 000012      20$:  MOV    #10,R0      ;NUMBER OF WORDS IN A READ STATUS BUFFER
2504 012762 004737 015436      JSR     PC,PRMSGEXP      ;PRINT EXPD/RECV MESSAGE BUFFERS
2505 012766      ENDMMSG
      L10014:
      TRAP   C:MSG
2506
2507 012770 013010 013063 013162 LOOPCOD: .WORD 1$,2$,3$,4$,5$,6$,7$,0
2508 013010 045 116 045 1$: .ASCIZ 'ENSA Tape Bus Loopback Signals in Word #8:'
2509 013063 045 116 045 2$: .ASCIZ 'ENSA PARERR<15> IRESV2<14> IRESV1<13>'
2510 013162 045 116 045 3$: .ASCIZ 'ENSA IHISP=>IEOT<12> IWRT=>IDENT<11> IREV =>ICER <10>'
2511 013261 045 116 045 4$: .ASCIZ 'ENSA IWM =>IFMK<09> IEDIT=>IHER <08> IFAD =>ISPEED<07>'
2512 013360 045 116 045 5$: .ASCIZ 'ENSA ITADO=>IRDY<06> ITAD1=>IONL <05> IERASE=>ILDP <04>'
2513 013457 045 116 045 6$: .ASCIZ 'ENSA IREW =>IDBY<03> IRWU =>IRWD <02> IFEN =>IFBY <01>'
2514 013556 045 116 045 7$: .ASCIZ 'ENSA IGO =>IFPT<00>'
2515      .EVEN
2516

```

```

2518 .SBTTL MSGSUB PRINT WRITE SUBSYSTEM MESSAGE BUFFER
2519 ;*
2520 ;
2521 ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RCV
2522 ;
2523 ;
2524 ;IMPLICIT INPUTS:
2525 ;
2526 ; EXPMSG - EXPECTED MESSAGE BUFFER
2527 ; RECMMSG - RECEIVED MESSAGE BUFFER
2528 ; RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2529 ; RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2530 ;
2531 ;-
2531 013604 BGNMSG MSGSUB
2532 013604 MSGSUB::
2532 013604 012700 000012 MOV #10.,R0 ;SIZE OF WRITE SUBSYSTEM BUFFER
2533 013610 004737 015436 JSR PC,PRMSGEXP ;PRINT EXPD/RCV MESSAGE BUFFERS
2534 013614 ENDMMSG
2534 013614 L10015:
2534 013614 104423 TRAP C$MSG
2535
2536
2537
2538
2539
2540 .SBTTL MEMADD - PRINT MEMORY ADDRESS DATA ERROR
2541 ;*
2542 ;
2543 ;PRINT ROUTINE TO PRINT MEMORY ADDRESS DATA COMPARE ERROR
2544 ;
2545 ;IMPLICIT INPUTS:
2546 ;
2547 ; ERRHI - MEMORY ERROR HIGH ORDER ADDRESS
2548 ; ERRLO - MEMORY ERROR LOW ORDER ADDRESS
2549 ; EXP - EXPECTED DATA
2550 ; RECV - RECEIVED DATA
2551 ;
2552 ;-
2552 013616 BGNMSG MEMADD
2553 013616 MEMADD::
2553 013616 004737 007716 JSR PC,PRIADD ;PRINT MEMORY ADDRESS IN ERROR
2554 013622 013701 002176 MOV EXPD,R1 ;GET EXPD DATA
2555 013626 013702 002200 MOV RECV,R2 ;GET RECEIVED DATA
2556 013632 004737 007500 JSR PC,PRIXOR ;PRINT EXPD/RCV
2557 013636 ENDMMSG
2557 013636 L10016:
2557 013636 104423 TRAP C$MSG
2558

```

```

2560 .SBTTL PRAMPKT - PRINT RAM AND PACKET DATA
2561 ;
2562 ;
2563 ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
2564 ;WHEN THE RAM DATA DOES NOT MATCH.
2565 ;
2566 ;INPUTS:
2567 ;
2568 ; R4 POINTER TO COMMAND PACKET
2569 ;
2570 ;IMPLICIT INPUTS:
2571 ;
2572 ; RAMDATA DATA AS READ FROM THE RAM
2573 ; RAMSIZ NUMBER OF BYTES IN PACKET
2574 ; IF RAMSIZ=0 THEN DEFAULT TO 8.
2575 ;
2576 ;IMPLICIT OUTPUTS:
2577 ;
2578 ; RAMSIZ SET TO 0
2579 ;
2580 PRAMPKT:
2581 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2582 MOV #RAMDATA,R1 ;DATA FROM THE RAM
2583 CLR R2 ;INIT BYTE NUMBER
2584 5$: CMPB (R1),.(R4). ;COMPARE EXPECTED, RECEIVED
2585 BNE 7$ ;BR IF NO MATCH
2586 7$: MOVB -1(R1),R5 ;GET RECV RAM DATA
2587 MOVB -1(R4),R3 ;GET EXPD PACKET DATA
2588 XOR R5,R3 ;XOR EXPD/RECV
2589 BIC #177400,R3 ;LOW BYTE ONLY
2590 MOVB -1(R1),RECV ;GET RECEIVED RAM DATA
2591 MOVB -1(R4),EXPD ;GET EXPECTED RAM DATA
2592 PRINTB #RAMASC,R2,RECV,EXPD,R3
2593 MOV R3,-(SP)
2594 MOV EXPD,-(SP)
2595 MOV RECV,-(SP)
2596 MOV R2,-(SP)
2597 MOV #RAMASC,-(SP)
2598 MOV #5,-(SP)
2599 MOV SP,R0
2600 TRAP C#PNTB
2601 ADD #14,SP
2602 10$: INC R2 ;UPDATE BYTE COUNT
2603 TST RAMSIZ ;DEFAULT TO 8.?
2604 BEQ 15$ ;BR IF YES
2605 CMP R2,RAMSIZ ;DONE ALL BYTES?
2606 BLE 5$ ;BR IF NO
2607 BR 25$ ;
2608 15$: CMP R2,#8. ;DONE DEFAULT NUMBER OF BYTES?
2609 BLT 5$ ;BR IF NO
2610 25$: CLR RAMSIZ ;SET DEFAULT RAMSIZ
2611 RTS PC ;RETURN
2612 045 RAMASC: .ASCIZ 'N#A BYTE: #D2#A RAM: #03#A Packet: #03#A XOR:#03
2613 .EVEN

```

```

2607          .SBTTL PRMESS - PRINT CONTENTS OF MESSAGE BUFFER
2608          ;*
2609          ;
2610          ; THIS ROUTINE PRINTS THE CONTENTS OF
2611          ; THE 7 WORD MESSAGE BUFFER RETURNED BY THE
2612          ; TK-25.
2613          ;
2614          ; INPUT:
2615          ;
2616          ;     R0      LOW ORDER ADDRESS OF MESSAGE BUFFER
2617          ;     R1      HIGH ORDER ADDRESS OF MESSAGE BUFFER
2618          ;     NOTE: R1 IS IGNORED IF KTENABLE FLAG IS CLEAR
2619          ;
2620          ; THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
2621          ;
2622          ; -
2623
2624 014072 PRMESS:
2625 014072 SAVR: G          ;SAVE THE REGISTERS
2626 014076 010537 011032 MOV      R5,RAMR5H      ;SAVE DEVICE REGISTER POINTER
2627 014102 010005      MOV      R0,R5          ;SAVE LOW ORDER ADDRESS
2628 014104 005737 003102 TST      KTENABLE      ;ADDRESS ABOVE 28K?
2629 014110 001001      BNE      10$          ;BR IF YES
2630 014112 005001      CLR      R1          ;SET HIGH ORDER ADDRESS TO 0
2631 014114 010103 10$: MOV      R1,R3          ;SAVE HIGH ORDER ADDRESS
2632 014116 006100      ROL      R0          ;SHIFT BIT15 TO C BIT
2633 014120 006101      ROL      R1          ;SHIFT TO HIGH ORDER FOR PRINTOUT
2634 014122      PRINTX  @PROASC,R1,R5      ;PRINT MESSAGE BUFFER ADDRESS
2635 014122 010546      MOV      R5,-(SP)
2636 014124 010146      MOV      R1,-(SP)
2637 014126 012746 014730 MOV      @PROASC,-(SP)
2638 014132 012746 000003 MOV      @3,-(SP)
2639 014136 010600      MOV      SP,R0
2640 014140 104415      TRAP     C$PNTX
2641 014142 062706 000010 ADD      @10,SP
2642 014146 022715 177777 CMP      @177777,(R5)      ;MESSAGE BUFFER FULL OF ONES
2643 014152 001010      BNE      15$          ;BR IF BUFFER IS PROBABLY OKAY
2644 014154      PRINTX  @MESBFN          ;"MESSAGE BUFFER PROBABLY NOT VALID"
2645 014154 012746 014650 MOV      @MESBFN,-(SP)
2646 014160 012746 000001 MOV      @1,-(SP)
2647 014164 010600      MOV      SP,R0
2648 014166 104415      TRAP     C$PNTX
2649 014170 062706 000004 ADD      @4,SP
2650 014174      PRINTX  @PR1ASC          ;PRINT HEADER FOR CONTENTS
2651 014174 012746 014775 MOV      @PR1ASC,-(SP)
2652 014200 012746 000001 MOV      @1,-(SP)
2653 014204 010600      MOV      SP,R0
2654 014206 104415      TRAP     C$PNTX
2655 014210 062706 000004 ADD      @4,SP
2656 014214 005004      CLR      R4          ;NUMBER OF THE NEXT WORD
2657 014216 010501      MOV      R5,R1          ;COPY LOW ORDER ADDRESS
2658 014220 010300      MOV      R3,R0          ;COPY HIGH ORDER ADDRESS
2659 014222 001403      BEQ      20$          ;BR IF NOT ABOVE 28K
2660 014224 004737 020304 JSR      PC,SETMAP      ;SETUP PAR ADDRESS IN R0
2661 014230 010005      MOV      R0,R5          ;GET PAR FORMAT ADDRESS ABOVE 28K
2662 014232 20$: PRINTX  @MESHEA,(R5).      ;PRINT "MESSAGE BUFFER HEADER"
2663 014232

```

	014232	012546		MOV	(R5), -(SP)	
	014234	012746	015033	MOV	#MESHEA, -(SP)	
	014240	012746	000002	MOV	#2, -(SP)	
	014244	010600		MOV	SP, R0	
	014246	104415		TRAP	C:PNTX	
	014250	062706	0000	ADD	#6, SP	
2647	014254			PRINTX	#DATAFL, (R5),	;PRINT "DATA FIELD LENGTH"
	014254	012546		MOV	(R5), -(SP)	
	014256	012746	015100	MOV	#DATAFL, -(SP)	
	014262	012746	000002	MOV	#2, -(SP)	
	014266	010600		MOV	SP, R0	
	014270	104415		TRAP	C:PNTX	
	014272	062706	000006	ADD	#6, SP	
2648	014276			PRINTX	#RBPORA, (R5),	;PRINT "RESIDUAL BYTE COUNTER"
	014276	012546		MOV	(R5), -(SP)	
	014300	012746	015145	MOV	#RBPORA, -(SP)	
	014304	012746	000002	MOV	#2, -(SP)	
	014310	010600		MOV	SP, R0	
	014312	104415		TRAP	C:PNTX	
	014314	062706	000006	ADD	#6, SP	
2649	014320			PRINTX	#XS0CON, (R5),	;PRINT "XSTAT0 CONTENTS"
	014320	012546		MOV	(R5), -(SP)	
	014322	012746	015212	MOV	#XS0CON, -(SP)	
	014326	012746	000002	MOV	#2, -(SP)	
	014332	010600		MOV	SP, R0	
	014334	104415		TRAP	C:PNTX	
	014336	062706	000006	ADD	#6, SP	
2650	014342			PRINTX	#XS1CON, (R5),	;PRINT "XSTAT1 CONTENTS"
	014342	012546		MOV	(R5), -(SP)	
	014344	012746	015257	MOV	#XS1CON, -(SP)	
	014350	012746	000002	MOV	#2, -(SP)	
	014354	010600		MOV	SP, R0	
	014356	104415		TRAP	C:PNTX	
	014360	062706	000006	ADD	#6, SP	
2651	014364			PRINTX	#XS2CON, (R5),	;PRINT "XSTAT2 CONTENTS"
	014364	012546		MOV	(R5), -(SP)	
	014366	012746	015324	MOV	#XS2CON, -(SP)	
	014372	012746	000002	MOV	#2, -(SP)	
	014376	010600		MOV	SP, R0	
	014400	104415		TRAP	C:PNTX	
	014402	062706	000006	ADD	#6, SP	
2652	014406			PRINTX	#XS3CON, (R5),	;PRINT "XSTAT3 CONTENTS"
	014406	012546		MOV	(R5), -(SP)	
	014410	012746	015371	MOV	#XS3CON, -(SP)	
	014414	012746	000002	MOV	#2, -(SP)	
	014420	010600		MOV	SP, R0	
	014422	104415		TRAP	C:PNTX	
	014424	062706	000006	ADD	#6, SP	
2653	014430	022737	000001	CMP	#1, TRANSTST	;CHECK FOR RAM DUMP REQUIRED
2654	014436	001402		BEQ	40\$	;BR, IF DUMP REQUIRED
2655	014440	000137	014550	JMP	50\$	;NO DUMP
2656	014444			PRINTX	#RAMFHR	
	014444	012746	014552	MOV	#RAMFHR, -(SP)	
	014450	012746	000001	MOV	#1, (SP)	
	014454	010600		MOV	SP, R0	
	014456	104415		TRAP	C:PNTX	
	014460	062706	000004	ADD	#4, SP	

002134

40\$:

```

2657 014464 012737 000010 002246      MOV      #8.,RAMSIZ      ;RAM FIELD IS 8 BYTES LONG
2658 014472 012737 000020 011030      MOV      #20,RAMHLD     ;FIELD STARTS AT 20 OCTAL (10 HEX)
2659 014500 004737 010646              JSR      PC,RAMER       ;READ AND PRINT THEM
2660 014504 012737 000040 011030      MOV      #40,RAMHLD     ;FIELD STARTS AT 40 OCTAL (20 HEX)
2661 014512 004737 010646              JSR      PC,RAMER       ;READ AND PRINT THEM
2662 014516 012737 000060 011030      MOV      #60,RAMHLD     ;FIELD STARTS AT 60 OCTAL (30 HEX)
2663 014524 004737 010646              JSR      PC,RAMER       ;READ AND PRINT THEM
2664 014530 012737 000020 002246      MOV      #16.,RAMSIZ     ;RAM FIELD IS SIXTEEN BYTES LONG
2665 014536 012737 000100 011030      MOV      #100,RAMHLD     ;FIELD STARTS AT 100 OCTAL (40 HEX)
2666 014544 004737 010646              JSR      PC,RAMER       ;READ AND PRINT THEM
2667 014550 000207                    S01:      RTS          PC      ;RETURN
2668 014552      045      116      045  RAMFHR: .ASCIZ 'SNA ***** SPECIAL CONTROLLER RAM MEMORY DUMP *****'
2669 014650      045      116      045  MESBFN: .ASCIZ 'SNA MESSAGE BUFFER CONTENTS PROBABLY NOT VALID'
2670 014730      045      116      045  PROASC: .ASCIZ 'SNA Message Buffer Address = #01#05'
2671 014775      045      116      045  PRIASC: .ASCIZ 'SNA Message Buffer Contents:'
2672
2673 015033      045      116      045  MESHEA: .ASCIZ 'SNA Message Buffer Header      = #06'
2674 015100      045      116      045  DATAFL: .ASCIZ 'SNA Data Field Length      = #06'
2675 015145      045      116      045  RBPCRA: .ASCIZ 'SNA Residual Byte Counter    = #06'
2676 015212      045      116      045  XSOCON: .ASCIZ 'SNA XSTAT0 Contents        = #06'
2677 015257      045      116      045  XS1CON: .ASCIZ 'SNA XSTAT1 Contents        = #06'
2678 015324      045      116      045  XS2CON: .ASCIZ 'SNA XSTAT2 Contents        = #06'
2679 015371      045      116      045  XS3CON: .ASCIZ 'SNA XSTAT3 Contents        = #06'
2680                                .EVEN

```

```

2682          .SBTTL PRMSGEXP PRINT EXPD/RCV MESSAGE BUFFERS
2683      ;*[B
2684      ;
2685      ;ROUTINE TO PRINT EXPECTED AND RECEIVED MESSAGE BUFFERS
2686      ;
2687      ;      RO      - NUMBER OF WORDS IN BUFFER
2688      ;
2689      ;IMPLICIT INPUTS:
2690      ;
2691      ;      EXPMSG - EXPECTED MESSAGE BUFFER
2692      ;      RECMSG - RECEIVED MESSAGE BUFFER
2693      ;      RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2694      ;      RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2695      ;
2696      PRMSGEXP::
2697          SAVREG                                ;SAVE R1-R5 UNTIL NEXT RETURN
2698          MOV      RO,R5                        ;SAVE NUMBER OF WORDS
2699          MOV      RCVLOADD,RO                  ;GET RECV LOW ADDRESS
2700          MOV      RO,R4                        ;COPY LOW ADDRESS
2701          MOV      RCVHIADD,R1                 ;GET RECV HIGH ADDRESS
2702          ROL      RO                          ;SHIFT BIT15 TO C BIT
2703          ROL      R1                          ;SHIFT TO HIGH ORDER FOR PRINTOUT
2704          PRINTX  #PRMSG0,R1,R4                ;PRINT MESSAGE BUFFER ADDRESS
2705          MOV      R4,-(SP)
2706          MOV      R1,-(SP)
2707          MOV      #PRMSG0,-(SP)
2708          MOV      #3,-(SP)
2709          MOV      SP,RO
2710          TRAP     C#PNTX
2711          ADD      #10,SP
2712          PRINTX  #PRMSG1,-(SP)                ;PRINT HEADER FOR CONTENTS
2713          MOV      #PRMSG1,-(SP)
2714          MOV      #1,-(SP)
2715          MOV      SP,RO
2716          TRAP     C#PNTX
2717          ADD      #4,SP
2718          CLR      R4                          ;NUMBER OF THE CURRENT WORD
2719          MOV      #EXPMSG,R1                   ;GET EXPD BUFFER ADDRESS
2720          MOV      #RECMSG,R2                   ;GET RECV BUFFER ADDRESS
2721          MOV      (R1),R0                      ;GET EXPD
2722          MOV      (R2),R3                      ;GET RECV
2723          XOR      R0,R3                       ;XOR EXPD/RCV
2724          PRINTX  #PRMSG2,R4,(R1)+,(R2)+,R3
2725          MOV      R3,-(SP)
2726          MOV      (R2)+,-(SP)
2727          MOV      (R1)+,-(SP)
2728          MOV      R4,-(SP)
2729          MOV      #PRMSG2,-(SP)
2730          MOV      #5,-(SP)
2731          MOV      SP,RO
2732          TRAP     C#PNTX
2733          ADD      #14,SP
2734          INC      R4                          ;NUMBER OF THE NEXT
2735          CMP      R4,R5                       ;DONE ALL YET?
2736          BGE      50$                         ;BR IF YES
2737          BR       20$                         ;DO ANOTHER
2738          RTS      PC                         ;RETURN
2739
2740      20$:
2741
2742      50$:

```

M6

CZTKGA TK 25 FRI END FUNC #3 MACRO M1200 20-APR-84 08:13 PAGE 56-1
PRMSGEXP PRINT EXPD/RECV MESSAGE BUFFERS

SEQ 77

2718
2719 015616 045 116 045 PRMSG0: .ASCIZ 'N#A Message Buffer Address = #01#05'
2720 015663 045 116 045 PRMSG1: .ASCIZ 'N#A Message Buffer Contents:'
2721 015721 045 116 045 PRMSG2: .ASCIZ 'N#A WORD #D2#A EXPD: #06#A RECV: #06#A XOR: #06'
2722 .EVEN
2723

```

2725 .SBTTL PRBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER
2726
2727
2728 ;ROUTINE TO PRINT ERROR BYTES IN MESSAGE BUFFERS
2729 ; ONLY THE FIRST 8 ERRORS ENCOUNTERED ARE PRINTED DUE TO SCREEN SPACE
2730
2731 ; R0 - NUMBER OF BYTES IN BUFFER
2732
2733 ;IMPLICIT INPUTS:
2734
2735 ; EXPMSG - EXPECTED MESSAGE BUFFER
2736 ; RECMMSG - RECEIVED MESSAGE BUFFER
2737
2738 PRBYTEXP::
2739 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2740 MOV R0,R5 ;SAVE NUMBER OF BYTES
2741 CLR PRMNO ;INIT ERROR COUNT
2742 CLR R4 ;NUMBER OF THE CURRENT BYTE
2743 MOV #EXPMSG,R1 ;GET EXPD BUFFER ADDRESS
2744 MOV #RECMMSG,R2 ;GET RECV BUFFER ADDRESS
2745 20$: MOV (R1),R0 ;GET EXPD BYTE
2746 BIC #C<377>,R0 ;CLEAR UPPER BYTE
2747 MOV R0,PRBEXP ;SAVE FOR ERROR REPORT
2748 MOV (R2),R3 ;GET RECV BYTE
2749 BIC #C<377>,R3 ;CLEAR UPPER BYTE
2750 MOV R3,PRBREC ;FOR ERROR REPORT
2751 XOR R0,R3 ;XOR EXPD/RECV
2752 CMPB (R1)+,(R2)+ ;EXPD = RECV?
2753 BEQ 30$ ;BR IF YES
2754 INC PRMNO ;UPDATE ERROR COUNT
2755 000010 CMP PRMNO,#8. ;PRINTED 8?
2756 BHI 30$ ;BR IF YES
2757 27$: PRINTX #PRBMSG,R4,PRBEXP,PRBREC,R3
2758 MOV R3,-(SP)
2759 MOV PRBREC,-(SP)
2760 MOV PRBEXP,-(SP)
2761 MOV R4,-(SP)
2762 MOV #PRBMSG,-(SP)
2763 MOV #5,-(SP)
2764 MOV SP,R0
2765 TRAP C$PNTX
2766 ADD #14,SP
2767 30$: FORCEEXIT 50$ ;880
2768 BR 35$ ;880
2769 30$: FORCERRR 27$,NOTSSR ;880
2770 35$:
2771 INC R4 ;NUMBER OF THE NEXT
2772 CMP R4,R5 ;DONE ALL YET?
2773 BGE 50$ ;BR IF YES
2774 BR 20$ ;DO ANOTHER
2775 50$: PRINTX #PRBTOT,PRMNO ;PRINT TOTAL ERROR COUNT
2776 MOV PRMNO,-(SP)
2777 MOV #PRBTOT,-(SP)
2778 MOV #2,-(SP)
2779 MOV SP,R0
2780 TRAP C$PNTX

```

	016214	062706	000006		ADD	06,5P	
2768	016220	000207			RTS	PC	,RETURN
2769							
2770	016222	045	116	045	PRBMSG: .ASCIZ	'#N#A	BYTE #D2#A EXPD: #03#A RECV: #03#A XOR: #03'
2771	016307	045	116	045	PRBTOT: .ASCIZ	'#N#A	NUMBER OF BYTES IN ERROR = #D2'
2772					.EVEN		
2773	016354	000000			PRBEXP: .WORD	0	,EXPD
2774	016356	000000			PRBREC: .WORD	0	,RECV
2775							

```
2777 .SBTTL EXPREC - PRINT EXPD/RECV WORD DATA
2778 ;
2779 ;
2780 ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
2781 ;
2782 ;INPUTS:
2783 ;
2784 ; R1 RECEIVED DATA
2785 ; R2 EXPECTED DATA
2786 ;
2787 ;
2788 ;
2789 016360 BGNMSG EXPREC
016360 EXPREC::
2790 016360 004737 007500 JSR PC,PRIOR ;PRINT THE DATA
2791 016364 ENDMMSG
016364 L10017:
016364 104423 TRAP CMSG
2792
2793
```

```

2795          .SBTTL  EXPBREC - PRINT EXPD/RECV BYTE DATA
2796          ;*
2797          ;
2798          ;PRINT ROUTINE TO DISPLAY BYTE EXPD/RECV DATA
2799          ;
2800          ;
2801          ;INPUTS:
2802          ;
2803          ;      R1      RECEIVED DATA BYTE
2804          ;      R2      EXPECTED DATA BYTE
2805          ;
2806          ;-
2807
2808          BGNMSG  EXPBREC
2809          EXPBREC:  JSR      PC,PRIBXOR      ;PRINT THE DATA
2810                   ENDMMSG
2811          L10020:  TRAP      C#MSG
2812
2813
2814
2815          .SBTTL  RAMERR - PRINT RAM AND PACKET DATA
2816          ;*
2817          ;
2818          ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
2819          ;
2820          ;INPUTS:
2821          ;
2822          ;      R4      POINTER TO COMMAND PACKET
2823          ;
2824          ;IMPLICIT INPUTS:
2825          ;
2826          ;      RAMDATA      DATA AS READ FROM THE RAM
2827          ;      RAMSIZ      NUMBER OF BYTES IN PACKET
2828          ;                  IF RAMSIZ=0 THEN DEFAULT TO 8.
2829          ;
2830          ;IMPLICIT OUTPUTS:
2831          ;
2832          ;      RAMSIZ  SET TO 0
2833          ;-
2834
2835          BGNMSG  RAMERR
2836          RAMERR:  JSR      PC,PRAMPKT      ;PRINT RAM/PACKET DATA
2837                   ENDMMSG
2838          L10021:  TRAP      C#MSG
2839
2840          .SBTTL  RAMTADD - PRINT TEST ADDRESS, RAM AND PACKET DATA
2841          ;*
2842          ;
2843          ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
2844          ;
2845          ;INPUTS:

```

```

016366
016366
2809 016366 004737 007350
2810 016372
016372 104423

```

```

016374
016374
2836 016374 004737 013640
2837 016400
016400
016400 104423

```

```

2846
2847      ;          RA      POINTER TO COMMAND PACKET
2848
2849      ;IMPLICIT INPUTS:
2850
2851      ;          RAMDATA    DATA AS READ FROM THE RAM
2852      ;          RAMSIZ     NUMBER OF BYTES IN PACKET
2853      ;                      IF RAMSIZ=0 THEN DEFAULT TO 8.
2854      ;          ERRHI      HIGH ORDER TEST ADDRESS
2855      ;          ERRLO      LOW ORDER TEST ADDRESS
2856
2857      ;IMPLICIT OUTPUTS:
2858
2859      ;          RAMSIZ     SET TO 0
2860
2861      ;-
2862      016402      BGNMSG    RAMTADD
2863      016402      RAMTADD:  ;
2864      016402      004737    010032      JSR      PC,PRITADD      ;PRINT TEST ADDRESS
2865      016406      004737    013640      JSR      PC,PRAMPKT     ;PRINT RAM/PACKET DATA
2866      016412      ENDMSG
2867      016412      L10022:
2868      016412      104423      TRAP      C#MSG
2869
2870      .SBTTL      RAMEXP    - PRINT RAM EXPD/RECV DATA
2871
2872      ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
2873
2874      ;IMPUTS:
2875
2876      ;          R1          RECEIVED DATA
2877      ;          R2          EXPECTED DATA
2878      ;          R4          CONTROLLER RAM ADDRESS
2879      ;-
2880      016414      BGNMSG    RAMEXP
2881      016414      RAMEXP:  ;
2882      016414      042701    177400      BIC      #C<377>,R1      ;SAVE EXPD RAM DATA BYTE
2883      016420      042702    177400      BIC      #C<377>,R2      ;SAVE EXPD RAM DATA BYTE
2884      016424      004737    007624      JSR      PC,PRIRAM      ;PRINT THE RAM ADDRESS
2885      016430      004737    007500      JSR      PC,PRIXOR      ;PRINT THE DATA
2886      016434      ENDMSG
2887      016434      L10023:
2888      016434      104423      TRAP      C#MSG
2889
2890      .SBTTL      TIMEXP    - PRINT TIMER A,B AND EXP/REC
2891
2892      ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
2893      ;AND TIMER A,B HEADER MESSAGE
2894
2895      ;IMPUTS:
2896
2897      ;          R1          RECEIVED DATA
2898      ;          R2          EXPECTED DATA

```

```

2897
2898
2899 016436          BGNMSG  TIMEXP
      016436          TIMEXP::
2900 016436          PRINTX  #TIMSG0      ;PRINT HEADER
      016436 012746 016464      MOV      #TIMSG0,-(SP)
      016442 012746 000001      MOV      #1,-(SP)
      016446 010600      MOV      SP,R0
      016450 104415      TRAP      C#PNTX
      016452 062706 000004      ADD      #4,SP
2901 016456 004737 007500      JSR      PC,PRIXOR      ;PRINT THE DATA
2902 016462          ENDMSG
      016462          L10024:
      016462 104423      TRAP      C#MSG
2903
2904
2905 016464      045      116      045  TIMSG0: .ASCIZ  'TIMER A STATUS IS IN BIT 3'
2906                                     .EVEN      'TIMER B STATUS IS IN BIT 2'

```

```

2908 .SBTTL BADSSR PRINT TSSR ERRORS ON DATA TRANSFERS
2909
2910 ;*
2911 ;
2912 ;PRINT ROUTINE FOR TSSR ERRORS ON DATA TRANSFERS
2913 ;
2914 ;INPUTS:
2915 ;
2916 ; R1 CONTENTS OF TSSR
2917 ; R2 DATA WRITTEN (8 BITS)
2918 ;
2919 ;-
2920
2921 BGNMSG BADSSR
2922 BADSSR::
2923     MOV R2,-(SP) ;SAVE DATA TRANSFERRED
2924     BIC #177400,R2 ;GET JUST ONE BYTE
2925     PRINTB #XFERASC,R2
2926     MOV R2,-(SP)
2927     MOV #XFERASC,-(SP)
2928     MOV #2,-(SP)
2929     MOV SP,R0
2930     TRAP C#PNTB
2931     ADD #6,SP
2932     MOV (SP)+,R2 ;RESTORE R2
2933     JSR PC,PRITSSR ;DECODE TSSR CONTENTS
2934     ENDMMSG
2935
2936 L10025:
2937     TRAP C#MSG
2938     .ASCIIZ '###A Data Transferred = #03'
2939
2940 016564 010246
2941 016564 042702 177400
2942 016572 010246
2943 016574 012746 016624
2944 016600 012746 000002
2945 016604 010600
2946 016606 104414
2947 016610 062706 000006
2948 016614 012602
2949 016616 004737 005264
2950 016622
2951 016622 104423
2952 016624 045 116 045 XFERASC:

```

2931
2932
2933
2934
2935
2936
2937

.SBTTL GLOBAL SUBROUTINES SECTION

; **
; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
; THAT ARE USED IN MORE THAN ONE TEST.
;

```

2939          .SBTTL  SOFINIT - SOFT INITIALIZE OF CONTROLLER
2940
2941          ;*
2942          ;
2943          ;ROUTINE TO DO A SOFT INITIALIZE OF THE CONTROLLER
2944          ;BY WRITING INTO THE TSSR REGISTER. AFTER THE INIT,
2945          ;THE TSSR REGISTER IS TESTED FOR ERRORS. ANY ERRORS
2946          ;DETECTED SHOULD BE TREATED AS DEVICE FATAL ERRORS.
2947          ;
2948          ;INPUTS:
2949          ;
2950          ;      R5      ADDRESS OF FIRST REGISTER
2951          ;
2952          ;OUTPUTS:
2953          ;
2954          ;      R0      CONTENTS OF TSSR, IF ERROR
2955          ;      CARRY   SET IF INIT WAS OKAY
2956          ;              CLEAR IF FATAL ERROR
2957          ;
2958          ;CALLING SEQUENCE:
2959          ;
2960          ;      MOV      @ADDRESS,R5
2961          ;      JSR      PC,SOFINIT
2962          ;      BCS      CONTINUE
2963          ;      ERROF                    ;REPORT FATAL ERROR
2964          ;
2965          ;-
2966
2967 016660      SOFINIT::
2968 016660      SAVREG
2969 016664      012765 000000 000000      MOV      @0,TSSR(R5)      ; SAVE THE REGISTERS
2970 016672      004737 017134      JSR      PC,WAITF      ; DO THE INIT.
2971 016676      016500 000000      MOV      TSSR(R5),R0      ; WAIT FOR SSR
2972 016702      010004      MOV      R0,R4      ;GET THE TSSR REGISTER
2973 016704      042704 176277      BIC      @C<HIADDR:OFL>,R4 ;START SETUP OF EXPECTED TSSR
2974 016710      052704 002200      BIS      @SSR:NBA,R4      ;CLEAR OUT UNUSED BITS
2975 016714      020400      CMP      R4,R0      ;R4 HAS EXPECTED CONTENTS
2976 016716      001402      BEQ      5$      ;ONLY EXPECTED BITS SET ?
2977 016720      000241      CLC      ;BRANCH IF OKAY
2978 016722      000401      BR      10$      ;CLEAR THE CARRY FOR ERROR
2979 016724      000261      5$: SEC      ;GO TO EXIT
2980 016726      000207      10$: RTS      PC      ;SET THE CARRY BIT
                ;RETURN TO CALLER

```

```

2982          .SBTTL  CHKAMB  - CHECK TSSR FOR AMBIGUITY
2983
2984          ;
2985          ;
2986          ; THIS ROUTINE TESTS THE CONTENTS OF THE TSSR REGISTER
2987          ; FOR AMBIGUITY
2988          ;
2989          ; INPUT:
2990          ;
2991          ;      RO      CONTENTS OF TSSR
2992          ;
2993          ; OUTPUT:
2994          ;
2995          ;      RO      CONTENTS OF TSSR
2996          ;
2997          ;      CARRY   SET - NO AMBIGUITY
2998          ;              CLR - AMBIGUOUS CONTENTS
2999          ;
3000          ;
3001          ;
3002          CHKAMB:
3003          SAVREG          ;SAVE THE GENERAL REGISTERS
3004          MOV      RO,R4  ;CONTENTS OF TSSR
3005          BIT      #SC,RO ;IS BIT 15 SET ?
3006          BNE     5$     ;BRANCH IF YES
3007          BIT      #C<NBA!OFL!SSR!HIADDR>,RO ;ANY OTHER BITS SET ?
3008          BNE     40$    ;MUST BE AN ERROR
3009          BR      45$    ;RETURN WITH SUCCESS
3010          5$:  BIT      #SSR,RO ;IS READY BIT SET ?
3011          BNE     10$    ;BRANCH IF READY BIT IS SET.
3012          BIT      #BIT5,RO ;IS FATAL ERROR BIT SET ?
3013          BEQ     40$    ;ERROR IF NOT
3014          BIC      #CTERCLS,R4 ;CLEAR ALL BUT TERMINATION CODE
3015          CMP     R4,#16 ;ALL THREE BITS MUST BE SET
3016          BNE     40$    ;ERROR IF NOT SET
3017          BR      45$    ;OK IF ALL ARE SET
3018          10$: BIT      #BIT5,RO ;IS FATAL ERROR BIT SET ?
3019          BEQ     45$    ;ERROR IF BIT IS SET WITH SSR
3020          BIT      #BIT2!BIT1,RO ;IS THIS A FUNCTION REJECT
3021          BNE     45$    ;BR. IF TSSR IS OK
3022          40$:  CLC      ;AMBIGUOUS CONTENTS
3023          BR      50$
3024          45$:  SEC      ;SHOW SUCCESS NO AMBIGUITY
3025          50$:  RTS     PC ;RETURN TO CALLER
3026
3002 016730
3003 016730
3004 016734 010004
3005 016736 032700 100000
3006 016742 001004
3007 016744 032700 174077
3008 016750 001023
3009 016752 000424
3010 016754 032700 000200
3011 016760 001011
3012 016762 032700 000040
3013 016766 001414
3014 016770 042704 177761
3015 016774 020427 000016
3016 017000 001007
3017 017002 000410
3018 017004 032700 000040
3019 017010 001405
3020 017012 032700 000006
3021 017016 001002
3022 017020 000241
3023 017022 000401
3024 017024 000261
3025 017026 000207
3026

```

```

3028          .SBTTL ENAINIT,DSBINT - ENABLE/DISABLE INTERRUPTS
3029          ;
3030          ; DEFAULT DISPLAY INTERRUPT HANDLERS.
3031          ; IF DISPLAY TIME-OUT, REPORT DEV FATAL, AND ABORT PASS.
3032          ; OTHERWISE, SAVE DPU REGISTERS AND DISMISS.
3033          ;
3034          ;
3035          ; BIT DEFINITIONS FOR "INTMASK" AND "INTFLAG" BYTES:
3036          ;
3037          000200          IOKCKIN=BIT7          ; DON'T CHECK FOR BAD INTERRUPTS -- TEST WILL.
3038          000001          IOKSTP=BIT0          ; EXPECT "STOP" INTERRUPT.
3039          ;
3040          ; INTERRUPT MASK -- SAYS EXPECTING INTERRUPTS
3041          017030          000          INTMASK: .BYTE 0
3042          ; INTERRUPT FLAG -- SAYS WE GOT ONE (IF POSITIVE)
3043          017031          000          INTFLAG: .BYTE 0
3044          ;
3045          ; SAVED INTERRUPT VECTOR:
3046          017032          000000          INTVEC: .WORD 0
3047          ; SAVE CPU PC
3048          017034          000000          INTCPC: .WORD 0
3049          ;
3050          ; SUBROUTINE TO ENABLE INTERRUPTS:
3051          017036          010046          ENAINIT: MOV RO, -(SP)          ; SAVE RO
3052          017040          013700          002156          MOV IVEC, RO          ; GET POINTER TO VECTORS
3053          017044          012720          017102          MOV @INTR, (RO),          ; SET UP INTERRUPT VECTOR
3054          017050          012720          000340          MOV @PRI07, (RO),
3055          017054          012600          MOV (SP)+, RO          ; RESTORE RO
3056          017056          011646          MOV (SP), -(SP)
3057          017060          012766          000000          000002          MOV @0.2(SP)          ; SET CPU TO LEVEL 0
3058          017066          000002          RTI
3059          ;
3060          ; SUBROUTINE TO DISABLE INTERRUPTS (RAISE PRIORITY TO LEVEL 7)
3061          017070          011646          DSBINT: MOV (SP), (SP)
3062          017072          012766          000340          000002          MOV @PRI07, 2(SP)
3063          017100          000002          RTI
3064

```

```

3066          .SBTTL  INTR      - INTERRUPT HANDLERS
3067
3068 017102      BGNSRV  INTR          ;DEFINE INTERRUPT ENTRY
          017102      INTR::
3069 017102 012737 000001 002172      MOV    #1,INTRECV    ;SET FLAG TO SHOW INTERRUPT RECEIVED
3070 017110 105037 017031              CLR    INTFLAG      ;CLEAR FLAG TO SAY WE GOT INTERRUPT
3071 017114 132737 000001 017030      BIT    #IOKSTP,INTMASK ;EXPECTING STOP INTERRUPT?
3072 017122 001003              BNE     1$                ;BR IF YES
3073 017124 152737 000001 017031      BIS    #IOKSTP,INTFLAG ;NO. SET THE ERROR FLAG.
3074
3075          ;SAVE REGISTERS, MSG BUFFER, ETC.
3076 017132      1$:
3077 017132              ENDSRV
          017132      L10026:
          017132 000002              RTI
3078
3079

```

```

3081      .SBTTL WAITF - WAIT FOR SUBSYSTEM READY
3082      ;
3083      ; SUBROUTINE TO WAIT FOR THE SUBSYSTEM READY FLAG
3084      ;
3085      ; INPUTS:
3086      ;
3087      ; R5      ADDRESS OF FIRST DEVICE REGISTER
3088      ;
3089      ; OUTPUTS:
3090      ;
3091      ; R0      CONTENTS OF LAST TSSR READ
3092      ; CARRY   SET - READY BIT SET
3093      ;         CLR - TIMEOUT WAITING FOR READY
3094      ;
3095      WAITF:: BREAK      ; DO A SUPVSR BREAK FIRST.
3096      017134 104422      TRAP      C$BRK
3097      017136 012746 177776 MOV      @177776, -(SP) ;BIG MSEC TIMER
3098      017142 012727 000001 DELAY    1 ;DELAY 100US
3099      017142 012727 000001 MOV      @1,(PC)+
3100      017146 000000      .WORD    0
3101      017150 013727 002116 MOV      L$DLY,(PC)+
3102      017154 000000      .WORD    0
3103      017156 005367 177772 DEC      -6(PC)
3104      017162 001375      BNE      .-4
3105      017164 005367 177756 DEC      -22(PC)
3106      017170 001367      BNE      .-20
3107      017172 016500 000000 2$: MOV    TSSR(R5),R0 ;READ THE TSSR REGISTER
3108      017176 105700      TSTB     R0 ;TEST FOR READY BIT SET
3109      017200 100421      BMI      3$ ; EXIT ON STOP FLAG.
3110      017202 012727 000001 DELAY    1 ; WAIT 100 USEC
3111      017206 000000      MOV      @1,(PC)+
3112      017210 013727 002116      .WORD    0
3113      017214 000000      MOV      L$DLY,(PC)+
3114      017216 000000      .WORD    0
3115      017216 005367 177772 DEC      -6(PC)
3116      017222 001375      BNE      .-4
3117      017224 005367 177756 DEC      -22(PC)
3118      017230 001367      BNE      .-20
3119      017232 012732 104422 BREAK     ; DO A SUPVSR BREAK FIRST.
3120      017234 005316      TRAP      C$BRK
3121      017236 001355      DEC      (SP) ;REDUCE DELAY COUNT
3122      017240 000241      BNE      2$ ;RETRY UNTIL TIMER EXPIRES
3123      017242 000401      CLC      ; C = 0, CONTROLLER STILL RUNNING...
3124      017244 000261      BR      4$ ;...OR HUNG-UP AFTER 300 MSEC.
3125      017246 005326      3$: SEC      ; C = 1, CONTROLLER IS STOPPED.
3126      017250 000207      4$: DEC      (SP)+ ;RESTORE STACK WITHOUT CHANGING CARRY BIT
3127      RTS      PC

```

```

3112          .SBTTL  CHKTSSR - CHECK TSSR FOR READY
3113
3114          ;*
3115          ;
3116          ;THIS ROUTINE WAITS FOR READY IN THE TSSR
3117          ;AND TESTS FOR AMBIGUOUS BIT SETTINGS IN TSSR.
3118          ;
3119          ;INPUT:
3120          ;
3121          ;      R5      ADDRESS OF CSR REGISTERS
3122          ;
3123          ;OUTPUT:
3124          ;
3125          ;      R0      CONTENTS OF TSSR
3126          ;      CARRY   SET - OKAY
3127          ;              CLR - NOT READY AMBIGUOUS, OR SC SET
3128          ;
3129          ;-
3130
3131          CHKTSSR:
3132          JSR    PC, WAITF      ;WAIT FOR READY
3133          BCC    20$           ;BRANCH IF TIME OUT
3134          JSR    PC, CHKAMB     ;TSSR AMBIGUOUS?
3135          BCC    10$           ;BR IF YES
3136          BIT    #SC, R0       ;SPECIAL CONDITION SET?
3137          BEQ    15$           ;BR IF NO
3138          BIT    #<SCE!BIE!RMR!NXM>, R0 ;ANY ERROR BITS SET?
3139          BEQ    15$           ;BR IF NO
3140          10$:  CLC             ;SET FAILURE
3141          BR     20$           ;
3142          15$:  SEC             ;SET SUCCESS
3143          20$:  RTS    PC      ;RETURN TO CALLER

```

```

3145 .SBTTL XNXM - CHECK FOR NONEXISTENT MEMORY
3146
3147 ; ROUTINE TO TEST FOR A NEXM IN THE RANGE (R1) THRU (R2).
3148 ; ON RETURN, IF "C" = 1, (R1) = NEXM ADDRESS.
3149 ; "C" = 0, ALL ADDRESSES OK.
3150
3151 ; CALL: MOV ADR1,R1
3152 ; MOV ADR2,R2
3153 ; JSR PC,NXM
3154 ; RETURN
3155 ; TEST "C" AND PROCEED.
3156 017312 012737 017344 000004 XNXM: MOV #21,004 ; SET BUSERR VECTOR.
3157 017320 012737 000200 000006 MOV #PRI04,006
3158 017326 005003 CLR R3 ; FLAG.
3159 017330 005711 11: TST (R1) ; TEST THE ADDRESS(ES).
3160 ; IF ANY TRAP, CONTINUE AT 21.
3161 017332 020102 CMP R1,R2 ; OTHERWISE, CONTINUE HERE.
3162 017334 001407 BEQ 31 ; BR IF FINISHED (NO NEXM'S).
3163 017336 062701 000002 ADD #2,R1 ; SET NEXT ADDRESS...
3164 017342 000772 BR 11 ; ...AND CONTINUE.
3165
3166 017344 005103 21: COM R3 ; GOT ONE, SET FLAG...
3167 017346 012716 017354 MOV #31,(SP)
3168 017352 000002 RTI ; ...AND DISMISS INTERRUPT...
3169 017354 000004 31: CLAVEC #4 ; ...AND GIVE BACK THE VECTOR.
3170 017362 005703 MOV #4,R0
3171 017364 001401 TRAP C1CVEC
3172 017366 000261 TST R3 ; DID WE CATCH ONE ??
3173 017370 000207 BEQ .+4 ; NO, "C" = 0, SKIP NEXT.
3174 ; YES, "C" = 1, (R1) = NEXM ADDR.
3175
3176
3177
3178 .SBTTL TSTLOOP - CHECK ITERATION COUNT
3179
3180 ; SUBROUTINE TO EXECUTE TEST ITERATIONS.
3181 ; EXIT WITH "C" SET IF LOOPS ALLOWED AND LOOP COUNT NON-ZERO.
3182 ; LOOP COUNTER IS SET BY "BEGIN.TEST" MACRO.
3183
3184 ; CALL: LOOPTO ARG
3185
3186 017372 005737 002136 TSTLOOP: TST NOITS ; ITERATIONS INHIBITED?
3187 017376 001006 BNE 11 ; YES.
3188 017400 005737 002152 TST QVP ; NO.
3189 017404 100403 BMI 11 ; LOOPS DISALLOWED IN QUICK PASS.
3190 017406 005337 002164 DEC LOOPCNT ; BUMP LOOP COUNTER.
3191 017412 001002 BNE 21
3192 017414 000241 11: CLC ; LOOP DISALLOWED, OR DONE.
3193 017416 000401 BR 31
3194 017420 000261 21: SEC ; LOOP ENABLED.
3195 017422 000207 31: RTS PC

```

3198
3199
3200
3201
3202
3203
3204
3205
3206
3207
3208
3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3240
3241
3242
3243
3244

017424
017424 010046
017426 005037 003106
017432 005037 017672
017436 005037 005232
017442 105037 017030
017446 013700 002150
017452 006300
017454 005737 003062
017460 001430
017462 100010
017464 052760 160000 003130
017472
017472 104455
017474 000001
017476 003636
017500 005176
017502 000407
017504 052760 160001 003130 3:
017512
017512 104455
017514 000002
017516 004233
017520 000000
017522 012737 177777 003060 2:
017530
017530 013700 002150
017534 104451
017536

```
.SBTTL TSTSETUP - PRINT TEST NAME AND INIT ERROR COUNTS
;
; PRINT THE NUMBER AND NAME OF EACH TEST AS WE GO ALONG.
; INCREMENT "TESTK" TO INDICATE THE NUMBER OF TESTS
; IN THE CURRENT RUN SEQUENCE.
; CLEAR THE ERROR COUNTER AND SIGNATURE EXTENSION FLAGS.
;
; INPUT:
;
; R0      POINTER TO TEST ID ASCIZ STRING
;
; OUTPUT:
;
; R5      ADDRESS OF FIRST DEVICE REGISTER
;
; IMPLICIT OUTPUTS:
;
; TSTCNT  UPDATED TO COUNT TESTS PERFORMED SINCE START OR RESTART
;
; SIDE EFFECTS:
;
; INTERRUPT LEVEL IS RAISED TO LEVEL OF
; THE DEVICE UNDER TEST
;
; -
TSTSETUP::
MOV     R0, -(SP)      ; SAVE THE TEST ID MESSAGE
CLR     SIFLAG          ; CLEAR "SOFT INIT" FLAG
CLR     ERRK            ; CLEAR LOCAL ERROR COUNTER.
CLR     EXTA            ; CLEAR ERROR EXTENSION FLAG.
CLRB    INTMASK         ; CLEAR INTERRUPT MASK (CHECK ERROR)
MOV     UNITN, R0       ; GET THE UNIT NUMBER.
ASL     R0              ; ... AND MAKE IT A WORD OFFSET.
TST     NODEV           ; DID STARTUP FIND THE DEVICE?
BEQ     4$              ; BR IF YES
BPL     3$              ; BR IF NOT IDLE
BIS     @160000,ERTABL(R0) ; FLAG ERROR IN THE ERROR TABLE
ERRDF   1,NXR,NXRERR    ; NO DEVICE HERE -- PRINT IT
TRAP    C$ERRDF
        .WORD 1
        .WORD NXR
        .WORD NXRERR
BR      2$
BIS     @160001,ERTABL(R0) ; FLAG ERROR IN THE ERROR TABLE
ERRDF   2,NOINIT        ; DEVICE NOT IDLE
TRAP    C$ERRDF
        .WORD 2
        .WORD NOINIT
        .WORD 0
MOV     @-1,DUFLG       ; DROP THE UNIT
DODU    UNITN
MOV     UNITN, R0
TRAP    C$DODU
DOCLN                    ; ABORT THE PASS
```

	017536	104444			TRAP	CIDCLN	
3245	017540	000423			BR	5:	
3246							
3247	017542			4:	RFLAGS	R0	; GET THE OPERATOR FLAGS.
	017542	104421			TRAP	CIRFLA	
3248	017544	032700	001000		BIT	#PNT,R0	; PRINT THE TEST NUMBERS?
3249	017550	001412			BEQ	1:	; BR IF NO
3250	017552	011600			MOV	(SP),R0	;GET THE ID MESSAGE
3251	017554				PRINTF	#TNAM,R0	;DISPLAY THE TEST ID
	017554	010046			MOV	R0,-(SP)	
	017556	012746	017620		MOV	#TNAM,-(SP)	
	017562	012746	000002		MOV	#2,-(SP)	
	017566	010600			MOV	SP,R0	
	017570	104417			TRAP	CIPNTF	
	017572	062706	000006		ADD	#6,SP	
3252	017576	005237	002162	1:	INC	TSTCNT	; BUMP TEST COUNTER.
3253	017602				SETPRI	IPRI	;PRIORITY THAT OF DEVICE
	017602	013700	002160		MOV	IPRI,R0	
	017606	104441			TRAP	CISPRI	
3254	017610	005726		5:	TST	(SP),	;FIX UP THE STACK
3255	017612	013705	002154		MOV	CSRADDR,R5	; ADDRESS OF TSV REGISTERS ON UNIBUS
3256	017616	000207			RTS	PC	
3257	017620	045	123	045	TNAM:	.ASCIZ	'\$S\$T\$A Test'
3258						.EVEN	

```

3260          .SBTTL  TSTEND  - PRINT ERRORS RECEIVED
3261          ;
3262          ; AT END OF EACH TEST, PRINT THE NUMBER OF ERRORS RECEIVED
3263          ; IF NORMAL ERROR REPORTING IS DISABLED (FLA:IER).
3264          ;
3265          TSTEND: RFLAGS  RO
3266          TRAP      C#RFLA
3267          BIT        RO,#IER
3268          BEQ        1$      ; BR IF "IER" NOT SET.
3269          PRINTF     #ESUM,ERRK      ; PRINT ERROR COUNT.
3270          MOV        ERRK,-(SP)
3271          MOV        #ESUM,-(SP)
3272          MOV        #2,-(SP)
3273          MOV        SP,RO
3274          TRAP      C#PNTF
3275          ADD        #6,SP
3276          1$:      RTS        PC
3277
3278          ERRK:      0          ; LOCAL ERROR COUNT.
3279          ESUM:      .ASCIZ    /#A #D#A ERRORS/
3280          EMAXDU:    .ASCIZ    /ERROR LIMIT REACHED -- DROPPING UNIT/
3281          .EVEN
3282
3283          .SBTTL  INCERK  - INCREMENT LOCAL ERROR COUNT
3284          ;
3285          ; ROUTINES TO INCREMENT LOCAL ERROR COUNT AND CHECK FOR LIMIT:
3286          ;
3287          INCERK: INC        ERRK      ; INCREMENT LOCAL ERROR COUNT
3288          MOV        RO,-(SP)      ; SAVE RO
3289          MOV        UNITN,RO      ; GET UNIT NUMBER
3290          ASL        RO            ; ... AND MAKE IT A WORD OFFSET.
3291          ADD        #ERTABL,RO      ; RO GETS ADDRESS OF ERROR TABLE ENTRY.
3292          INC        (RO)           ; INCREMENT THE DEVICE ERROR COUNT
3293          BIT        #7777,(RO)      ; DID WE OVERFLOW THE FIELD?
3294          BNE        1$            ; BR IF NO.
3295          DEC        (RO)           ; YES -- BACK IT UP TO 7777.
3296          1$:      MOV        (SP)+,RO      ; RESTORE RO
3297          RTS        PC            ; RETURN TO CALLER.
3298
3299          CKEMAX: MOV        RO,-(SP)      ; SAVE RO
3300          MOV        UNITN,RO      ; GET UNIT NUMBER
3301          ASL        RO            ; ... AND MAKE IT A WORD OFFSET
3302          MOV        ERTABL(RO),RO      ; GET ERROR TABLE ENTRY
3303          BIC        #170000,RO      ; EXTRACT ERROR COUNT FIELD
3304          CMP        RO,GERRMAX      ; IS GLOBAL LIMIT EXCEEDED FOR THIS UNIT?
3305          BHS        1$            ; BR IF YES
3306          CMP        ERRK,LERRMAX     ; IS LOCAL LIMIT EXCEEDED FOR THIS TEST?
3307          BLO        2$            ; BR IF NO
3308          1$:      RFLAGS  RO      ; GET OPERATOR FLAGS
3309          TRAP      C#RFLA
3310          BIT        #IDU,RO
3311          BNE        2$            ; IS DROPPING INHIBITED?
3312          MOV        #-1,DUFLG      ; BR IF YES.
3313          ERROF     4,EMAXDU      ; NO -- DROP THE UNIT
3314          TRAP      C#ERDF
3315          .WORD      4
3316          .WORD      EMAXDU

```

```

3306 020100 000000          .WORD 0
      020102          DODU UNITN
      020102 013700 002150 MOV UNITN,R0
      020106 104451      TRAP C#DODU
3307 020110          DOCLN
      020110 104444      TRAP C#DCLN
3308 020112 012600      2$: MOV (SP)+,R0      ; RESTORE R0
3309 020114 000207      RTS PC      ; RETURN TO CALLER
3310          .SBTTL FATCHK - INC FATAL ERRORS AND CHECK FOR LIMIT
3311          ;
3312          ;
3313          ; CHECK FATAL COUNTER, AFTER INC, FOR MORE THAN 25
3314          ; ERRORS AND IF OVER CALL UNIT DROP ROUTINE
3315          ;
3316          ;
3317 020116          FATCHK:
3318 020116          SAVREG
3319 020122 013701 002150 MOV UNITN,R1      ;BETTER SAVE THE REGISTERS
3320 020126 006301      ASL R1      ;PICK UP THE UNIT NUMBER
3321 020130 062761 000001 003130 ADD #1,ERTABL(R1) ;MAKE IT INTO A BYTE OFFSET
3322 020136 005237 002170      INC FATFLG ;ADD 1 TO THE PROPER UNIT'S ERROR COUNTER
3323 020142 023727 002170 000031 CMP FATFLG,#25. ;BUMP FATAL ERROR COUNTER
3324 020150 002406      BLT 9$      ;CHECK AGAINST 25
3325 020152          RFLAGS R0      ;BR, IF LESS THAN 25 ERRORS
      020152 104421      TRAP C#RFLA ;READ THE FLAGS INTO R0
3326 020154 032700 040000      BIT #BIT14,R0 ;BR, IF LOOP ON ERROR IS SET
3327 020160 001002      BNE 9$      ;OTHERWISE NEVER BE ABLE TO SCOPE ETC.
3328 020162 004737 020170      JSR PC,CKDROP ;DROP UNIT IF ALLOWED
3329 020166 000207      RTS PC      ;RETURN ETC.
3330          ;
3331          ;
3332          ;

```

```

3334                                     .SBTTL CKDROP - CHECK IF UNIT SHOULD BE DROPPED
3335                                     ;
3336                                     ; CHECK IF UNIT SHOULD BE DROPPED
3337                                     ;
3338 020170 010046 CKDROP: MOV      R0, -(SP)
3339 020172 FORCERROR 1$, NOTSSR
3340 020202 RFLAGS R0
3341 020204 104421 TRAP      C$RFLA
3342 020210 032700 000040 BIT      @IDU, R0
3343 020212 001010 BNE      1$
3344 020214 011600 MOV      (SP), R0
3345 020222 012737 177777 003060 MOV      @-1, DUFLG
3346 020222 013700 002150 DODU     UNITN
3347 020226 104451 MOV      UNITN, R0
3348 020230 TRAP      C$DODU
3349 020230 DOCLN                                     ; ABORT THE PASS
3350 020230 104444 TRAP      C$DCLN
3351 020232 012600 1$: MOV      (SP)+, R0
3352 020234 000207 RTS      PC
3353                                     .SBTTL CONFIG - DETERMINE CONFIGURATION OF SYSTEM
3354                                     ;
3355                                     ; SUBROUTINE - DETERMINE CONFIGURATION OF TK 25 SYSTEM.
3356                                     ;
3357 020236 CONFIG:
3358 020236 004737 016660 JSR      PC, SOFINIT
3359 020242 000207 RTS      PC
3360
3361
3362

```

```

3364 .SBTTL KTON,KTOFF          ENABLE/DISABLE MEMORY MANAGEMENT
3365
3366 ; SUBROUTINE - ENABLE MEM MGT.
3367 ;
3368 020244 005737 003100      KTON:  TST      KTFLG      ; GOT KT?
3369 020250 001403              BEQ      1$           ; NO.
3370 020252 012737 000001 177572  MOV     @1,SRO      ; YES. ENABLE KT11.
3371 020260 000207          1$:   RTS      PC
3372
3373
3374
3375 ; SUBROUTINE - DISABLE MEM MGT.
3376 ;
3377 ;
3378 020262 005737 003100      KTOFF: TST      KTFLG      ; GOT KT11?
3379 020266 001405              BEQ      1$           ; NO.
3380 020270 000240              NOP
3381 020272 000240              NOP
3382 020274 012737 000000 177572  MOV     @0,SRO      ; DISABLE KT.
3383 020302 000207          1$:   RTS      PC
3384
3385

```

```

3387 .SBTTL SETMAP - SETUP PAR6 MAPPING
3388
3389
3390
3391 ; THIS ROUTINE SETS UP KERNEL PAR6 TP HANDLE
3392 ; AN 18 BIT ADDRESS. THE OFFSET INTO THE PAGE
3393 ; IS RETURNED BIASED TO PAR6.
3394
3395 ; INPUTS:
3396
3397 ; R0 HIGH ORDER ADDRESS BITS
3398 ; R1 LOW ORDER ADDRESS BITS
3399
3400 ; OUTPUTS:
3401
3402 ; R0 OFFSET INTO BLOCK WITH PAR6 BIAS (I.E. THE ADDRESS)
3403 ; CARRY SET IF SUCCESS
3404 ; CLR IF ERROR
3405
3406 SETMAP:
3407 SAVREG R1-R4 UNTIL NEXT RETURN
3408 TST R0 ; SYSTEM HAVE ABOVE 28K?
3409 BEQ 10$ ; BR IF NO
3410 MOV R1,R2 ; SAVE LOW ORDER BITS
3411 .REPT 6
3412 ASR R0
3413 ROR R1 ; CONVERT WORD ADDRESS TO 32W BLOCKS
3414 .ENDR ; MAKE IT DOUBLE PRECISION
3415 BIC #177,R1 ; ALINE FOR LOWER 4K BOUNDARY
3416 CMP R1,KIFLG ; HIGHER THAN EXISTING MEMORY?
3417 BMIS 10$ ; BR IF YES
3418 MOV R1,#KIPAR6 ; SETUP MAPPING REGISTER PAR6
3419 BIC #160000,R2 ; SETUP DISPLACEMENT IN PAGE
3420 ADD #140000,R2 ; ADD IN PAR6 BIAS
3421 MOV R2,R0 ; RETURN IN R0
3422 SEC ; SET SUCCESS
3423 BR 15$ ;
3424 10$: CLC ; SET FAILURE
3425 15$: RTS PC ; RETURN
3426

```

```

3428 .SBTTL FILLMEM - FILL MEMORY WITH BACKGROUND PATTERN
3429
3430 ; FILL MEMORY WITH A BACKGROUND PATTERN
3431
3432 ; INPUTS:
3433
3434 ; RO = BACKGROUND PATTERN
3435 ; FREE = FIRST LOCATION AVAILABLE TO DIAGNOSTIC
3436 ; KTFLG = SET TO HIGHEST MEMORY LOCATION IF > 28K.
3437
3438 ; OUTPUTS:
3439
3440 ; NONE
3441
3442 ;
3443 FILLMEM:
3444 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
3445 JSR PC,KTOFF ;DISABLE KT.
3446 MOV R0,R3 ;COPY TEST PATTERN
3447 MOV FREE,R1 ;GET FIRST FREE LOCATION
3448 MOV FRESIZ,R2 ;SIZE OF FREE SPACE BELOW 28K.
3449 10$: MOV R3,(R1)+ ;STORE A BACKGROUND WORD
3450 DEC R2 ;DONE ALL MEMORY IN FREE SPACE?
3451 BGT 10$ ;BR IF NO
3452 TST KTFLG ; GOT KT?
3453 BEQ 55$ ; NO. GET OUT.
3454 JSR PC,KTON ; YES. ENABLE KT.
3455 CLR R0 ;HIGH ORDER ADDRESS START
3456 020454 MOV PST32W,R1 ;GET >28K START ADDRESS (IN 32W BLOCKS)
3457 000006 .REPT 6
3458 CLC ;CLEAR C BIT
3459 ROL R1 ;CONVERT BLOCKS TO WORDS
3460 ROL R0 ;MAKE IT DOUBLE PRECISION
3461 .ENDR
3462 020524 004737 020304 JSR PC,SETMAP ;SETUP PAR6 MAPPING REGISTER
3463 020530 010320 30$: MOV R3,(R0)+ ;STORE TEST PATTERN IN >28K ADDRESS
3464 020532 020027 160000 CMP R0,#160000 ;END OF PAR6 MAPPING AREA?
3465 020536 103774 30$ BLO 30$ ;BR IF NO
3466 020540 162700 020000 SUB #20000,R0 ;BACKUP INTO PAR6 MAPPING BEGIN
3467 020544 062737 000200 172354 ADD #200,#KIPAR6 ;POINT TO NEXT 4K BLOCK >28K.
3468 020552 023737 172354 003100 CMP #KIPAR6,KTFLG ;END OF MEMORY?
3469 020560 001402 BEQ 50$ ;BR IF YES
3470 020562 000137 020530 JMP 30$ ;KEEP GOING ON ETC.
3471 020566 004737 020262 50$: JSR PC,KTOFF ;DISABLE KT.
3472 020572 000207 55$: RTS PC
3473
3474

```

```

3476 .SBTTL CMPMEM COMPARE MEMORY TO BACKGROUND PATTERN
3477 ;
3478 ; COMPARE MEMORY WITH A BACKGROUND PATTERN
3479 ;
3480 ; INPUTS:
3481 ;
3482 ; RO = BACKGROUND PATTERN
3483 ; FREE = FIRST LOCATION AVAILABLE TO DIAGNOSTIC
3484 ; KTFLG = SET TO HIGHEST MEMORY LOCATION IF > 28K.
3485 ;
3486 ; OUTPUTS:
3487 ;
3488 ; CARRY - SET IF NO ERROR
3489 ; CARRY - CLR IF ERROR
3490 ;
3491 ; IMPLICIT OUTPUTS:
3492 ;
3493 ; ERRHI - ERROR HIGH ADDRESS
3494 ; ERRLO - ERROR LOW ADDRESS
3495 ; EXPD - EXPECTED DATA
3496 ; RECV - RECEIVED DATA
3497 ;
3498 CMPMEM:
3499 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
3500 MOV R0,R3 ;COPY TEST PATTERN
3501 JSR PC,KTOFF ;DISABLE KT.
3502 MOV FREE,R1 ;GET FIRST FREE LOCATION
3503 MOV FRESIZ,R2 ;SIZE OF FREE SPACE BELOW 28K.
3504 10$: CMP R3,(R1) ;FREE SPACE LOCATION EQUAL TO EXPD?
3505 BEQ 15$ ;BR IF YES
3506 MOV R1,ERRLO ;SAVE ADDRESS IN ERROR
3507 CLR ERRHI ;NO HIGH ADDRESS
3508 MOV R3,EXPD ;SAVE EXPD FOR ERROR REPORT
3509 MOV (R1),RECV ;SAVE RECV FOR ERROR REPORT
3510 BR 50$ ;
3511 15$: TST (R1)+ ;POINT TO NEXT ADDRESS
3512 DEC R2 ;DONE ALL MEMORY IN FREE SPACE?
3513 BGT 10$ ;BR IF NO
3514 TST KTFLG ; GOT KT?
3515 BEQ 55$ ; NO. GET OUT.
3516 JSR PC,KTON ; YES. ENABLE KT.
3517 CLR R0 ;HIGH ORDER ADDRESS START
3518 MOV PST32W,R1 ;GET >28K START ADDRESS (IN 32W BLOCKS)
3519 .REPT 6
3520 ROL R1 ;CONVERT BLOCKS TO WORDS
3521 ROL R0 ;MAKE IT DOUBLE PRECISION
3522 .ENDR
3523 BIC #177,R1 ;ALINE 4K BOUNDARY
3524 MOV R0,-(SP) ;SAVE HIGH ORDER
3525 MOV R1,-(SP) ;SAVE LOW ORDER
3526 JSR PC,SETMAP ;SETUP PAR6 MAPPING REGISTER
3527 MOV R0,R4 ;COPY ADDRESS BIASED TO PAR6
3528 MOV (SP)+,R1 ;RESTORE LOW ORDER IN NON PAR6 FORMAT
3529 MOV (SP)+,R0 ;RESTORE HIGH ORDER IN NON PAR6 FORMAT
3530 30$: CMP R3,(R4) ;ABOVE 28K LOCATION EQUAL EXPD?
3531 BEQ 32$ ;BR IF YES
3532 MOV R0,ERRHI ;SAVE HIGH ORDER IN ERROR

```

```

3533 020754 010137 002204      MOV      R1,ERRLO      ;SAVE LOW ORDER IN ERROR
3534 020760 010337 002176      MOV      R3,EXPD      ;SAVE EXPD FOR ERROR REPORT
3535 020764 011437 002200      MOV      (R4),RECV      ;SAVE RECV FOR ERROR REPORT
3536 020770 000421              BR          50$              ;
3537 020772 062701 000002      32$:  ADD      #2,R1          ;UPDATE NON PAR6 ADDRESS
3538 020776 005500              ADC      R0              ;MAKE IT DOUBLE PRECISION ADD
3539 021000 062704 000002      ADD      #2,R4          ;UPDATE PAR FORMAT ADDRESS
3540 021004 020427 160000      CMP      R4,#160000      ;END OF PAR6 MAPPING AREA?
3541 021010 103755              BLO      30$              ;BR IF NO
3542 021012 162704 020000      SUB      #20000,R4        ;BACKUP INTO PAR6 MAPPING BEGIN
3543 021016 062737 000200 172354  ADD      #200,#KIPAR6      ;POINT TO NEXT 4K BLOCK >28K.
3544 021024 023737 172354 003100  CMP      #KIPAR6,KTFLG      ;END OF MEMORY?
3545 021032 101744              BLOS     30$              ;BR IF NO
3546 021034 004737 020262      50$:  JSR      PC,KTOFF        ;TURN OFF MEMORY MAPPING
3547 021040 000241              CLC              ;SET FAILURE
3548 021042 000403              BR          60$              ;
3549 021044 004737 020262      55$:  JSR      PC,KTOFF        ;TURN OFF MEMORY MAPPING
3550 021050 000261              SEC              ;SET SUCCESS
3551 021052 000207      60$:  RTS      PC
3552

```

```

3554 .SBTTL REGSAV - SAVE R1-R5 ON STACK
3555 ;*
3556 ;
3557 ;ROUTINE TO
3558 ;SAVE R1 THROUGH R5 ON THE STACK
3559 ;
3560 ;CALLING SEQUENCE:
3561 ;
3562 ; JSR R5,REGSAV
3563 ;
3564 ;THIS IS A COOROUTINE WHICH TRANSFER CONTROL BACK TO
3565 ;THE CALLING ROUTINE. AT THE END OF THE CALLING ROUTINE,
3566 ;THE RTS PC RETURNS CONTROL TO THIS ROUTINE TO RESTOR
3567 ;REGISTERS.
3568 ;
3569 ;THIS ROUTINE SHOULD ONLY BE CALLED FROM ROUTINES WHICH ARE
3570 ;CALLED VIA A JSR PC INSTRUCTION
3571 ;
3572 ;-
3573
3574 021054 REGSAV:
3575 021054 BREAK ;LOOK FOR CNTL C
3576 021054 104422 TRAP C#BRK
3577 021056 010446 MOV R4,-(SP)
3578 021060 010346 MOV R3,-(SP)
3579 021062 010246 MOV R2,-(SP)
3580 021064 010146 MOV R1,-(SP)
3581 021066 010546 MOV R5,-(SP)
3582 021070 016605 000012 MOV 10.(SP),R5
3583 021074 004736 JSR PC,8(SP)+
3584 021076 012601 MOV (SP)+,R1
3585 021100 012602 MOV (SP)+,R2
3586 021102 012603 MOV (SP)+,R3
3587 021104 012604 MOV (SP)+,R4
3588 021106 012605 MOV (SP)+,R5
3589 021110 104422 BREAK ;LOOK FOR CNTL C
3590 021112 000207 TRAP C#BRK
RTS PC

```

```

3592          .SBTTL  GETPAT  - GET 8 BIT PATTERN FROM OPERATOR
3593          ;*
3594          ;
3595          ;ROUTINE TO REQUEST AN 8 BIT DATA PATTERN FROM THE OPERATOR
3596          ;
3597          ;INPUTS:
3598          ;
3599          ;      NONE.
3600          ;
3601          ;OUTPUTS:
3602          ;
3603          ;      RO      OCTAL NUMBER FROM THE OPERATOR
3604          ;
3605          ;CALLING SEQUENCE:
3606          ;
3607          ;      JSR      PC,GETPAT
3608          ;
3609          ;-
3610
3611  GETPAT::
3612          SAVREG          ;SAVE THE GENERAL REGISTERS
3613  1$:      GMANID  DATASC,PATDAT,0,377,0,377,NO
3614          TRAP    C$GMAN
3615          BR      10000$
3616          .WORD   PATDAT
3617          .WORD   T$CODE
3618          .WORD   DATASC
3619          .WORD   377
3620          .WORD   T$LOLIM
3621          .WORD   T$HILIM
3622          10000$:
3623          BNCOMPLETE      1$      ;RETRY IF ERROR
3624          PCC            1$
3625          MOV            PATDAT,RO  ;DATA PATTERN FROM OPERATOR
3626          RTS            PC        ;RETURN TO CALLER
3627
3628          ;*
3629          ;LOCAL DATA AREA
3630          ;-
3631
3632  PATDAT:  .WORD   0              ;TEMPORARY STORAGE FOR DATA
3633  DATASC:  .ASCIZ  'ENTER DATA PATTERN'
3634          .EVEN

```

```

3626                                     .SBTTL GETSEL - ISSUE MENU AND GET OPERATOR RESPONSE
3627
3628 ;ROUTINE TO ISSUE A MENU AND GET
3629 ;THE OPERATOR'S RESPONSE.
3630 ;INPUTS:
3631 ;
3632 ;      R0      ADDRESS OF ASCIZ STRING OF MENU
3633 ;      R1      MAXIMUM ALLOWABLE OPERATOR RESPONSE
3634 ;OUTPUTS:
3635 ;
3636 ;      R0      NUMBER OF THE OPERATOR'S SELECTION
3637 ;
3638 GETSEL::
3639     SAVREG                                ;SAVE GENERAL REGISTERS
3640     MOV     R0,R2                        ;SAVE THE MENU ADDRESS
3641     MOV     R2,R3                        ;START OF MENU STRING
3642     TST     (R3)                        ;END OF ASCII ?
3643     BEQ     3$                          ;BRANCH IF ALL LINES DISPLAYED
3644     PRINTF  @SELASC,(R3),                ;DISPLAY THE MENU
3645     MOV     (R3),-(SP)
3646     MOV     @SELASC,-(SP)
3647     MOV     @2,-(SP)
3648     MOV     SP,R0
3649     TRAP    C$PRINTF
3650     ADD     @6,SP
3651     BR      2$
3652     3$:    GMANID MENASC,MENRES,0,-1,0, 1,NO
3653     TRAP    C$GMAN
3654     BR      10001$
3655     .WORD   MENRES
3656     .WORD   T$CODE
3657     .WORD   MENASC
3658     .WORD   -1
3659     .WORD   T$LOLIM
3660     .WORD   T$HILIM
3661     10001$: BNCOMPLETE 1$              ;RETRY IF ERROR
3662     BCC     1$
3663     MOV     MENRES,R0                    ;GET THE OPERATOR'S REPLY
3664     CMP     R0,R1                        ;COMPARE TO MAXIMUM ALLOWED
3665     BLOS    5$                          ;BRANCH IF OK
3666     PRINTF  @MENERR                      ;DISPLAY ERROR MESSAGE
3667     MOV     @MENERR,-(SP)
3668     MOV     @1,-(SP)
3669     MOV     SP,R0
3670     TRAP    C$PRINTF
3671     ADD     @4,SP
3672     BR      1$                          ;RETRY
3673     5$:    RTS     PC                    ;RETURN TO CALLER
3674     MENERR: .ASCIZ  '*** Menu Selection Too Large ***'
3675     SELASC: .ASCIZ  'SEL'
3676     MENASC: .ASCIZ  'Enter Menu Selection: '
3677     .EVEN
3678     MENRES: .WORD   0

```

```

3660 .SBTTL CHKMAN - CHECK MANUAL INTERVENTION LEGALITY
3661
3662 ;
3663 ;ROUTINE TO TEST FOR MANUAL INTERVENTION LEGALITY.
3664 ;
3665 ;INPUT:
3666 ;
3667 ;      NONE.
3668 ;
3669 ;OUTPUT:
3670 ;
3671 ;      CARRY    0      MANUAL INTERVENTION NOT ALLOWED
3672 ;              1      MANUAL INTERVENTION IS OK
3673 ;
3674 ;SIDE EFFECTS:
3675 ;
3676 ;      A MESSAGE IS DISPLAYED WARNING THAT TEST IS
3677 ;      NOT EXECUTED IF MANUAL INTERVENTION IS NOT
3678 ;      ALLOWED.
3679 ;
3680 ;-
3681
3682 021420 CHKMAN:
3683 021420      SAVREG      ;SAVE THE REGISTERS
3684 021424      MANUAL    ;SEE IF MANUAL INTERVENTION OK
3685 021424      TRAP      C1MANI
3686 021426      BCOMPLETE 11      ;BRANCH IF ALLOWED
3687 021430      BCS       11
3688 021430      PRINTF    @NOMAN    ;PRINT THE WARNING MESSAGE
3689 021430      MOV        @NOMAN, -(SP)
3690 021434      MOV        @1, -(SP)
3691 021440      MOV        SP, R0
3692 021442      TRAP      C1PNTF
3693 021444      ADD        @4, SP
3694 021450      CLC
3695 021452      RTS       PC      ;CLEAR CARRY FOR ERROR
3696                                ;RETURN
3697
3698 021454      045      116      045  NOMAN: .ASCIZ 'NSA *** Manual Intervention not Allowed  Test Aborted ***
3699                                .even

```

```

3693                                     .SBTTL  ENVIRN  - SETUP FREE DIAGNOSTIC SPACE
3694                                     ;
3695                                     ; SUBROUTINE TO SET-UP VARIOUS ENVIRONMENTAL PARAMETERS.
3696                                     ;
3697 021550  ENVIRN: MEMORY  R0
      021550 104431      TRAP  C:MEM
3698 021552 010037 003072      MOV  R0,FREE      ; GET 1ST FREE ADDRESS...
3699 021556 062737 000002 003072      ADD  #2,FREE
3700 021564 011037 003074      MOV  (R0),FRESIZ ; ...AND WORD COUNT.
3701 021570 162737 000004 003074      SUB  #4,FRESIZ
3702 021576 013702 002012      MOV  L:UNIT,R2  ; GET NUMBER OF UNITS
3703 021602 162737 000007 003074 104:  SUB  #7,FRESIZ ; TAKE AWAY 7 WORDS PER UNIT
3704 021610 005302      DEC  R2
3705 021612 001373      BNE  104
3706 021614 013700 003072      MOV  FREE,R0    ;GET FIRST FREE ADDRESS
3707 021620 063700 003074      ADD  FRESIZ,R0  ;POINT TO LAST FREE ADDRESS
3708 021624 162700 000002      SUB  #2,R0    ;BACKUP 1 WORD
3709 021630 010037 003076      MOV  R0,FREEHI ;STORE LAST FREE ADDRESS
3710 021634 000207      RTS  PC              ;RETURN
3711

```

```

3713                                     .SBTTL KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS
3714                                     ;*
3715                                     ;
3716                                     ;ROUTINE TO INIT KT-11
3717                                     ;
3718                                     ;-
3719
3720 021636                               KTINIT:
3721 021636 005037 003100                 CLR     KTFLG           ; INIT >28K MEMORY FLAG
3722 021642 005037 003102                 CLR     KTENABLE        ; INIT TEST >28K FLAG
3723 021646 023727 002120 001577         CMP     L$HIME,#1577      ; GOT ENOUGH MEMORY (>28K)?
3724 021654 101444                       ELOS    9#               ; NO.
3725 021656 013700 000004                 MOV     @ERRVEC,R0       ; SAVE OLD ERR VEC PTR.
3726 021662 012737 021754 000004         MOV     #21,@ERRVEC     ; SET ERR VEC PTR.
3727 021670 005737 177572                 TST     @SRO           ; GOT KT11?
3728 021674 000240                       NOP                     ; (TRAP IF NO).
3729 021676 013737 002120 003100         MOV     L$HIME,KTFLG      ; YES. SET KT FLAG.
3730 021704 042737 000177 003100         BIC     #177,KTFLG       ;
3731 021712 010037 000004                 MOV     R0,@ERRVEC       ; RESTORE OLD ERR VEC PTR.
3732 021716 005000                       CLR     R0              ; R0 = AR DATA.
3733 021720 012701 172340                 MOV     #KIPAR0,R1       ; R1 = KI REGS PTR.
3734 021724 012761 077406 177740 1# :   MOV     #77406,-40(R1)   ; SET DESCRIPTOR REG.
3735 021732 010021                       MOV     R0,(R1)+         ; SET KIPAR REG.
3736 021734 062700 000200                 ADD     #200,R0         ; BUMP AR DATA BY "4K".
3737 021740 020027 002000                 CMP     R0,#2000        ; AT "I/O"?
3738 021744 001367                       BNE     1#               ; NO.
3739 021746 012741 177600                 MOV     #177600,-(R1)   ; YES. SET KIPAR7 FOR I/O.
3740 021752 000405                       BR      9#               ;
3741
3742 021754 012716 021762                 2# :   MOV     #61,(SP)   ; SET UP RETURN
3743 021760 000002                       RTI                     ; RTI TO NEXT LOCATION
3744
3745 021762 010037 000004                 6# :   MOV     R0,@ERRVEC   ; RESTORE OLD ERR VEC PTR.
3746
3747 021766 000207                 9# :   RTS     PC
3756
3757
3763

```

```
3765  
3766 021770 .SBTTL PROTECTION TABLE  
          021770 BGNPROT  
3767 021770 177777 177777 177777 L$PROT::  
3768 022000 .WORD -1, -1, -1, -1 ;NO DEVICE PROTECTION REQUIRED.  
3769 ENDPROT
```

```

3771                                     .SBTTL  INITIALIZE SECTION
3772
3773                                     ;**
3774                                     ;THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
3775                                     ;AT THE BEGINNING OF EACH PASS.
3776                                     ;
3777                                     ;IF "START" OR "RESTART", SET QUICK-PASS FLAG AND BUS-INIT.
3778                                     ;IF "CONTINUE", NOTHING IS REQUIRED.
3779                                     ;
3780                                     ;--
3781                                     ;*
3782                                     ;INSERT TEMPORARY JUMP TO ODT
3783                                     ;-
3784 022000                                     BGNINIT
3785 022000 L$INIT::
3786 022000 012737 005672 002146 40$: MOV     #EPR1,EPR1SW ;SET UP PRIMARY MESSAGE FOR REPLACEMENT
3787 022006 005037 003106      CLR     SIFLAG ;CLEAR "SOFT INIT" FLAG
3788 022012 005037 003102      CLR     KTENABLE ;CLEAR TEST ABOVE 28K FLAG
3789 022016 005037 002246      CLR     RAMSIZ ;CLEAR RAM SIZE FOR RAMERR ROUTINE
3790 022022      READDEF #EF.CONTINUE
3791 022022 012700 000036      MOV     #EF.CONTINUE,RO
3792 022026 104447      TRAP     C$REFG
3793 022030      BNCOMPLETE 1$
3794 022030 103023      BCC      1$
3795 022032 023737 002150 002012      CMP     UNITN,L$UNIT ;UNIT IN RANGE?
3796 022040 103064      BMIS     4$ ;BR IF NO.
3797 022042 005737 003060      TST     DUFLG ;DROPPED UNIT?
3798 022046 100466      BMI      NXTU ;BR IF YES
3799 022050 013701 002150      MOV     UNITN,R1
3800 022054 006301      ASL      R1
3801 022056 005761 003130      TST     ERTABL(R1)
3802 022062 001512      BEQ      SETU
3803 022064 032761 040000 003130      BIT     #BIT14,ERTABL(R1) ;DROPPED?
3804 022072 001054      BNE      NXTU
3805 022074      EXIT      INIT ;DO NOTHING IF "CONTINUE".
3806 022074 104432      TRAP     C$EXIT
3807 022076 000412      .WORD    L10030-.
3808 022100      READDEF #EF.NEW
3809 022100 012700 000035 1$: MOV     #EF.NEW,RO
3810 022104 104447      TRAP     C$REFG
3811 022106      BNCOMPLETE NXTU ;TAKE NEXT UNIT IF NOT NEW PASS.
3812 022106 103046      BCC      NXTU
3813 022110      READDEF #EF.START
3814 022110 012700 000040      MOV     #EF.START,RO
3815 022114 104447      TRAP     C$REFG
3816 022116      BCOMPLETE 2$
3817 022116 103404      BCS      2$
3818 022120      READDEF #EF.RESTART
3819 022120 012700 000037      MOV     #EF.RESTART,RO
3820 022124 104447      TRAP     C$REFG
3821 022126      BNCOMPLETE 31$
3822 022126 103025      BCC      31$
3823 022130      BRESET
3824 022130      TRAP     C$RESET ;1ST PASS, BUS-INIT...
3825 022130 104433      CLR      TSTCNT ;BUS RESET.
3826 022132 005037 002162      ;NUMBER OF TESTS RUN IN PASS

```

```

3812 022136 005037 002170          CLR    FATFLG      ;RESET FLAG TO ZERO "FATAL ERRORS"
3813 022142 005037 003332          CLR    SKIPT      ;CLEAR THE SUBTEST "SKIPPER"
3814 022146                                194:
3815 022146 012737 177777 002152 204:      MOV    @-1,QVP      ;...QUICK VERIFY...
3816 022154 004737 021550          JSR    PC,ENVIRN      ;SET ENVIRONMENT.
3817 022160 004737 021636          JSR    PC,KTINIT      ;INITIALIZE KT MEMORY MANAGEMENT
3818 022164 012700 003130          MOV    @ERTABL,R0
3819 022170 005020 304:      CLR    (R0).      ;CLEAR THE ERROR TABLE
3820 022172 020027 003330          CMP    R0,@ERTABE
3821 022176 103774          BLO    304
3822 022200 000404          BR    44
3823 022202 005037 002152 314:      CLR    QVP
3824 022206 000137 022256          JMP    PASRPT      ;GO REPORT THE STATUS
3825
3826 022212                                44:
3827 022212 012737 177777 002150 NEWPAS: MOV    @-1,UNITN      ;INIT UNIT NUMBER...
3828 022220 005037 002166          CLR    DEVCNT      ;CLEAR COUNT OF DEVICES RUNNING
3829 022224                                NXTU:
3830 022224 005237 002150          BREAK
3831 022232 023737 002150 002012      TRAP    C:BRK
3832 022240 103423          INC    UNITN      ;...AND SET NEXT UNIT NUMBER.
3833 022242 012737 177777 003060      CMP    UNITN,L$UNIT
3834 022250 000401          BLO    SETU
3835 022252          MOV    @-1,DUFLG
3836 022252 104444          BR    114
3837 022254 000240          DOCLN
3838 022256 023727 002012 000001      TRAP    C:DOCLN      ;ABORT, NO MORE UNITS.
3839 022264 101752          NOP
3840 022266 005737 002166 114:
3841 022272 001747          PASRPT:
3842 022274 104421          CMP    L$UNIT,@1      ;HOW MANY UNITS SELECTED?
3843 022276 032700 000100          BLOS    NEWPAS      ;BR IF ONLY 1
3844 022302 001343          TST    DEVCNT      ;ARE ANY STILL RUNNING?
3845          BEQ    NEWPAS      ;BR IF NO
3846 022304          RFLAGS    R0
3847 022306 000741          TRAP    C:RFLA
3848 022310          BIT    @ISR,R0      ;SHOULD WE PRINT STATISTICS
3849          BNE    NEWPAS      ;BR IF NO
3850          DORPT
3851          TRAP    C:DRPT
3852          BR    NEWPAS
3853          104:
3854          SETU:
3855          GPHARD    UNITN,R0      ;GET UNIT N P TABLE POINTER.
3856          MOV    UNITN,R0
3857          TRAP    C:GPHRD
3858          BNCOMPLETE NXTU      ;BR IF UNIT NOT AVAILABLE.
3859          BCC    NXTU
3860          CLR    DUFLG      ;CLEAR "DROPPED" FLAG.
3861          INC    DEVCNT
3862          MOV    (R0),R1      ;GET 1ST REGISTER ADDRESS.
3863          MOV    R1,CSRADDR      ;ADDRESS OF REGISTERS OF UNIT UNDER TEST
3864
3865          MOV    (R0),R1      ;GET VECTOR ADDRESS.
3866          MOV    (R0),R2      ;GET INTERRUPT PRIORITY
3867          MOV    R2,IPRI      ;SET INTERRUPT PRIORITY.
3868          MOV    R1,IVEC      ;SET INTERRUPT VECTOR POINTER...
3869          MOV    @INTR,(R1).      ;...VECTOR...

```

```

3862 022356 010221          MOV      R2,(R1),      ;...AND PRIORITY.
3863
3864 022360          1$:
3865          ;          TST      QVP              ;1ST PASS ??
3866          ;          BEQ      5$              ;NO, SKIP THE PASS 1 STUFF.
3867
3868          ;
3869          ;1ST PASS, CHECK THAT DEVICE ADDRESSES ARE VALID, AND
3870          ;THAT THE DISPLA STATUS IS PROPERLY INITIALIZED.
3871          ;
3872 022360 013701 002150      MOV      UNITN,R1
3873 022364 006301          ASL      R1
3874 022366 052761 100000 003130  BIS      #BIT15,ERTABL(R1) ;SAY DEVICE RUNNING
3875 022374 005037 005232          CLR      EXTA              ;CLEAR ERROR EXTENSION FLAG.
3876 022400 023727 002012 000001  CMP      L$UNIT,#1          ;ARE WE TESTING MULTIPLE UNITS?
3877 022406 101416          BLOS     10$              ;BR IF NO.
3878 022410          RFLAGS     R0              ;YES -- GET OPERATOR FLAGS.
3879 022412 032700 001000      TRAP     C$RFLA
3880 022416 001412          BIT      #PNT,R0          ;SHOULD WE PRINT UNIT #?
3881 022420          BEQ      10$              ;BR IF NOT.
3882 022420 013746 002150      PRINTF   #PUNIT,UNITN      ;PRINT THE UNIT #
3883 022424 012746 022512      MOV      UNITN,-(SP)
3884 022430 012746 000002      MOV      #PUNIT,-(SP)
3885 022434 010600          MOV      #2,-(SP)
3886 022436 104417          MOV      SP,R0
3887 022440 062706 000006      TRAP     C$PNTF
3888 022444          ADD      #6,SP
3889 022444 005037 003062      10$:      CLR      NODEV
3890 022450 013701 002154      MOV      CSRADDR,R1      ;ADDRESS OF FIRST REGISTER
3891 022454 010102          MOV      R1,R2              ;START OF REGISTERS
3892 022456 062702 000000      ADD      #TSSR,R2          ;ADDRESS OF TSSR REGISTER
3893 022462 004737 017312      JSR      PC,XNXM          ;TEST BOTH CONTROLLER REGISTERS...
3894 022466 103005          BCC      2$              ;...AND BR IF ALL OK.
3895 022470 010137 003062      MOV      R1,NODEV        ;FLAG DEVICE AS NON EXISTENT
3896 022474 012737 177777 003060      MOV      #-1,DUFLG      ;DROP THIS UNIT.
3897 022502          2$:
3898          ;
3899          ;FINALLY, SET CPU PRIORITY AND WE'RE DONE.
3900          ;
3901 022502          5$:      SETPRI   #PRI00          ;ENABLE INTERRUPTS.
3902 022506 012700 000000      MOV      #PRI00,R0
3903 022506 104441          TRAP     C$SPRI
3904 022510          ENDINIT
3905 022510 104411          L10030:  TRAP     C$INIT
3906 022512 045 116 045 PUNIT: .ASCIZ  /***** TESTING UNIT #02#A *****/
3907          .EVEN

```

```

3901 .SBTTL ADD AND DROP UNITS SECTIONS
3902
3903
3904 ;**
3905 ; THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
3906 ; TO BE (A) ADDED TO THE TEST LIST FOR THE FIRST TIME,
3907 ; OR (B) RE-INSERTED IF IT HAD BEEN PREVIOUSLY DROPPED.
3908 ; -
3909 BGNAU
3910 L$AU::
3911 MOV R0,R1 ; GET UNIT TO BE ADDED (R0)
3912 ASL R1 ; MAKE IT A WORD INDEX
3913 BIS #100000,ERTABL(R1) ; SET THE "ACTIVE" BIT
3914 BIC #40000,ERTABL(R1) ; CLEAR THE "DROPPED" BIT
3915 PRINTF #1$,R0
3916 MOV R0,-(SP)
3917 MOV #1$,-(SP)
3918 MOV #2,-(SP)
3919 MOV SP,R0
3920 TRAP C$PNTF
3921 ADD #6,SP
3922 EXIT AU
3923 .WORD J$JMP
3924 .WORD L10031-2-
3925 .ASCIZ /%N$A UNIT %D$A ADDED/
3926 .EVEN
3927 ENDAU ; UNUSED.
3928 L10031:
3929 TRAP C$AU
3930 ;**
3931 ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
3932 ; TO BE REMOVED FROM THE TEST LIST.
3933 ;
3934 ; SUPVSR DOES THE "DROPPING". THIS IS JUST TO TELL THE MAN.
3935 ; "DROPPED" UNITS ARE RE-SELECTED ON OPERATOR "STA" OR "ADD"
3936 ; COMMAND, OTHERWISE REMAIN INACTIVE. THE "DISPLAY" COMMAND
3937 ; WILL PRINT ALL DROPPED UNITS, AND THE P-TABLES OF THOSE
3938 ; WHICH ARE STILL ACTIVE.
3939 ; UPON ENTRY, R0 CONTAINS THE UNIT TO BE DROPPED.
3940 BGNDU
3941 L$DU::
3942 MOV # -1,DUFLG
3943 MOV R0,R1
3944 ASL R1
3945 BIS #140000,ERTABL(R1) ; SAY DROPPED
3946 240,240,240 ; ?????????
3947 PRINTF #1$,R0
3948 MOV R0,-(SP)
3949 MOV #1$,-(SP)
3950 MOV #2,-(SP)
3951 MOV SP,R0
3952 TRAP C$PNTF
3953 ADD #6,SP
3954 EXIT DU
3955 .WORD J$JMP
3956 .WORD L10032 2

```

3938	022732		045	116	045	1\$: .ASCIZ /\$N\$A UNIT \$D\$A DROPPED/ .EVEN ENDDU	
3939							
3940	022762				L10032:	TRAP C\$DU	
	022762						
	022762	104453					
3941					***		
3942					; AUTO-DROP CODE SECTION.		
3943					--		
3944	022764				BGNAUTO		
	022764				L\$AUTO::		
3945	022764	012703	000550		MOV #360.,R3	;ENOUGH TIME FOR 2400' REEL TO REWIND	
3946	022770	004737	017134		10\$: JSR PC,WAITF	;WAIT FOR SSR TO SET	
3947	022774	103420			BCS 20\$	;LEAVE WHEN SSR IS SET	
3948	022776				DELAY 250.	;WAIT FOR .25 SECONDS	
	022776	012727	000372		MOV #250.,(PC)+		
	023002	000000			.WORD 0		
	023004	013727	002116		MOV L\$DLY,(PC)+		
	023010	000000			.WORD 0		
	023012	005367	177772		DEC -6(PC)		
	023016	001375			BNE -.4		
	023020	005367	177756		DEC -22(PC)		
	023024	001367			BNE -.20		
3949	023026	005303			DEC R3	;BUMP COUNTER DOWN	
3950	023030	001357			BNE 10\$	;KEEP GOING	
3951	023032	004737	020170		JSR PC,CKDROP	;TRY AND DROP UNIT	
3952	023036				20\$: ENDAUTO	; UNUSED.	
3953	023036				L10033:		
	023036	104461			TRAP C\$AUTO		

```

3955          .SBTTL  CLEAN-UP AND REPORT CODING SECTIONS
3956
3957          ;**
3958          ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS
3959          ; EXECUTED AT THE END OF EACH PASS (OR SUB-PASS).
3960          ; USE TO RETURN DEVICE UNDER TEST TO A NEUTRAL STATE.
3961          ;--
3962          BGNCN
3963          L$CLEAN::
3964          TST      DUFLG          ; "DROPPED" FLAG IS SET ON...
3965          BMI      1$            ; ...AND GROSS CONTROLLER FAULT...
3966          ; ...DON'T TRY TO XCT CLEANUP CODE.
3967          023040 005737 003060
3968          023040 100407
3969          023046 013705 002154
3970          023052 012765 000000 000000
3971          023060 004737 017134
3972          023064
3973          023064
3974          023064 104412
3975          1$:
3976          2$:
3977          L10034:
3978          TRAP     C$CLEAN
3979          ;**
3980          ; THE REPORT CODING SECTION CONTAINS THE
3981          ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
3982          ;--
3983          BGNRPT
3984          L$RPT::
3985          PRINTS   #DEVSUM
3986          MOV      #DEVSUM, -(SP)
3987          MOV      #1, -(SP)
3988          MOV      SP, R0
3989          TRAP     C$PNTS
3990          ADD      #4, SP
3991          MOV      R2, -(SP)
3992          MOV      R3, -(SP)
3993          MOV      R4, -(SP)
3994          MOV      #ERTABL, R4          ; GET START OF ERROR TABLE.
3995          CLR      R3                  ; CLEAR UNIT NUMBER
3996          1$:
3997          MOV      (R4), R2          ; GET ERROR TABLE ENTRY & TEST IT.
3998          BEQ      4$                ; ZERO IF UNIT NOT RUN
3999          BPL      4$
4000          BIT      #BIT14, R2        ; WAS UNIT DROPPED?
4001          BNE      2$                ; BR IF YES
4002          BIC      #C7777, R2        ; GET ERROR COUNT FIELD
4003          PRINTS   #DEVONL, R3, R2   ; PRINT
4004          MOV      R2, -(SP)
4005          MOV      R3, -(SP)
4006          MOV      #DEVONL, -(SP)
4007          MOV      #3, -(SP)
4008          MOV      SP, R0
4009          TRAP     C$PNTS
4010          ADD      #10, SP
4011          BR       4$
4012          2$:
4013          CMP      R2, #160000        ; WAS UNIT NON EXISTENT?
4014          BNE      3$                ; BR IF NO
4015          PRINTS   #DEVNXR, R3
4016          MOV      R3, -(SP)
4017          MOV      #DEVNXR, -(SP)

```

```

023204 012746 000002      MOV      #2,-(SP)
023210 010600      MOV      SP,R0
023212 104416      TRAP     C#PNTS
023214 062706 000006      ADD      #6,SP
3994 023220 000431      BR      4$
3995 023222 020227 160001      3$:    CMP      R2,#160001      ; WAS UNIT NOT READY AT STARTUP?
3996 023226 001012      BNE     30$      ; BR IF NO.
3997 023230      PRINTS   #DEVNRD,R3
023230 010346      MOV      R3,-(SP)
023232 012746 023517      MOV      #DEVNRD,-(SP)
023236 012746 000002      MOV      #2,-(SP)
023242 010600      MOV      SP,R0
023244 104416      TRAP     C#PNTS
023246 062706 000006      ADD      #6,SP
3998 023252 000414      BR      4$
3999 023254 042702 170000      30$:   BIC      #C7777,R2
4000 023260      PRINTS   #DEVDR0,R3,R2
023260 010246      MOV      R2,-(SP)
023262 010346      MOV      R3,-(SP)
023264 012746 023600      MOV      #DEVDR0,-(SP)
023270 012746 000003      MOV      #3,-(SP)
023274 010600      MOV      SP,R0
023276 104416      TRAP     C#PNTS
023300 062706 000010      ADD      #10,SP
4001 023304 062704 000002      4$:    ADD      #2,R4
4002 023310 005203      INC      R3
4003 023312 020427 003330      CMP      R4,#ERTABE
4004 023316 103701      BLO     1$
4005 023320 012604      MOV      (SP)+,R4
4006 023322 012603      MOV      (SP)+,R3
4007 023324 012602      MOV      (SP)+,R2
4008 023326      ENDRPT      ; UNUSED.
023326      L10035:
023326 104425      TRAP     C#RPT
4009
4010
4011 023330      045      116      045  DEVSUM: .ASCIZ  /#N#ADEVICE STATUS SUMMARY:#N/
4012 023365      045      101      040  DEVONL: .ASCIZ  /#A UNIT #D3#A ONLINE, ERRORS = #D#N/
4013 023435      045      101      040  DEVNXR: .ASCIZ  /#A UNIT #D3#A DROPPED, NON-EXISTENT REGISTER#N/
4014 023517      045      101      040  DEVNRD: .ASCIZ  /#A UNIT #D3#A DROPPED, NOT READY AT STARTUP#N/
4015 023600      045      101      040  DEVDR0: .ASCIZ  /#A UNIT #D3#A DROPPED, ERRORS = #D#N/
4016
4019
4026
4032
4040

```

[illegible]

```

4100
4101
4102 023740 004737 016660
4103 023744 103427
4104 023746
      023746 012727 000250
      023752 000000
      023754 013727 002116
      023760 000000
      023762 005367 177772
      023766 001375
      023770 005367 177756
      023774 001367
4105 023776 005337 030272
4106 024002 001356
4107 024004 004737 020116
4111 024010 016501 000000
4112 024014
      024014 104455
      024016 000145
      024020 003550
      024022 011676
4113 024024
4114
4115 024024 012704 030110
4116
4117
4118
4119
4120
4121
4122
4123 024030 004737 010342
4124 024034 103407
4125 024036 004737 020116
4129 024042 010001
4130 024044
      024044 104456
      024046 000146
      024050 004754
      024052 011676
4131
4132
4133
4134
4135 024054 016501 000000
4136 024060 032701 000100
4137 024064 001406
4141 024066
      024066 104455
      024070 000147
      024072 030274
      024074 016360
4142 024076 004737 020170
4143
4144
4145

```

```

;*****
51:      JSR      PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
      BCS      101              ;BR IF INIT WAS OK
      DELAY     250              ;DELAY IF REQUIRED
                                  MOV      #250,(PC)+
                                  .WORD   0
                                  MOV      L1DLY,(PC)+
                                  .WORD   0
                                  DEC      -6(PC)
                                  BNE      -4
                                  DEC      -22(PC)
                                  BNE      -20
      DEC      T25DLY            ;DEC DELAY COUNTER
      BNE      51                ;BR, IF LOOP IS REQUIRED
      JSR      PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
      MOV      TSSR(R5),R1      ;CONTENTS OF TSSR REGISTER
      ERDIF    ERRNO,SFIERR,SFIMSG ;FATAL ERROR TS R WAS NOT OK
                                  TRAP     C1ERDIF
                                  .WORD    101
                                  .WORD    SFIERR
                                  .WORD    SFIMSG
101:
      MOV      #T25PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
;*****
;
;WRITE CHARACTERISTICS COMMAND (CALL TO WRTHCR)
;
;*****
      JSR      PC,WRTHCR          ;ISSUE WRITE CHARACTERISTICS
      BCS      141                ;BR, IF COMMAND ISSUED OK
      JSR      PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
      MOV      R0,R1             ;SAVE CONTENTS OF TSSR
      ERHRD    ERRNO,WRTHMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
                                  TRAP     C1ERHRD
                                  .WORD    102
                                  .WORD    WRTHMSG
                                  .WORD    SFIMSG
;
;      CHECK FOR DRIVE OFF-LINE
141:     MOV      TSSR(R5),R1      ;READ THE TSSR
      BIT      #OFL,R1           ;CHECK FOR DRIVE OFF LINE
      BEQ      151                ;BR, IF DRIVE IS ON LINE (GOOD)
      ERDIF    ERRNO,T21OFL,EXPREC ;"DRIVE IS OFF-LINE" (BAD)
                                  TRAP     C1ERDIF
                                  .WORD    103
                                  .WORD    T21OFL
                                  .WORD    EXPREC
      JSR      PC,CKDROP          ;TRY AND DROP UNIT
;*****
;

```

```

4146      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4147      ;
4148      ;*****
4149
4150 024102 004737 010444      151:      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
4151 024106 103407      BCS      301      ;BR, IF NO PROBLEM
4152 024110 010001      MOV      R0,R1      ;SAVE TSSR
4153 024112 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4157 024116      ERRMRD  ERNO,T25RWN,PKTSSR      ;REWIND NOT ACCEPTED
      024116 104456      TRAP      C1ERMRD
      024120 000150      .WORD    104
      024122 031267      .WORD    T25RWN
      024124 011710      .WORD    PKTSSR
4158 024126      301:      CKLOOP      ;LOOP IF SELECTED
      024126 104406      TRAP      C1CLP1
4159
4160      ;*****
4161      ;
4162      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4163      ;
4164      ;*****
4165
4166 024130 013701 030136      MOV      T258FR+6,R1      ;PICK UP XSTO
4167 024134 010102      MOV      R1,R2      ;SET UP EXPECTED
4168 024136 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
4169 024142 020102      CMP      R1,R2      ;DOES EXP = REC'D
4170 024144 001406      BEQ      401      ;BR, IF EQUAL (OK)
4171 024146 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4175 024152      ERRMRD  ERNO,T25BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      024152 104456      TRAP      C1ERMRD
      024154 000151      .WORD    105
      024156 030457      .WORD    T25BOT
      024160 016360      .WORD    EXPREC
4176 024162      401:      CKLOOP      ;LOOP IF SELECTED
      024162 104406      TRAP      C1CLP1
4177 024164 012703 000400      MOV      #256.,R3      ;RECORD SIZE
4178 024170 013737 003072 030242      MOV      FREE,T25RB      ;STARTING WRITE BUFFER ADDRESS
4179
4180      ;*****
4181      ;
4182      ;WRITE DATA,ACK,CVC=1 COMMAND
4183      ;
4184      ;*****
4185
4186 024176 012737 140005 030240      MOV      #140005,T25PK3      ;WRITE DATA,ACK,CVC=1 COMMAND
4187 024204 012704 030240      MOV      #T25PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
4188 024210
4189 024210 010337 030246      651:      MOV      R3,T25SZ      ;SET UP RECORD SIZE IN PACKET
4190 024214 013777 030270 156650      MOV      T25CNT,0FREE      ;LOAD UP RECORD COUNTER IN WRT BUFFER
4191 024222 062737 000001 030270      ADD      #1,T25CNT      ;GET READY FOR NEXT RECORD
4192 024230 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
4193 024234 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
4194 024240 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
4195 024244 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
4196 024250 020102      CMP      R1,R2      ;ARE THEY EQUAL
4197 024252 001411      BEQ      751      ;BR, IF OK
4198 024254 032701 000004      BIT      #BIT2,R1      ;CHECK FOR TAPE STATUS ALERT

```

```

4199 024260 001014          BNE      1201          ;BR, IF TSA IS SET (SUSPECT IS EOT)
4200 024262 004737 020116    JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
4204                                ;SOFT ERROR GENERATED BECAUSE THE
4205                                ;WRITE COMMAND IS NOT BEING CHECKED
4206                                ;HERE. IT WAS CHECKED IN LEAH2
4207 024266          ERRSOF ERRNO,WRERR,PKTSSR      ;TSSR INCORRECT AFTER WRITE DATA
          024266 104457          TRAP      C$ERSOFT
          024270 000152          .WORD    106
          024272 005011          .WORD    WRERR
          024274 011710          .WORD    PKTSSR
4208 024276          751:      CKLOOP              ;LOOP IF SELECTED
          024276 104406          TRAP      C$CLP1
4209 024300 005203          INC      R3              ;BUMP RECORD SIZE
4210 024302 022703 001000    CMP      #512.,R3      ;END OF RECORD YET
4211 024306 001340          BNE      651            ;BR, IF MORE RECORDS TO WRITE
4212 024310 000415          BR       1251           ;ENOUGH RECORDS
4213 024312          1201:
4214          ;*****
4215          ;
4216          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4217          ;
4218          ;*****
4219
4220 024312 013701 030136    MOV      T25BFR+6,R1    ;QUICK CHECK FOR EOT SET
4221 024316 010102          MOV      R1,R2           ;SET UP EXPECTED
4222 024320 052702 000001    BIS      @BIT0,R2       ;SET THE EOT BIT XSTO
4223 024324 020102          CMP      R1,R2           ;IS THE EOT BIT SET IN XSTO
4224 024326 001406          BEQ      1251           ;BR, IF SET (GOOD)
4225 024330 004737 020116    JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
4229 024334          ERRDF ERRNO,T25NET,EXPREC      ;DEVICE FATAL NOT EOT FOUND ETC.
          024334 104455          TRAP      C$ERDF
          024336 000153          .WORD    107
          024340 030613          .WORD    T25NET
          024342 016360          .WORD    EXPREC

```

4231	024344			125:		
4232				;*****		
4233				:		
4234				;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE		
4235				:		
4236				;*****		
4237						
4238	024344	004737	010444	JSR PC,REWIND	;CALL TAPE REWIND COMMAND	
4239	024350	103407		BCS 130:	;BR, IF NO PROBLEM	
4240	024352	010001		MOV RO,R1	;SAVE TSSR	
4241	024354	004737	020116	JSR PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS	
4245	024360			ERRHRD ERNO,T25RW,N,PktTSSR	;REWIND NOT ACCEPTED	
	024360	104456				TRAP C\$ERHRO
	024362	000154			.WORD	108
	024364	031267			.WORD	T25RW,N
	024366	011710			.WORD	PktTSSR
4246	024370			130:	CKLOOP	;LOOP IF SELECTED
	024370	104406				TRAP C\$CLP1
4247	024372			ENDSUB	;>>>>>>>> END SUBTEST >>>>>>>	L10037:
	024372					TRAP C\$ESUB
	024372	104403				
4248	024374	023727	002170 000031	CMP FATFLG,#25.	;IS ERROR COUNT AT 25	
4249	024402	002402		BLT 999:	;BR, IF LESS THAN 25	
4250	024404	004737	020170	JSR PC,CMDROP	;TRY TO DROP THE UNIT	
4251	024410			999:		

```

4253      ;+
4254      ;
4255      ;TEST 1, SUBTEST 2
4256      ;
4257      ;VERIFIES THAT SPACE RECORDS FORWARD CAN SPACE ONE
4258      ;RECORD OFF BOT AND CAUSES BOT STATUS TO BE CLEARED.
4259      ;
4260      ;
4261      ;
4262      ;-
4263      024410      BGNSUB                                ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
                     024410                                T1.2:
                     024410      104402                  TRAP          C#BSUB
4264      024412      004737      031500                JSR          PC,T25REST        ;SET COMMAND PACKET
4265      024416      004737      031572                JSR          PC,T25RT2       ;SET UP OTHER COMMAND PACKET
4266      024422      004737      031634                JSR          PC,T25RT3       ;SET UP OTHER COMMAND PACKET
4267
4268      ;*****
4269      ;
4270      ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
4271      ;
4272      ;*****
4273
4274      024426      004737      016660                JSR          PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
4275      024432      103407                        BCS          10$             ;BR IF INIT WAS OK
4276      024434      004737      020116                JSR          PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
4280      024440      010001                        MOV          R0,R1           ;CONTENTS OF TSSR REGISTER
4281      024442                        ERDZF         ERRNO,SFIERR,SFMSG         ;FATAL ERROR TSSR WAS NOT OK
                     024442      104455                                TRAP          C#ERDF
                     024444      000155                                .WORD          109
                     024446      003550                                .WORD          SFIERR
                     024450      011676                                .WORD          SFMSG
4282      024452
4283      10$:
4284      024452      012704      030110                MOV          @T25PACKET,R4            ;SUBROUTINE NEEDS PACKET ADDRESS
4285
4286      ;*****
4287      ;
4288      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPHR)
4289      ;
4290      ;*****
4291
4292      024456      004737      010342                JSR          PC,WRTPHR        ;ISSUE WRITE CHARACTERISTICS
4293      024462      103407                        BCS          15$             ;BR, IF COMMAND ISSUED OK
4294      024464      004737      020116                JSR          PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
4298      024470      010001                        MOV          R0,R1           ;SAVE CONTENTS OF TSSR
4299      024472                        ERMRD         ERRNO,WRTMSG,SFMSG         ;WRITE CHARACTERISTICSC FAILED
                     024472      104456                                TRAP          C#ERMRD
                     024474      000156                                .WORD          110
                     024476      004754                                .WORD          WRTMSG
                     024500      011676                                .WORD          SFMSG
4300
4301      ;*****
4302      ;
4303      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4304      ;
4305      ;*****

```

```

4306
4307 024502 004737 010444      15: JSR PC,REWIND      ;CALL TAPE REWIND COMMAND
4308 024506 103407              BCS 30:          ;BR, IF NO PROBLEM
4309 024510 010001              MOV R0,R1        ;SAVE TSSR
4310 024512 004737 020116      JSR PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
4314 024516              ERRMRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP C:ERMRD
                                .WORD 111
                                .WORD T25RWN
                                .WORD PKTSSR
                                TRAP C:CLP1
4315 024526 104406      30: CKLOOP              ;LOOP IF SELECTED
                                TRAP C:CLP1
4316
4317 ;*****
4318 ;
4319 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4320 ;
4321 ;*****
4322
4323 024530 013701 030136      MOV T25BFR+6,R1    ;PICK UP XSTO
4324 024534 010102              MOV R1,R2        ;SET UP EXPECTED
4325 024536 052702 000002      BIS #BIT1,R2      ;SET BOT BIT IN EXPECTED
4326 024542 020102              CMP R1,R2        ;DOES EXP = REC'D
4327 024544 001406              BEQ 40:          ;BR, IF EQUAL (OK)
4328 024546 004737 020116      JSR PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
4332 024552              ERRMRD ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP C:ERMRD
                                .WORD 112
                                .WORD T25BOT
                                .WORD EXPREC
4333 024562              40: CKLOOP              ;LOOP IF SELECTED
                                TRAP C:CLP1
4334 024564 012737 000001 030242 MOV #000001,T25RB ;NUMBER OF RECORDS TO SPACE OVER
4335
4336 ;*****
4337 ;
4338 ;SPACE FORWARD,ACK,CVC=1 COMMAND
4339 ;
4340 ;*****
4341
4342 024572 012737 140010 030240 MOV #140010,T25PK3 ;SPACE FORWARD,ACK,CVC=1 COMMAND
4343 024600 012704 030240      MOV #T25PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
4344 024604
4345 024604 010465 177776      65: MOV R4,T5DB(R5) ;ISSUE COMMAND
4346 024610 004737 017134      JSR PC,WAITF    ;WAIT FOR SSR TO SET
4347 024614 016501 000000      MOV TSSR(R5),R1 ;GET TSSR CONTENTS
4348 024620 012702 000200      MOV #SSR,R2      ;SET UP EXPECTED
4349 024624 020102              CMP R1,R2        ;ARE THEY EQUAL
4350 024626 001411              BEQ 75:          ;BR, IF OK
4351 024630 032701 000004      BIT #BIT2,R1     ;CHECK FOR TAPE STATUS ALERT
4352 024634 001006              BNE 75:          ;BR, IF TSA IS SET (SUSPECT IS EOT)
4353 024636 004737 020116      JSR PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
4357 024642              ERRMRD ERRNO,T25WDE,EXPREC ;TSSR INCORR CT AFTER READ DATA
                                TRAP C:ERMRD
                                .WORD 113
                                .WORD T25WDE
                                .WORD EXPREC
4358 024642 104456
4359 024644 000161
4360 024646 030377
4361 024650 016360

```

```

4358 024652      758:  CKLOOP                      ;LOOP IF SELECTED
      024652 104406                                TRAP  C8CLP1
4359 024654
4360
4361 ;*****
4362 ;
4363 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4364 ;
4365 ;*****
4366
4367 024654 013701 030136      MOV  T25BFR+6,R1      ;QUICK CHECK FOR BOT SET
4368 024660 010102      MOV  R1,R2      ;SET UP EXPECTED
4369 024662 042702 000002      BIC  #BIT1,R2      ;CLEAR THE BOT BIT (XSTO)
4370 024666 020102      CMP  R1,R2      ;IS THE EOT BIT SET IN XSTO
4371 024670 001406      BEQ  1258      ;BR, IF SET (GOOD)
4372 024672 004737 020116      JSR  PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4376 024676      ERRHRD  ERRNO,T25BNC,EXPREC      ;BOT NOT CLEARED AFTER SPACE FROM BOT
      024676 104456                                TRAP  C8ERHRD
      024700 000162                                .WORD  114
      024702 030752                                .WORD  T25BNC
      024704 016360                                .WORD  EXPREC
4377 024706 004737 031634      1258:  JSR  PC,T25RT3      ;CLEAN UP PACKET
4378 024712 012737 000401 030246      MOV  #257.,T25SZ      ;SET THE CORRECT SIZE UP
4379
4380 ;*****
4381 ;
4382 ;READ DATA COMMAND IN PLACE
4383 ;
4384 ;*****
4385
4386 024720 012737 140001 030240      MOV  #140001,T25PK3      ;READ DATA COMMAND IN PLACE
4387 024726 013737 003072 030242      MOV  FREE,T25R8      ;READ BUFFER ADDRESS TO PACKET
4388 024734 012704 030240      MOV  #T25PK3,R4      ;R4 = POINTER TO PACKET
4389 024740 010465 177776      MOV  R4,TSD8(R5)      ;ISSUE COMMAND
4390 024744 004737 017134      JSR  PC,WAITF      ;WAIT FOR SSR TO SET
4391 024750 016501 000000      MOV  TSSR(R5),R1      ;GET TSSR CONTENTS
4392 024754 012702 000200      MOV  #SSR,R2      ;SET UP EXPECTED
4393 024760 020102      CMP  R1,R2      ;ARE THEY EQUAL
4394 024762 001406      BEQ  1908      ;BR, IF OK ESP. FUNCTION REJECT
4395 024764 004737 020116      JSR  PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4399 024770      ERRHRD  ERRNO,RDERR,PKTSSR      ;TSSR INCORRECT AFTER READ DATA CMD
      024770 104456                                TRAP  C8ERHRD
      024772 000163                                .WORD  115
      024774 005104                                .WORD  RDERR
      024776 011710                                .WORD  PKTSSR
4400 025000      1908:  CKLOOP                      ;LOOP IF SELECTED
      025000 104406                                TRAP  C8CLP1
4401 025002 017701 156064      MOV  #FREE,R1      ;GET FIRST WORD FROM BUFFER
4402 025006 012702 000001      MOV  #1,R2      ;SET UP EXPECTED
4403 025012 020102      CMP  R1,R2      ;WAS RECORD NUMBERED 1
4404 025014 001406      BEQ  2008      ;BR, IF CORRECT RECORD
4405 025016 004737 020116      JSR  PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4409 025022      ERRHRD  ERRNO,T25WNG,EXPREC      ;SHOULD HAVE BEEN RECORD NUMBER 1
      025022 104456                                TRAP  C8ERHRD
      025024 000164                                .WORD  116
      025026 030667                                .WORD  T25WNG
      025030 016360                                .WORD  EXPREC

```

CZTKGA TK 25 FRI END FUNC 03
TEST 1: SPACE RECORDS

MACRO M1200 20 APR 84 08:13 PAGE 88 3

TEST 1: SPACE RECORDS

4410 025032

2008:

4411 025032

ENDSUB

```

;,,,,,,,,, END SUBTEST ,,,,,,,,,

```

025032

L10040:

025032 104403

TRAP

CBSUB

4412 025034 023727 002170 000031

CMP FATFLG. 025.

IS ERROR COUNT AT 25

4413 025042 002402

BLT 9998

BR. IF LESS THAN 25

4414 025044 004737 020170

JSR PC, CKDROP

TRY TO DROP THE UNIT

4415 025050

9991:

```

4417      ;*
4418      ;
4419      ;TEST 1. SUBTEST 3
4420      ;
4421      ;VERIFIES THAT SPACE RECORDS REVERSE CAN SPACE BACK
4422      ;OVER THE FIRST RECORD ON TAPE.
4423      ;
4424      ;
4425      ;-
4426
4427      025050          BGNSUB                      ,>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
4428      025050                                     T1.3:
4429      025050 104402                                     TRAP    C$BSUB
4430      025052 004737 031500        JSR     PC,T25REST       ;SET COMMAND PACKET
4431      025056 004737 031572        JSR     PC,T25RT2         ;SET UP OTHER COMMAND PACKET
4432      025062 004737 031634        JSR     PC,T25RT3         ;SET UP OTHER COMMAND PACKET
4433
4434      ;*****
4435      ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
4436      ;
4437      ;*****
4438      025066 004737 016660        JSR     PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
4439      025072 103407                BCS     20$              ;BR IF INIT WAS OK
4440      025074 004737 020116        JSR     PC,FATCHK         ;INC AND CHECK FOR MORE THAN 25 ERRORS
4441      025100 010001                MOV     R0,R1             ;CONTENTS OF TSSR REGISTER
4442      025102                                ERRDF   ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
4443      025102 104455                                     TRAP    C$ERDF
4444      025104 000165                                     .WORD   117
4445      025106 003550                                     .WORD   SFIERR
4446      025110 011676                                     .WORD   SFIMSG
4447      025112
4448      20$:
4449      025112 012704 030110        MOV     @T25PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
4450
4451      ;*****
4452      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPCH)
4453      ;
4454      ;*****
4455
4456      025116 004737 010342        JSR     PC,WRTPCH          ;ISSUE WRITE CHARACTERISTICS
4457      025122 103407                BCS     25$              ;BR, IF COMMAND ISSUED OK
4458      025124 004737 020116        JSR     PC,FATCHK         ;INC AND CHECK FOR MORE THAN 25 ERRORS
4459      025130 010001                MOV     R0,R1             ;SAVE CONTENTS OF TSSR
4460      025132                                ERHWD   ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
4461      025132 104456                                     TRAP    C$ERHWD
4462      025134 000166                                     .WORD   118
4463      025136 004754                                     .WORD   WRTMSG
4464      025140 011676                                     .WORD   SFIMSG
4465      025142
4466      25$: CKLOOP                                  ;LOOP IF SELECTED
4467      025142 104406                                     TRAP    C$CLP1
4468
4469      ;*****
4470      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE

```

```

4469
4470
4471
4472 025144 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
4473 025150 103407      BCS      30$      ;BR, IF NO PROBLEM
4474 025152 010001      MOV      R0,R1      ;SAVE TSSR
4475 025154 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4479 025160      ERRHRD  ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD     119
                                .WORD     T25RWN
                                .WORD     PKTSSR
                                TRAP      C$CLP1
4480 025170      30$:      CKLOOP      ;LOOP IF SELECTED
4481 025170 104406
4482
4483
4484 ;*****
4485 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4486 ;*****
4487
4488 025172 013701 030136      MOV      T25BFR+6,R1      ;PICK UP XSTO
4489 025176 010102      MOV      R1,R2      ;SET UP EXPECTED
4490 025200 052702 000002      BIS      @BIT1,R2      ;SET BOT BIT IN EXPECTED
4491 025204 020102      CMP      R1,R2      ;DOES EXP = REC'D
4492 025206 001406      BEQ      40$      ;BR, IF EQUAL (OK)
4493 025210 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4497 025214      ERRHRD  ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD     120
                                .WORD     T25BOT
                                .WORD     EXPREC
                                TRAP      C$CLP1
4498 025224      40$:      CKLOOP      ;LOOP IF SELECTED
4499 025224 104406
4500
4501 ;*****
4502 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
4503 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
4504 ;*****
4505
4506
4507 025226 012703 000001      MOV      @000001,R3      ;NUMBER OF RECORDS TO SPACE FORWARD
4508 025232 004737 010144      JSR      PC,SPACE      ;CALL SPACE COMMAND
4509 025236 103410      BCS      50$      ;CHECK FOR ERROR
4510 025240 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
4511 025244 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4515 025250      ERRHRD  ERRNO,T25WDE,SFFMSG      ;SPACE FORWARD FAILED
                                TRAP      C$ERHRD
                                .WORD     121
                                .WORD     T25WDE
                                .WORD     SFFMSG
                                TRAP      C$CLP1
4516 025260      50$:      CKLOOP      ;LOOP IF SELECTED
4517 025260 104406      MOV      @1,T25RB      ;NUMBER OF RECORDS TO SPACE OVER
4518 025262 012737 000001 030242
4519 ;*****

```

```

4520
4521      ; SPACE REVERSE,ACK,CVC=1 COMMAND
4522      ;
4523      ; *****
4524
4525 025270 012737 140410 030240      MOV      #140410,T25PK3      ; SPACE REVERSE,ACK,CVC=1 COMMAND
4526 025276 012704 030240      MOV      #T25PK3,R4      ; SET UP R4 WITH PACKET ADDRESS
4527 025302
4528 025302 010465 177776      65$:      MOV      R4,TSDB(R5)      ; ISSUE COMMAND
4529 025306 004737 017134      JSR      PC,WAITF      ; WAIT FOR SSR TO SET
4530 025312 016501 000000      MOV      TSSR(R5),R1      ; GET TSSR CONTENTS
4531 025316 012702 000200      MOV      #SSR,R2      ; SET UP EXPECTED
4532 025322 020102      CMP      R1,R2      ; ARE THEY EQUAL
4533 025324 001406      BEQ      75$      ; BR, IF OK
4534 025326 004737 020116      JSR      PC,FATCHK      ; INC AND CHECK FOR MORE THAN 25 ERRORS
4538 025332      ERRHRD      ERRNO,T25WDE,PKTSSR      ; TSSR INCORRECT AFTER READ DATA
      025332 104456      TRAP      C$ERHRD
      025334 000172      .WORD      122
      025336 030377      .WORD      T25WDE
      025340 011710      .WORD      PKTSSR
4539 025342      75$:      CKLOOP      ; LOOP IF SELECTED
      025342 104406      TRAP      C$CLP1
4540 025344
4541 025344 012703 000400      120$:      MOV      #256.,R3      ; RECORD SIZE
4542 025350 013737 003072 030242      MOV      FREE,T25R8      ; STARTING READ BUFFER ADDRESS
4543
4544      ; *****
4545      ;
4546      ; READ DATA,ACK,CVC=1 COMMAND
4547      ;
4548      ; *****
4549
4550 025356 012737 140001 030240      MOV      #140001,T25PK3      ; READ DATA,ACK,CVC=1 COMMAND
4551 025364 012704 030240      165$:      MOV      #T25PK3,R4      ; SET UP R4 WITH PACKET ADDRESS
4552 025370 010337 030246      MOV      R3,T25S2      ; SET UP RECORD SIZE IN PACKET
4553 025374 010465 177776      MOV      R4,TSDB(R5)      ; ISSUE COMMAND
4554 025400 004737 017134      JSR      PC,WAITF      ; WAIT FOR SSR TO SET
4555 025404 016501 000000      MOV      TSSR(R5),R1      ; GET TSSR CONTENTS
4556 025410 012702 000200      MOV      #SSR,R2      ; SET UP EXPECTED
4557 025414 020102      CMP      R1,R2      ; ARE THEY EQUAL
4558 025416 001406      BEQ      170$      ; BR, IF OK
4559 025420 004737 020116      JSR      PC,FATCHK      ; INC AND CHECK FOR MORE THAN 25 ERRORS
4563 025424      ERRHRD      ERRNO,RDERR,PKTSSR      ; TSSR INCORRECT AFTER READ DATA
      025424 104456      TRAP      C$ERHRD
      025426 000173      .WORD      123
      025430 005104      .WORD      RDERR
      025432 011710      .WORD      PKTSSR
4564 025434      170$:      CKLOOP      ; LOOP IF SELECTED
      025434 104406      TRAP      C$CLP1
4565 025436 017701 155430      MOV      @FREE,R1      ; GET FIRST WORD FROM BUFFER
4566 025442 012702 000000      MOV      #0,R2      ; SET UP EXPECTED
4567 025446 020102      CMP      R1,R2      ; WAS RECORD NUMBERED 1
4568 025450 001406      BEQ      200$      ; BR, IF CORRECT RECORD
4569 025452 004737 020116      JSR      PC,FATCHK      ; INC AND CHECK FOR MORE THAN 25 ERRORS
4573 025456      ERRHRD      ERRNO,T25WNG,EXPREC      ; SHOULD HAVE BEEN RECORD NUMBER 1
      025456 104456      TRAP      C$ERHRD
      025460 000174      .WORD      124

```

CZTKGA TK 25 FRT END FUNC #3 MACRO M1200 20-APR-84 08:13 PAGE 89-3
TEST 1: SPACE RECORDS

	025462	030667					
	025464	016360					
4574	025466				2001:		
4575	025466					ENDSUB	
	025466						
	025466	104403					
4576	025470	023727	002170	000031		CMP	FATFLG.#25.
4577	025476	002402				BLT	9991
4578	025500	004737	020170			JSR	PC,CKDROP
4579	025504				9991:		

```
.WORD      T25WNG  
.WORD      EXPREC
```

; >>>>>> END SUBTEST >>>'>>>>>
 L10041:

```
              TRAP           C$ESUB  
    ;IS ERROR COUNT AT 25  
    ;BR. IF LESS THAN 25  
    ;TRY TO DROP THE UNIT
```

```

4581      ;+
4582      ;
4583      ;TEST 1, SUBTEST 4
4584      ;
4585      ;VERIFIES THAT SPACE RECORDS FORWARD CAN SPACE A
4586      ;MULTIPLE NUMBER OF RECORDS (2 THROUGH 64K OR THE
4587      ;MAXIMUM NUMBER OF RECORDS WRITTEN ON THE TAPE,
4588      ;WHICH EVER IS LESS.
4589      ;
4590      ;
4591      ;
4592      ;-
4593
4594 025504          BGNSUB                                ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
025504                                     T1.4:
025504 104402                                         TRAP    C$BSUB
4595 025506 004737 031500        JSR      PC,T25REST        ;SET COMMAND PACKET
4596 025512 004737 031572        JSR      PC,T25RT2         ;SET UP OTHER COMMAND PACKET
4597 025516 004737 031634        JSR      PC,T25RT3         ;SET UP OTHER COMMAND PACKET
4598 025522 013737 030270 030266   MOV      T25CNT,T25CN2     ;HOLD NUMBER OF RECORDS
4599 025530 162737 000002 030266   SUB      #2,T25CN2        ;ACTUAL NUMBER OF RECORDS ON TAPE
4600 025536 013737 030270 030272   MOV      T25CNT,T25DLY     ;SET UP REWIND DELAY COUNTER
4601
4602      ;*****
4603      ;
4604      ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
4605      ;
4606      ;*****
4607
4608 025544 004737 016660 10$:      JSR      PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
4609 025550 103427                BCS      20$              ;BR IF INIT WAS OK
4610 025552                DELAY      250                    ;WAIT ABOUT .25 SECONDS
025552 012727 000250                MOV      #250,(PC)+
025556 000000                .WORD      0
025560 013727 002116                MOV      L$DLY,(PC)+
025564 000000                .WORD      0
025566 005367 177772                DEC      -6(PC)
025572 001375                BNE      -.4
025574 005367 177756                DEC      -22(PC)
025600 001367                BNE      .-20
4611 025602 005337 030272        DEC      T25DLY            ;DEC COUNTER
4612 025606 001356                BNE      10$              ;BR, IF MORE LOOPS REQUIRED
4613 025610 016501 000000        MOV      TSSR(R5),R1        ;CONTENTS OF TSSR REGISTER
4614 025614 004737 020116        JSR      PC,FATCHK         ;INC AND CHECK FOR MORE THAN 25 ERRORS
4618 025620                ERRDF   ERNO,SFIERR,SFIMSG        ;FATAL ERROR TSSR WAS NOT OK
025620 104455                TRAP      C$ERDF
025622 000175                .WORD      125
025624 003550                .WORD      SFIERR
025626 011676                .WORD      SFIMSG
4619 025630 20$:
4620
4621 025630 012704 030110        MOV      #T25PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
4622
4623      ;*****
4624      ;
4625      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4626      ;

```

```
4627
4628
4629 025634 004737 010342      JSR      PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
4630 025640 103407      BCS      254      ;BR, IF COMMAND ISSUED OK
4631 025642 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4635 025646 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
4636 025650      ERRMRD  ERRNO,WRTMSG,SFMSG      ;WRITE CHARACTERISTICS FAILED
                                TRAP      C1ERMRD
                                .WORD     126
                                .WORD     WRTMSG
                                .WORD     SFMSG
                                025650 104456
                                025652 000176
                                025654 004754
                                025656 011676
4637 025660      254:      CKLOOP      ;LOOP IF SELECTED
                                TRAP      C1CLP1
                                025660 104406
4638
4639
4640
4641
4642
4643
4644
4645 025662 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
4646 025666 103407      BCS      304      ;BR, IF NO PROBLEM
4647 025670 010001      MOV      R0,R1      ;SAVE TSSR
4648 025672 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4652 025676      ERRMRD  ERRNO,T25RWN,PKTSSR      ;REWIND NOT ACCEPTED
                                TRAP      C1ERMRD
                                .WORD     127
                                .WORD     T25RWN
                                .WORD     PKTSSR
                                025676 104456
                                025700 000177
                                025702 031267
                                025704 011710
4653 025706      304:      CKLOOP      ;LOOP IF SELECTED
                                TRAP      C1CLP1
                                025706 104406
4654 025710 013701 030266      MOV      T25CN2,R1      ;NUMBER OF RECORDS ON TAPE
4655 025714 012702 177776      MOV      #65534.,R2      ;MAX IT CAN SPACE OVER
4656 025720 020201      CMP      R2,R1      ;WHICH VALUE CAN WE USE
4657 025722 003002      BGT      464      ;BR, IF # WRITTEN > 64K
4658 025724 010103      MOV      R1,R3      ;# WRITTEN CAN BE USED
4659 025726 000401      BR      474      ;MOVE ON
4660 025730 010203      464:      MOV      R2,R3      ;USE MAX NUMBER
4661 025732 162703 000001      474:      SUB      #1,R3      ;DON'T GO ALL THE WAY YET
4662 025736 010337 030242      MOV      R3,T25RB      ;NUMBER OF RECORDS TO SPACE OVER
4663
4664
4665
4666
4667
4668
4669
4670 025742 012737 140010 030240      MOV      #140010,T25PK3      ;SPACE FORWARD,ACK,CVC=1 COMMAND
4671 025750 012704 030240      MOV      #T25PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
4672 025754
4673 025754 013737 030266 030272      654:      MOV      T25CN2,T25DLY      ;NUMBER OF RECORDS USED AS DELAY COUNTER
4674 025762 010465 177776      MOV      R4,T5D8(R5)      ;ISSUE COMMAND
4675 025766 004737 017134      674:      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
4676 025772 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
4677 025776 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
4678 026002 020102      CMP      R1,R2      ;ARE THEY EQUAL
4679 026004 001425      BEQ      754      ;BR, IF OK
```

```

4680 026006          DELAY 250          ;DELAY .25 SECONDS
      026006 012727 000250              MOV    #250,(PC).
      026012 000000                      .WORD 0
      026014 013727 002116              MOV    L$DLY,(PC).
      026020 000000                      .WORD 0
      026022 005367 177772              DEC     -6(PC)
      026026 001375                      BNE     -4
      026030 005367 177756              DEC     -22(PC)
      026034 001367                      BNE     -20
4681 026036 005337 030272              DEC     T25DLY          ;BUMP DOWN COUNTER
4682 026042 001351                      BNE     67$            ;BR, IF NOT AT END OF DELAY
4683 026044 004737 020116              JSR     PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
4687 026050                      ERRMRD  ERRNO,T25WDE,PKTSSR    ;TSSR INCORRECT AFTER READ DATA
      026050 104456                      TRAP    C$ERRMRD
      026052 000200                      .WORD 128
      026054 030377                      .WORD T25WDE
      026056 011710                      .WORD PKTSSR
4688 026060          75$: CKLOOP          ;LOOP IF SELECTED
      026060 104406                      TRAP    C$CLP1
4689 026062 012703 010000              MOV     #4096.,R3          ;RECORD SIZE
4690 026066 013737 003072 030242      MOV     FREE,T25RB        ;STARTING READ BUFFER ADDRESS
4691
4692 ;*****
4693 ;
4694 ;READ DATA,ACK COMMAND
4695 ;
4696 ;*****
4697
4698 026074 012737 100001 030240      165$: MOV     #100001,T25PK3      ;READ DATA,ACK COMMAND
4699 026102 012704 030240              MOV     #T25PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
4700 026106 010337 030246              MOV     R3,T25SZ        ;SET UP RECORD SIZE IN PACKET
4701 026112 010465 177776              MOV     R4,TSD8(R5)      ;ISSUE COMMAND
4702 026116 004737 017134              JSR     PC,WAITF        ;WAIT FOR SSR TO SET
4703 026122 016501 000000              MOV     TSSR(R5),R1    ;GET TSSR CONTENTS
4704 026126 012702 000200              MOV     #SSR,R2        ;SET UP EXPECTED
4705 026132 020102                      CMP     R1,R2          ;ARE THEY EQUAL
4706 026134 001411                      BEQ     170$          ;BR, IF OK
4707 026136 032701 000004              BIT     #BIT2,R1        ;CHECK FOR TAPE STATUS ALERT
4708 026142 001006                      BNE     170$          ;IF SET ALL IS WELL
4709 026144 004737 020116              JSR     PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
4713 026150                      ERRMRD  ERRNO,RDERR,PKTSSR    ;TSSR INCORRECT AFTER READ DATA
      026150 104456                      TRAP    C$ERRMRD
      026152 000201                      .WORD 129
      026154 005104                      .WORD RDERR
      026156 011710                      .WORD PKTSSR
4714 026160          170$: CKLOOP        ;LOOP IF SELECTED
      026160 104406                      TRAP    C$CLP1
4715 026162 017701 154704              MOV     #FREE,R1          ;GET FIRST WORD FROM BUFFER
4716 026166 013702 030266              MOV     T25CN2,R2        ;SET UP EXPECTED
4717 026172 162702 000001              SUB     #1,R2          ;SHOULD BE LAST RECORD
4718 026176 020102                      CMP     R1,R2          ;WAS RECORD NUMBERED R3
4719 026200 001406                      BEQ     200$          ;BR, IF CORRECT RECORD
4720 026202 004737 020116              JSR     PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
4724 026206                      ERRMRD  ERRNO,T25WNG,EXPREC    ;SHOULD HAVE BEEN RECORD NUMBER 1
      026206 104456                      TRAP    C$ERRMRD
      026210 000202                      .WORD 130
      026212 030667                      .WORD T25WNG

```

CZIKGA, TK-25 ERI END EUNC 03
TEST 1: SPACE RECORDS

MACRO M1200 20 APR-84 08:13 PAGE 90 3

026214 016360

4725 026216

2008:

ENDSUB

4726 026216
026216

026216 104403

4727 026220 023727 002170 000031

CMP FATFLG, 025.

4728 026226 002402
4729 026230 004737

BLT 9998
ISB BC CH

4730 026234

9991:

JSR PC,CKDROP

```
.WORD    EXPREC
```

1>>>>>>>>>> END SUBTEST >>>>>>>>>>

L10042:

TRAP

CIESUB

THIS ERROR COUNT AT 25

BR, IF LESS THAN 25
TRY TO DROP THE BALL

TRY TO DROP THE UNIT

```

4733      ; *
4734      ;
4735      ; TEST 1, SUBTEST 5
4736      ;
4737      ; VERIFIES THAT SPACE RECORDS REVERSE CAN SPACE A
4738      ; MULTIPLE NUMBER OF RECORDS (2 THROUGH 64K OR THE
4739      ; MAXIMUM NUMBER OF RECORDS WRITTEN ON THE TAPE
4740      ; WHICH EVER IS LESS.
4741      ;
4742      ;
4743      ;
4744      ; -
4745      026234      BGNSUB                                ; >>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>
4746      026234      104402                                T1.5:
4747      026236      004737      031500                      JSR          PC,T25REST                ;SET COMMAND PACKET
4748      026242      013737      030270      030266          MOV          T25CNT,T25CN2            ;SET UP RECORD COUNTER
4749      026250      004737      031572                      JSR          PC,T25RT2              ;SET UP OTHER COMMAND PACKET
4750      026254      004737      031634                      JSR          PC,T25RT3              ;SET UP OTHER COMMAND PACKET
4751      026260      013737      030270      030272          MOV          T25CNT,T25DLY            ;SET UP REWIND DELAY COUNTER
4752      ; *****
4753      ;
4754      ; ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
4755      ;
4756      ; *****
4757      4758      026266      004737      016660          10$:    JSR          PC,SUFINIT             ;DO INITIALIZE ON CONTROLLER
4759      026272      103427                                     BCS          20$                  ;BR IF INIT WAS OK
4760      026274      012727      000250                     DELAY          250                 ;WAIT ABOUT .25 SECONDS
4761      026300      000000                                     MOV          #250,(PC).
4762      026302      013727      002116                                     .WORD        0
4763      026306      000000                                     MOV          L$DLY,(PC).
4764      026310      005367      177772                                     .WORD        0
4765      026314      001375                                     DEC          -6(PC)
4766      026316      005367      177756                                     BNE          . 4
4767      026322      001367                                     DEC          -22(PC)
4768      026324      005337      030272                                     BNE          .-20
4769      026330      001356                      DEC          T25DLY                ;DEC COUNTER
4770      026332      016501      000000                      BNE          10$                 ;BR, IF MORE LOOPS REQUIRED
4771      026336      004737      020116                      MOV          TSSR(R5),R1         ;CONTENTS OF TSSR REGISTER
4772      026342      104455                      JSR          PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
4773      026344      000203                      ERDF          ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
4774      026346      003550                                     TRAP          C$ERDF
4775      026350      011676                                     .WORD        131
4776      026352                                     .WORD        SFIERR
4777      20$:                                               .WORD        SFIMSG
4778      026352      012704      030110                      MOV          @T25PACKET,R4       ;SUBROUTINE NEEDS PACKET ADDRESS
4779      ; *****
4780      ;
4781      ; WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4782      ;
4783      ; *****

```

```

4779 026356 004737 010342      JSR      PC,WRCHMR      ;ISSUE WRITE CHARACTERISTICS
4780 026362 103407      BCS      25$      ;BR, IF COMMAND ISSUED OK
4781 026364 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4785 026370 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
4786 026372      ERRMRD  ERRNO,WRMSG,SFMSG      ;WRITE CHARACTERISTICS FAILED
                                TRAP      C$ERRMRD
                                .WORD     132
                                .WORD     WRMSG
                                .WORD     SFMSG
                                TRAP      C$CLP1
4787 026402      25$: CKLOOP      ;LOOP IF SELECTED
                                TRAP      C$CLP1
4788
4789      ;*****
4790      ;
4791      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4792      ;
4793      ;*****
4794
4795 026404 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
4796 026410 103407      BCS      30$      ;BR, IF NO PROBLEM
4797 026412 010001      MOV      R0,R1      ;SAVE TSSR
4798 026414 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4802 026420      ERRMRD  ERRNO,T25RWN,PKTSSR      ;REWIND NOT ACCEPTED
                                TRAP      C$ERRMRD
                                .WORD     133
                                .WORD     T25RWN
                                .WORD     PKTSSR
                                TRAP      C$CLP1
4803 026430      30$: CKLOOP      ;LOOP IF SELECTED
                                TRAP      C$CLP1
4804
4805      ;*****
4806      ;
4807      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4808      ;
4809      ;*****
4810
4811 026432 013701 030136      MOV      T25BFR+6,R1      ;PICK UP XSTO
4812 026436 010102      MOV      R1,R2      ;SET UP EXPECTED
4813 026440 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
4814 026444 020102      CMP      R1,R2      ;DOES EXP = REC'D
4815 026446 001406      BEQ      40$      ;BR, IF EQUAL (OK)
4816 026450 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4820 026454      ERRMRD  ERRNO,T25BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERRMRD
                                .WORD     134
                                .WORD     T25BOT
                                .WORD     EXPREC
                                TRAP      C$CLP1
4821 026464      40$: CKLOOP      ;LOOP IF SELECTED
                                TRAP      C$CLP1
4822 026466 013701 030266      MOV      T25CN2,R1      ;NUMBER OF RECORDS ON TAPE
4823 026472 012702 177776      MOV      #65534.,R2      ;MAX IT CAN SPACE OVER
4824 026476 020201      CMP      R2,R1      ;WHICH VALUE CAN WE USE
4825 026500 003002      BGT      46$      ;BR, IF # WRITTEN > 64K
4826 026502 010103      MOV      R1,R3      ;# WRITTEN CAN BE USED
4827 026504 000401      BR      47$      ;MOVE ON
4828 026506 010203      46$: MOV      R2,R3      ;USE MAX NUMBER
4829 026510      47$:

```

```

4830 026510 010337 030242          MOV      R3,T25RB          ;NUMBER OF RECORDS TO SPACE OVER
4831
4832          ;*****
4833          ;
4834          ;SPACE FORWARD,ACK,CVC=1 COMMAND
4835          ;
4836          ;*****
4837
4838 026514 012737 140010 030240      MOV      #140010,T25PK3      ;SPACE FORWARD,ACK,CVC=1 COMMAND
4839 026522 012704 030240              MOV      #T25PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
4840 026526 010465 177776              MOV      R4,TSD8(R5)      ;ISSUE COMMAND
4841 026532 013737 030266 030272      MOV      T25CN2,T25DLY      ;SET UP DELAY COUNTER
4842 026540 004737 017134 484:      JSR      PC,WAITF          ;WAIT FOR SSR TO SET
4843 026544 016501 000000              MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
4844 026550 012702 000200              MOV      #SSR,R2          ;SET UP EXPECTED
4845 026554 020102                      CMP      R1,R2          ;ARE THEY EQUAL
4846 026556 001425                      BEQ      50$          ;BR, IF OK
4847 026560                      DELAY      250          ;WAIT .25 SECONDS
                                MOV      #250,(PC)+
                                .WORD    0
                                MOV      L#DLY,(PC)+
                                .WORD    0
                                DEC      -6(PC)
                                BNE      -4
                                DEC      -22(PC)
                                BNE      -20
4848 026610 005337 030272      DEC      T25DLY          ;DEC THE DELAY COUNTER
4849 026614 001351 020116      BNE      48$          ;BR, IF COUNTER HASN'T EXPIRED
4850 026616 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4854 026622                      ERRMRD      ERRNO,T25WDE,EXPREC ;TSSR INCORRECT AFTER READ DATA
                                TRAP      C#ERRMRD
                                .WORD    135
                                .WORD    T25WDE
                                .WORD    EXPREC
4855 026632 013701 030266 50$:      MOV      T25CN2,R1          ;NUMBER OF RECORDS ON TAPE
4856 026636 012702 177776              MOV      #65534.,R2      ;MAX IT CAN SPACE OVER
4857 026642 020201                      CMP      R2,R1          ;WHICH VALUE CAN WE USE
4858 026644 003002                      BGT      55$          ;BR, IF # WRITTEN > 64K
4859 026646 010103                      MOV      R1,R3          ;# WRITTEN CAN BE USED
4860 026650 000401                      BR      60$          ;MOVE ON
4861 026652 010203 55$:      MOV      R2,R3          ;USE MAX NUMBER
4862 026654 162703 000001 60$:      SUB      #1,R3          ;DON'T GO ALL THE WAY YET
4863 026660 010337 030242      MOV      R3,T25RB          ;NUMBER OF RECORDS TO SPACE OVER
4864
4865          ;*****
4866          ;
4867          ;SPACE REVERSE,ACK,CVC=1 COMMAND
4868          ;
4869          ;*****
4870
4871 026664 012737 140410 030240      MOV      #140410,T25PK3      ;SPACE REVERSE,ACK,CVC=1 COMMAND
4872 026672 012704 030240              MOV      #T25PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
4873 026676 010465 177776              MOV      R4,TSD8(R5)      ;ISSUE COMMAND
4874 026702 013737 030266 030272      MOV      T25CN2,T25DLY      ;SET UP COUNTER
4875 026710 004737 017134 70$:      JSR      PC,WAITF          ;WAIT FOR SSR TO SET
4876 026714 016501 000000              MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
4877 026720 012702 000200              MOV      #SSR,R2          ;SET UP EXPECTED

```

```

4878 026724 020102      CMP      R1,R2      ;ARE THEY EQUAL
4879 026726 001425      BEQ       75$      ;BR, IF OK
4880 026730      DELAY    250      ;WAIT ABOUT .25 SECONDS
      026730 012727 000250      MOV      #250,(PC)+
      026734 000000      .WORD    0
      026736 013727 002116      MOV      L$DLY,(PC)+
      026742 000000      .WORD    0
      026744 005367 177772      DEC      6(PC)
      026750 001375      BNE      .-4
      026752 005367 177756      DEC      -22(PC)
      026756 001367      BNE      .-20
4881 026760 005337 030272      DEC      T25DLY      ;BUMP COUNTER DOWN
4882 026764 001351      BNE      70$      ;BR, IF COUNTER HASN'T EXPIRED
4883 026766 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4887 026772      ERRHRD  ERRNO,T25WDE,EXPREC      ;TSSR INCORRECT AFTER READ DATA
      026772 104456      TRAP      C$ERHRD
      026774 000210      .WORD    136
      026776 030377      .WORD    T25WDE
      027000 016360      .WORD    EXPREC
4888 027002      75$: CKLOOP      ;LOOP IF SELECTED
      027002 104406      TRAP      C$CLP1
4889 027004 012703 010000      MOV      #4096,R3      ;RECORD SIZE
4890 027010 013737 003072 030242      MOV      FREE,T25RB      ;STARTING READ BUFFER ADDRESS
4891
4892      ;*****
4893      ;
4894      ;READ DATA,ACK COMMAND
4895      ;
4896      ;*****
4897
4898 027016 012737 100001 030240      MOV      #100001,T25PK3
4899 027024 012704 030240 165$: MOV      #T25PK3,R4      ;READ DATA,ACK COMMAND
4900 027030 012700 177777      MOV      #177777,R0      ;SET UP R4 WITH PACKET ADDRESS
4901 027034 004737 020410      JSR      PC,FILLMEM      ;SET ALL ONES INTO CORRECT REGISTER
4902 027040 010337 030246      MOV      R3,T25SZ      ;FILL MEMORY WITH RECORD SIZE
4903 027044 010465 177776      MOV      R4,TSD8(R5)      ;SET UP RECORD SIZE IN PACKET
4904 027050 004737 017134      JSR      PC,WAITF      ;ISSUE COMMAND
4905 027054 016501 000000      MOV      TSSR(R5),R1      ;WAIT FOR SSR TO SET
4906 027060 012702 000200      MOV      #SSR,R2      ;GET TSSR CONTENTS
4907 027064 020102      CMP      R1,R2      ;SET UP EXPECTED
4908 027066 001411      BEQ      170$      ;ARE THEY EQUAL
4909 027070 032701 000004      BIT      #BIT2,R1      ;BR, IF OK
4910 027074 001006      BNE      170$      ;CHECK FOR TAPE STATUS ALERT
4911 027076 004737 020116      JSR      PC,FATCHK      ;BR, IF BIT SET
4915 027102      ERRHRD  ERRNO,RDERR,EXPREC      ;INC AND CHECK FOR MORE THAN 25 ERRORS
      027102 104456      ;TSSR INCORRECT AFTER READ DATA
      027104 000211      TRAP      C$ERHRD
      027106 005104      .WORD    137
      027110 016360      .WORD    RDERR
      4916 027112      170$: CKLOOP      ;LOOP IF SELECTED
      027112 104406      TRAP      C$CLP1
4917 027114 017701 153752      MOV      $FREE,R1      ;GET FIRST WORD FROM BUFFER
4918 027120 012702 000001      MOV      #1,R2      ;SET UP EXPECTED
4919 027124 020102      CMP      R1,R2      ;WAS RECORD NUMBERED R3
4920 027126 001406      BEQ      200$      ;BR, IF CORRECT RECORD
4921 027130 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4925 027134      ERRHRD  ERRNO,T25WNH,EXPREC      ;SHOULD HAVE BEEN RECORD NUMBER 1

```

```

                                TRAP          C$ERHRD
                                .WORD          138
                                .WORD          T25,NH
                                .WORD          EXPREC
;,,,,,,,,,,,,, END SUBTEST ,,,,,,,,,,,,,,
                                L10043:
                                TRAP          C$ESUB
;IS ERROR COUNT AT 25
;BR, IF LESS THAN 25
;TRY TO DROP THE UNIT
```

```

4933      ;+
4934      ;
4935      ;TEST 1, SUBTEST 6
4936      ;
4937      ;VERIFIES THAT SPACE RECORDS REVERSE ISSUED WHILE THE
4938      ;TAPE IS AT BOT RESULTS IN FUNCTION REJECT TERMINATION
4939      ;WITH THE NONEXECUTABLE FUNCTION (NEF) ERROR BIT SET.
4940      ;
4941      ;
4942      ;
4943      ;-
4944      027162          BGNSUB                                ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
4945      027162                                     T1.6:
4946      027162    104402                                         TRAP      C$BSUB
4947      027164    004737    031500        JSR      PC,T25REST        ;SET COMMAND PACKET
4948      027170    004737    031572        JSR      PC,T25RT2         ;SET UP OTHER COMMAND PACKET
4949      027174    004737    031634        JSR      PC,T25RT3         ;SET UP OTHER COMMAND PACKET
4950      ;*****
4951      ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
4952      ;
4953      ;*****
4954      027200    004737    016660        JSR      PC,SOFI'IT         ;DO INITIALIZE ON CONTROLLER
4955      027204    103407                BCS      20$               ;BR IF INIT WAS OK
4956      027206    004737    020116        JSR      PC,FATCHK         ;INC AND CHECK FOR MORE THAN 25 ERRORS
4957      027212    010001                MOV      R0,R1             ;CONTENTS OF TSSR REGISTER
4958      027214                ERDFF   ERNO,SFIERR,SFIMSG           ;FATAL ERROR TSSR WAS NOT OK
4959      027214    104455                                         TRAP      C$ERDF
4960      027216    000213                                         .WORD     139
4961      027220    003550                                         .WORD     SFIERR
4962      027222    011676                                         .WORD     SFIMSG
4963      027224
4964      20$:
4965      027224    012704    030110        MOV      @T25PACKET,R4       ;SUBROUTINE NEEDS PACKET ADDRESS
4966      ;*****
4967      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4968      ;
4969      ;*****
4970      027230    004737    010342        JSR      PC,WRTCHR         ;ISSUE WRITE CHARACTERISTICS
4971      027234    103407                BCS      25$               ;BR, IF COMMAND ISSUED OK
4972      027236    004737    020116        JSR      PC,FATCHK         ;INC AND CHECK FOR MORE THAN 25 ERRORS
4973      027242    010001                MOV      R0,R1             ;SAVE CONTENTS OF TSSR
4974      027244                ERHRD   ERNO,WRTMSG,SFIMSG           ;WRITE CHARACTERISTICSC FAILED
4975      027244    104456                                         TRAP      C$ERHRD
4976      027246    000214                                         .WORD     140
4977      027250    004754                                         .WORD     WRTMSG
4978      027252    011676                                         .WORD     SFIMSG
4979      027254
4980      25$:    CKLOOP                                ;LOOP IF SELECTED
4981      027254    104406                                         TRAP      C$CLP1
4982      ;*****
4983      ;
4984      ;

```

```

4985 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4986 ;
4987 ;*****
4988
4989 027256 004737 010444 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
4990 027262 103407 BCS 30$ ;BR, IF NO PROBLEM
4991 027264 010001 MOV R0,R1 ;SAVE TSSR
4992 027266 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4996 027272 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
      027272 104456 TRAP C$ERHRD
      027274 000215 .WORD 141
      027276 031267 .WORD T25RWN
      027300 011710 .WORD PKTSSR
4997 027302 30$: CKLOOP ;LOOP IF SELECTED
      027302 104406 TRAP C$CLP1
4998
4999 ;*****
5000 ;
5001 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
5002 ;
5003 ;*****
5004
5005 027304 013701 030136 MOV T25BFR+6,R1 ;PICK UP XST0
5006 027310 010102 MOV R1,R2 ;SET UP EXPECTED
5007 027312 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
5008 027316 020102 CMP R1,R2 ;DOES EXP = REC'D
5009 027320 001406 BEQ 40$ ;BR, IF EQUAL (OK)
5010 027322 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5014 027326 ERRHRD ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      027326 104456 TRAP C$ERHRD
      027330 000216 .WORD 142
      027332 030457 .WORD T25BOT
      027334 016360 .WORD EXPREC
5015 027336 40$: CKLOOP ;LOOP IF SELECTED
      027336 104406 TRAP C$CLP1
5016 027340 012737 000001 030242 MOV #1,T25RB ;NUMBER OF RECORDS TO SPACE OVER
5017
5018 ;*****
5019 ;
5020 ;SPACE REVERSE,ACK COMMAND
5021 ;
5022 ;*****
5023
5024 027346 012737 100410 030240 MOV #100410,T25PK3 ;SPACE REVERSE,ACK COMMAND
5025 027354 012704 030240 MOV #T25PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
5026 027360 65$:
5027 027360 010465 177776 MOV R4,TSDB(R5) ;ISSUE COMMAND
5028 027364 004737 017134 JSR PC,WAITF ;WAIT FOR SSR TO SET
5029 027370 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
5030 027374 012702 100206 MOV #SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
5031 027400 020102 CMP R1,R2 ;ARE THEY EQUAL
5032 027402 001406 BEQ 75$ ;BR, IF OK
5033 027404 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5037 027410 ERRHRD ERRNO,T25WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      027410 104456 TRAP C$ERHRD
      027412 000217 .WORD 143
      027414 030377 .WORD T25WDE

```

```

5038 027416 011710          75$:  CKLOOP .                ;LOOP IF SELECTED      .WORD  PKTSSR
      027420 104406          ;*****
5039                                     ;
5040                                     ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
5041                                     ;*****
5042                                     ;
5043                                     ;
5044                                     ;
5045                                     ;
5046 027422 013701 030136      MOV      T25BFR+6,R1          ;GET XSTO STATUS WORD
5047 027426 010102            MOV      R1,R2                ;SET UP EXPECTED
5048 027430 052702 002000      BIS      @BIT10,R2           ;SET THE NEF BIT
5049 027434 020102            CMP      R1,R2                ;ARE THEY EQ ?
5050 027436 001406            BEQ      170$                 ;BR, IF EQUAL (GOOD)
5051 027440 004737 020116      JSR      PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
5055 027444            ERRMRD  ERRNO,T25NEF,EXPREC          ;NEF SHOULD BE SET
      027444 104456                                     TRAP  C$ERRMRD
      027446 000220                                     .WORD 144
      027450 031125                                     .WORD T25NEF
      027452 016360                                     .WORD EXPREC
5056 027454            170$:
5057 027454            ENDSUB
      027454            L10044:
      027454 104403            TRAP  C$ESUB
5058 027456 023727 002170 000031  CMP      FATFLG,#25.      ;IS ERROR COUNT AT 25
5059 027464 002402            BLT      999$                 ;BR, IF LESS THAN 25
5060 027466 004737 020170      JSR      PC,CKDROP           ;TRY TO DROP THE UNIT
5061 027472            999$:

```



```

5115 ;*****
5116 ;
5117 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5118 ;
5119 ;*****
5120
5121 027566 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
5122 027572 103407          BCS      30$      ;BR, IF NO PROBLEM
5123 027574 010001          MOV      R0,R1      ;SAVE TSSR
5124 027576 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
5128 027602          ERRHRD  ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
5129 027612          30$:      CKLOOP          ;LOOP IF SELECTED
5130 027612 104406          TRAP      C$ERHRD
5131          .WORD      147
5132          .WORD      T25RWN
5133          .WORD      PKTSSR
5134          TRAP      C$CLP1
5135 ;*****
5136 ;
5137 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
5138 ;
5139 ;*****
5140
5141 027614 013701 030136      MOV      T25BFR+6,R1      ;PICK UP XSTO
5142 027620 010102          MOV      R1,R2      ;SET UP EXPECTED
5143 027622 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
5144 027626 020102          CMP      R1,R2      ;DOES EXP = REC'D
5145 027630 001406          BEQ      40$      ;BR, IF EQUAL (OK)
5146 027632 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
5147 027636          ERRHRD  ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
5148          TRAP      C$ERHRD
5149          .WORD      148
5150          .WORD      T25BOT
5151          .WORD      EXPREC
5152 027646 012737 000001 030242 40$:      MOV      #1,T25RB      ;NUMBER OF RECORDS TO SPACE OVER
5153 ;*****
5154 ;
5155 ;SPACE FORWARD,IE,ACK,CVC=1 COMMAND
5156 ;
5157 ;*****
5158
5159 027654 012737 140210 030240      MOV      #140210,T25PK3      ;SPACE FORWARD,IE,ACK,CVC=1 COMMAND
5160 027662 012704 030240          MOV      #T25PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
5161 027666 010465 177776          MOV      R4,TSDR(R5)      ;ISSUE COMMAND
5162 027672 004737 017134          JSR      PC,WAITF      ;WAIT FOR SSR TO SET
5163 027676 016501 000000          MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
5164 027702 012702 000200          MOV      #SSR,R2      ;SET UP EXPECTED
5165 027706 020102          CMP      R1,R2      ;ARE THEY EQUAL
5166 027710 001406          BEQ      75$      ;BR, IF OK
5167 027712 004737 020116          JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
5168 027716          ERRHRD  ERRNO,T25WDE,EXPREC ;TSSR INCORRECT AFTER READ DATA
5169          TRAP      C$ERHRD
5170          .WORD      149
5171          .WORD      T25WDE
5172          .WORD      EXPREC

```

```

5168 027726          754:  CKLOOP          ;LOOP IF SELECTED
      027726 104406          ;NUMBER OF RECORDS TO SPACE OVER
5169 027730 012737 000020 030242      MOV    #20,T25RB          TRAP    C1CLP1
5170                                     ;*****
5171                                     ;SPACE REVERSE,IE,ACK COMMAND
5172                                     ;*****
5173
5174
5175
5176
5177 027736 012737 100610 030240      MOV    #100610,T25PK3      ;SPACE REVERSE,IE,ACK COMMAND
5178 027744 012704 030240      MOV    #T25PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
5179 027750 010465 177776      MOV    R4,TSD8(R5)          ;ISSUE COMMAND
5180 027754 004737 017134      JSR     PC,WAITF          ;WAIT FOR SSR TO SET
5181 027760 016501 000000      MOV    TSSR(R5),R1          ;GET TSSR CONTENTS
5182 027764 012702 100204      MOV    #SSR!BIT2!SC,R2        ;SET UP EXPECTED
5183 027770 020102      CMP     R1,R2          ;ARE THEY EQUAL
5184 027772 001406      BEQ     1754          ;BR, IF OK
5185 027774 004737 020116      JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
5189 030000      ERRMRD  ERRNO,T25WDE,EXPREC          ;TSSR INCORRECT AFTER READ DATA
      030000 104456          TRAP    C1ERMRD
      030002 000226          .WORD  150
      030004 030377          .WORD  T25WDE
      030006 016360          .WORD  EXPREC
5190 030010          1754:  CKLOOP          ;LOOP IF SELECTED
      030010 104406          TRAP    C1CLP1
5191 030012 013701 030144      MOV    T25BFR+14,R1          ;GET XST3 STATUS WORD
5192 030016 010102      MOV    R1,R2          ;SET UP EXPECTED
5193 030020 052702 000001      BIS    #BIT0,R2          ;SET THE RIB BIT
5194 030024 020102      CMP     R1,R2          ;ARE THEY EQUAL
5195 030026 001406      BEQ     1804          ;BR, IF EQUAL (GOOD)
5196 030030 004737 020116      JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
5200 030034      ERRMRD  ERRNO,T25NEF,EXPREC          ;NEF SHOULD BE SET
      030034 104456          TRAP    C1ERMRD
      030036 000227          .WORD  151
      030040 031125          .WORD  T25NEF
      030042 016360          .WORD  EXPREC
5201 030044          1804:
5202 030044      ENDSUB          ;***** END SUBTEST *****
      030044 104403          L10045:
      030046 023727 002170 000031      CMP    FATFLG,#25.          ;IS ERROR COUNT AT 25
5204 030054 002402      BLT     9994          ;BR, IF LESS THAN 25
5205 030056 004737 020170      JSR     PC,CKDROP          ;TRY TO DROP THE UNIT
5206 030062          9994:
5207                                     ;
5208                                     ;
5209                                     ;
5210 030062 004737 017372      JSR     PC,TSTLOOP          ;DO WE NEED TO ITERATE TEST
5211 030066 103002      BCC     1934          ;BR, IF NO LOOP REQUIRED
5212 030070 000137 023710      JMP     T25LOOP          ;EXECUTE AGAIN
5213 030074          1934:
5214 030074      EXIT    TST          ;ALL DONE THIS TEST
      030074 104432          TRAP    C1EXIT
      030076 001566          .WORD  L10036

```

5216		;			
5217		;	LOCAL STORAGE FOR THIS TEST		
5218		;			
5220 030100			.BLKB 10-<. TUV2A&7>		
5222 030110		T25PACKET:			;COMMAND PACKET FOR TEST
5223 030110 100004		.WORD	100004		;WRITE CHARACTERISTICS COMMAND, WITH ACK
5224 030112 030120		.WORD	T25DATA		;ADDRESS OF CHARACTERISTICS BLOCK
5225 030114 000000		.WORD	0		
5226 030116 000010		.WORD	8.		;STARTING VALUE OF BLOCK SIZE
5227 030120		T25DATA:			;CHARACTERISTICS DATA BLOCK
5228 030120 030130		.WORD	T25BFR		;ADDRESS OF MESSAGE BUFFER
5229 030122 000000		.WORD	0		
5230 030124 000012		.WORD	10.		;LENGTH OF MESSAGE BUFFER
5231 030126 000000		.WORD	0		
5232 030130		T25BFR: .BLKW	25.		;MESSAGE BUFFER
5233		;			
5234		;	WRITE SUBSYSTEM MEMORY COMMAND PACKET		
5235		;			
5237 030212			.BLKB 10-<. -TUV2A&7>		
5239 030220		T25PK2:			
5240 030220 100006		.WORD	100006		;WRITE SUB SYS MEM COMMAND, AND ACK
5241 030222 030250		.WORD	T25BF2		;ADDRESS OF SELECT BLOCK DATA
5242 030224 000000		.WORD	0		
5243 030226 000006		.WORD	6.		;SIZE OF DATA PACKET
5244					
5246 030230			.BLKB 10-<. -TUV2A&7>		
5248 030240		T25PK3:			
5249 030240 100005		.WORD	100005		;READ COMMAND, AND ACK
5250 030242		T25RB:			
5251 030242 003072		T25WB: .WORD	FREE		;ADDRESS OF WRITE BUFFER
5252 030244 000000		.WORD	0		
5253 030246 000000		T25SZ: .WORD	0		;SIZE OF BUFFER (EXTENT)
5254		.EVEN			
5255		;			
5256		;			
5257		;			
5258 030250		T25BF2:			
5259 030250 010		T25BS0: .BYTE	10		;BSELO AREA
5260 030251 200		T25BS1: .BYTE	200		;BSEL1 AREA
5261 030252 000000		T25S2: .WORD	0		;SEL 2 AREA
5262 030254 000000		T25S3: .WORD	0		;DATA AREA
5263		;			
5264		;			
5265		.EVEN			
5266		;	TAPE MOTION PACKET COMMAND VALUES		
5267					
5268 030256 100005		T25RN: .WORD	100005		;READ DATA (NEXT)
5269 030260 100405		T25WR: .WORD	100405		;READ DATA RETRY
5270 030262 102005		T25CON: .WORD	102005		;WRITE CONTINUOUS
5271 030264 177777		.WORD	177777		;END OF DATA
5272					
5273					

```
5275 030266 000000      T25CN2: .WORD 0      ;COUNTER FOR RECORDS
5276 030270 000000      T25CNT: .WORD 0      ;COUNTER FOR RECORDS
5277 030272 000000      T25DLY: .WORD 0      ;COUNTER FOR RECORDS
5278
5279
5280      ;*
5281      ;LOCAL TEXT MESSAGES FOR TEST
5282      ;-
5283
5284 030274      104      122      111 T210FL: .ASCIIZ 'DRIVE IS OFF-LINE'
5285 030316      127      122      111 T25SSR: .ASCIIZ 'WRITE SUBSYSTEM Miscellaneous Read Status Failed'
5286 030377      124      123      123 T25WDE: .ASCIIZ 'TSSR Not Correct After POSITION (SPACE) Command'
5287 030457      124      141      160 T25BOT: .ASCIIZ 'Tape Not At BOT After REWIND Command'
5288 030524      124      123      123 T25TM: .ASCIIZ 'TSSR Not Correct After POSITION (Space Command) Reject
5289 030613      127      162      151 T25NET: .ASCIIZ 'Write Tape, Status Alert, But No EOT Sensed'
5290 030667      123      160      141 T25WNG: .ASCIIZ 'Space Forward Failed To Position On Correct Record'
5291 030752      123      160      141 T25BNC: .ASCIIZ 'Space Forward, From BOT, Failed To Clear BOT Indication'
5292 031042      123      160      141 T25WNH: .ASCIIZ 'Space Reverse Failed To Position On Correct Record'
5293 031125      123      160      141 T25NEF: .ASCIIZ 'Space Reverse, At BOT, Failed To Set NEF (XST0)'
5294 031205      123      160      141 T25RIB: .ASCIIZ 'Space Reverse, Into BOT, Failed To Set RIB (XST3)'
5295 031267      122      145      167 T25RWN: .ASCIIZ 'Rewind (POSITION) Command Not Accepted'
5296 031336      104      162      151 T25OFL: .ASCIIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
5297 031411      124      123      123 T25WDC: .ASCIIZ 'TSSR Not Correct After READ DATA Command'
5298 031462      123      160      141 T25ID: .ASCIIZ 'Space Records'
5299      .EVEN
5300      ;*
5301      ;
5302      ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
5303      ;WRITE SUBSYSTEM MEMORY COMMAND
5304      ;
5305      ;-
5306
5307 031500      T25REST:
5308 031500      SAVREG
5309 031504      012701 030110      MOV      #T25PACKET,R1      ;SAVE THE REGISTERS
5310 031510      012721 100004      MOV      #100004,(R1).      ;START OF THE PACKET
5311 031514      012721 030120      MOV      #T25DATA,(R1).      ;WRITE SUBSYSTEM MEM. WITH ACK
5312 031520      005021      CLR      (R1).      ;ADDRESS OF CHARAISTICS DATA BLOCK
5313 031522      012721 000012      MOV      #10.,(R1).      ;EXTENDED ADDRESS
5314 031526      012721 030130      MOV      #T25BFR,(R1).      ;SIZE OF DATA BLOCK IN BYTES
5315 031532      005021      CLR      (R1).      ;ADDRESS OF MESSAGE BUFFER
5316 031534      012721 000024      MOV      #20.,(R1).      ;LENGTH OF MESSAGE BUFFER
5317 031540      005021      CLR      (R1).
5318 031542      012711 000000      MOV      #0,(R1)      ;SELECT DRIVE ZERO
5319 031546      012702 000030      MOV      #24.,R2      ;NUMBER OF LOCATIONS TO BE CLEARED
5320 031552      012762 177777 030130 64: MOV      #177777,T25BFR(R2) ;ALL ONES TO MESSAGE BUFFER
5321 031560      005742      TST      -(R2)      ;NEXT LOCATION
5322 031562      022702 000000      CMP      #0,R2      ;IS R2 AT ZERO YET
5323 031566      001371      BNE      64:      ;KEEP GOING UNTIL DONE
5324 031570      000207      RTS      PC      ;RETURN
5325
5326 031572      T25RT2:
5327 031572      SAVREG
5328 031576      012701 030220      MOV      #T25PK2,R1      ;SAVE THE REGISTERS
5329 031602      012721 100006      MOV      #100006,(R1).      ;START OF THE PACKET
5330 031606      012721 030250      MOV      #T25BF2,(R1).      ;WRITE SUBSYSTEM MEM. WITH ACK.
5331 031612      005021      CLR      (R1).      ;ADDRESS OF DATA BLOCK
                    ;EXTENDED ADDRESS
```

5332 031614 012721 000006
 5333 031620 005021
 5334 031622 012701 030250
 5335 031626 005021
 5336 031630 005011
 5337 031632 000207
 5338 031634
 5339 031634
 5340 031640 012701 030240
 5341 031644 012721 000000
 5342 031650 012721 000000
 5343 031654 005021
 5344 031656 012721 000000
 5345 031662 000207
 5346 031664
 031664
 031664 104401

T25RT3:

MOV #6.,(R1).
 CLR (R1).
 MOV #T25BF2,R1
 CLR (R1).
 CLR (R1)
 RTS PC
 SAVREG
 MOV #T25PK3,R1
 MOV #0,(R1).
 MOV #0,(R1).
 CLR (R1).
 MOV #0,(R1).
 RTS PC
 ENDTST

;SIZE OF DATA BLOCK IN BYTES

;POINT TO DATA SEL AREA

;RETURN

;SAVE THE REGISTERS

;START OF THE PACKET

;WRITE SUBSYSTEM MEM. WITH ACK.

;ADDRESS OF DATA BLOCK

;EXTENDED ADDRESS

;SIZE OF DATA BLOCK IN BYTES

;RETURN

L10036:

TRAP

C#ETST

```

5348 .SBTTL TEST 2: REREADS
5349 ;*
5350 ;
5351 ;THIS TEST VERIFIES THAT THE REREAD PREVIOUS AND REREAD NEXT
5352 ;COMMANDS OPERATE PROPERLY. VARIOUS COMBINATIONS OF ODD AND EVEN
5353 ;DATA BUFFER BOUNDRIES, RECORD SIZES (UP TO 64K BYTES IF MEMORY
5354 ;SPACE IS AVAILIABLE), AND BYTE-SWAP (SWP) AND OPPOSITE (OPP)
5355 ;CONRTOL ARE USED. ALSO TESTED ARE PROPER TERMINATIONS ON
5356 ;EXCEPTIONAL OR ERROR CONDITIONS: RECORD LENGTH LONG, RECORD
5357 ;LENGTH SHORT, READ REVERSE AT BOT, ILLEGAL DATA BUFFER ADDRESSES,
5358 ;AND DATA BUFFERS IN NONEXISTENT MEMORY.
5359 ;
5360 ;
5361 ;THE TEST CONSISTS OF THE FOLLOWING 15 SUBTESTS
5362 ;
5363 ;
5364 ;
5365 ;
5366 BGNTST
5367 031666 T2::
5368 031666 CLR FATFLG ;CLEAR FATAL ERROR FLAG
5369 031672 005037 003100 CLR KTFLG ;HOLD OFF KT11
5370 031676 012737 005771 002146 MOV #EPRT2,EPRTSW ;SECONDARY ERROR MESSAGE
5375 031704 004737 020262 JSR PC,KTOFF ;DON'T NEED KT11
5376 031710 012700 050637 MOV #TST26ID,R0 ;ASCII MESSAGE TO IDENTIFY TEST
5377 031714 004737 017424 JSR PC,TSTSETUP ;DO INITIAL TEST SETUP
5378 031720 012737 000001 002164 MOV #1,LOOPCNT ;PERFORM 1 ITERATIONS
5379 031726 005037 046106 CLR T26CNT ;CLEAR TAPE RECORD COUNTER
5380 ;*
5381 ;
5382 ;TEST 2. SUBTEST 1
5383 ;
5384 ;
5385 ;VERIFIES THAT THE REREAD PREVIOUS COMMAND
5386 ;OPERATES PROPERLY. THE TAPE IS FIRST
5387 ;REWOUND AND THEN WRITTEN WITH A SERIES OF TEST
5388 ;RECORDS VARYING IN LENGTH AND DATA CONTENT. THE TAPE
5389 ;IS THEN REWOUND AGAIN. FOR EACH TEST RECORD, THE
5390 ;TAPE IS SPACED FORWARD ONE RECORD AND A REREAD
5391 ;PREVIOUS COMMAND ISSUED. RESULTS (STATUS, DATA,
5392 ;ETC.) ARE VERIFIED. THE BYTE COUNT ON EACH REREAD
5393 ;PREVIOUS COMMAND IS SET TO THE LENGTH OF THE EXPECTED
5394 ;RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD OCCUR.
5395 ;
5396 ;
5397 ;-
5398 031732 T26LOOP:
5399
5400 031732 BGNSUB ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
5401 031732 T2.1: TRAP C$BSUB
5402 031732 104402 JSR PC,T26REST ;SET COMMAND PACKET
5403 031734 004737 050650 JSR PC,T26RT2 ;SET UP OTHER COMMAND PACKET
5404 031740 004737 050742 JSR PC,T26RT3 ;SET UP OTHER COMMAND PACKET
5405 031744 004737 051004 MOV #65000.,T26DLY ;SET UP DELAY COUNTER

```

```

5406 ;*****
5407 ;
5408 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
5409 ;
5410 ;*****
5411
5412 031756 004737 016660 10$: JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
5413 031762 103426 BCS 20$ ;BR IF INIT WAS OK
5414 031764 DELAY 250 ;DELAY FOR A REWIND TO FINISH
                                MOV #250,(PC)+
                                .WORD 0
                                MOV L$DLY,(PC)+
                                .WORD 0
                                DEC -6(PC)
                                BNE . 4
                                DEC 22(PC)
                                BNE . 20
5415 032014 005337 046114 DEC T26DLY ;DEC COUNTER
5416 032020 001356 BNE 10$ ;BR, IF DELAY NOT READY
5417 032022 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5421 032026 010001 MOV R0,R1 ;CONTENTS OF TSSR REGISTER
5422 032030 ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP C$ERDF
                                .WORD 201
                                .WORD SFIERR
                                .WORD SFIMSG
5423 032040
5424
5425 032040 012704 045730 20$: MOV #T26PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
5426
5427 ;*****
5428 ;
5429 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
5430 ;
5431 ;*****
5432
5433 032044 004737 010342 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
5434 032050 103407 BCS 26$ ;BR, IF COMMAND ISSUED OK
5435 032052 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5439 032056 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
5440 032060 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
                                TRAP C$ERHRD
                                .WORD 202
                                .WORD WRTMSG
                                .WORD SFIMSG
5441 032070 26$: CKLOOP ;LOOP IF SELECTED
5442 032070 104406 TRAP C$CLP1
5443
5444 ;*****
5445 ;
5446 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5447 ;
5448 ;*****
5449 032072 004737 010444 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
5450 032076 103413 BCS 30$ ;BR, IF NO PROBLEM
5451 032100 016501 000000 MOV TSSR(R5),R1 ;GET TSSR

```

```

5452 032104 012702 000200      MOV    #SSR,R2      ;SET UP EXPECTED TSSR
5453 032110 010004      MOV    R0,R4      ;PACKET ADDRESS SET UP
5454 032112 004737 020116      JSR    PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
5458 032116      ERRMRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      032116 104456      TRAP    C1ERRMRD
      032120 000313      .WORD  203
      032122 047414      .WORD  T26RWN
      032124 011710      .WORD  PKTSSR
5459 032126      30$:    CKLOOP      ;LOOP IF SELECTED
      032126 104406      TRAP    C1CLP1
5460
5461      ;*****
5462      ;
5463      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
5464      ;
5465      ;*****
5466
5467 032130 013701 045756      MOV    T26BFR+6,R1    ;PICK UP XSTO
5468 032134 010102      MOV    R1,R2      ;SET UP EXPECTED
5469 032136 052702 000002      BIS    #BIT1,R2    ;SET BOT BIT IN EXPECTED
5470 032142 020102      CMP    R1,R2      ;DOES EXP = REC'D
5471 032144 001406      BEQ    40$      ;BR, IF EQUAL (OK)
5472 032146 004737 020116      JSR    PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
5476 032152      ERRMRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      032152 104456      TRAP    C1ERRMRD
      032154 000314      .WORD  204
      032156 047125      .WORD  T26BOT
      032160 016360      .WORD  EXPREC
5477 032162      40$:    CKLOOP      ;LOOP IF SELECTED
      032162 104406      TRAP    C1CLP1
5478 032164 012703 000400      MOV    #256.,R3    ;RECORD SIZE
5479 032170 013737 003072 046062  MOV    FREE,T26RB    ;STARTING WRITE BUFFER ADDRESS
5480
5481      ;*****
5482      ;
5483      ;WRITE DATA,ACK,CVC=1 COMMAND
5484      ;
5485      ;*****
5486
5487 032176 012737 140005 046060  MOV    #140005,T26PK3    ;WRITE DATA,ACK,CVC=1 COMMAND
5488 032204 012704 046060      MOV    #T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
5489 032210      65$:    MOV    R3,R0      ;SET PATTERN IN CORRECT REGISTER
5490 032210 010300      JSR    PC,FILLMEM    ;FILL MEMORY WITH RECORD SIZE
5491 032212 004737 020410      MOV    R3,T26SZ    ;SET UP RECORD SIZE IN PACKET
5492 032216 010337 046066      MOV    R4,TSD8(R5) ;ISSUE COMMAND
5493 032222 010465 177776      JSR    PC,WAITF    ;WAIT FOR SSR TO SET
5494 032226 004737 017134      MOV    TSSR(R5),R1 ;GET TSSR CONTENTS
5495 032232 016501 000000      MOV    #SSR,R2      ;SET UP EXPECTED
5496 032236 012702 000200      CMP    R1,R2      ;ARE THEY EQUAL
5497 032242 020102      BEQ    75$      ;BR, IF OK
5498 032244 001406      JSR    PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
5499 032246 004737 020116      ;SOFT ERROR GENERATED BECAUSE THE
5503      ;WRITE COMMAND IS NOT BEING CHECKED
5504      ;HERE. IT WAS CHECKED IN LEAH2
5505      ;TSSR INCORRECT AFTER WRITE DATA
5506 032252      ERRSOFT  ERRNO,WRTErr,EXPREC ;
      032252 104457      TRAP    C1ERSOFT

```

```

032254 000315 .WORD 205
032256 005011 .WORD WRTERR
032260 016360 .WORD EXPREC
5507 032262 104406 75$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
032262 104406 TST (R3)+ ;BUMP RECORD SIZE
5508 032264 005723 CMP #268.,R3 ;END OF RECORD YET
5509 032266 022703 000414 BNE 65$ ;BR, IF MORE RECORDS TO WRITE
5510 032272 001346 80$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
5511 032274 104406 120$:
032276 104406 ;*****
;
;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
;
;*****
5512 032276 004737 010444 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
5513 032302 103413 BCS 130$ ;BR, IF NO PROBLEM
5514 032304 016501 000000 MOV TSSR(R5),R1 ;GET TSSR
5515 032310 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
5516 032314 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
5517 032316 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5518 032322 104456 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED TRAP C$ERHRD
032322 104456 .WORD 206
032324 000316 .WORD T26RWN
032326 047414 .WORD PKTSSR
032330 011710
5530 032332 104406 130$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
032332 104406 ;*****
;
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
;
;*****
5531 032334 013701 045756 MOV T26BFR+6,R1 ;PICK UP XST0
5532 032340 010102 MOV R1,R2 ;SET UP EXPECTED
5533 032342 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
5534 032346 020102 CMP R1,R2 ;DOES EXP = REC'D
5535 032350 001406 BEQ 140$ ;BR, IF EQUAL (OK)
5536 032352 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5537 032356 104456 ERRHRD ERRNO,T26BOT,PKTSSR ;TAPE NOT AT BOT AFTER REWIND TRAP C$ERHRD
032356 104456 .WORD 207
032360 000317 .WORD T26BOT
032362 047125 .WORD PKTSSR
032364 011710
5548 032366 104406 140$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
032366 104406 MOV #256.,T26RSZ ;SET RECORD SIZE
5549 032370 012737 000400 046112 ;*****
;
;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
;BIT 15 SETS DIRECTION 0=FORWARD 1=REVERSE

```

```

5555
5556
5557
5558 032376 012703 000001
5559 032402 004737 010144
5560 032406 103412
5561 032410 016501 000000
5562 032414 012702 000200
5563 032420 004737 020116
5567 032424
      032424 104456
      032426 000320
      032430 046527
      032432 016360
5568 032434
5569 032434 013703 046112
5570 032440 013737 003072 046062
5571
5572
5573
5574
5575
5576
5577
5578 032446 012737 141001 046060
5579 032454 012704 046060
5580 032460 010337 046066
5581 032464 010465 177776
5582 032470 004737 017134
5583 032474 016501 000000
5584 032500 012702 000200
5585 032504 020102
5586 032506 001406
5587 032510 004737 020116
5591 032514
      032514 104456
      032516 000321
      032520 047750
      032522 011710
5592 032524
      032524 104406
5593 032526 013702 003072
5594 032532 010304
5595 032534 162704 000400
5596 032540 060204
5597 032542 021403
5598 032544 001410
5599 032546 011401
5600 032550 010302
5601 032552 004737 020116
5605 032556
      032556 104456
      032560 000322
      032562 047172
      032564 016360
5606 032566
      032566 104406

;
; *****
145$:  MOV    #1,R3                ;SPACE ONE RECORD PARAMETER
      JSR     PC,SPACE            ;CALL SPACE ROUTINE
      BCS     150$               ;BR, IF NO PROBLEM WITH SPACE COMMAND
      MOV     TSSR(R5),R1        ;GET TSSR
      MOV     #SSR,R2            ;SET UP EXPECTED TSSR
      JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
      ERRHRD  ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
                                   TRAP    C$ERHRD
                                   .WORD   208
                                   .WORD   T26SC
                                   .WORD   EXPREC
150$:  MOV     T26RSZ,R3          ;RECORD SIZE
      MOV     FREE,T26RB         ;STARTING READ BUFFER ADDRESS
; *****
;
;REREREAD DATA,CVC=1,ACK COMMAND
;
; *****
165$:  MOV     #141001,T26PK3     ;REREREAD DATA,CVC=1,ACK COMMAND
      MOV     #T26PK3,R4         ;SET UP R4 WITH PACKET ADDRESS
      MOV     R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
      MOV     R4,TSDB(R5)        ;ISSUE COMMAND
      JSR     PC,WAITF           ;WAIT FOR SSR TO SET
      MOV     TSSR(R5),R1        ;GET TSSR CONTENTS
      MOV     #SSR,R2            ;SET UP EXPECTED
      CMP     R1,R2              ;ARE THEY EQUAL
      BEQ     170$              ;BR, IF OK
      JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
      ERRHRD  ERRNO,T26WDC,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
                                   TRAP    C$ERHRD
                                   .WORD   209
                                   .WORD   T26WDC
                                   .WORD   PKTSSR
170$:  CKLOOP                    ;LOOP IF SELECTED
                                   TRAP    C$CLP1
      MOV     FREE,R2            ;CURRENT BUFFER ADDRESS TO R2
      MOV     R3,R4              ;CURRENT RECORD SIZE
      SUB     #256.,R4           ;FIRST LOCATION IN BUFFER
173$:  ADD     R2,R4              ;SET UP POINTER
      CMP     (R4),R3            ;CHECK DATA READ (R3=DATA ALSO)
      BEQ     180$              ;BR, IF ALL IS WELL
      MOV     (R4),R1            ;RECD DATA
      MOV     R3,R2              ;EXPECTED DATA
      JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
      ERRHRD  ERRNO,T26DTA,EXPREC ;DATA READ NOT = WRITTEN
                                   TRAP    C$ERHRD
                                   .WORD   210
                                   .WORD   T26DTA
                                   .WORD   EXPREC
180$:  CKLOOP                    ;LOOP IF SELECTED
                                   TRAP    C$CLP1

```

MACRO M1200 20-APR 84 08:13 PAGE 97-5

SEQ 153

5607	032570	005724			TST	(R4),
5608	032572	160204			SUB	R2,R4
5609	032574	020403			CMP	R4,R3
5610	032576	001360			BNE	173\$
5611	032600	005723			TST	(R3),
5612	032602	010337	046112		MOV	R3,T26RSZ
5613	032606	022703	000412		CMP	#266.,R3
5614	032612	001271			BNE	145\$
5615	032614			190\$:	CKLOOP	
	032614	104406				
5616	032616				ENDSUB	
	032616					
	032616	104403				
5617	032620	023727	002170	000031	CMP	FATFLG,#25.
5618	032626	002402			BLT	999\$
5619	032630	004737	020170		JSR	PC,CKDROP
5620	032634			999\$:		

```
;BUMP TO NEXT LOCATION  
;CORRECT RECORDS SIZE VALUE  
;END OF RECORD YET  
;BR, IF NOT AT END OF RECORD  
;BUMP RECORD SIZE  
;RESET RECORD SIZE  
;END OF RECORD YET  
;BR, IF MORE RECORDS TO READ  
;LOOP IF SELECTED  
  
                                TRAP      C$CLP1  
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>  
                                L10047:  
  
                                TRAP      C$ESUB  
                ;IS ERROR COUNT AT 25  
                ;BR, IF LESS THAN 25  
                ;TRY TO DROP THE UNIT
```

L12

[illegible]

```
5674 ;*****
5675 ;
5676 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5677 ;
5678 ;*****
5679
5680 032730 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
5681 032734 103413          BCS      30$      ;BR, IF NO PROBLEM
5682 032736 016501 000000    MOV      TSSR(R5),R1    ;GET TSSR
5683 032742 012702 000200    MOV      #SSR,R2      ;SET UP EXPECTED TSSR
5684 032746 010004          MOV      R0,R4      ;PACKET ADDRESS SET UP
5685 032750 004737 020116    JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
5689 032754          ERRHRD  ERRNO,T26RWN,PKTSSR    ;REWIND NOT ACCEPTED
          032754 104456          TRAP      C$ERHRD
          032756 000325          .WORD    213
          032760 047414          .WORD    T26RWN
          032762 011710          .WORD    PKTSSR
5690 032764          30$:      CKLOOP          ;LOOP IF SELECTED
          032764 104406          TRAP      C$CLP1
5691 ;*****
5692 ;
5693 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
5694 ;
5695 ;*****
5696
5697
5698 032766 013701 045756      MOV      T26BFR+6,R1    ;PICK UP XSTO
5699 032772 010102          MOV      R1,R2      ;SET UP EXPECTED
5700 032774 052702 000002    BIS      #BIT1,R2    ;SET BOT BIT IN EXPECTED
5701 033000 020102          CMP      R1,R2      ;DOES EXP = REC'D
5702 033002 001406          BEQ      40$      ;BR, IF EQUAL (OK)
5703 033004 004737 020116    JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
5707 033010          ERRHRD  ERRNC T26BOT,EXPREC    ;TAPE NOT AT BOT AFTER REWIND
          033010 104456          TRAP      C$ERHRD
          033012 000326          .WORD    214
          033014 047125          .WORD    T26BOT
          033016 016360          .WORD    EXPREC
5708 033020          40$:      CKLOOP          ;LOOP IF SELECTED
          033020 104406          TRAP      C$CLP1
5709 033022 012703 000400      MOV      #256.,R3    ;RECORD SIZE
5710 033026 013737 003072 046062  MOV      FREE,T26RB    ;STARTING WRITE BUFFER ADDRESS
5711 ;*****
5712 ;
5713 ;WRITE DATA,ACK,SWB COMMAND
5714 ;
5715 ;*****
5716
5717
5718 033034 012737 110005 046060  MOV      #110005,T26PK3    ;WRITE DATA,ACK,SWB COMMAND
5719 033042 012704 046060      MOV      #T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
5720 033046          65$:      MOV      R3,R0      ;SET PATTERN IN CORRECT REGISTER
5721 033046 010300          JSR      PC,FILLMEM    ;FILL MEMORY WITH RECORD SIZE
5722 033050 004737 020410      MOV      R3,T26SZ    ;SET UP RECORD SIZE IN PACKET
5723 033054 010337 046066      MOV      R4,TSD8(R5)  ;ISSUE COMMAND
5724 033060 010465 177776      JSR      PC,WAITF    ;WAIT FOR SSR TO SET
5725 033064 004737 017134      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
5726 033070 016501 000000
```

```

5727 033074 012702 000200      MOV    #SSR,R2      ;SET UP EXPECTED
5728 033100 020102      CMP     R1,R2      ;ARE THEY EQUAL
5729 033102 001406      BEQ     75$      ;BR, IF OK
5730 033104 004737 020116      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
5734                                ;SOFT ERROR GENERATED BECAUSE THE
5735                                ;WRITE COMMAND IS NOT BEING CHECKED
5736                                ;HERE. IT WAS CHECKED IN LEAH2
5737 033110      ERRSOF ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      033110 104457      TRAP     C$ERSOFT
      033112 000327      .WORD    215
      033114 005011      .WORD    WRTErr
      033116 011710      .WORD    PKTSSR
5738 033120      75$:    CKLOOP      ;LOOP IF SELECTED
      033120 104406      TRAP     C$CLP1
5739 033122 005723      TST     (R3)+    ;BUMP RECORD SIZE
5740 033124 022703 000414      CMP     #268.,R3 ;END OF RECORD YET
5741 033130 001346      BNE     65$      ;BR, IF MORE RECORDS TO WRITE
5742 033132      80$:    CKLOOP      ;LOOP IF SELECTED
      033132 104406      TRAP     C$CLP1
5743 033134      120$:
5744                                ;*****
5745                                ;
5746                                ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5747                                ;
5748                                ;*****
5749
5750      JSR     PC,REWIND    ;CALL TAPE REWIND COMMAND
5751 033134 004737 010444      BCS     130$    ;BR, IF NO PROBLEM
5752 033140 103413      MOV     TSSR(R5),R1 ;GET TSSR
5753 033142 016501 000000      MOV     #SSR,R2 ;SET UP EXPECTED TSSR
5754 033146 012702 000200      MOV     R0,R4 ;PACKET ADDRESS SET UP
5755 033152 010004      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
5756 033154 004737 020116      ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
5760 033160      104456      TRAP     C$ERHRD
      033162 000330      .WORD    216
      033164 047414      .WORD    T26RWN
      033166 011710      .WORD    PKTSSR
5761 033170      130$:    CKLOOP      ;LOOP IF SELECTED
      033170 104406      TRAP     C$CLP1
5762                                ;*****
5763                                ;
5764                                ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
5765                                ;
5766                                ;*****
5767
5768      MOV     T26BFR+6,R1    ;PICK UP XSTO
5769 033172 013701 045756      MOV     R1,R2 ;SET UP EXPECTED
5770 033176 010102      BIS     #BIT1,R2    ;SET BOT BIT IN EXPECTED
5771 033200 052702 000002      CMP     R1,R2 ;DOES EXP = REC'D
5772 033204 020102      BEQ     140$    ;BR, IF EQUAL (OK)
5773 033206 001406      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
5774 033210 004737 020116      ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
5778 033214      104456      TRAP     C$ERHRD
      033216 000331      .WORD    217
      033220 047125      .WORD    T26BOT

```

```

033222 016360
5779 033224 104406 000400 046112 1401: CKLOOP ;LOOP IF SELECTED .WORD EXPREC
033224 104406 ;TRAP C1CLP1
5780 033226 012737 000400 046112 MOV #256.,T26RSZ ;SET UP RECORD SIZE
5781
5782 ;*****
5783 ;
5784 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
5785 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
5786 ;
5787 ;*****
5788
5789 033234 012703 000001 1451: MOV #1,R3 ;SPACE ONE RECORD PARAMETER
5790 033240 004737 010144 JSR PC,SPACE ;CALL SPACE ROUTINE
5791 033244 103412 BCS 1501 ;BR, IF NO PROBLEM WITH SPACE COMMAND
5792 033246 016501 000000 MOV TSSR(R5),R1 ;GET TSSR
5793 033252 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
5794 033256 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5798 033262 ERRHRD ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
033262 104456 TRAP C1ERHRD
033264 000332 .WORD 218
033266 046527 .WORD T26SC
033270 016360 .WORD EXPREC
5799 033272
5800 033272 013703 046112 1501: MOV T26RSZ,R3 ;RECORD SIZE
5801 033276 013737 003072 046062 MOV FREE,T26RB ;STARTING READ BUFFER ADDRESS
5802
5803 ;*****
5804 ;
5805 ;REREAD DATA,CVC=1,ACK,SWB COMMAND
5806 ;
5807 ;*****
5808
5809 033304 012737 151001 046060 1651: MOV #151001,T26PK3 ;REREAD DATA,CVC=1,ACK,SWB COMMAND
5810 033312 012704 046060 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
5811 033316 010337 046066 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
5812 033322 010465 177776 MOV R4,TSD8(R5) ;ISSUE COMMAND
5813 033326 004737 017134 JSR PC,WAITF ;WAIT FOR SSR TO SET
5814 033332 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
5815 033336 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
5816 033342 020102 CMP R1,R2 ;ARE THEY EQUAL
5817 033344 001406 BEQ 1701 ;BR, IF OK
5818 033346 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5822 033352 ERRHRD ERRNO,T26WDC,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
033352 104456 TRAP C1ERHRD
033354 000333 .WORD 219
033356 047750 .WORD T26WDC
033360 011710 .WORD PKTSSR
5823 033362
5824 033362 104406 003072 1701: CKLOOP ;LOOP IF SELECTED TRAP C1CLP1
5825 033370 010304 MOV FREE,R2 ;CURRENT BUFFER ADDRESS TO R2
5826 033372 162704 000400 MOV R3,R4 ;CURRENT RECORD SIZE
5827 033376 060204 SUB #256.,R4 ;FIRST LOCATION IN BUFFER
5828 033400 021403 1731: ADD R2,R4 ;SET UP POINTER
5829 033402 001410 CMP (R4),R3 ;CHECK DATA READ (R3=DATA ALSO)
5830 033404 011401 BEQ 1801 ;BR, IF ALL IS WELL
MOV (R4),R1 ;RECD DATA

```

Address	Instruction	Op-Code	Register	Value	Comments
5831	MOV	033406	R3,R2	010302	EXPECTED DATA
5832	JSR	033410	PC,FATCHK	004737	INC AND CHECK FOR MORE THAN 25 ERRORS
5836	ERRHRD	033414	ERRNO,T26DTA,EXPREC		DATA READ NOT WRITTEN
		033414		104456	TRAP C\$ERHRD
		033416		000334	.WORD 220
		033420		047172	.WORD T26DTA
		033422		016360	.WORD EXPREC
5837	180\$: CKLOOP	033424			LOOP IF SELECTED
		033424		104406	TRAP C\$CLP1
5838	TST	033426	(R4).	005724	BUMP TO NEXT LOCATION
5839	SUB	033430	R2,R4	160204	CORRECT RECORDS SIZE VALUE
5840	CMP	033432	R4,R3	020403	END OF RECORD YET
5841	BNE	033434	173\$	001360	BR, IF NOT AT END OF RECORD
5842	TST	033436	(R3).	005723	BUMP RECORD SIZE
5843	MOV	033440	R3,T26RSZ	010337	STORE RECORD SIZE
5844	CMP	033444	#266.,R3	022703	END OF RECORD YET
5845	BNE	033450	145\$	001271	BR, IF MORE RECORDS TO READ
5846	190\$: CKLOOP	033452			LOOP IF SELECTED
		033452		104406	TRAP C\$CLP1
5847	ENDSUB	033454			END SUBTEST
		033454			L10050:
5848	CMP	033456	FATFLG,#25.	002170	IS ERROR COUNT AT 25
5849	BLT	033464	999\$	002402	BR, IF LESS THAN 25
5850	JSR	033466	PC,CKDROP	004737	TRY TO DROP THE UNIT
5851	999\$:	033472			

				:	:*
				:	:
5853				:	TEST 2, SUBTEST 3
5854				:	:
5855				:	VERIFIES THAT THE REREAD PREVIOUS COMMAND WITH OPP=1
5856				:	(READ REVERSE, SPACE FORWARD) AND SWB=0 OPERATES
5857				:	PROPERLY. THE TAPE IS FIRST REWOUND AND THEN WRITTEN
5858				:	WITH A SERIES OF TEST RECORDS VARYING IN LENGTH AND
5859				:	DATA CONTENT; THE FIRST FOUR BYTES OF EACH RECORD
5860				:	CONTAIN ITS RECORD NUMBER (INDICATING POSITION ON
5861				:	TAPE). THE TAPE IS THEN REWOUND AGAIN. FOR EACH
5862				:	TEST RECORD, THE FOLLOWING SEQUENCE IS EXECUTED.
5863				:	:
5864				:	1. THE REREAD PREVIOUS COMMAND WITH OPP=1 IS ISSUED
5865				:	AND THE RESULTS CHECKED
5866				:	:
5867				:	2. A READ FORWARD COMMAND IS THEN ISSUED AND THE
5868				:	DATA IS CHECKED TO VERIFY THAT THE TAPE WAS
5869				:	POSITIONED PROPERLY AFTER THE REREAD PREVIOUS
5870				:	COMMAND (E.G. THE TAPE SHOULD HAVE BEEN LEFT
5871				:	POSITIONED AT THE START OF THE TEST RECORD). THE
5872				:	READ FORWARD COMMAND LEAVES THE TAPE POSITIONED
5873				:	PROPERLY AT THE START OF THE NEXT TEST RECORD.
5874				:	:
5875				:	THE BYTE COUNT ON EACH REREAD PREVIOUS COMMAND IS SET
5876				:	TO THE LENGTH OF THE EXPECTED RECORD, SO NO
5877				:	EXCEPTIONAL CONDITIONS SHOULD OCCUR.
5878				:	:
5879				:	:
5880				:	:
5881				:	:
5882				:	:
5883				:	:
5884				:	:
5885				:	-:
5886	033472		BGNSUB		;***** BEGIN SUBTEST *****
	033472				T2.3:
	033472	104402			TRAP C#BSUB
5887	033474	004737	050650	JSR PC,T26REST	;SET COMMAND PACKET
5888	033500	005037	046106	CLR T26CNT	;CLEAR TAPE RECORD COUNTER
5889	033504	004737	050742	JSR PC,T26RT2	;SET UP OTHER COMMAND PACKET
5890	033510	004737	051004	JSR PC,T26RT3	;SET UP OTHER COMMAND PACKET
5891					
5892					;*****
5893					:
5894					;ISSUE CONTROLLER "SOFT" INITIALIZE CARRY BIT CLEAR IF ERROR
5895					:
5896					;*****
5897					
5898	033514	004737	016660	JSR PC,SOFINIT	;DO INITIALIZE ON CONTROLLER
5899	033520	103407		BCS 20;	;BR IF INIT WAS OK
5900	033522	004737	020116	JSR PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS
5904	033526	010001		MOV RO,R1	;CONTENTS OF ICSR REGISTER
5905	033530			ERRDF ERNO,SFIERR,SFIMSG	;FATAL ERROR ICSR WAS NOT OK
	033530	104455			TRAP C#ERDF
	033532	000335			.WORD 221
	033534	003550			.WORD SFIERR
	033536	011676			.WORD SFIMSG
5906	033540		201:		

```

5907
5908 033540 012704 045730          MOV      #T26PACKET,R4          ;SUBROUTINE NEEDS PACKET ADDRESS
5909
5910          ;*****
5911          ;
5912          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
5913          ;
5914          ;*****
5915
5916 033544 004737 010342          JSR      PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
5917 033550 103407          BCS      26$          ;BR, IF COMMAND ISSUED OK
5918 033552 004737 020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
5922 033556 010001          MOV      R0,R1          ;SAVE CONTENTS OF TSSR
5923 033560          ERRHRD  ERRNO,WRTMSG,SFMSG          ;WRITE CHARACTERISTICS FAILED
          033560 104456          TRAP      C$ERHRD
          033562 000336          .WORD    222
          033564 004754          .WORD    WRTMSG
          033566 011676          .WORD    SFMSG
5924 033570          26$:      CKLOOP          ;LOOP IF SELECTED
          033570 104406          TRAP      C$CLP1
5925
5926          ;*****
5927          ;
5928          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5929          ;
5930          ;*****
5931
5932 033572 004737 010444          JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
5933 033576 103413          BCS      30$          ;BR, IF NO PROBLEM
5934 033600 016501 000000          MOV      TSSR(R5),R1          ;GET TSSR
5935 033604 012702 000200          MOV      #SSR,R2          ;SET UP EXPECTED TSSR
5936 033610 010004          MOV      R0,R4          ;PACKET ADDRESS SET UP
5937 033612 004737 020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
5941 033616          ERRHRD  ERRNO,T26RWN,PKTSSR          ;REWIND NOT ACCEPTED
          033616 104456          TRAP      C$ERHRD
          033620 000337          .WORD    223
          033622 047414          .WORD    T26RWN
          033624 011710          .WORD    PKTSSR
5942 033626          30$:      CKLOOP          ;LOOP IF SELECTED
          033626 104406          TRAP      C$CLP1
5943
5944          ;*****
5945          ;
5946          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
5947          ;
5948          ;*****
5949
5950 033630 013701 045756          MOV      T26BFR+6,R1          ;PICK UP XST0
5951 033634 010102          MOV      R1,R2          ;SET UP EXPECTED
5952 033636 052702 000002          BIS      #BIT1,R2          ;SET BOT BIT IN EXPECTED
5953 033642 020102          CMP      R1,R2          ;DOES EXP = REC'D
5954 033644 001406          BEQ      40$          ;BR, IF EQUAL (OK)
5955 033646 004737 020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
5959 033652          ERRHRD  ERRNO,T26BOT,EXPREC          ;TAPE NOT AT BOT AFTER REWIND
          033652 104456          TRAP      C$ERHRD
          033654 000340          .WORD    224
          033656 047125          .WORD    T26BOT

```

```

033660 016360
5960 033662 40$: CKLOOP ;LOOP IF SELECTED .WORD EXPREC
033662 104406 ;WRITE DATA,CVC=1,ACK COMMAND TRAP C$CLP1
5961 033664 012703 000400 MOV #256.,R3 ;RECORD SIZE
5962 033670 013737 003072 046062 MOV FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
5963
5964 ;*****
5965 ;
5966 ;WRITE DATA,CVC=1,ACK COMMAND
5967 ;
5968 ;*****
5969
5970 033676 012737 140005 046060 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
5971 033704 012704 046060 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
5972 033710
5973 033710 010300 65$: MOV R3,R0 ;SET PATTERN IN CORRECT REGISTER
5974 033712 004737 020410 JSR PC,FILLMEM ;FILL MEMORY WITH RECORD SIZE
5975 033716 010337 046066 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
5976 033722 013777 046106 147142 MOV T26CNT,8FREE ;MOVE TAPE RECORD NUMBER TO BUFFER
5977 033730 062737 000001 046106 ADD #1,T26CNT ;NUMBER READY FOR NEXT RECORD
5978 033736 010465 177776 MOV R4,TSD8(R5) ;ISSUE COMMAND
5979 033742 004737 017134 JSR PC,WAITF ;WAIT FOR SSR TO SET
5980 033746 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
5981 033752 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
5982 033756 020102 CMP R1,R2 ;ARE THEY EQUAL
5983 033760 001406 BEQ 75$ ;BR, IF OK
5984 033762 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5988 ;SOFT ERROR GENERATED BECAUSE THE
5989 ;WRITE COMMAND IS NOT BEING CHECKED
5990 ;HERE. IT WAS CHECKED IN LEAH2
5991 033766 ERRSOF ERRNO,WRITER,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
033766 104457 TRAP C$ERSOFT
033770 000341 .WORD 225
033772 005011 .WORD WRITER
033774 011710 .WORD PKTSSR
5992 033776 75$: CKLOOP ;LOOP IF SELECTED .WORD C$CLP1
033776 104406 ;BUMP THE RECORD SIZE
5993 034000 005723 CMP #268.,R3 ;MAXIMUM SIZE YET
5994 034002 022703 000414 BEQ 120$ ;BR, IF AT END OF WRITE SEQUENCE
5995 034006 001401 BR 65$ ;WRITE MORE RECORDS
5996 034010 000737
5997 034012
5998 034012 005037 046106 120$: CLR T26CNT ;SET RECORD COUNTER BACK TO ZERO
5999
6000 ;*****
6001 ;
6002 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6003 ;
6004 ;*****
6005
6006 034016 004737 010444 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
6007 034022 103413 BCS 130$ ;BR, IF NO PROBLEM
6008 034024 016501 000000 MOV TSSR(R5),R1 ;GET TSSR
6009 034030 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
6010 034034 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
6011 034036 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6015 034042 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED

```

034042	104456						TRAP	C\$ERHRD
034044	000342						.WORD	226
034046	047414						.WORD	T26RWN
034050	011710						.WORD	PKTSSR
6016	034052			130:	CKLOOP			
	034052	104406						
6017							TRAP	C\$CLP1
6018								
6019								
6020								
6021								
6022								
6023								
6024	034054	013701	045756					
6025	034060	010102						
6026	034062	052702	000002					
6027	034066	020102						
6028	034070	001406						
6029	034072	004737	020116					
6033	034076							
	034076	104456						
	034100	000343						
	034102	047125						
	034104	016360						
6034	034106			140:	CKLOOP			
	034106	104406						
6035							TRAP	C\$CLP1
6036								
6037								
6038								
6039								
6040								
6041								
6042								
6043	034110	012703	000001					
6044	034114	004737	010144					
6045	034120	012703	000400					
6046	034124	013737	003072	046062	150:			
6047								
6048								
6049								
6050								
6051								
6052								
6053								
6054	034132	012737	161001	046060				
6055	034140	012704	046060					
6056	034144	010337	046066					
6057	034150	010465	177776					
6058	034154	004737	017134					
6059	034160	016501	000000					
6060	034164	012702	000200					
6061	034170	020102						
6062	034172	001406						
6063	034174	004737	020116					
6067	034200							
	034200	104456						

	034202	000344					.WORD	228
	034204	046432					.WORD	T26RRG
	034206	011710					.WORD	PKTSSR
6068	034210		170\$:	CKLOOP		; LOOP IF SELECTED		
	034210	104406					TRAP	C\$CLP1
6069	034212	005723			TST (R3),	; BUMP RECORD SIZE		
6070	034214	062737	000001	046106	ADD #1,T26CNT	; BUMP TAPE RECORD COUNTER		
6071								
6072								
6073								
6074								
6075								
6076								
6077								
6078	034222	012737	140001	046060	MOV #140001,T26PK3	; READ DATA, CVC=1, ACK COMMAND		
6079	034230	010337	046066		MOV R3,T26SZ	; SET SIZE INTO PACKET		
6080	034234	010465	177776		MOV R4,TSD8(R5)	; ISSUE READ DATA COMMAND		
6081	034240	004737	017134		JSR PC,WAITF	; WAIT FOR SSR		
6082	034244	016501	000000		MOV TSSR(R5),R1	; PICK UP THE TSSR		
6083	034250	012702	000200		MOV #SSR,R2	; SET UP EXPECTED		
6084	034254	020102			CMP R1,R2	; IS THE TSSR OK		
6085	034256	001406			BEQ 195\$	; BR, IF TSSR OK (GOOD)		
6086	034260	004737	020116		JSR PC,FATCHK	; INC AND CHECK FOR MORE THAN 25 ERRORS		
6090	034264				ERRHRD ERRNO,RDERR,PKTSSR	; READ DATA COMMAND FAILED		
	034264	104456					TRAP	C\$ERHRD
	034266	000345					.WORD	229
	034270	005104					.WORD	RDERR
	034272	011710					.WORD	PKTSSR
6091	034274		195\$:	CKLOOP		; LOOP IF SELECTED		
	034274	104406					TRAP	C\$CLP1
6092	034276	017701	146570		MOV #FREE,R1	; FIRST WORD FROM READ BUFFER		
6093	034302	013702	046106		MOV T26CNT,R2	; SET UP EXPECTED		
6094	034306	020102			CMP R1,R2	; IS TAPE POSITION CORRECT		
6095	034310	001406			BEQ 197\$	; KEEP GOING POSITION OK		
6096	034312	004737	020116		JSR PC,FATCHK	; INC AND CHECK FOR MORE THAN 25 ERRORS		
6100	034316				ERRHRD ERRNO,T26WNG,EXPREC	; TAPE POSITION INCORRECT		
	034316	104456					TRAP	C\$ERHRD
	034320	000346					.WORD	230
	034322	046116					.WORD	T26WNG
	034324	016360					.WORD	EXPREC
6101	034326		197\$:	CKLOOP				
	034326	104406					TRAP	C\$CLP1
6102	034330	022703	000412		CMP #266.,R3	; AT MAX SIZE YET		
6103	034334	001401			BEQ 200\$	; BR, IF AT END OF THE SUBTEST		
6104	034336	000672			BR 150\$	; KEEP GOING MORE RECORDS		
6105	034340		200\$:					
6106	034340			ENDSUB		; >>>>>>>>> END SUBTEST >>>>>>>>>		
	034340					L10051:		
	034340	104403					TRAP	C\$ESUB
6107	034342	023727	002170	000031	CMP FATFLG,#25.	; IS ERROR COUNT AT 25		
6108	034350	002402			BLT 999\$	; BR, IF LESS THAN 25		
6109	034352	004737	020170		JSR PC,CKDROP	; TRY TO DROP THE UNIT		
6110	034356		999\$:					

[illegible]

```

6165 034454 104406 26$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
034454
6166
6167 ;*****
6168 ;
6169 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6170 ;
6171 ;*****
6172
6173 034456 004737 010444 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
6174 034462 103413 BCS 30$ ;BR, IF NO PROBLEM
6175 034464 016501 000000 MOV TSSR(R5),R1 ;GET TSSR
6176 034470 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
6177 034474 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
6178 034476 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6182 034502 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
034502 104456 TRAP C$ERHRD
034504 000351 .WORD 233
034506 047414 .WORD T26RWN
034510 011710 .WORD PKTSSR
6183 034512 30$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
034512 104406
6184
6185 ;*****
6186 ;
6187 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
6188 ;
6189 ;*****
6190
6191 034514 013701 045756 MOV T26BFR+6,R1 ;PICK UP XST0
6192 034520 010102 MOV R1,R2 ;SET UP EXPECTED
6193 034522 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
6194 034526 020102 CMP R1,R2 ;DOES EXP = REC'D
6195 034530 001406 BEQ 40$ ;BR, IF EQUAL (OK)
6196 034532 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6200 034536 ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
034536 104456 TRAP C$ERHRD
034540 000352 .WORD 234
034542 047125 .WORD T26BOT
034544 016360 .WORD EXPREC
6201 034546 40$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
034546 104406
6202 034550 012703 000400 MOV #256.,R3 ;RECORD SIZE
6203 034554 013737 003072 046062 MOV FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
6204
6205 ;*****
6206 ;
6207 ;WRITE DATA,CVC=1,ACK COMMAND
6208 ;
6209 ;*****
6210
6211 034562 012737 140005 046060 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
6212 034570 012704 046060 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
6213 034574
6214 034574 010300 65$: MOV R3,R0 ;SET PATTERN IN CORRECT REGISTER
6215 034576 004737 020410 JSR PC,FILLMEM ;FILL MEMORY WITH RECORD SIZE
6216 034602 010337 046066 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET

```

```

6217 034606 013777 046106 146256      MOV      T26CNT,0FREE      ;MOVE TAPE RECORD NUMBER TO BUFFER
6218 034614 062737 000001 046106      ADD      #1,T26CNT      ;NUMBER READY FOR NEXT RECORD
6219 034622 010465 177776              MOV      R4,TSD8(R5)      ;ISSUE COMMAND
6220 034626 004737 017134              JSR      PC,WAITF      ;WAIT FOR SSR TO SET
6221 034632 016501 000000              MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
6222 034636 012702 000200              MOV      #SSR,R2      ;SET UP EXPECTED
6223 034642 020102                    CMP      R1,R2      ;ARE THEY EQUAL
6224 034644 001406                    BEQ      75$      ;BR, IF OK
6225 034646 004737 020116              JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
6229                                     ;SOFT ERROR GENERATED BECAUSE THE
6230                                     ;WRITE COMMAND IS NOT BEING CHECKED
6231                                     ;HERE. IT WAS CHECKED IN LEAH2
6232 034652                                ERRSOF ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
6233 034652 104457                                TRAP      C$ERSOFT
6234 034654 000353                                .WORD      235
6235 034656 005011                                .WORD      WRERR
6236 034660 011710                                .WORD      PKTSSR
6237 034662 104406      75$:      CKLOOP      ;LOOP IF SELECTED
6238 034664 005723      TST      (R3)+      ;BUMP THE RECORD SIZE
6239 034666 022703 000412      CMP      #266.,R3      ;MAXIMUM SIZE YET
6240 034672 001401      BEQ      120$      ;BR, IF AT END OF WRITE SEQUENCE
6241 034674 000737      BR      65$      ;WRITE MORE RECORDS
6242                                     ;
6243      120$:      CLR      T26CNT      ;SET RECORD COUNTER BACK TO ZERO
6244                                     ;
6245      ;*****
6246      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6247      ;*****
6248 034702 004737 010444              JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
6249 034706 103413              BCS      130$      ;BR, IF NO PROBLEM
6250 034710 016501 000000              MOV      TSSR(R5),R1      ;GET TSSR
6251 034714 012702 000200              MOV      #SSR,R2      ;SET UP EXPECTED TSSR
6252 034720 010004              MOV      R0,R4      ;PACKET ADDRESS SET UP
6253 034722 004737 020116              JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
6254 034726                                ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
6255 034726 104456                                TRAP      C$ERHRD
6256 034730 000354                                .WORD      236
6257 034732 047414                                .WORD      T26RWN
6258 034734 011710                                .WORD      PKTSSR
6259      130$:      CKLOOP      ;LOOP IF SELECTED
6260                                     ;
6261      ;*****
6262      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
6263      ;*****
6264 034740 013701 045756              MOV      T26BFR+6,R1      ;PICK UP XST0
6265 034744 010102              MOV      R1,R2      ;SET UP EXPECTED
6266 034746 052702 000002              BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
6267 034752 020102              CMP      R1,R2      ;DOES EXP = REC'D
6268 034754 001406              BEQ      140$      ;BR, IF EQUAL (OK)

```

```

6270 034756 004737 020116          JSR    PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6274 034762          ERRHRD    ERRNO,T26BOT,EXPREC          ;TAPE NOT AT BOT AFTER REWIND
        034762 104456          TRAP    C$ERHRD
        034764 000355          .WORD    237
        034766 047125          .WORD    T26BOT
        034770 016360          .WORD    EXPREC
6275 034772          140$:    CKLOOP          ;LOOP IF SELECTED
        034772 104406          TRAP    C$CLP1
6276
6277          ;*****
6278          ;
6279          ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
6280          ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
6281          ;
6282          ;*****
6283
6284 034774 012703 000001          MOV     #1,R3          ;SET UP SPACE FORWARD 1
6285 035000 004737 010144          JSR     PC,SPACE          ;ISSUE SPACE COMMAND
6286 035004 012703 000400          MOV     #256.,R3          ;RECORD SIZE
6287 035010 013737 003072 046062 150$:    MOV     FREE,T26RB          ;STARTING READ BUFFER ADDRESS
6288
6289          ;*****
6290          ;
6291          ;REREAD DATA,CVC=1,ACK, OPP COMMAND
6292          ;
6293          ;*****
6294
6295 035016 012737 171001 046060 165$:    MOV     #171001,T26PK3          ;REREAD DATA,CVC=1,ACK, OPP COMMAND
6296 035024 012704 046060          MOV     #T26PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
6297 035030 010337 046066          MOV     R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
6298 035034 010465 177776          MOV     R4,T5DB(R5)          ;ISSUE COMMAND
6299 035040 004737 017134          JSR     PC,WAITF          ;WAIT FOR SSR TO SET
6300 035044 016501 000000          MOV     TSSR(R5),R1          ;GET TSSR CONTENTS
6301 035050 012702 000200          MOV     #SSR,R2          ;SET UP EXPECTED
6302 035054 020102          CMP     R1,R2          ;ARE THEY EQUAL
6303 035056 001406          BEQ     170$          ;BR, IF OK
6304 035060 004737 020116          JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6308 035064          ERRHRD    ERRNO,T26RRF,PKTSSR          ;TSSR INCORRECT AFTER REREAD DATA
        035064 104456          TRAP    C$ERHRD
        035066 000356          .WORD    238
        035070 046335          .WORD    T26RRF
        035072 011710          .WORD    PKTSSR
6309 035074          170$:    CKLOOP          ;LOOP IF SELECTED
        035074 104406          TRAP    C$CLP1
6310 035076 017701 145770          MOV     #FREE,R1          ;FIRST WORD FROM READ BUFFER
6311 035102 013702 046106          MOV     T26CNT,R2          ;SET UP EXPECTED
6312 035106 000302          SWAB    R2          ;SWAP BYTES IN EXPECTED
6313 035110 020102          CMP     R1,R2          ;IS TAPE POSITION CORRECT
6314 035112 001406          BEQ     190$          ;KEEP GOING POSITION OK
6315 035114 004737 020116          JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6319 035120          ERRHRD    ERRNO,T26WNG,EXPREC          ;TAPE POSITION INCORRECT
        035120 104456          TRAP    C$ERHRD
        035122 000357          .WORD    239
        035124 046116          .WORD    T26WNG
        035126 016360          .WORD    EXPREC
6320 035130          190$:    CKLOOP
        035130 104406          TRAP    C$CLP1

```

Address	Op Code	Register	Value	Label	Comment	Trap	Value
6321	035132	005723					
6322	035134	062737	000001	046106	TST ADD	(R3),	
6323						#1,T26CNT	
6324							
6325							
6326							
6327							
6328							
6329							
6330	035142	012737	140001	046060	MOV	#140001,T26PK3	
6331	035150	010337	046066		MOV	R3,T26SZ	
6332	035154	010465	177776		MOV	R4,T26DB(R5)	
6333	035160	004737	017134		JSR	PC,WAITF	
6334	035164	016501	000000		MOV	TSSR(R5),R1	
6335	035170	012702	000200		MOV	#SSR,R2	
6336	035174	020102			CMP	R1,R2	
6337	035176	001406			BEQ	215\$	
6338	035200	004737	020116		JSR	PC,FATCHK	
6342	035204				ERRHRD	ERRNO,T26RDF,PKTSSR	
	035204	104456					TRAP C\$ERHRD
	035206	000360					.WORD 240
	035210	046266					.WORD T26RDF
	035212	011710					.WORD PKTSSR
6343	035214				215\$: CKLOOP		
	035214	104406					TRAP C\$CLP1
6344	035216	017701	145650		MOV	#FREE,R1	
6345	035222	013702	046106		MOV	T26CNT,R2	
6346	035226	020102			CMP	R1,R2	
6347	035230	001406			BEQ	217\$	
6348	035232	004737	020116		JSR	PC,FATCHK	
6352	035236				ERRHRD	ERRNO,T26WNG,EXPREC	
	035236	104456					TRAP C\$ERHRD
	035240	000361					.WORD 241
	035242	046116					.WORD T26WNG
	035244	016360					.WORD EXPREC
6353	035246				217\$: CKLOOP		
	035246	104406					TRAP C\$CLP1
6354	035250	022703	000410		CMP	#264.,R3	
6355	035254	001401			BEQ	220\$	
6356	035256	000654			BR	150\$	
6357	035260				220\$:		
6358	035260				ENDSUB		
	035260						; >>>>>>>>> END SUBTEST >>>>>>>>>
	035260	104403					L10052:
6359	035262	023727	002170	000031	CMP	FATFLG,#25.	
6360	035270	002402			BLT	999\$	
6361	035272	004737	020170		JSR	PC,CKDROP	
6362	035276				999\$:		


```

035370 104406                                TRAP      C1CLP1
6417
6418
6419
6420
6421
6422
6423
6424 035372 004737 010444          JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
6425 035376 103413          BCS      301          ;BR, IF NO PROBLEM
6426 035400 016501 000000          MOV      TSSR(R5),R1          ;GET TSSR
6427 035404 012702 000200          MOV      #SSR,R2          ;SET UP EXPECTED TSSR
6428 035410 010004          MOV      R0,R4          ;PACKET ADDRESS SET UP
6429 035412 004737 020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6433 035416          ERRHRD  ERRNO,T26RWN,PKTSSR          ;REWIND NOT ACCEPTED
        035416 104456                                TRAP      C1ERRHRD
        035420 000364                                .WORD      244
        035422 047414                                .WORD      T26RWN
        035424 011710                                .WORD      PKTSSR
6434 035426          301:      CKLOOP                                ;LOOP IF SELECTED
        035426 104406                                TRAP      C1CLP1
6435
6436
6437
6438
6439
6440
6441
6442 035430 013701 045756          MOV      T26BFR+6,R1          ;PICK UP XSTO
6443 035434 010102          MOV      R1,R2          ;SET UP EXPECTED
6444 035436 052702 000002          BIS      #BIT1,R2          ;SET BOT BIT IN EXPECTED
6445 035442 020102          CMP      R1,R2          ;DOES EXP = REC'D
6446 035444 001406          BEQ      401          ;BR, IF EQUAL (OK)
6447 035446 004737 020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6451 035452          ERRHRD  ERRNO,T26BOT,EXPREC          ;TAPE NOT AT BOT AFTER REWIND
        035452 104456                                TRAP      C1ERRHRD
        035454 000365                                .WORD      245
        035456 047125                                .WORD      T26BOT
        035460 016360                                .WORD      EXPREC
6452 035462          401:      CKLOOP                                ;LOOP IF SELECTED
        035462 104406                                TRAP      C1CLP1
6453 035464 012703 001000          MOV      #512,R3          ;RECORD SIZE
6454 035470 013737 003072 046062  MOV      FREE,T26RB          ;STARTING WRITE BUFFER ADDRESS
6455
6456
6457
6458
6459
6460
6461
6462 035476 012737 140005 046060  MOV      #140005,T26PK3          ;WRITE DATA,CVC=1,ACK COMMAND
6463 035504 012704 046080          MOV      #T26PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
6464 035510          651:      MOV      R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
6465 035510 010337 046066          MOV      R4,TSD8(R5)          ;ISSUE COMMAND
6466 035514 010465 177776          JSR      PC,WAITF          ;WAIT FOR SSR TO SET
6467 035520 004737 017134          MOV      TSSR(R5),R1          ;GET TSSR CONTENTS
6468 035524 016501 000000

```

```

6469 035530 012702 000200      MOV    #SSR,R2      ;SET UP EXPECTED
6470 035534 020102      CMP     R1,R2      ;ARE THEY EQUAL
6471 035536 001406      BEQ     751      ;BR, IF OK
6472 035540 004737 020116      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
6476                               ;SOFT ERROR GENERATED BECAUSE THE
6477                               ;WRITE COMMAND IS NOT BEING CHECKED
6478                               ;HERE. IT WAS CHECKED IN LEAH2
6479 035544      ERRSOF1 ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
        035544 104457      TRAP    C1ERSOFT
        035546 000366      .WORD   246
        035550 005011      .WORD   WRTErr
        035552 011710      .WORD   PKTSSR
6480 035554      751:    CKLOOP      ;LOOP IF SELECTED
        035554 104406      TRAP    C1CLP1
6481 035556 005303      DEC     R3      ;SET RECORD SIZE TO 511.
6482 035560 013737 003072 046062      MOV     FREE,T26RB    ;STARTING READ BUFFER ADDRESS
6483
6484      ;*****
6485      ;
6486      ;REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
6487      ;
6488      ;*****
6489
6490 035566 012737 161001 046060      MOV     #161001,T26PK3    ;REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
6491 035574 012704 046060      1651:  MOV     #T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
6492 035600 010337 046066      MOV     R3,T26SZ      ;SET UP RECORD SIZE IN PACKET
6493 035604 010465 177776      MOV     R4,TSD8(R5)    ;ISSUE COMMAND
6494 035610 004737 017134      JSR     PC,WAITF      ;WAIT FOR SSR TO SET
6495 035614 016501 000000      MOV     TSSR(R5),R1    ;GET TSSR CONTENTS
6496 035620 012702 100204      MOV     #SSR!SC!BIT2,R2    ;SET UP EXPECTED
6497 035624 020102      CMP     R1,R2      ;ARE THEY EQUAL
6498 035626 001406      BEQ     1701      ;BR, IF OK
6499 035630 004737 020116      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
6503 035634      ERRMRD  ERRNO,T26TRL,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
        035634 104456      TRAP    C1ERRMRD
        035636 000367      .WORD   247
        035640 050472      .WORD   T26TRL
        035642 011710      .WORD   PKTSSR
6504 035644      1701:  CKLOOP      ;LOOP IF SELECTED
        035644 104406      TRAP    C1CLP1
6505
6506      ;*****
6507      ;
6508      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6509      ;
6510      ;*****
6511
6512 035646 013701 045756      MOV     T26BFR+6,R1    ;GET MESSAGE BUFFER
6513 035652 010102      MOV     R1,R2      ;SET UP EXPECTED
6514 035654 052702 010000      BIS     #81112,R2    ;SET THE ALL BIT IN EXPECTED
6515 035660 020102      CMP     R1,R2      ;ARE THEY EQUAL
6516 035662 001406      BEQ     1801      ;BR, IF EQUAL (ALL IS WELL)
6517 035664 004737 020116      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
6521 035670      ERRMRD  ERRNO,T26LON,EXPREC ;THE ALL BIT WAS NOT SET IN XSTO
        035670 104456      TRAP    C1ERRMRD
        035672 000370      .WORD   248
        035674 050240      .WORD   T26LON

```

	035676	016360								.WORD	EXPREC
6522	035700				1804:	CKLOOP			;LOOP IF SELECTED		
	035700	104406								TRAP	C\$CLP1
6523	035702	012703	000777			MOV	#511.,R3		;SET RECORD SIZE		
6524	035706	013737	003072	046062		MOV	FREE,T26RB		;STARTING READ BUFFER ADDRESS		
6525											
6526											
6527											
6528											
6529											
6530											
6531											
6532	035714	012737	141001	046060		MOV	#141001,T26PK3		;REREAD DATA,CVC=1,ACK COMMAND		
6533	035722	012704	046060		3654:	MOV	#T26PK3,R4		;SET UP R4 WITH PACKET ADDRESS		
6534	035726	010337	046066			MOV	R3,T26SZ		;SET UP RECORD SIZE IN PACKET		
6535	035732	010465	177776			MOV	R4,TSDB(R5)		;ISSUE COMMAND		
6536	035736	004737	017134			JSR	PC,WAITF		;WAIT FOR SSR TO SET		
6537	035742	016501	000000			MOV	TSSR(R5),R1		;GET TSSR CONTENTS		
6538	035746	012702	100204			MOV	#SSR!SC!BIT2,R2		;SET UP EXPECTED		
6539	035752	020102				CMP	R1,R2		;ARE THEY EQUAL		
6540	035754	001406				BEQ	3704		;BR, IF OK		
6541	035756	004737	020116			JSR	PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS		
6545	035762					ERRHRD	ERRNO,T26TRL,PKTSSR		;TSSR INCORRECT AFTER REREAD DATA		
	035762	104456								TRAP	C\$ERHRD
	035764	000371								.WORD	249
	035766	050472								.WORD	T26TR.
	035770	011710								.WORD	PKTSSR
6546	035772				3704:	CKLOOP			;LOOP IF SELECTED		
	035772	104406								TRAP	C\$CLP1
6547											
6548											
6549											
6550											
6551											
6552											
6553											
6554	035774	013701	045756			MOV	T26BFR+6,R1		;GET MESSAGE BUFFER		
6555	036000	010102				MOV	R1,R2		;SET UP EXPECTED		
6556	036002	052702	010000			BIS	#BIT12,R2		;SET THE RLL BIT IN EXPECTED		
6557	036006	020102				CMP	R1,R2		;ARE THEY EQUAL		
6558	036010	001406				BEQ	3804		;BR, IF EQUAL (ALL IS WELL)		
6559	036012	004737	020116			JSR	PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS		
6563	036016					ERRHRD	ERRNO,T26LON,EXPREC		;THE RLL BIT WAS NOT SET IN XSTO		
	036016	104456								TRAP	C\$ERHRD
	036020	000372								.WORD	250
	036022	050240								.WORD	T26LON
	036024	016360								.WORD	EXPREC
6564	036026				3804:						
6565	036026					ENDSUB			;***** END SUBTEST *****		
	036026								L10053:		
	036026	104403								TRAP	C\$ESUB
6566	036030	023727	002170	000031		CMP	FATFLG,#25.		;IS ERROR COUNT AT 25		

[illegible]

```

036134 011676
6625 036136 104406      261:  CKLOOP                      ;LOOP IF SELECTED      .WORD  SFIMSG
                                TRAP  C$CLP1
036136 104406
6626
6627 ;*****
6628 ;
6629 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6630 ;
6631 ;*****
6632
6633 036140 004737 010444      JSR    PC,REWIND          ;CALL TAPE REWIND COMMAND
6634 036144 103413      BCS    301          ;BR, IF NO PROBLEM
6635 036146 016501 000000      MOV    TSSR(R5),R1      ;GET TSSR
6636 036152 012702 000200      MOV    #SSR,R2          ;SET UP EXPECTED TSSR
6637 036156 010004      MOV    R0,R4          ;PACKET ADDRESS SET UP
6638 036160 004737 020116      JSR    PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
6642 036164      ERRHRD  ERRNO,T26RWN,PKTSSR      ;REWIND NOT ACCEPTED
                                TRAP  C$ERHRD
                                .WORD  253
                                .WORD  T26RWN
                                .WORD  PKTSSR
036164 104456
036166 000375
036170 047414
036172 011710
6643 036174 104406      301:  CKLOOP                      ;LOOP IF SELECTED      .WORD  C$CLP1
                                TRAP  C$CLP1
036174 104406
6644
6645 ;*****
6646 ;
6647 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6648 ;
6649 ;*****
6650
6651 036176 013701 045756      MOV    T26BFR+6,R1      ;PICK UP XSTO
6652 036202 010102      MOV    R1,R2          ;SET UP EXPECTED
6653 036204 052702 000002      BIS    #BIT1,R2      ;SET BOT BIT IN EXPECTED
6654 036210 020102      CMP    R1,R2          ;DOES EXP = REC'D
6655 036212 001406      BEQ    401          ;BR, IF EQUAL (OK)
6656 036214 004737 020116      JSR    PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
6660 036220      ERRHRD  ERRNO,T26BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
                                TRAP  C$ERHRD
                                .WORD  254
                                .WORD  T26BOT
                                .WORD  EXPREC
036220 104456
036222 000376
036224 047125
036226 016360
6661 036230 104406      401:  CKLOOP                      ;LOOP IF SELECTED      .WORD  C$CLP1
                                TRAP  C$CLP1
036230 104406
6662 036232 012703 000400      MOV    #256.,R3          ;RECORD SIZE
6663 036236 013737 003072 046062  MOV    FREE,T26RB      ;STARTING WRITE BUFFER ADDRESS
6664
6665 ;*****
6666 ;
6667 ;WRITE DATA,CVC=1,ACK COMMAND
6668 ;
6669 ;*****
6670
6671 036244 012737 140005 046060      MOV    #140005,T26PK3      ;WRITE DATA,CVC=1,ACK COMMAND
6672 036252 012704 046060      MOV    #T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
6673 036256
6674 036256 010337 046066      651:  MOV    R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
6675 036262 010465 177776      MOV    R4,TSDB(R5)      ;ISSUE COMMAND

```

```

6676 036266 004737 017134      JSR      PC, WAITF      ;WAIT FOR SSR TO SET
6677 036272 016501 000000      MOV      TSSR(R5), R1    ;GET TSSP CONTENTS
6678 036276 012702 000200      MOV      #SSR, R2      ;SET UP EXPECTED
6679 036302 020102              CMP      R1, R2      ;ARE THEY EQUAL
6680 036304 001406              BEQ      75$      ;BR, IF OK
6681 036306 004737 020116      JSR      PC, FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
6685                                ;SOFT ERROR GENERATED BECAUSE THE
6686                                ;WRITE COMMAND IS NOT BEING CHECKED
6687                                ;HERE. IT WAS CHECKED IN LEAH2
6688 036312              ERRSOF ERRNO, WRTERR, PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
        036312 104457              TRAP      C$ERSOFT
        036314 000377              .WORD    255
        036316 005011              .WORD    WRTERR
        036320 011710              .WORD    PKTSSR
6689 036322 104406 001000      75$: CKLOOP      ;LOOP IF SELECTED
        036322 012703 003072 046062      MOV      #512., R3      ;RECORD SIZE
        036330 013737 003072 046062      MOV      FREE, T26RB    ;STARTING READ BUFFER ADDRESS
6692                                ;*****
6693                                ;
6694                                ; REREAD PREVIOUS, ACK, CVC=1, OPP=1
6695                                ;
6696                                ;*****
6697                                ;
6698                                ;
6699 036336 012737 161001 046060      MOV      #161001, T26PK3    ;REREAD PREVIOUS, ACK, CVC=1, OPP=1
6700 036344 012704 046060      165$: MOV      #T26PK3, R4      ;SET UP R4 WITH PACKET ADDRESS
6701 036350 010337 046066      MOV      R3, T26SZ      ;SET UP RECORD SIZE IN PACKET
6702 036354 010465 177776      MOV      R4, TSD8(R5)    ;ISSUE COMMAND
6703 036360 004737 017134      JSR      PC, WAITF      ;WAIT FOR SSR TO SET
6704 036364 016501 000000      MOV      TSSR(R5), R1    ;GET TSSR CONTENTS
6705 036370 012702 100204      MOV      #SSR!SC!BIT2, R2 ;SET UP EXPECTED
6706 036374 020102              CMP      R1, R2      ;ARE THEY EQUAL
6707 036376 001406              BEQ      170$      ;BR, IF OK
6708 036400 004737 020116      JSR      PC, FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
6712 036404              ERRHRD ERRNO, T26TRL, PKTSSR ;TSSR INCORRECT AFTER READ DATA
        036404 104456              TRAP      C$ERHRD
        036406 000400              .WORD    256
        036410 050472              .WORD    T26TRL
        036412 011710              .WORD    PKTSSR
6713 036414 104406 001000      170$: CKLOOP      ;LOOP IF SELECTED
        036414 104406              TRAP      C$CLP1
6714                                ;*****
6715                                ;
6716                                ; READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
6717                                ;
6718                                ;*****
6719                                ;
6720                                ;
6721 036416 013701 045756      MOV      T26BFR+6, R1    ;GET MESSAGE BUFFER
6722 036422 010102              MOV      R1, R2      ;SET UP EXPECTED
6723 036424 052702 040000      BIS      #BIT14, R2    ;SET THE RLS BIT IN EXPECTED
6724 036430 020102              CMP      R1, R2      ;ARE THEY EQUAL
6725 036432 001406              BEQ      180$      ;BR, IF EQUAL (ALL IS WELL)
6726 036434 004737 020116      JSR      PC, FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
6730 036440 104456              ERRHRD ERRNO, T26LOP, EXPREC ;THE RLL BIT WAS NOT SET IN XST0
        036440 104456              TRAP      C$ERHRD

```

```

036442 000401
036444 050322
036446 016360
6731 036450 180$:
6732 036450 013701 045754 MOV T26BFR+4,R1 ;PICK UP RESIDUAL BYTE COUNTER
6733 036454 012702 000400 MOV #256.,R2 ;THIS SHOULD BE THE DIFFERENCE
6734 036460 020102 CMP R1,R2 ;IS THE DIFFERENCE CORRECT
6735 036462 001406 BEQ 190$ ;BR, IF CORRECT
6736 036464 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6740 036470 ERRHRD ERRNO,T26PBP,EXPREC ;RBPCT NOT CORRECT
036470 104456 TRAP C$ERHRD
036472 000402 .WORD 258
036474 050404 .WORD T26PBP
036476 016360 .WORD EXPREC
6741 036500 190$: CKLOOP ;LOOP IF SELECTED
036500 104406 TRAP C$CLP1
6742 036502 012703 001000 MOV #512.,R3 ;RECORD SIZE
6743 036506 013737 003072 046062 MOV FREE,T26RB ;STARTING READ BUFFER ADDRESS
6744
6745 ;*****
6746 ;
6747 ;REREAD PREVIOUS,ACK,CVC=1,OPP=0
6748 ;
6749 ;*****
6750
6751 036514 012737 141001 046060 MOV #141001,T26PK3 ;REREAD PREVIOUS,ACK,CVC=1,OPP=0
6752 036522 012704 046060 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
6753 036526 010337 046066 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
6754 036532 010465 177776 MOV R4,TSD8(R5) ;ISSUE COMMAND
6755 036536 004737 017134 JSR PC,WAITF ;WAIT FOR SSR TO SET
6756 036542 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
6757 036546 012702 100204 MOV #SSR!SC!BIT2,R2 ;SET UP EXPECTED
6758 036552 020102 CMP R1,R2 ;ARE THEY EQUAL
6759 036554 001406 BEQ 270$ ;BR, IF OK
6760 036556 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6764 036562 ERRHRD ERRNO,T26TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
036562 104456 TRAP C$ERHRD
036564 000403 .WORD 259
036566 050472 .WORD T26TRL
036570 011710 .WORD PKTSSR
6765 036572 270$: CKLOOP ;LOOP IF SELECTED
036572 104406 TRAP C$CLP1
6766
6767 ;*****
6768 ;
6769 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6770 ;
6771 ;*****
6772
6773 036574 013701 045756 MOV T26BFR+6,R1 ;GET MESSAGE BUFFER
6774 036600 010102 MOV R1,R2 ;SET UP EXPECTED
6775 036602 052702 040000 BIS #BIT14,R2 ;SET THE RLS BIT IN EXPECTED
6776 036606 020102 CMP R1,R2 ;ARE THEY EQUAL
6777 036610 001406 BEQ 280$ ;BR, IF EQUAL (ALL IS WELL)
6778 036612 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6782 036616 ERRHRD ERRNO,T26LOP,EXPREC ;THE RLL BIT WAS NOT SET IN XSTO
036616 104456 TRAP C$ERHRD

```

	036620	000404					.WORD	260
	036622	050322					.WORD	T26LOP
	036624	016360					.WORD	EXPREC
6783	036626			280\$:				
6784	036626	013701	045754		MOV	T26BFR+4,R1		;PICK UP RESIDUAL BYTE COUNTER
6785	036632	012702	000400		MOV	#256.,R2		;THIS SHOULD BE THE DIFFERENCE
6786	036636	020102			CMP	R1,R2		;IS THE DIFFERENCE CORRECT
6787	036640	001405			BEQ	290\$		;BR, IF CORRECT
6791	036644				ERRHRD	ERRNO,T26PBP,EXPREC		;RBPCR NOT CORRECT
	036644	104456					TRAP	C\$ERHRD
	036646	000404					.WORD	260
	036650	050404					.WORD	T26PBP
	036652	016360					.WORD	EXPREC
6792	036654			290\$:	CKLOOP			;LOOP IF SELECTED
	036654	104406					TRAP	C\$CLP1
6793	036656				ENDSUB			; >>>>>>>>> END SUBTEST >>>>>>>>>
	036656						L10054:	
	036656	104403					TRAP	C\$ESUB
6794	036660	023727	002170	000031	CMP	FATFLG,#25.		;IS ERROR COUNT AT 25
6795	036666	002402			BLT	999\$		;BR, IF LESS THAN 25
6796	036670	004737	020170		JSR	PC,CKDROP		;TRY TO DROP THE UNIT
6797	036674			999\$:				

```

6799      ; *
6800      ;
6801      ; TEST 6, SUBTEST 7
6802      ;
6803      ;
6804      ; VERIFIES THAT THE REREAD NEXT COMMAND WITH OPP=0
6805      ; AND SWB=0 OPERATES PROPERLY. THE TAPE IS FIRST
6806      ; REWOUND AND THEN WRITTEN WITH A SERIES OF TEST
6807      ; RECORDS VARYING IN LENGTH AND DATA CONTENT. THE TAPE
6808      ; IS THEN REWOUND AGAIN. FOR EACH TEST RECORD, THE
6809      ; TAPE IS SPACED FORWARD ONE RECORD AND A REREAD
6810      ; NEXT COMMAND ISSUED. RESULTS (STATUS, DATA,
6811      ; ETC.) ARE VERIFIED. THE BYTE COUNT ON EACH REREAD
6812      ; NEXT COMMAND IS SET TO THE LENGTH OF THE EXPECTED
6813      ; RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD OCCUR.
6814      ;
6815      ;
6816      ; -
6817
6818
6819      036674          BGNSUB                                ; >>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
        036674                                          T2.7:
        036674    104402                                     TRAP      C$BSUB
6820      036676    004737    050650          JSR      PC,T26REST          ;SET COMMAND PACKET
6821      036702    004737    050742          JSR      PC,T26RT2         ;SET UP OTHER COMMAND PACKET
6822      036706    004737    051004          JSR      PC,T26RT3         ;SET UP OTHER COMMAND PACKET
6823
6824      ; *****
6825      ;
6826      ; ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6827      ;
6828      ; *****
6829
6830      036712    004737    016J60          JSR      PC,SOFINIT          ;DO INITIALIZE ON CONTROLLER
6831      036716    103407                    BCS      20$              ;BR IF INIT WAS OK
6832      036720    004737    020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6833      036724    010001                    MOV      R0,R1             ;CONTENTS OF TSSR REGISTER
6834      036726                    ERDZF   ERRNO,SFIERR,SFIMSG          ;FATAL ERROR TSSR WAS NOT OK
        036726    104455                                     TRAP      C$ERDF
        036730    000405                                     .WORD     261
        036732    003550                                     .WORD     SFIERR
        036734    011676                                     .WORD     SFIMSG
6835      036736
6836      20$:
6837
6838      036736    012704    045730          MOV      @T26PACKET,R4       ;SUBROUTINE NEEDS PACKET ADDRESS
6839
6840      ; *****
6841      ;
6842      ; WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6843      ;
6844      ; *****
6845
6846      036742    004737    010342          JSR      PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
6847      036746    103407                    BCS      26$              ;BR, IF COMMAND ISSUED OK
6848      036750    004737    020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6849      036754    010001                    MOV      R0,R1             ;SAVE CONTENTS OF TSSR
6850      036756                    ERHRD   ERRNO,WRTMSG,SFIMSG          ;WRITE CHARACTERISTIC FAILED

```

036756	104456					TRAP	C\$ERHRD
036760	000406					.WORD	262
036762	004754					.WORD	WRTMSG
036764	011676					.WORD	SFMSG
6856	036766			26\$: CKLOOP	;LOOP IF SELECTED	TRAP	C\$CLP1
	036766	104406					
6857							
6858							
6859							
6860							
6861							
6862							
6863							
6864	036770	004737	010444	JSR	PC,REWIND		
6865	036774	103413		BCS	30\$		
6866	036776	016501	000000	MOV	TSSR(R5),R1		
6867	037002	012702	000200	MOV	\$SSR,R2		
6868	037006	010004		MOV	R0,R4		
6869	037010	004737	020116	JSR	PC,FATCHK		
6873	037014			ERRHRD	ERRNO,T26RWN,PKTSSR		
	037014	104456					
	037016	000407				TRAP	C\$ERHRD
	037020	047414				.WORD	263
	037022	011710				.WORD	T26RWN
6874	037024					.WORD	PKTSSR
	037024	104406		30\$: CKLOOP	;LOOP IF SELECTED	TRAP	C\$CLP1
6875							
6876							
6877							
6878							
6879							
6880							
6881							
6882	037026	013701	045756	MOV	T26BFR+6,R1		
6883	037032	010102		MOV	R1,R2		
6884	037034	052702	000002	BIS	\$BIT1,R2		
6885	037040	020102		CMP	R1,R2		
6886	037042	001406		BEQ	40\$		
6887	037044	004737	020116	JSR	PC,FATCHK		
6891	037050			ERRHRD	ERRNO,T26BOT,EXPREC		
	037050	104456					
	037052	000410				TRAP	C\$ERHRD
	037054	047125				.WORD	264
	037056	016360				.WORD	T26BOT
6892	037060					.WORD	EXPREC
	037060	104406		40\$: CKLOOP	;LOOP IF SELECTED	TRAP	C\$CLP1
6893	037062	012703	000400	MOV	\$256.,R3		
6894	037066	013737	003072	MOV	FREE,T26R8		
6895			046062				
6896							
6897							
6898							
6899							
6900							
6901							
6902	037074	012737	140005	MOV	\$140005,T26PK3		
6903	037102	012704	046060	MOV	\$T26PK3,R4		

```

6904 037106
6905 037106 010300
6906 037110 004737 020410
6907 037114 010337 046066
6908 037120 010465 177776
6909 037124 004737 017134
6910 037130 016501 000000
6911 037134 012702 000200
6912 037140 020102
6913 037142 001406
6914 037144 004737 020116
6918
6919
6920
6921 037150
        037150 104457
        037152 000411
        037154 005011
        037156 011710
6922 037160
        037160 104406
6923 037162 005723
6924 037164 022703 000414
6925 037170 001346
6926 037172
        037172 104406
6927 037174
6928
6929
6930
6931
6932
6933
6934
6935 037174 004737 010444
6936 037200 103413
6937 037202 016501 000000
6938 037206 012702 000200
6939 037212 010004
6940 037214 004737 020116
6944 037220
        037220 104456
        037222 000412
        037224 047414
        037226 011710
6945 037230
        037230 104406
6946
6947
6948
6949
6950
6951
6952
6953 037232 013701 045756
6954 037236 010102
6955 037240 052702 000002

65$:
        MOV      R3,R0
        JSR      PC,FILLMEM
        MOV      R3,T26SZ
        MOV      R4,TSD8(R5)
        JSR      PC,WAITF
        MOV      TSSR(R5),R1
        MOV      #SSR,R2
        CMP      R1,R2
        BEQ      75$
        JSR      PC,FATCHK
;SET PATTERN IN CORRECT REGISTER
;FILL MEMORY WITH RECORD SIZE
;SET UP RECORD SIZE IN PACKET
;ISSUE COMMAND
;WAIT FOR SSR TO SET
;GET TSSR CONTENTS
;SET UP EXPECTED
;ARE THEY EQUAL
;BR, IF OK
;INC AND CHECK FOR MORE THAN 25 ERRORS
;SOFT ERROR GENERATED BECAUSE THE
;WRITE COMMAND IS NOT BEING CHECKED
;HERE. IT WAS CHECKED IN LEAH2
;TSSR INCORRECT AFTER WRITE DATA
        TRAP     C$ERSOFT
        .WORD    265
        .WORD    WRTERR
        .WORD    PKTSSR

75$:  CKLOOP
;LOOP IF SELECTED
        TRAP     C$CLP1
        TST      (R3)+
        CMP      #268.,R3
        BNE      65$
;BUMP RECORD SIZE
;END OF RECORD YET
;BR, IF MORE RECORDS TO WRITE
;LOOP IF SELECTED
        TRAP     C$CLP1

80$:  CKLOOP
120$:
;*****
;
;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
;
;*****
        JSR      PC,REWIND
        BCS      130$
        MOV      TSSR(R5),R1
        MOV      #SSR,R2
        MOV      R0,R4
        JSR      PC,FATCHK
        ERRHRD   ERRNO,T26RWN,PKTSSR
;CALL TAPE REWIND COMMAND
;BR, IF NO PROBLEM
;GET TSSR
;SET UP EXPECTED TSSR
;PACKET ADDRESS SET UP
;INC AND CHECK FOR MORE THAN 25 ERRORS
;REWIND NOT ACCEPTED
        TRAP     C$ERHRD
        .WORD    266
        .WORD    T26RWN
        .WORD    PKTSSR

130$:  CKLOOP
;LOOP IF SELECTED
        TRAP     C$CLP1
;*****
;
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
;
;*****
        MOV      T268FR+6,R1
        MOV      R1,R2
        BIS      #BIT1,R2
;PICK UP XST0
;SET UP EXPECTED
;SET BOT BIT IN EXPECTED

```

```

6956 037244 020102      CMP      R1,R2      ;DOES EXP = REC'D
6957 037246 001406      BEQ      140$      ;BR, IF EQUAL (OK)
6958 037250 004737 020116 JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
6962 037254      ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
        037254 104456      TRAP      C$ERHRD
        037256 000413      .WORD    267
        037260 047125      .WORD    T26BOT
        037262 016360      .WORD    EXPREC
6963 037264      140$: CKLOOP      ;LOOP IF SELECTED
        037264 104406      TRAP      C$CLP1
6964 037266 012737 000400 046112 MOV      #256.,T26RSZ      ;STORE START RECORD SIZE
6965 037274 000420      BR      150$      ;SKIP THE SPACE THIS TIME
6966
6967 ;*****
6968 ;
6969 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
6970 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
6971 ;
6972 ;*****
6973
6974 037276 012703 000001 145$: MOV      #1,R3      ;SPACE ONE RECORD PARAMETER
6975 037302 004737 010144 JSR      PC,SPACE      ;CALL SPACE ROUTINE
6976 037306 103413 BCS      150$      ;BR, IF NO PROBLEM WITH SPACE COMMAND
6977 037310 016501 000000 MOV      TSSR(R5),R1      ;GET TSSR
6978 037314 012702 000200 MOV      #SSR,R2      ;SET UP EXPECTED TSSR
6979 037320 010004 MOV      R0,R4      ;PACKET ADDRESS SET UP
6980 037322 004737 020116 JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
6984 037326      ERRHRD  ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
        037326 104456      TRAP      C$ERHRD
        037330 000414      .WORD    268
        037332 046527      .WORD    T26SC
        037334 016360      .WORD    EXPREC
6985 037336      150$:
6986 037336 013703 046112 MOV      T26RSZ,R3      ;RECORD SIZE
6987 037342 013737 003072 046062 MOV      FREE,T26RB      ;STARTING READ BUFFER ADDRESS
6988
6989 ;*****
6990 ;
6991 ;REREREAD DATA,CVC=1,ACK COMMAND
6992 ;
6993 ;*****
6994
6995 037350 012737 141401 046060 165$: MOV      #141401,T26PK3      ;REREREAD DATA,CVC=1,ACK COMMAND
6996 037356 012704 046060 MOV      #T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
6997 037362 010337 046066 MOV      R3,T26SZ      ;SET UP RECORD SIZE IN PACKET
6998 037366 010465 177776 MOV      R4,TSDR(R5)      ;ISSUE COMMAND
6999 037372 004737 017134 JSR      PC,WAITF      ;WAIT FOR SSR TO SET
7000 037376 016501 000000 MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
7001 037402 012702 000200 MOV      #SSR,R2      ;SET UP EXPECTED
7002 037406 020102 CMP      R1,R2      ;ARE THEY EQUAL
7003 037410 001406 BEQ      170$      ;BR, IF OK
7004 037412 004737 020116 JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
7008 037416      ERRHRD  ERRNO,T26WDC,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
        037416 104456      TRAP      C$ERHRD
        037420 000415      .WORD    269
        037422 047750      .WORD    T26WDC
        037424 011710      .WORD    PKTSSR

```

7009	037426			170\$:	CKLOOP	; LOOP IF SELECTED			
	037426	104406					TRAP	C\$CLP1	
7010	037430	013702	003072		MOV FREE,R2	; CURRENT BUFFER ADDRESS TO R2			
7011	037434	010304			MOV R3,R4	; CURRENT RECORD SIZE			
7012	037436	162704	000400		SUB #256.,R4	; FIRST LOCATION IN BUFFER			
7013	037442	060204		173\$:	ADD R2,R4	; SET UP POINTER			
7014	037444	021403			CMP (R4),R3	; CHECK DATA READ (R3=DATA ALSO)			
7015	037446	001410			BEG 180\$	; BR, IF ALL IS WELL			
7016	037450	011401			MOV (R4),R1	; RECD DATA			
7017	037452	010302			MOV R3,R2	; EXPECTED DATA			
7018	037454	004737	020116		JSR PC,FATCHK	; INC AND CHECK FOR MORE THAN 25 ERRORS			
7022	037460				ERRHRD	ERRNO,T26DTA,EXPREC	; DATA READ NOT = WRITTEN		
	037460	104456					TRAP	C\$ERHRD	
	037462	000416					.WORD	270	
	037464	047172					.WORD	T26DTA	
	037466	016360					.WORD	EXPREC	
7023	037470			180\$:	CKLOOP	; LOOP IF SELECTED			
	037470	104406					TRAP	C\$CLP1	
7024	037472	005724			TST (R4)+	; BUMP TO NEXT LOCATION			
7025	037474	160204			SUB R2,R4	; CORRECT RECORDS SIZE VALUE			
7026	037476	020403			CMP R4,R3	; END OF RECORD YET			
7027	037500	001360			BNE 173\$	; BR, IF NOT AT END OF RECORD			
7028	037502	005723			TST (R3)+	; BUMP RECORD SIZE			
7029	037504	010337	046112		MOV R3,T26RSZ	; STORE PRESENT RECORD SIZE			
7030	037510	022703	000410		CMP #264.,R3	; END OF RECORD YET			
7031	037514	001270			BNE 145\$	; BR, IF MORE RECORDS TO READ			
7032	037516			190\$:	CKLOOP	; LOOP IF SELECTED			
	037516	104406					TRAP	C\$CLP1	
7033	037520				ENDSUB	; >>>>>>>>> END SUBTEST >>>>>>>>			
	037520					L10055:			
	037520	104403					TRAP	C\$ESUB	
7034	037522	023727	002170	000031	CMP FATFLG,#25.	; IS ERROR COUNT AT 25			
7035	037530	002402			BLT 999\$	; BR, IF LESS THAN 25			
7036	037532	004737	020170		JSR PC,CKDROP	; TRY TO DROP THE UNIT			
7037	037536			999\$:					

Address	Op Code	Op 1	Op 2	Op 3	Op 4	Op 5	Op 6	Op 7	Op 8	Op 9	Op 10	Op 11	Op 12	Op 13	Op 14	Op 15	Op 16	Op 17	Op 18	Op 19	Op 20	Op 21	Op 22	Op 23	Op 24	Op 25	Op 26	Op 27	Op 28	Op 29	Op 30	Op 31	Op 32	Op 33	Op 34	Op 35	Op 36	Op 37	Op 38	Op 39	Op 40	Op 41	Op 42	Op 43	Op 44	Op 45	Op 46	Op 47	Op 48	Op 49	Op 50	Op 51	Op 52	Op 53	Op 54	Op 55	Op 56	Op 57	Op 58	Op 59	Op 60	Op 61	Op 62	Op 63	Op 64	Op 65	Op 66	Op 67	Op 68	Op 69	Op 70	Op 71	Op 72	Op 73	Op 74	Op 75	Op 76	Op 77	Op 78	Op 79	Op 80	Op 81	Op 82	Op 83	Op 84	Op 85	Op 86	Op 87	Op 88	Op 89	Op 90	Op 91	Op 92	Op 93	Op 94	Op 95	Op 96	Op 97	Op 98	Op 99	Op 100	Op 101	Op 102	Op 103	Op 104	Op 105	Op 106	Op 107	Op 108	Op 109	Op 110	Op 111	Op 112	Op 113	Op 114	Op 115	Op 116	Op 117	Op 118	Op 119	Op 120	Op 121	Op 122	Op 123	Op 124	Op 125	Op 126	Op 127	Op 128	Op 129	Op 130	Op 131	Op 132	Op 133	Op 134	Op 135	Op 136	Op 137	Op 138	Op 139	Op 140	Op 141	Op 142	Op 143	Op 144	Op 145	Op 146	Op 147	Op 148	Op 149	Op 150	Op 151	Op 152	Op 153	Op 154	Op 155	Op 156	Op 157	Op 158	Op 159	Op 160	Op 161	Op 162	Op 163	Op 164	Op 165	Op 166	Op 167	Op 168	Op 169	Op 170	Op 171	Op 172	Op 173	Op 174	Op 175	Op 176	Op 177	Op 178	Op 179	Op 180	Op 181	Op 182	Op 183	Op 184	Op 185	Op 186	Op 187	Op 188	Op 189	Op 190	Op 191	Op 192	Op 193	Op 194	Op 195	Op 196	Op 197	Op 198	Op 199	Op 200	Op 201	Op 202	Op 203	Op 204	Op 205	Op 206	Op 207	Op 208	Op 209	Op 210	Op 211	Op 212	Op 213	Op 214	Op 215	Op 216	Op 217	Op 218	Op 219	Op 220	Op 221	Op 222	Op 223	Op 224	Op 225	Op 226	Op 227	Op 228	Op 229	Op 230	Op 231	Op 232	Op 233	Op 234	Op 235	Op 236	Op 237	Op 238	Op 239	Op 240	Op 241	Op 242	Op 243	Op 244	Op 245	Op 246	Op 247	Op 248	Op 249	Op 250	Op 251	Op 252	Op 253	Op 254	Op 255	Op 256	Op 257	Op 258	Op 259	Op 260	Op 261	Op 262	Op 263	Op 264	Op 265	Op 266	Op 267	Op 268	Op 269	Op 270	Op 271	Op 272	Op 273	Op 274	Op 275	Op 276	Op 277	Op 278	Op 279	Op 280	Op 281	Op 282	Op 283	Op 284	Op 285	Op 286	Op 287	Op 288	Op 289	Op 290	Op 291	Op 292	Op 293	Op 294	Op 295	Op 296	Op 297	Op 298	Op 299	Op 300	Op 301	Op 302	Op 303	Op 304	Op 305	Op 306	Op 307	Op 308	Op 309	Op 310	Op 311	Op 312	Op 313	Op 314	Op 315	Op 316	Op 317	Op 318	Op 319	Op 320	Op 321	Op 322	Op 323	Op 324	Op 325	Op 326	Op 327	Op 328	Op 329	Op 330	Op 331	Op 332	Op 333	Op 334	Op 335	Op 336	Op 337	Op 338	Op 339	Op 340	Op 341	Op 342	Op 343	Op 344	Op 345	Op 346	Op 347	Op 348	Op 349	Op 350	Op 351	Op 352	Op 353	Op 354	Op 355	Op 356	Op 357	Op 358	Op 359	Op 360	Op 361	Op 362	Op 363	Op 364	Op 365	Op 366	Op 367	Op 368	Op 369	Op 370	Op 371	Op 372	Op 373	Op 374	Op 375	Op 376	Op 377	Op 378	Op 379	Op 380	Op 381	Op 382	Op 383	Op 384	Op 385	Op 386	Op 387	Op 388	Op 389	Op 390	Op 391	Op 392	Op 393	Op 394	Op 395	Op 396	Op 397	Op 398	Op 399	Op 400	Op 401	Op 402	Op 403	Op 404	Op 405	Op 406	Op 407	Op 408	Op 409	Op 410	Op 411	Op 412	Op 413	Op 414	Op 415	Op 416	Op 417	Op 418
---------	---------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------

```

7091
7092
7093
7094
7095
7096
7097 037632 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
7098 037636 103413      BCS      30$           ;BR, IF NO PROBLEM
7099 037640 016501 000000      MOV      TSSR(R5),R1    ;GET TSSR
7100 037644 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED TSSR
7101 037650 010004      MOV      R0,R4      ;PACKET ADDRESS SET UP
7102 037652 004737 020116      JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7106 037656      ERRMRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      037656 104456      TRAP      C$ERRMRD
      037660 000421      .WORD    273
      037662 047414      .WORD    T26RWN
      037664 011710      .WORD    PKTSSR
7107 037666      30$:    CKLOOP           ;LOOP IF SELECTED      TRAP      C$CLP1
      037666 104406
7108
7109
7110
7111
7112
7113
7114
7115 037670 013701 045756      MOV      T26RFR+6,R1    ;PICK UP XSTO
7116 037674 010102      MOV      R1,R2      ;SET UP EXPECTED
7117 037676 052702 000002      BIS      #BIT1,R2    ;SET BOT BIT IN EXPECTED
7118 037702 020102      CMP      R1,R2      ;DOES EXP = REC'D
7119 037704 001406      BEQ      40$         ;BR, IF EQUAL (OK)
7120 037706 004737 020116      JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7124 037712      ERRMRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      037712 104456      TRAP      C$ERRMRD
      037714 000422      .WORD    274
      037716 047125      .WORD    T26BOT
      037720 016360      .WORD    EXPREC
7125 037722      40$:    CKLOOP           ;LOOP IF SELECTED      TRAP      C$CLP1
      037722 104406
7126 037724 012703 000400      MOV      #256.,R3    ;RECORD SIZE
7127 037730 013737 003072 046062 MOV      FREE,T26RB    ;STARTING WRITE BUFFER ADDRESS
7128
7129
7130
7131
7132
7133
7134
7135 037736 012737 150005 046060 MOV      #150005,T26PK3 ;WRITE DATA,CVC=1,ACK,SWB COMMAND
7136 037744 012704 046060      MOV      #T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
7137 037750      65$:    MOV      R3,R0      ;SET PATTERN IN CORRECT REGISTER
7138 037750 010300      JSR      PC,FILLMEM    ;FILL MEMORY WITH RECORD SIZE
7139 037752 004737 020410      MOV      R3,T26SZ    ;SET UP RECORD SIZE IN PACKET
7140 037756 010337 046066      MOV      R4,TSD8(R5)  ;ISSUE COMMAND
7141 037762 010465 177776      JSR      PC,WAITF    ;WAIT FOR SSR TO SET
7142 037766 004737 017134      MOV      TSSR(R5),R1 ;GET TSSR CONTENTS
7143 037772 016501 000000

```

```

7144 037776 012702 000200      MOV    #SSR,R2      ;SET UP EXPECTED
7145 040002 020102      CMP     R1,R2      ;ARE THEY EQUAL
7146 040004 001406      BEQ     758      ;BR, IF OK
7147 040006 004737 020116      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7151                                ;SOFT ERROR GENERATED BECAUSE THE
7152                                ;WRITE COMMAND IS NOT BEING CHECKED
7153                                ;HERE. IT WAS CHECKED IN LEAM2
7154 040012      ERRSOF ERRNO,WRTERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP    C$ERSOFT
                                .WORD   275
                                .WORD   WRTERR
                                .WORD   PKTSSR
                                TRAP    C$CLP1
7155 040022      758:    CKLOOP      ;LOOP IF SELECTED
                                TRAP    C$CLP1
                                .WORD   275
                                .WORD   WRTERR
                                .WORD   PKTSSR
7156 040024 005723      TST     (R3)+    ;BUMP RECORD SIZE
7157 040026 022703 000414      CMP     #268.,R3    ;END OF RECORD YET
7158 040032 001346      BNE     658      ;BR, IF MORE RECORDS TO WRITE
7159 040034      808:    CKLOOP      ;LOOP IF SELECTED
                                TRAP    C$CLP1
                                .WORD   275
                                .WORD   WRTERR
                                .WORD   PKTSSR
7160 040036      1208:
;*****
;
;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
;
;*****
7161
7162
7163
7164
7165
7166
7167
7168 040036 004737 010444      JSR     PC,REWIND    ;CALL TAPE REWIND COMMAND
7169 040042 103413      BCS     1308      ;BR, IF NO PROBLEM
7170 040044 016501 000000      MOV     TSSR(R5),R1    ;GET TSSR
7171 040050 012702 000200      MOV     #SSR,R2      ;SET UP EXPECTED TSSR
7172 040054 010004      MOV     R0,R4      ;PACKET ADDRESS SET UP
7173 040056 004737 020116      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7177 040062      ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C$ERHRD
                                .WORD   276
                                .WORD   T26RWN
                                .WORD   PKTSSR
7178 040072      1308:    CKLOOP      ;LOOP IF SELECTED
                                TRAP    C$CLP1
                                .WORD   276
                                .WORD   T26RWN
                                .WORD   PKTSSR
7179
7180
7181
7182
7183
7184
7185
7186 040074 013701 045756      MOV     T26BFR+6,R1    ;PICK UP XSTO
7187 040100 010102      MOV     R1,R2      ;SET UP EXPECTED
7188 040102 052702 000002      BIS     #BIT1,R2    ;SET BOT BIT IN EXPECTED
7189 040106 020102      CMP     R1,R2      ;DOES EXP = REC'D
7190 040110 001406      BEQ     1408      ;BR, IF EQUAL (OK)
7191 040112 004737 020116      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7195 040116      ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP    C$ERHRD
                                .WORD   277
                                .WORD   T26BOT
7196 040116 104456
7197 040120 000425
7198 040122 047125

```

```

7196 040124 016360
7196 040126 104406
7197 040130 012737 000400 046112
7198 040136 000420
7199
7200
7201
7202
7203
7204
7205
7206
7207 040140 012703 000001
7208 040144 004737 010144
7209 040150 103413
7210 040152 016501 000000
7211 040156 012702 000200
7212 040162 010004
7213 040164 004737 020116
7217 040170
040170 104456
040172 000426
040174 046527
040176 016360
7218 040200
7219 040200 013703 046112
7220 040204 013737 003072 046062
7221
7222
7223
7224
7225
7226
7227
7228 040212 012737 151401 046060
7229 040220 012704 046060
7230 040224 010337 046066
7231 040230 010465 177776
7232 040234 004737 017134
7233 040240 016501 000000
7234 040244 012702 000200
7235 040250 020102
7236 040252 001406
7237 040254 004737 020116
7241 040260
040260 104456
040262 000427
040264 047750
040266 011710
7242 040270
040270 104406
7243 040272 013702 003072
7244 040276 010304
7245 040300 162704 000400
7246 040304 060204
7247 040306 021403

140: CKLOOP                                ;LOOP IF SELECTED                .WORD  EXPREC
                                           ;START RECORD SIZE              TRAP    C$CLP1
                                           ;SKIP SPACE THIS TIME

;*****
;
;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
;
;*****

145:  MOV    #1,R3                        ;SPACE ONE RECORD PARAMETER
      JSR    PC,SPACE                    ;CALL SPACE ROUTINE
      BCS    150:                        ;BR, IF NO PROBLEM WITH SPACE COMMAND
      MOV    TSSR(R5),R1                 ;GET TSSR
      MOV    #SSR,R2                     ;SET UP EXPECTED TSSR
      MOV    R0,R4                       ;PACKET ADDRESS SET UP
      JSR    PC,FATCHK                   ;INC AND CHECK FOR MORE THAN 25 ERRORS
      ERHRD  ERRNO,T26SC,EXPREC          ;POSITION (SPACE RECORDS) FAILED
                                           TRAP    C$ERHRD
                                           .WORD    278
                                           .WORD    T26SC
                                           .WORD    EXPREC

150:  MOV    T26RSZ,R3                    ;RECORD SIZE
      MOV    FREE,T26RB                  ;STARTING READ BUFFER ADDRESS

;*****
;
;REREAD DATA,ACK,CVC=1,SWB COMMAND
;
;*****

165:  MOV    #151401,T26PK3              ;REREAD DATA,ACK,CVC=1,SWB COMMAND
      MOV    #T26PK3,R4                  ;SET UP R4 WITH PACKET ADDRESS
      MOV    R3,T26SZ                     ;SET UP RECORD SIZE IN PACKET
      MOV    R4,TSD8(R5)                  ;ISSUE COMMAND
      JSR    PC,WAITF                     ;WAIT FOR SSR TO SET
      MOV    TSSR(R5),R1                 ;GET TSSR CONTENTS
      MOV    #SSR,R2                     ;SET UP EXPECTED
      CMP    R1,R2                       ;ARE THEY EQUAL
      BEQ    170:                        ;BR, IF OK
      JSR    PC,FATCHK                   ;INC AND CHECK FOR MORE THAN 25 ERRORS
      ERHRD  ERRNO,T26WDC,PKTSSR         ;TSSR INCORRECT AFTER REREAD DATA
                                           TRAP    C$ERHRD
                                           .WORD    279
                                           .WORD    T26WDC
                                           .WORD    PKTSSR

170:  CKLOOP                                ;LOOP IF SELECTED                TRAP    C$CLP1
                                           ;CURRENT BUFFER ADDRESS TO R2
      MOV    FREE,R2                     ;CURRENT RECORD SIZE
      MOV    R3,R4                       ;FIRST LOCATION IN BUFFER
      SUB    #256.,R4                    ;SET UP POINTER
      ADD    R2,R4                       ;CHECK DATA READ (R3=DATA ALSO)
      CMP    (R4),R3

```

7248	040310	001410		BEG	180\$		;BR, IF ALL IS WELL		
7249	040312	011401		MOV	(R4),R1		;RECD DATA		
7250	040314	010302		MOV	R3,R2		;EXPECTED DATA		
7251	040316	004737	020116	JSR	PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS		
7255	040322			ERRHRD	ERRNO,T26DTA,EXPREC		;DATA READ NOT = WRITTEN		
	040322	104456						TRAP	C\$ERHRD
	040324	000430						.WORD	280
	040326	047172						.WORD	T26DTA
	040330	016360						.WORD	EXPREC
7256	040332			180\$:	CKLOOP		;LOOP IF SELECTED		
	040332	104406						TRAP	C\$CLP1
7257	040334	005724		TST	(R4).		;BUMP TO NEXT LOCATION		
7258	040336	160204		SUB	R2,R4		;CORRECT RECORDS SIZE VALUE		
7259	040340	020403		CMP	R4,R3		;END OF RECORD YET		
7260	040342	001360		BNE	173\$		;BR, IF NOT AT END OF RECORD		
7261	040344	005723		TST	(R3).		;BUMP RECORD SIZE		
7262	040346	010337	046112	MOV	R3,T26RSZ		;STORE RECORD SIZE		
7263	040352	022703	000410	CMP	#264.,R3		;END OF RECORD YET		
7264	040356	001270		BNE	145\$		;BR, IF MORE RECORDS TO WRITE		
7265	040360			190\$:	CKLOOP		;LOOP IF SELECTED		
	040360	104406						TRAP	C\$CLP1
7266	040362			ENDSUB			; >>>>>>>>> END SUBTEST >>>>>>>>>		
	040362						L10056:		
	040362	104403						TRAP	C\$ESUB
7267	040364	023727	002170	000031	CMP	FATFLG,#25.	;IS ERROR COUNT AT 25		
7268	040372	002402			BLT	999\$	;BR, IF LESS THAN 25		
7269	040374	004737	020170		JSR	PC,CKDROP	;TRY TO DROP THE UNIT		
7270	040400			999\$:					

[illegible]

```

7326
7327 040446 012704 045730          MOV      #T26PACKET,R4          ;SUBROUTINE NEEDS PACKET ADDRESS
7328
7329          ;*****
7330          ;
7331          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTHCR)
7332          ;
7333          ;*****
7334
7335 040452 004737 010342          JSR      PC,WRTHCR          ;ISSUE WRITE CHARACTERISTICS
7336 040456 103407          BCS      26$          ;BR, IF COMMAND ISSUED OK
7337 040460 004737 020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
7341 040464 010001          MOV      R0,R1          ;SAVE CONTENTS OF TSSR
7342 040466          ERRHRD  ERRNO,WRTHMSG,SFMSG          ;WRITE CHARACTERISTICS FAILED
          040466 104456          TRAP      C$ERHRD
          040470 000432          .WORD    282
          040472 004754          .WORD    WRTHMSG
          040474 011676          .WORD    SFMSG
7343 040476          26$:      CKLOOP          ;LOOP IF SELECTED
          040476 104406          TRAP      C$CLP1
7344
7345          ;*****
7346          ;
7347          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7348          ;
7349          ;*****
7350
7351 040500 004737 010444          JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
7352 040504 103413          BCS      30$          ;BR, IF NO PROBLEM
7353 040506 016501 000000          MOV      TSSR(R5),R1          ;GET TSSR
7354 040512 012702 000200          MOV      #SSR,R2          ;SET UP EXPECTED TSSR
7355 040516 010004          MOV      R0,R4          ;PACKET ADDRESS SET UP
7356 040520 004737 020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
7360 040524          ERRHRD  ERRNO,T26RWN,PKTSSR          ;REWIND NOT ACCEPTED
          040524 104456          TRAP      C$ERHRD
          040526 000433          .WORD    283
          040530 047414          .WORD    T26RWN
          040532 011710          .WORD    PKTSSR
7361 040534          30$:      CKLOOP          ;LOOP IF SELECTED
          040534 104406          TRAP      C$CLP1
7362
7363          ;*****
7364          ;
7365          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
7366          ;
7367          ;*****
7368
7369 040536 013701 045756          MOV      T26BFR+6,R1          ;PICK UP XST0
7370 040542 010102          MOV      R1,R2          ;SET UP EXPECTED
7371 040544 052702 000002          BIS      #BIT1,R2          ;SET BOT BIT IN EXPECTED
7372 040550 020102          CMP      R1,R2          ;DOES EXP = REC'D
7373 040552 001406          BEQ      40$          ;BR, IF EQUAL (OK)
7374 040554 004737 020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
7378 040560          ERRHRD  ERRNO,T26BOT,EXPREC          ;TAPE NOT AT BOT AFTER REWIND
          040560 104456          TRAP      C$ERHRD
          040562 000434          .WORD    284
          040564 047125          .WORD    T26BOT

```

```

040566 016360
7379 040570 104406 401: CKLOOP ;LOOP IF SELECTED .WORD EXPREC
040570 104406 ;TRAP C$CLP1
7380 040572 012703 000400 MOV #256.,R3 ;RECORD SIZE
7381 040576 013737 003072 046062 MOV FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
7382
7383 ;*****
7384 ;
7385 ;WRITE DATA,CVC=1,ACK COMMAND
7386 ;
7387 ;*****
7388
7389 040604 012737 140005 046060 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
7390 040612 012704 046060 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
7391 040616
7392 040616 010337 046066 651: MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
7393 040622 013777 046106 142242 MOV T26CNT,8FREE ;MOVE TAPE RECORD NUMBER TO BUFFER
7394 040630 062737 000001 046106 ADD #1,T26CNT ;NUMBER READY FOR NEXT RECORD
7395 040636 010465 177776 MOV R4,TSD8(R5) ;ISSUE COMMAND
7396 040642 004737 017134 JSR PC,WAITF ;WAIT FOR SSR TO SET
7397 040646 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
7398 040652 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
7399 040656 020102 CMP R1,R2 ;ARE THEY EQUAL
7400 040660 001406 BEQ 751 ;BR, IF OK
7401 040662 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7405 ;SOFT ERROR GENERATED BECAUSE THE
7406 ;WRITE COMMAND IS NOT BEING CHECKED
7407 ;HERE. IT WAS CHECKED IN LEAH2
7408 040666 ERRSOF ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
040666 104457 TRAP C$ERSOFT
040670 000435 .WORD 285
040672 005011 .WORD WRERR
040674 011710 .WORD PKTSSR
7409 040676 751: CKLOOP ;LOOP IF SELECTED .WORD C$CLP1
040676 104406 ;TRAP C$CLP1
7410 040700 005723 TST (R3). ;BUMP THE RECORD SIZE
7411 040702 022703 000414 CMP #268.,R3 ;MAXIMUM SIZE YET
7412 040706 001401 BEQ 1201 ;BR, IF AT END OF WRITE SEQUENCE
7413 040710 000742 BR 651 ;WRITE MORE RECORDS
7414 040712
7415 040712 005037 046106 1201: CLR T26CNT ;SET RECORD COUNTER BACK TO ZERO
7416
7417 ;*****
7418 ;
7419 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7420 ;
7421 ;*****
7422
7423 040716 004737 010444 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
7424 040722 103411 BCS 1301 ;BR, IF NO PROBLEM
7425 040724 016501 000000 MOV TSSR(R5),R1 ;GET TSSR
7426 040730 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
7427 040732 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7431 040736 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
040736 104456 TRAP C$ERHRD
040740 000436 .WORD 286
040742 047414 .WORD T26RWN

```

```

040744 011710
7432 040746 104406 130$: CKLOOP ;LOOP IF SELECTED .WORD PKTSSR
040746 104406 TRAP C$CLP1
7433
7434 ;*****
7435 ;
7436 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7437 ;
7438 ;*****
7439
7440 040750 013701 045756 MOV T26BFR+6,R1 ;PICK UP XSTO
7441 040754 010102 MOV R1,R2 ;SET UP EXPECTED
7442 040756 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
7443 040762 020102 CMP R1,R2 ;DOES EXP = REC'D
7444 040764 001406 BEQ 135$ ;BR, IF EQUAL (OK)
7445 040766 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7449 040772 ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
040772 104456 TRAP C$ERHRD
040774 000437 .WORD 287
040776 047125 .WORD T26BOT
041000 016360 .WORD EXPREC
7450 041002 135$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
041002 104406
7451 041004 012737 000400 046112 MOV #256.,T26RSZ ;STARTING RECORD SIZE
7452 041012 000420 BR 140$ ;SKIP OVER THE SPACE THIS TIME
7453
7454 ;*****
7455 ;
7456 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
7457 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
7458 ;
7459 ;*****
7460
7461 041014 012703 000001 132$: MOV #000001,R3 ;SET UP SPACE COMMAND (1 FORWARD)
7462 041020 004737 010144 JSR PC,SPACE ;CALL SPACE ROUTINE
7463 041024 103413 BCS 140$ ;BR, IF NO TROUBLE
7464 041026 016501 000000 MOV TSSR(R5),R1 ;GET TSSR
7465 041032 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
7466 041036 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
7467 041040 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7471 041044 ERRHRD ERRNO,T26SC,PKTSSR ;SPACE (FORWARD) FAILED
041044 104456 TRAP C$ERHRD
041046 000440 .WORD 288
041050 046527 .WORD T26SC
041052 011710 .WORD PKTSSR
7472 041054 140$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
041054 104406
7473 041056 013703 046112 MOV T26RSZ,R3 ;RECORD SIZE
7474 041062 013737 003072 046062 150$: MOV FREE,T26RB ;STARTING READ BUFFER ADDRESS
7475
7476 ;*****
7477 ;
7478 ;REREAD DATA,CVC=1,ACK, OPP COMMAND
7479 ;
7480 ;*****
7481
7482 041070 012737 161401 046060 MOV #161401,T26PKZ ;REREAD DATA,CVC=1,ACK, OPP COMMAND

```

7483	041076	012704	046060		165\$:	MOV	#T26PK3,R4		;SET UP R4 WITH PACKET ADDRESS
7484	041102	010337	046066			MOV	R3,T26SZ		;SET UP RECORD SIZE IN PACKET
7485	041106	010465	177776			MOV	R4,TSD8(R5)		;ISSUE COMMAND
7486	041112	004737	017134			JSR	PC,WAITF		;WAIT FOR SSR TO SET
7487	041116	016501	000000			MOV	TSSR(R5),R1		;GET TSSR CONTENTS
7488	041122	012702	000200			MOV	#SSR,R2		;SET UP EXPECTED
7489	041126	020102				CMP	R1,R2		;ARE THEY EQUAL
7490	041130	001406				BEG	170\$		;BR, IF OK
7491	041132	004737	020116			JSR	PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS
7495	041136					ERRHRD	ERRNO,T26RRF,PKTSSR		;TSSR INCORRECT AFTER REREAD DATA
	041136	104456							TRAP C\$ERHRD
	041140	000441							.WORD 289
	041142	046335							.WORD T26RRF
	041144	011710							.WORD PKTSSR
7496	041146				170\$:	CKLOOP			;LOOP IF SELECTED
	041146	104406							TRAP C\$CLP1
7497	041150	017701	141716			MOV	\$FREE,R1		;FIRST WORD FROM READ BUFFER
7498	041154	013702	046106			MOV	T26CNT,R2		;SET UP EXPECTED
7499	041160	020102				CMP	R1,R2		;IS TAPE POSITION CORRECT
7500	041162	001406				BEG	190\$		;KEEP GOING POSITION OK
7501	041164	004737	020116			JSR	PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS
7505	041170					ERRHRD	ERRNO,T26WNG,EXPREC		;TAPE POSITION INCORRECT
	041170	104456							TRAP C\$ERHRD
	041172	000442							.WORD 290
	041174	046116							.WORD T26WNG
	041176	016360							.WORD EXPREC
7506	041200				190\$:	CKLOOP			
	041200	104406							TRAP C\$CLP1
7507	041202	062737	000001	046106		ADD	#1,T26CNT		;BUMP TAPE RECORD COUNTER
7508	041210	005723				TST	(R3).		;NEXT RECORD SIZE
7509	041212	010337	046112			MOV	R3,T26RSZ		;STORE RECORD SIZE
7510	041216	022703	000412			CMP	#266.,R3		;MAX SIZE YET
7511	041222	001402				BEG	200\$		;BR, IF AT END OF THE SUBTEST
7512	041224	000137	041014			JMP	132\$		;KEEP GOING MORE RECORDS
7513	041230				200\$:				
7514	041230					ENDSUB			;>>>>>>>>> END SUBTEST >>>>>>>>>
	041230								L10057:
	041230	104403							TRAP C\$ESUB
7515	041232	023727	002170	000031		CMP	FATFLG,#25.		;IS ERROR COUNT AT 25
7516	041240	002402				BLT	999\$		;BR, IF LESS THAN 25
7517	041242	004737	020170			JSR	PC,CKDROP		;TRY TO DROP THE UNIT
7518	041246				999\$:				

7520		:	*
7521		:	
7522		:	TEST 2, SUBTEST 10
7523		:	
7524		:	VERIFIES THAT THE REREAD NEXT COMMAND WITH OPP=1
7525		:	AND SWB=1 OPERATES PROPERLY. THE TEST SEQUENCE IS
7526		:	THE SAME THAT IS USED IN SUBTEST 3. IT IS
7527		:	VERIFIED THAT DATA STORED BY THE COMMAND CONTAINS
7528		:	SWAPPED BYTES.
7529		:	
7530		:	
7531		:	
7532		:	
7533		:	-
7534		:	
7535	041246	BGNSUB	; >>>>>>>>> BEGIN SUBTEST >>>>>>>>>
	041246		T2.10:
	041246	104402	TRAP C\$BSUB
7536	041250	004737 050650	JSR PC,T26REST ;SET COMMAND PACKET
7537	041254	005037 046106	CLR T26CNT ;CLEAR TAPE RECORD COUNTER
7538	041260	004737 050742	JSR PC,T26RT2 ;SET UP OTHER COMMAND PACKET
7539	041264	004737 051004	JSR PC,T26RT3 ;SET UP OTHER COMMAND PACKET
7540			
7541			:*****
7542			:
7543			;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
7544			:
7545			:*****
7546			
7547	041270	004737 016660	JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
7548	041274	103407	BCS 20\$;BR IF INIT WAS OK
7549	041276	004737 020116	JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7553	041302	010001	MOV RO,R1 ;CONTENTS OF TSSR REGISTER
7554	041304		ERRDF ERNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
	041304	104455	TRAP C\$ERDF
	041306	000443	.WORD 291
	041310	003550	.WORD SFIERR
	041312	011676	.WORD SFIMSG
7555	041314	20\$:	
7556			
7557	041314	012704 045730	MOV @T26PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
7558			
7559			:*****
7560			:
7561			;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
7562			:
7563			:*****
7564			
7565	041320	004737 010342	JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
7566	041324	103407	BCS 26\$;BR, IF COMMAND ISSUED OK
7567	041326	004737 020116	JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7571	041332	010001	MOV RO,R1 ;SAVE CONTENTS OF TSSR
7572	041334		ERRHRD ERNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
	041334	104456	TRAP C\$ERRHD
	041336	000444	.WORD 292
	041340	004754	.WORD WRTMSG
	041342	011676	.WORD SFIMSG

```

7573 041344      26$: CKLOOP                      ;LOOP IF SELECTED
041344 104406                                TRAP C$CLP1
7574
7575 ;*****
7576 ;
7577 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7578 ;
7579 ;*****
7580
7581 041346 004737 010444      JSR PC,REWIND          ;CALL TAPE REWIND COMMAND
7582 041352 016501 000000      MOV TSSR(R5),R1        ;GET TSSR
7583 041356 012702 000200      MOV #SSR,R2            ;SET UP EXPECTED TSSR
7584 041362 103407              BCS 30$                ;BR, IF NO PROBLEM
7585 041364 010004              MOV R0,R4              ;PACKET ADDRESS SET UP
7586 041366 004737 020116      JSR PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
7590 041372              ERRHRD ERRNO,T26RWN,PKTSSR    ;REWIND NOT ACCEPTED
041372 104456                                TRAP C$ERHRD
041374 000445                                .WORD 293
041376 047414                                .WORD T26RWN
041400 011710                                .WORD PKTSSR
7591 041402      30$: CKLOOP                      ;LOOP IF SELECTED
041402 104406                                TRAP C$CLP1
7592
7593 ;*****
7594 ;
7595 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
7596 ;
7597 ;*****
7598
7599 041404 013701 045756      MOV T26BFR+6,R1          ;PICK UP XST0
7600 041410 010102              MOV R1,R2              ;SET UP EXPECTED
7601 041412 052702 000002      BIS #BIT1,R2            ;SET BOT BIT IN EXPECTED
7602 041416 020102              CMP R1,R2              ;DOES EXP = REC'D
7603 041420 001406              BEQ 40$                ;BR, IF EQUAL (OK)
7604 041422 004737 020116      JSR PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
7608 041426              ERRHRD ERRNO,T26BOT,EXPREC    ;TAPE NOT AT BOT AFTER REWIND
041426 104456                                TRAP C$ERHRD
041430 000446                                .WORD 294
041432 047125                                .WORD T26BOT
041434 016360                                .WORD EXPREC
7609 041436      40$: CKLOOP                      ;LOOP IF SELECTED
041436 104406                                TRAP C$CLP1
7610 041440 012703 000400      MOV #256.,R3            ;RECORD SIZE
7611 041444 013737 003072 046062      MOV FREE,T26RB    ;STARTING WRITE BUFFER ADDRESS
7612
7613 ;*****
7614 ;
7615 ;WRITE DATA,CVC=1,ACK COMMAND
7616 ;
7617 ;*****
7618
7619 041452 012737 140005 046060      MOV #140005,T26PK3    ;WRITE DATA,CVC=1,ACK COMMAND
7620 041460 012704 046060      MOV #T26PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
7621 041464
7622 041464 010337 046066      65$: MOV R3,T26SZ        ;SET UP RECORD SIZE IN PACKET
7623 041470 013777 046106 141374      MOV T26CNT,#FREE    ;MOVE TAPE RECORD NUMBER TO BUFFER
7624 041476 062737 000001 046106      ADD #1,T26CNT      ;NUMBER READY FOR NEXT RECORD

```

```

7625 041504 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
7626 041510 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
7627 041514 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
7628 041520 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
7629 041524 020102      CMP      R1,R2      ;ARE THEY EQUAL
7630 041526 001406      BEQ      75$      ;BR, IF OK
7631 041530 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
7635      ;SOFT ERROR GENERATED BECAUSE THE
7636      ;WRITE COMMAND IS NOT BEING CHECKED
7637      ;HERE. IT WAS CHECKED IN LEAH2
7638 041534      ERRSOF ERRNO,WRterr,PKTSSR      ;TSSR INCORRECT AFTER WRITE DATA
      041534 104457      TRAP      C$ERSOFT
      041536 000447      .WORD      29F
      041540 005011      .WORD      WRterr
      041542 011710      .WORD      PKTSSR
7639 041544      75$:      CKLOOP      ;LOOP IF SELECTED
      041544 104406      TRAP      C$CLP1
7640 041546 005723      TST      (R3)+      ;BUMP THE RECORD SIZE
7641 041550 022703 000414      CMP      #268.,R3      ;MAXIMUM SIZE YET
7642 041554 001401      BEQ      120$      ;BR, IF AT END OF WRITE SEQUENCE
7643 041556 000742      BR      65$      ;WRITE MORE RECORDS
7644 041560      120$:      CLR      T26CNT      ;SET RECORD COUNTER BACK TO ZERO
7645 041560 005037 046106
7646      ;*****
7647      ;
7648      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7649      ;
7650      ;*****
7651      ;
7652      ;*****
7653 041564 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
7654 041570 103411      BCS      130$      ;BR, IF NO PROBLEM
7655 041572 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR
7656 041576 010004      MOV      R0,R4      ;PACKET ADDRESS SET UP
7657 041600 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
7661 041604      ERRHRD ERRNO,T26RWN,PKTSSR      ;REWIND NOT ACCEPTED
      041604 104456      TRAP      C$ERHRD
      041606 000450      .WORD      296
      041610 047414      .WORD      T26RWN
      041612 011710      .WORD      PKTSSR
7662 041614      130$:      CKLOOP      ;LOOP IF SELECTED
      041614 104406      TRAP      C$CLP1
7663      ;*****
7664      ;
7665      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7666      ;
7667      ;*****
7668      ;
7669      ;*****
7670 041616 013701 045756      MOV      T26BFR+6,R1      ;PICK UP XSTO
7671 041622 010102      MOV      R1,R2      ;SET UP EXPECTED
7672 041624 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
7673 041630 020102      CMP      R1,R2      ;DOES EXP = REC'D
7674 041632 001406      BEQ      135$      ;BR, IF EQUAL (OK)
7675 041634 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
7679 041640      ERRHRD ERRNO,T26BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      041640 104456      TRAP      C$ERHRD

```

	041642	000451					.WORD	297
	041644	047125					.WORD	T26BOT
	041646	016360					.WORD	EXPREC
7680	041650				135:	CKLOOP		;LOOP IF SELECTED
	041650	104406						TRAP C1CLP1
7681	041652	012737	000400	046112		MOV #256.,T26RSZ		;START RECORD SIZE
7682	041660	000420				BR 140:		;SKIP OVER SPACE
7683								
7684								
7685								
7686								
7687								
7688								
7689								
7690								
7691	041662	012703	000001		136:	MOV #000001,R3		;SET UP SPACE COMMAND (1 FORWARD)
7692	041666	004737	010144			JSR PC,SPACE		;CALL SPACE ROUTINE
7693	041672	103413				BCS 140:		;BR, IF NO TROUBLE
7694	041674	016501	000000			MOV TSSR(R5),R1		;GET TSSR
7695	041700	012702	000200			MOV #SSR,R2		;SET UP EXPECTED TSSR
7696	041704	010004				MOV R0,R4		;PACKET ADDRESS SET UP
7697	041706	004737	020116			JSR PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS
7701	041712					ERRHRD ERNO,T26SC,PKTSSR		;SPACE (FORWARD) FAILED
	041712	104456						TRAP C1ERMRD
	041714	000452						.WORD 298
	G-1716	046527						.WORD T26SC
	041720	C11710						.WORD PKTSSR
7702	041722				140:	CKLOOP		;LOOP IF SELECTED
	041722	104406						TRAP C1CLP1
7703	041724	013703	046112			MOV T26RSZ,R3		;RECORD SIZE
7704	041730	013737	003072	046062	150:	MOV FREE,T26RB		;STARTING READ BUFFER ADDRESS
7705								
7706								
7707								
7708								
7709								
7710								
7711								
7712	041736	012737	161401	046060		MOV #161401,T26PK3		;REREAD DATA,CVC=1,ACK, OPP COMMAND
7713	041744	012704	046060		165:	MOV #T26PK3,R4		;SET UP R4 WITH PACKET ADDRESS
7714	041750	010337	046066			MOV R3,T26SZ		;SET UP RECORD SIZE IN PACKET
7715	041754	010465	177776			MOV R4,TSD8(R5)		;ISSUE COMMAND
7716	041760	004737	017134			JSR PC,WAITF		;WAIT FOR SSR TO SET
7717	041764	016501	000000			MOV TSSR(R5),R1		;GET TSSR CONTENTS
7718	041770	012702	000200			MOV #SSR,R2		;SET UP EXPECTED
7719	041774	020102				CMP R1,R2		;ARE THEY EQUAL
7720	041776	001406				BEQ 170:		;BR, IF OK
7721	042000	004737	020116			JSR PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS
7725	042004					ERRHRD ERNO,T26RRF,PKTSSR		;TSSR INCORRECT AFTER REREAD DATA
	042004	104456						TRAP C1ERMRD
	042006	000453						.WORD 299
	042010	046335						.WORD T26RRF
	042012	011710						.WORD PKTSSR
7726	042014				170:	CKLOOP		;LOOP IF SELECTED
	042014	104406						TRAP C1CLP1
7727	042016	017701	141050			MOV #FREE,R1		;FIRST WORD FROM READ BUFFER
7728	042022	013702	046106			MOV T26CNT,R2		;SET UP EXPECTED

```

7729 042026 020102      CMP      R1,R2      ;IS TAPE POSITION CORRECT
7730 042030 001406      BEQ       1901      ;KEEP GOING POSITION OK
7731 042032 004737 020116 JSR      PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7735 042036      ERRMRD  ERRNO,T26WNG,EXPREC ;TAPE POSITION INCORRECT
      042036 104456      TRAP      C1ERMRD
      042040 000454      .WORD    300
      042042 046116      .WORD    T26WNG
      042044 016360      .WORD    EXPREC
7736 042046      1901:   CKLOOP      TRAP      C1CLP1
      042046 104406
7737 042050 062737 000001 046106      ADD      #1,T26CNT ;BUMP TAPE RECORD COUNTER
7738 042056 005723      TST      (R3). ;NEXT RECORD SIZE
7739 042060 010337 046112      MOV      R3,T26RSZ ;STORE RECORD SIZE
7740 042064 022703 000412      CMP      #266.,R3 ;AT MAX SIZE YET
7741 042070 001402      BEQ       2201      ;BR, IF AT END OF THE SUBTEST
7742 042072 000137 041662      JMP      1361 ;KEEP GOING MORE RECORDS
7743 042076      2201:   ENDSUB
7744 042076
      042076 104403
      042076 023727 002170 000031      CMP      FATFLG,#25.
7745 042100 023727 002170 000031      BLT      9991
7746 042106 002402      JSR      PC,CKDROP ;IS ERROR COUNT AT 25
7747 042110 004737 020170      ;BR, IF LESS THAN 25
7748 042114      9991:   ;TRY TO DROP THE UNIT

```

7750					;*
7751					;
7752					;TEST 2, SUBTEST 11
7753					;
7754					;VERIFIES THAT A REREAD NEXT COMMAND READING A
7755					;RECORD LONGER THAN THE SPECIFIED BYTE COUNT CAUSES
7756					;TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH
7757					;LONG (RLL) BIT SET. RESULTS ARE VERIFIED FOR BOTH
7758					;STATES OF OPP (0 AND 1).
7759					;
7760					;
7761					;
7762					;-
7763					
7764					
7765	042114			BGNSUB	;***** BEGIN SUBTEST *****
	042114				T2.11:
	042114	104402			TRAP C#BSUB
7766	042116	004737	050650	JSR PC,T26REST	;SET COMMAND PACKET
7767	042122	004737	050742	JSR PC,T26RT2	;SET UP OTHER COMMAND PACKET
7768	042126	004737	051004	JSR PC,T26RT3	;SET UP OTHER COMMAND PACKET
7769					
7770					;*****
7771					;
7772					;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
7773					;
7774					;*****
7775					
7776	042132	004737	016660	JSR PC,SOFINIT	;DO INITIALIZE ON CONTROLLER
7777	042136	103407		BCS 20;	;BR IF INIT WAS OK
7778	042140	004737	020116	JSR PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS
7782	042144	010001		MOV RO,R1	;CONTENTS OF TSSR REGISTER
7783	042146			ERRDF ERNO,SFIERR,SFMSG	;FATAL ERROR TSSR WAS NOT OK
	042146	104455			TRAP C#ERDF
	042150	000455			.WORD 301
	042152	003550			.WORD SFIERR
	042154	011676			.WORD SFMSG
7784	042156			20\$:	
7785					
7786	042156	012704	045730	MOV #T26PACKET,R4	;SUBROUTINE NEEDS PACKET ADDRESS
7787					
7788					;*****
7789					;
7790					;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPHR)
7791					;
7792					;*****
7793					
7794	042162	004737	010342	JSR PC,WRTPHR	;ISSUE WRITE CHARACTERISTICS
7795	042166	103407		BCS 26\$	;BR, IF COMMAND ISSUED OK
7796	042170	004737	020116	JSR PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS
7800	042174	010001		MOV RO,R1	;SAVE CONTENTS OF TSSR
7801	042176			ERRMRD ERNO,WRTMSG,SFMSG	;WRITE CHARACTERISTICSC FAILED
	042176	104456			TRAP C#ERRMRD
	042200	000456			.WORD 302
	042202	004754			.WORD WRTMSG
	042204	011676			.WORD SFMSG
7802	042206			26\$: CKLOOP	;LOOP IF SELECTED

```

042206 104406                                TRAP    C:CLP1
7803
7804 ;*****
7805 ;
7806 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7807 ;
7808 ;*****
7809
7810 042210 004737 010444          JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
7811 042214 016501 000000          MOV      TSSR(R5),R1        ;GET TSSR
7812 042220 012702 000200          MOV      #SSR,R2          ;SET UP EXPECTED TSSR
7813 042224 103407                  BCS      30$              ;BR, IF NO PROBLEM
7814 042226 010004          MOV      R0,R4          ;PACKET ADDRESS SET UP
7815 042230 004737 020116          JSR      PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
7819 042234          ERRMRD  ERRNO,T26RWN,PKTSSR      ;REWIND NOT ACCEPTED
          042234 104456                                TRAP    C:ERRMRD
          042236 000457                                .WORD    303
          042240 047414                                .WORD    T26RWN
          042242 011710                                .WORD    PKTSSR
7820 042244          30$:      CKLOOP                    ;LOOP IF SELECTED
          042244 104406                                TRAP    C:CLP1
7821
7822 ;*****
7823 ;
7824 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
7825 ;
7826 ;*****
7827
7828 042246 013701 045756          MOV      T26BFR+6,R1        ;PICK UP XST0
7829 042252 010102          MOV      R1,R2          ;SET UP EXPECTED
7830 042254 052702 000002          BIS      #BIT1,R2        ;SET BOT BIT IN EXPECTED
7831 042260 020102          CMP      R1,R2          ;DOES EXP = REC'D
7832 042262 001406          BEQ      40$              ;BR, IF EQUAL (OK)
7833 042264 004737 020116          JSR      PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
7837 042270          ERRMRD  ERRNO,T26BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
          042270 104456                                TRAP    C:ERRMRD
          042272 000460                                .WORD    304
          042274 047125                                .WORD    T26BOT
          042276 016360                                .WORD    EXPREC
7838 042300          40$:      CKLOOP                    ;LOOP IF SELECTED
          042300 104406                                TRAP    C:CLP1
7839 042302 012703 001000          MOV      #512.,R3          ;RECORD SIZE
7840 042306 013737 003072 046062  MOV      FREE,T26RB        ;STARTING WRITE BUFFER ADDRESS
7841
7842 ;*****
7843 ;
7844 ;WRITE DATA,CVC=1,ACK COMMAND
7845 ;
7846 ;*****
7847
7848 042314 012737 140005 046060  MOV      #140005,T26PK3      ;WRITE DATA,CVC=1,ACK COMMAND
7849 042322 012704 046060          MOV      #T26PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
7850 042326          65$:
7851 042326 010337 046066          MOV      R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
7852 042332 010465 177776          MOV      R4,TSD8(R5)      ;ISSUE COMMAND
7853 042336 004737 017134          JSR      PC,WAITF          ;WAIT FOR SSR TO SET
7854 042342 016501 000000          MOV      TSSR(R5),R1        ;GET TSSR CONTENTS

```

```

7855 042346 012702 000200      MOV    #SSR,R2      ;SET UP EXPECTED
7856 042352 020102      CMP     R1,R2      ;ARE THEY EQUAL
7857 042354 001406      BEQ     75$      ;BR, IF OK
7858 042356 004737 020116      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7862                                ;SOFT ERROR GENERATED BECAUSE THE
7863                                ;WRITE COMMAND IS NOT BEING CHECKED
7864                                ;HERE. IT WAS CHECKED IN LEAM2
7865 042362      ERRSOF ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP     C$ERSOFT
                                .WORD    305
                                .WORD    WRTErr
                                .WORD    PKTSSR
7866 042372 104406      75$:   CKLOOP      ;LOOP IF SELECTED
                                TRAP     C$CLP1
7867                                ;*****
7868                                ;
7869                                ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7870                                ;
7871                                ;*****
7872                                ;
7873                                ;
7874 042374 004737 010444      JSR     PC,REWIND    ;CALL TAPE REWIND COMMAND
7875 042400 016501 000000      MOV     TSSR(R5),R1 ;GET TSSR
7876 042404 012702 000200      MOV     #SSR,R2      ;SET UP EXPECTED TSSR
7877 042410 103407      BCS     130$      ;BR, IF NO PROBLEM
7878 042412 010004      MOV     R0,R4      ;PACKET ADDRESS SET UP
7879 042414 004737 020116      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7883 042420      ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP     C$ERHRD
                                .WORD    306
                                .WORD    T26RWN
                                .WORD    PKTSSR
7884 042430 104406      130$:   CKLOOP      ;LOOP IF SELECTED
                                TRAP     C$CLP1
7885                                ;*****
7886                                ;
7887                                ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7888                                ;
7889                                ;*****
7890                                ;
7891                                ;
7892 042432 013701 045756      MOV     T26BFR+6,R1    ;PICK UP XSTO
7893 042436 010102      MOV     R1,R2      ;SET UP EXPECTED
7894 042440 052702 000002      BIS     #BIT1,R2    ;SET BOT BIT IN EXPECTED
7895 042444 020102      CMP     R1,R2      ;DOES EXP = REC'D
7896 042446 001406      BEQ     140$      ;BR, IF EQUAL (OK)
7897 042450 004737 020116      JSR     PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7901 042454      ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP     C$ERHRD
                                .WORD    307
                                .WORD    T26BOT
                                .WORD    EXPREC
7902 042464 104406      140$:   CKLOOP      ;LOOP IF SELECTED
                                TRAP     C$CLP1
7903 042466 005303      DEC     R3          ;SET RECORD SIZE TO 511.
7904 042470 013737 003072 046062      MOV     FREE,T26RB ;STARTING READ BUFFER ADDRESS
7905

```

```

7906 ;*****
7907 ;
7908 ;REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
7909 ;
7910 ;*****
7911
7912 042476 012737 161401 046060      MOV      #161401,T26PK3      ;REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
7913 042504 012704 046060      165$: MOV      #T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
7914 042510 010337 046066      MOV      R3,T26SZ      ;SET UP RECORD SIZE IN PACKET
7915 042514 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
7916 042520 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
7917 042524 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
7918 042530 012702 100204      MOV      #SSR!SC!BIT2,R2      ;SET UP EXPECTED
7919 042534 020102      CMP      R1,R2      ;ARE THEY EQUAL
7920 042536 001406      BEQ      170$      ;BR, IF OK
7921 042540 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
7925 042544      ERRHRD      ERRNO,T26TRL,PKTSSR      ;TSSR INCORRECT AFTER REREAD DATA
                                TRAP      C$ERRHRD
                                .WORD      308
                                .WORD      T26TRL
                                .WORD      PKTSSR
7926 042554      170$: CKLOOP      ;LOOP I' SELECTED
                                TRAP      C$CLP1
7927 042554 104406
7928 ;*****
7929 ;
7930 ;READ MESSAGE BUFFER EXTENDED STAT, REGISTER ZERO (XSTO)
7931 ;
7932 ;*****
7933
7934 042556 013701 045756      MOV      T268FR+6,R1      ;GET MESSAGE BUFFER
7935 042562 010102      MOV      R1,R2      ;SET UP EXPECTED
7936 042564 052702 010000      BIS      #BIT12,R2      ;SET THE RLL BIT IN EXPECTED
7937 042570 020102      CMP      R1,R2      ;ARE THEY EQUAL
7938 042572 001406      BEQ      180$      ;BR, IF EQUAL (ALL IS WELL)
7939 042574 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
7943 042600      ERRHRD      ERRNO,T26LON,EXPREC      ;THE RLL BIT WAS NOT SET IN XSTO
                                TRAP      C$ERRHRD
                                .WORD      309
                                .WORD      T26LON
                                .WORD      EXPREC
7944 042610      180$:
7945 042610 012703 000777      MOV      #511.,R3      ;SET UP SIZE OF RECORD
7946 042614 013737 003072 046062      MOV      FREE,T26RB      ;STARTING READ BUFFER ADDRESS
7947 ;*****
7948 ;
7949 ;REREAD DATA,CVC=1,ACK COMMAND
7950 ;
7951 ;*****
7952
7953
7954 042622 012737 141401 046060      365$: MOV      #141401,T26PK3      ;REREAD DATA,CVC=1,ACK COMMAND
7955 042630 012704 046060      MOV      #T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
7956 042634 010337 046066      MOV      R3,T26SZ      ;SET UP RECORD SIZE IN PACKET
7957 042640 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
7958 042644 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
7959 042650 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS

```

```

7960 042654 012702 100204      MOV    #SSR!SC!BIT2,R2      ;SET UP EXPECTED
7961 042660 020102      CMP     R1,R2                ;ARE THEY EQUAL
7962 042662 001406      BEQ     370$                 ;BR, IF OK
7963 042664 004737 020116      JSR     PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
7967 042670      ERRHRD  ERRNO,T26TRL,PKTSSR          ;TSSR INCORRECT AFTER REREAD DATA
                                TRAP     C$ERHRD
                                .WORD    310
                                .WORD    T26TRL
                                .WORD    PKTSSR
7968 042700      370$:  CKLOOP                        ;LOOP IF SELECTED
                                TRAP     C$CLP1
042700 104406
7969
7970      ;*****
7971      ;
7972      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
7973      ;
7974      ;*****
7975
7976 042702 013701 045756      MOV     T26BFR+6,R1        ;GET MESSAGE BUFFER
7977 042706 010102      MOV     R1,R2                ;SET UP EXPECTED
7978 042710 052702 010000      BIS     #BIT12,R2        ;SET THE RLL BIT IN EXPECTED
7979 042714 020102      CMP     R1,R2                ;ARE THEY EQUAL
7980 042716 001406      BEQ     380$                 ;BR, IF EQUAL (ALL IS WELL)
7981 042720 004737 020116      JSR     PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
7985 042724      ERRHRD  ERRNO,T26LON,EXPREC          ;THE RLL BIT WAS NOT SET IN XST0
                                TRAP     C$ERHRD
                                .WORD    311
                                .WORD    T26LON
                                .WORD    EXPREC
042724 104456
042726 000467
042730 050240
042732 016360
7986 042734      380$:  ENDSUB                        ;>>>>>>>>> END SUBTEST >>>>>>>>>
7987 042734      L10061:
                                TRAP     C$ESUB
042734 104403
7988 042736 023727 002170 000031      CMP     FATFLG,#25.      ;IS ERROR COUNT AT 25
7989 042744 002402      BLT     999$                 ;BR, IF LESS THAN 25
7990 042746 004737 020170      JSR     PC,CKDROP        ;RY TO DROP THE UNIT
7991 042752      999$:

```

	:	*				
7993	:					
7994	:					
7995	:	TEST 2, SUBTEST 12				
7996	:					
7997	:	VERIFIES THAT A REREAD NEXT COMMAND READING A				
7998	:	RECORD SHORTER THAN THE SPECIFIED BYTE COUNT CAUSES				
7999	:	TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH				
8000	:	SHORT (RLS) BIT SET. IT IS VERIFIED THAT THE				
8001	:	RESIDUAL BYTE COUNTER (RBPCR) IN THE MESSAGE BUFFER				
8002	:	CONTAINS THE PROPER NONZERO VALUE (E.G., THE				
8003	:	DIFFERENCE BETWEEN THE ORIGINAL BYTE COUNT AND THE				
8004	:	ACTUAL RECORD LENGTH). RESULTS ARE VERIFIED FOR BOTH				
8005	:	STATES OF OPP (0 AND 1).				
8006	:					
8007	:					
8008	:					
8009	:	-				
8010	042752	BGNBUB			:>>>>>>>>> BEGIN SUBTEST >>>>>>>	
	042752				T2.12:	
	042752	104402			TRAP	C#BSUB
8011	042754	004737	050650	JSR PC,T26REST	;SET COMMAND PACKET	
8012	042760	004737	050742	JSR PC,T26RT2	;SET UP OTHER COMMAND PACKET	
8013	042764	004737	051004	JSR PC,T26RT3	;SET UP OTHER COMMAND PACKET	
8014						
8015						
8016						
8017						
8018						
8019						
8020						
8021	042770	004737	016660	JSR PC,SOFINIT	;DO INITIALIZE ON CONTROLLER	
8022	042774	103407		BCS 20\$	;BR IF INIT WAS OK	
8023	042776	004737	020116	JSR PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS	
8027	043002	010001		MOV RO,R1	;CONTENTS OF TSSR REGISTER	
8028	043004			ERRDF ERRNO,SFIERR,SFIMSG	;FATAL ERROR TSSR WAS NOT OK	
	043004	104455			TRAP	C#ERDF
	043006	000470			.WORD	312
	043010	003550			.WORD	SFIERR
	043012	011676			.WORD	SFIMSG
8029	043014					
8030						
8031	043014	012704	045730	MOV @T26PACKET,R4	;SUBROUTINE NEEDS PACKET ADDRESS	
8032						
8033						
8034						
8035						
8036						
8037						
8038						
8039	043020	004737	010342	JSR PC,WRTCHR	;ISSUE WRITE CHARACTERISTICS	
8040	043024	103407		BCS 26\$	;BR, IF COMMAND ISSUED OK	
8041	043026	004737	020116	JSR PC FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS	
8045	043032	010001		MOV P?,R1	;SAVE CONTENTS OF TSSR	
8046	043034			ERRHRD ERRNO,WRTMSG,SFIMSG	;WRITE CHARACTERISTISC FAILED	
	043034	104456			TRAP	C#ERRHD
	043036	000471			.WORD	313
	043040	004754			.WORD	WRTMSG

8047	043042	011676		26\$:	CKLOOP		;LOOP IF SELECTED	.WORD	SFIMSG
	043044	104406						TRAP	C\$CLP1
8048									
8049									
8050									
8051									
8052									
8053									
8054									
8055	043046	004737	010444		JSR	PC,REWIND			
8056	043052	016501	000000		MOV	TSSR(R5),R1			
8057	043056	012702	000200		MOV	#SSR,R2			
8058	043062	103407			BCS	30\$			
8059	043064	010004			MOV	R0,R4			
8060	043066	004737	020116		JSR	PC,FATCHK			
8064	043072				ERRMRD	ERRNO,T26RWN,PKTSSR			
	043072	104456						TRAP	C\$ERMRD
	043074	000472						.WORD	314
	043076	047414						.WORD	T26RWN
	043100	011710						.WORD	PKTSSR
8065	043102			30\$:	CKLOOP		;LOOP IF SELECTED		
	043102	104406						TRAP	C\$CLP1
8066									
8067									
8068									
8069									
8070									
8071									
8072									
8073	043104	013701	045756		MOV	T26BFR+6,R1			
8074	043110	010102			MOV	R1,R2			
8075	043112	052702	000002		BIS	#BIT1,R2			
8076	043116	020102			CMP	R1,R2			
8077	043120	001406			BEQ	40\$			
8078	043122	004737	020116		JSR	PC,FATCHK			
8082	043126				ERRMRD	ERRNO,T26BOT,EXPREC			
	043126	104456						TRAP	C\$ERMRD
	043130	000473						.WORD	315
	043132	047125						.WORD	T26BOT
	043134	016360						.WORD	EXPREC
8083	043136			40\$:	CKLOOP		;LOOP IF SELECTED		
	043136	104406						TRAP	C\$CLP1
8084	043140	012703	000400		MOV	#256.,R3			
8085	043144	013737	003072	046062	MOV	FREE,T26RB			
8086									
8087									
8088									
8089									
8090									
8091									
8092									
8093	043152	012737	140005	046060	MOV	#140005,T26PK3			
8094	043160	012704	046060		MOV	#T26PK3,F4			
8095	043164								
8096	043164	010337	046066		MOV	R3,T26SZ			
8097	043170	010465	177776		MOV	R4,TSDB(R5)			

```
8098 043174 004737 017134      JSR      PC, WAITF      ;WAIT FOR SSR TO SET
8099 043200 016501 000000      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
8100 043204 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
8101 043210 020102              CMP      R1,R2      ;ARE THEY EQUAL
8102 043212 001406              BEQ      75$        ;BR, IF OK
8103 043214 004737 020116      JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
8107                                ;SOFT ERROR GENERATED BECAUSE THE
8108                                ;WRITE COMMAND IS NOT BEING CHECKED
8109                                ;HERE. IT WAS CHECKED IN LEAH2
8110                                ;TSSR INCORRECT AFTER WRITE DATA
      043220              ERRSOF ERRNO,WRTErr,PKTSSR
      043220 104457              TRAP      C$ERSOFT
      043222 000474              .WORD    316
      043224 005011              .WORD    WRTErr
      043226 011710              .WORD    PKTSSR
8111 043230 75$: CKLOOP              ;LOOP IF SELECTED
      043230 104406              TRAP      C$CLP1
8112 043232 120$:
8113                                ;*****
8114                                ;
8115                                ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8116                                ;
8117                                ;*****
8118
8119
8120 043232 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
8121 043236 016501 000000      MOV      TSSR(R5),R1    ;GET TSSR
8122 043242 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED TSSR
8123 043246 103407              BCS      130$        ;BR, IF NO PROBLEM
8124 043250 010004              MOV      R0,R4        ;PACKET ADDRESS SET UP
8125 043252 004737 020116      JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
8129 043256              ERRHRD ERRNO,T26RWN,PKTSSR    ;REWIND NOT ACCEPTED
      043256 104456              TRAP      C$ERHRD
      043260 000475              .WORD    317
      043262 047414              .WORD    T26RWN
      043264 011710              .WORD    PKTSSR
8130 043266 130$: CKLOOP              ;LOOP IF SELECTED
      043266 104406              TRAP      C$CLP1
8131
8132                                ;*****
8133                                ;
8134                                ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
8135                                ;
8136                                ;*****
8137
8138 043270 013701 045756      MOV      T26BFR+6,R1      ;PICK UP XST0
8139 043274 010102              MOV      R1,R2        ;SET UP EXPECTED
8140 043276 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
8141 043302 020102              CMP      R1,R2        ;DOES EXP = REC'D
8142 043304 001406              BEQ      135$        ;BR, IF EQUAL (OK)
8143 043306 004737 020116      JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
8147 043312              ERRHRD ERRNO,T26BOT,EXPREC    ;TAPE NOT AT BOT AFTER REWIND
      043312 104456              TRAP      C$ERHRD
      043314 000476              .WORD    318
      043316 047125              .WORD    T26BOT
      043320 016360              .WORD    EXPREC
8148 043322 135$: CKLOOP              ;LOOP IF SELECTED
      043322 104406              TRAP      C$CLP1
```

```

8149 043324 012703 001000      MOV      #512.,R3      ;RECORD SIZE
8150 043330 013737 003072 046062  MOV      FREE,T26RB      ;STARTING READ BUFFER ADDRESS
8151
8152      ;*****
8153      ;
8154      ;REREAD NEXT,ACK,CVC=1,OPP=1
8155      ;
8156      ;*****
8157
8158 043336 012737 161401 046060      MOV      #161401,T26PK3      ;REREAD NEXT,ACK,CVC=1,OPP=1
8159 043344 012704 046060      165$:  MOV      #T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
8160 043350 010337 046066      MOV      R3,T26SZ      ;SET UP RECORD SIZE IN PACKET
8161 043354 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
8162 043360 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
8163 043364 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
8164 043370 012702 100204      MOV      #SSR!SC!BIT2,R2      ;SET UP EXPECTED
8165 043374 020102      CMP      R1,R2      ;ARE THEY EQUAL
8166 043376 001406      BEQ      170$      ;BR, IF OK
8167 043400 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8171 043404      ERRHRD      ERRNO,T26TRL,PKTSSR      ;TSSR INCORRECT AFTER READ DATA
      043404 104456      TRAP      C$ERHRD
      043406 000477      .WORD      319
      043410 050472      .WORD      T26TRL
      043412 011710      .WORD      PKTSSR
8172 043414      170$:  CKLOOP      ;LOOP IF SELECTED
      043414 104406      TRAP      C$CLP1
8173
8174      ;*****
8175      ;
8176      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8177      ;
8178      ;*****
8179
8180 043416 013701 045756      MOV      T26BFR+6,R1      ;GET MESSAGE BUFFER
8181 043422 010102      MOV      R1,R2      ;SET UP EXPECTED
8182 043424 052702 040000      BIS      #BIT14,R2      ;SET THE RLS BIT IN EXPECTED
8183 043430 020102      CMP      R1,R2      ;ARE THEY EQUAL
8184 043432 001406      BEQ      180$      ;BR, IF EQUAL (ALL IS WELL)
8185 043434 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8189 043440      ERRHRD      ERRNO,T26LOP,EXPREC      ;THE RLL BIT WAS NOT SET IN XSTO
      043440 104456      TRAP      C$ERHRD
      043442 000500      .WORD      320
      043444 050322      .WORD      T26LOP
      043446 016360      .WORD      EXPREC
8190 043450      180$:
8191 043450 013701 045754      MOV      T26BFR+4,R1      ;PICK UP RESIDUAL BYTE COUNTER
8192 043454 012702 000400      MOV      #256.,R2      ;THIS SHOULD BE THE DIFFERENCE
8193 043460 020102      CMP      R1,R2      ;IS THE DIFFERENCE CORRECT
8194 043462 001405      BEQ      190$      ;BR, IF CORRECT
8198 043466      ERRHRD      ERRNO,T26PBP,EXPREC      ;RBPCT NOT CORRECT
      043466 104456      TRAP      C$ERHRD
      043470 000500      .WORD      320
      043472 050404      .WORD      T26PBP
      043474 016360      .WORD      EXPREC
8199 043476      190$:  CKLOOP      ;LOOP IF SELECTED
      043476 104406      TRAP      C$CLP1
8200 043500 012703 001000      MOV      #512.,R3      ;RECORD SIZE

```

Address	Offset	Index	Base	Label	Instruction	Comment
8201	043504	013737	003072	046062	MOV	FREE,T26RB ;STARTING READ BUFFER ADDRESS
8202						
8203						*****
8204						
8205						;REREAD NEXT,ACK,CVC=1,OPP=0
8206						
8207						*****
8208						
8209	043512	012737	141401	046060	MOV	#141401,T26PK3 ;REREAD NEXT,ACK,CVC=1,OPP=0
8210	043520	012704	046060		MOV	#T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
8211	043524	010337	046066		MOV	R3,T26SZ ;SET UP RECORD SIZE IN PACKET
8212	043530	010465	177776		MOV	R4,TSD8(R5) ;ISSUE COMMAND
8213	043534	004737	017134		JSR	PC,WAITF ;WAIT FOR SSR TO SET
8214	043540	016501	000000		MOV	TSSR(R5),R1 ;GET TSSR CONTENTS
8215	043544	012702	100204		MOV	#SSR!SC!BIT2,R2 ;SET UP EXPECTED
8216	043550	020102			CMP	R1,R2 ;ARE THEY EQUAL
8217	043552	001406			BEQ	270\$;BR, IF OK
8218	043554	004737	020116		JSR	PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
8222	043560				ERRHRD	ERRNO,T26TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
	043560	104456				TRAP C\$ERHRD
	043562	000501				.WORD 321
	043564	050472				.WORD T26TRL
	043566	011710				.WORD PKTSSR
8223	043570			270\$:	CKLOOP	;LOOP IF SELECTED
	043570	104406				TRAP C\$CLP1
8224						
8225						*****
8226						
8227						;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8228						
8229						*****
8230						
8231	043572	013701	045756		MOV	T26BFR+6,R1 ;GET MESSAGE BUFFER
8232	043576	010102			MOV	R1,R2 ;SET UP EXPECTED
8233	043600	052702	040000		BIS	#BIT14,R2 ;SET THE RLS BIT IN EXPECTED
8234	043604	020102			CMP	R1,R2 ;ARE THEY EQUAL
8235	043606	001406			BEQ	280\$;BR, IF EQUAL (ALL IS WELL)
8236	043610	004737	020116		JSR	PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
8240	043614				ERRHRD	ERRNO,T26LOP,EXPREC ;THE RLL BIT WAS NOT SET IN XSTO
	043614	104456				TRAP C\$ERHRD
	043616	000502				.WORD 322
	043620	050322				.WORD T26LOP
	043622	016360				.WORD EXPREC
8241	043624			280\$:		
8242	043624	013701	045754		MOV	T26BFR+4,R1 ;PICK UP RESIDUAL BYTE COUNTER
8243	043630	012702	000400		MOV	#256.,R2 ;THIS SHOULD BE THE DIFFERENCE
8244	043634	020102			CMP	R1,R2 ;IS THE DIFFERENCE CORRECT
8245	043636	001405			BEQ	290\$;BR, IF CORRECT
8249	043642				ERRHRD	ERRNO,T26PBP,EXPREC ;RBPCT NOT CORRECT
	043642	104456				TRAP C\$ERHRD
	043644	000502				.WORD 322
	043646	050404				.WORD T26PBP
	043650	016360				.WORD EXPREC
8250	043652			290\$:	CKLOOP	;LOOP IF SELECTED
	043652	104406				TRAP C\$CLP1
8251	043654				ENDSUB	; >>>>>>>>> END SUBTEST >>>>>>>>>
	043654					L10062:

B1

CZTKGA TK-25 FRT END FUNC #3 MACRO M1200 20-APR-84 08:13 PAGE 108-5
TEST 2: REREADS

SEQ 208

8252	043654	104403				
8253	043656	023727	002170	000031	CMP	FATFLG.#25.
8254	043664	002402			BLT	9991
8255	043666	004737	020170		JSR	PC,CKDROP
	043672			9991:		

TRAP C#ESUB
;IS ERROR COUNT AT 25
;BR, IF LESS THAN 25
;TRY TO DROP THE UNIT

D1

CZIKGA TK-25 FBT END FUNC #3 MACRO M1200 20-APR 84 08:13 PAGE 109 1

SEQ 210

```

8311 043764 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8315 043770 010001              MOV      R0,R1      ;SAVE CONTENTS OF TSSR
8316 043772              ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
                                TRAP      C$ERHRD
                                .WORD     324
                                .WORD     WRTMSG
                                .WORD     SFIMSG
8317 044002 104406      26$:  CKLOOP      ;LOOP IF SELECTED
                                TRAP      C$CLP1
8318
8319      ;*****
8320      ;
8321      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8322      ;
8323      ;*****
8324
8325 044004 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
8326 044010 103411      BCS      30$      ;BR, IF NO PROBLEM
8327 044012 016501 000000      MOV      TSSR(R5),R1 ;GET TSSR
8328 044016 010004      MOV      R0,R4      ;PACKET ADDRESS SET UP
8329 044020 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8333 044024              ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD     325
                                .WORD     T26RWN
                                .WORD     PKTSSR
8334 044034 104406      30$:  CKLOOP      ;LOOP IF SELECTED
                                TRAP      C$CLP1
8335
8336      ;*****
8337      ;
8338      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8339      ;
8340      ;*****
8341
8342 044036 013701 045756      MOV      T26BFR+6,R1 ;PICK UP XSTO
8343 044042 010102      MOV      R1,R2      ;SET UP EXPECTED
8344 044044 052702 000002      BIS      #BIT1,R2 ;SET BOT BIT IN EXPECTED
8345 044050 020102      CMP      R1,R2      ;DOES EXP = REC'D
8346 044052 001406      BEQ      40$      ;BR, IF EQUAL (OK)
8347 044054 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8351 044060              ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD     326
                                .WORD     T26BOT
                                .WORD     EXPREC
8352 044070 104406      40$:  CKLOOP      ;LOOP IF SELECTED
                                TRAP      C$CLP1
8353 044072 013737 003072 046062      MOV      FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
8354
8355      ;*****
8356      ;
8357      ;WRITE DATA,CVC=1,ACK COMMAND
8358      ;
8359      ;*****
8360
8361 044100 012737 140005 046060      MOV      #140005,T26PKZ ;WRITE DATA,CVC=1,ACK COMMAND

```

E1

```

8362 044106 012704 046060      MOV      #T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
8363 044112 012737 000400 046066 65:  MOV      #256.,T26SZ      ;SET UP RECORD SIZE IN PACKET
8364 044120 013777 046106 136744      MOV      T26CNT,BFREE      ;MOVE TAPE RECORD NUMBER TO BUFFER
8365 044126 062737 000001 046106      ADD      #1,T26CNT      ;NUMBER READY FOR NEXT RECORD
8366 044134 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
8367 044140 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
8368 044144 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
8369 044150 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
8370 044154 020102      CMP      R1,R2      ;ARE THEY EQUAL
8371 044156 001406      BEQ      75:      ;BR, IF OK
8372 044160 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8376      ;SOFT ERROR GENERATED BECAUSE THE
8377      ;WRITE COMMAND IS NOT BEING CHECKED
8378      ;HERE. IT WAS CHECKED IN LEAM2
8379      ;TSSR INCORRECT AFTER WRITE DATA
      ERRSOF ERRNO,WRTERR,PKTSSR      TRAP      C$ERSOFT
      .WORD      327
      .WORD      WRTERR
      .WORD      PKTSSR
8380 044174      75:  CKLOOP      ;LOOP IF SELECTED
      044174 104406      TRAP      C$CLP1
8381 044176 022737 000013 046106      CMP      #11.,T26CNT      ;CHECK NUMBER OF RECORDS WRITTEN
8382 044204 001401      BEQ      120:      ;BR, IF AT END OF WRITE SEQUENCE
8383 044206 000741      BR      65:      ;WRITE MORE RECORDS
8384 044210      120:  CLR      T26CNT      ;SET RECORD COUNTER BACK TO ZERO
8385 044210 005037 046106
8386      ;*****
8387      ;
8388      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8389      ;
8390      ;*****
8391      ;
8392      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
8393 044214 004737 010444      BCS      130:      ;BR, IF NO PROBLEM
8394 044220 103411      MOV      TSSR(R5),R1      ;GET TSSR
8395 044222 016501 000000      MOV      R0,R4      ;PACKET ADDRESS SET UP
8396 044226 010004      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8397 044230 004737 020116      ERRMRD ERRNO,T26RWN,PKTSSR      ;REWIND NOT ACCEPTED
8401 044234      TRAP      C$ERMRD
      044234 104456      .WORD      328
      044236 000510      .WORD      T26RWN
      044240 047414      .WORD      PKTSSR
      044242 011710
8402 044244      130:  CKLOOP      ;LOOP IF SELECTED
      044244 104406      TRAP      C$CLP1
8403      ;*****
8404      ;
8405      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8406      ;
8407      ;*****
8408      ;
8409      MOV      T26BFR+6,R1      ;PICK UP XSTO
8410 044246 013701 045756      MOV      R1,R2      ;SET UP EXPECTED
8411 044252 010102      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
8412 044254 052702 000002      CMP      R1,R2      ;DOES EXP = REC'D
8413 044260 020102      BEQ      140:      ;BR, IF EQUAL (OK)
8414 044262 001406

```

```

8415 044264 004737 020116          JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
8419 044270          ERRHRD      ERRNO,T26BOT,EXPREC          ;TAPE NOT AT BOT AFTER REWIND
          044270 104456          TRAP      C$ERHRD
          044272 000511          .WORD      329
          044274 047125          .WORD      T26BOT
          044276 016360          .WORD      EXPREC
8420 044300          140$: CKLOOP          ;LOOP IF SELECTED
          044300 104406          TRAP      C$CLP1
8421 044302 012703 046076          MOV      #T26RN,R3          ;COMMAND BUFFER ADDRESS
8422 044306 012737 177376 046062 150$: MOV      #177376,T26RB          ;STARTING READ BUFFER ADDRESS
8423 044314 012737 000077 046064          MOV      #000077,T26RB+2          ;SET UP HIGH ORDER ADDRESS BITS
8424
8425          ;*****
8426          ;
8427          ;REREAD DATA,IE,ACK, OPP COMMAND
8428          ;
8429          ;*****
8430
8431 044322 011337 046060          MOV      (R3),T26PK3          ;REREAD DATA,IE,ACK, OPP COMMAND
8432 044326 012704 046060          165$: MOV      #T26PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
8433 044332 012737 000400 046066          MOV      #256.,T26SZ          ;SET UP RECORD SIZE IN PACKET
8434 044340 010465 177776          MOV      R4,TSD8(R5)          ;ISSUE COMMAND
8435 044344 004737 017134          JSR      PC,WAITF          ;WAIT FOR SSR TO SET
8436 044350 016501 000000          MOV      TSSR(R5),R1          ;GET TSSR CONTENTS
8437 044354 012702 104210          MOV      #SSR!NXM!SC!BIT3,R2          ;SET UP EXPECTED
8438 044360 020102          CMP      R1,R2          ;ARE THEY EQUAL
8439 044362 001414          BEQ      170$          ;BR, IF OK
8440 044364 031327 001000          BIT      (R3),#BIT9          ;CHECK FOR A READ COMMAND
8441 044370 001403          BEQ      168$          ;BR, IF IT WAS A READ COMMAND
8442 044372 030127 000002          BIT      R1,#BIT1          ;WAS BIT1 SET
8443 044376 001006          BNE      170$          ;BR, IF REREAD AND BIT1 SET
8444 044400
8445 044400 004737 020116          168$: JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
8449 044404          ERRHRD      ERRNO,T26RRF,PKTSSR          ;TSSP INCORRECT AFTER REREAD DATA
          044404 104456          TRAP      C$ERHRD
          044406 000512          .WORD      330
          044410 046335          .WORD      T26RRF
          044412 011710          .WORD      PKTSSR
8450 044414          170$: CKLOOP          ;LOOP IF SELECTED
          044414 104406          TRAP      C$CLP1
8451
8452          ;*****
8453          ;
8454          ;READ DATA, ACK,CVC=1 COMMAND
8455          ;
8456          ;*****
8457
8458 044416 012737 140001 046060          MOV      #140001,T26PK3          ;READ DATA, ACK,CVC=1 COMMAND
8459 044424 012737 000400 046066          MOV      #256.,T26SZ          ;SET SIZE INTO PACKET
8460 044432 005037 046064          CLR      T26RB+2          ;CLEAR OUT HIGH ADDRESS BITS
8461 044436 013737 003072 046062          MOV      FREE,T26RB          ;GIVE READ A GOOD BUFFER
8462 044444 010465 177776          MOV      R4,TSD8(R5)          ;ISSUE READ DATA COMMAND
8463 044450 004737 017134          JSR      PC,WAITF          ;WAIT FOR SSR
8464 044454 016501 000000          MOV      TSSR(R5),R1          ;PICK UP THE TSSR
8465 044460 012702 000200          MOV      #SSR,R2          ;SET UP EXPECTED
8466 044464 020102          CMP      R1,R2          ;IS THE TSSR OK
8467 044466 001406          BEQ      180$          ;BR, IF TSSR OK (GOOD)

```

```

8468 044470 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8472 044474      ERRHRD      ERRNO,RDERR,PKTSSR      ;READ DATA COMMAND FAILED
      044474 104456      TRAP      C$ERHRD
      044476 000513      .WORD      331
      044500 005104      .WORD      RDERR
      044502 011710      .WORD      PKTSSR
8473 044504      180$: CKLOOP      ;LOOP IF SELECTED
      044504 104406      TRAP      C$CLP1
8474 044506 017701 136360      MOV      @FREE,R1      ;FIRST WORD FROM READ BUFFER
8475 044512 012702 000001      MOV      #1,R2      ;SET UP EXPECTED
8476 044516 020102      CMP      R1,R2      ;IS TAPE POSITION CORRECT
8477 044520 001406      BEQ      190$      ;KEEP GOING POSITION OK
8478 044522 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8482 044526      ERRHRD      ERRNO,T26WNG,XPREC      ;TAPE POSITION INCORRECT
      044526 104456      TRAP      C$ERHRD
      044530 000514      .WORD      332
      044532 046116      .WORD      T26WNG
      044534 016360      .WORD      XPREC
8483 044536      190$: CKLOOP
      044536 104406      TRAP      C$CLP1
8484
8485      ;*****
8486      ;
8487      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8488      ;
8489      ;*****
8490
8491 044540 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
8492 044544 103411      BCS      194$      ;BR, IF NO PROBLEM
8493 044546 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR
8494 044552 010004      MOV      R0,R4      ;PACKET ADDRESS SET UP
8495 044554 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8499 044560      ERRHRD      ERRNO,T26RWN,PKTSSR      ;REWIND NOT ACCEPTED
      044560 104456      TRAP      C$ERHRD
      044562 000515      .WORD      333
      044564 047414      .WORD      T26RWN
      044566 011710      .WORD      PKTSSR
8500 044570      194$: CKLOOP      ;LOOP IF SELECTED
      044570 104406      TRAP      C$CLP1
8501
8502      ;*****
8503      ;
8504      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
8505      ;
8506      ;*****
8507
8508 044572 013701 045756      MOV      T26BFR+6,R1      ;PICK UP XST0
8509 044576 010102      MOV      R1,R2      ;SET UP EXPECTED
8510 044600 052702 000002      BIS      @BIT1,R2      ;SET BOT BIT IN EXPECTED
8511 044604 020102      CMP      R1,R2      ;DOES EXP = REC'D
8512 044606 001406      BEQ      196$      ;BR, IF EQUAL (OK)
8513 044610 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8517 044614      ERRHRD      ERRNO,T26BOT,XPREC      ;TAPE NOT AT BOT AFTER REWIND
      044614 104456      TRAP      C$ERHRD
      044616 000516      .WORD      334
      044620 047125      .WORD      T26BOT
      044622 016360      .WORD      XPREC

```

H1

CZTKGA TK 25 FRI END FUNC #3 MACRO M1200 20-APR 84 08:13 PAGE 109 5

SEQ 214

```

8518 044624      196$: CKLOOP                      ;LOOP IF SELECTED
      044624 104406                                TRAP      C$CLP1
8519 044626 010302                                MOV      R3,R2      ;SAVE R3 FOR A MOMENT
8520
8521 ;*****
8522 ;
8523 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
8524 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
8525 ;
8526 ;*****
8527
8528 044630 012703 000001      MOV      #1,R3      ;SPACE ONE RECORD
8529 044634 004737 010144      JSR      PC,SPACE      ;CALL SPACE ROUTINE
8530 044640 010203      MOV      R2,R3      ;RESTORE R3
8531 044642 005723      TST      (R3)+      ;BUMP COUNTER
8532 044644 021327 177777      CMP      (R3),#177777      ;END OF COMMAND BUFFER YET
8533 044650 001216      BNE      150$      ;MORE COMMANDS KEEP GOING
8534 044652
8535 044652      200$: ENDSUB                      ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
      044652                                L10063:
      044652 104403                                TRAP      C$ESUB
8536 044654 023727 002170 000031      CMP      FATFLG,#25.      ;IS ERROR COUNT AT 25
8537 044662 002402      BLT      999$      ;BR. IF LESS THAN 25
8538 044664 004737 020170      JSR      PC,CKDROP      ;TRY TO DROP THE UNIT
8539 044670      999$:

```



```

8593
8594
8595
8596
8597
8598
8599
8600 044766 004737 010444
8601 044772 016501 000000
8602 044776 012702 000200
8603 045002 103407
8604 045004 010004
8605 045006 004737 020116
8609 045012
      045012 104456
      045014 000521
      045016 047414
      045020 011710
8610 045022
      045022 104406
8611
8612
8613
8614
8615
8616
8617
8618 045024 013701 045756
8619 045030 010102
8620 045032 052702 000002
8621 045036 020102
8622 045040 001406
8623 045042 004737 020116
8627 045046
      045046 104456
      045050 000522
      045052 047125
      045054 016360
8628 045056
      045056 104406
8629 045060 012737 000400 046066
8630 045066 013737 003072 046062
8631 045074 005703
8632 045076 001404
8633
8634
8635
8636
8637
8638
8639
8640 045100 012737 161001 046060
8641 045106 000403
8642
8643
8644
8645

;*****
;
;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
;
;*****
26$: JSR PC,REWIND ;CALL TAPE REWIND COMMAND
      MOV TSSR(R5),R1 ;GET TSSR
      MOV #SSR,R2 ;SET UP EXPECTED TSSR
      BCS 30$ ;BR, IF NO PROBLEM
      MOV R0,R4 ;PACKET ADDRESS SET UP
      JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
      ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                     TRAP C$ERHRD
                                     .WORD 337
                                     .WORD T26RWN
                                     .WORD PKTSSR
30$: CKLOOP ;LOOP IF SELECTED
                                     TRAP C$CLP1

;*****
;
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
;
;*****
      MOV T26BFR+6,R1 ;PICK UP XSTO
      MOV R1,R2 ;SET UP EXPECTED
      BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
      CMP R1,R2 ;DOES EXP = REC'D
      BEQ 40$ ;BR, IF EQUAL (OK)
      JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
      ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                     TRAP C$ERHRD
                                     .WORD 338
                                     .WORD T26BOT
                                     .WORD EXPREC
40$: CKLOOP ;LOOP IF SELECTED
                                     TRAP C$CLP1
      MOV #256.,T26SZ ;SET UP RECORD SIZE IN PACKET
      MOV FREE,T26RB ;ADDRESS OF READ BUFFER
      TST R3 ;CHECK NUMBER OF TIMES THROUGH HERE
      BEQ 50$ ;BR, IF FIRST TIME THROUGH HERE

;*****
;
;REREAD,CVC=1,ACK COMMAND
;
;*****
      MOV #161001,T26PK3 ;REREAD,CVC=1,ACK COMMAND
      BR 55$ ;SKIP NEXT COMMAND

;*****
;
;REREAD,ACK COMMAND

```

```

8646
8647
8648
8649 045110 012737 141001 046060 50$: MOV #141001,T26PK3 ;REREAD,ACK COMMAND
8650 045116 55$:
8651 045116 012704 046060 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
8652 045122 65$:
8653 045122 010465 177776 MOV R4,TSDB(R5) ;ISSUE COMMAND
8654 045126 004737 017134 JSR PC,WAITF ;WAIT FOR SSR TO SET
8655 045132 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
8656 045136 012702 100206 MOV #SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
8657 045142 020102 CMP R1,R2 ;ARE THEY EQUAL
8658 045144 001406 BEQ 75$ ;BR, IF OK
8659 045146 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
8663 045152 ERRHRD ERRNO,T26WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      045152 104456 TRAP C$ERHRD
      045154 000523 .WORD 339
      045156 047053 .WORD T26WDE
      045160 011710 .WORD PKTSSR
8664 045162 75$: CKLOOP ;LOOP IF SELECTED
      045162 104406 TRAP C$CLP1
8665
8666
8667
8668
8669
8670
8671
8672 045164 013701 045756 MOV T26BFR+6,R1 ;GET XSTO STATUS WORD
8673 045170 010102 MOV R1,R2 ;SET UP EXPECTED
8674 045172 052702 002000 BIS #BIT10,R2 ;SET THE NEF BIT
8675 045176 020102 CMP R1,R2 ;ARE THEY EQUAL
8676 045200 001406 BEQ 170$ ;BR, IF EQUAL (GOOD)
8677 045202 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
8681 045206 ERRHRD ERRNO,T26NEF,EXPREC ;NEF SHOULD BE SET
      045206 104456 TRAP C$ERHRD
      045210 000524 .WORD 340
      045212 046204 .WORD T26NEF
      045214 016360 .WORD EXPREC
8682 045216 170$:
8683 045216 005103 COM R3 ;RESET THE SWITCH
8684 045220 001262 BNE 26$ ;BR, IF FIRST TIME THROUGH HERE
8685 045222 ENDSUB
      045222 L10064:
      045222 104403 TRAP C$ESUB
8686 045224 023727 002170 000031 CMP FATFLG,#25. ;IS ERROR COUNT AT 25
8687 045232 002402 BLT 999$ ;BR, IF LESS THAN 25
8688 045234 004737 020170 JSR PC,CKDROP ;TRY TO DROP THE UNIT
8689 045240 999$:

```

8691				;*
8692				;
8693				;TEST 2, SUBTEST 15
8694				;
8695				;VERIFIES THAT REREAD PREVIOUS WITH OPP=1 (SPACE
8696				;REVERSE, READ FORWARD) AND REREAD PREVIOUS WITH OPP=1
8697				;(READ REVERSE, SPACE FORWARD) ISSUED WHEN THE TAPE
8698				;POSITIONED JUST BEFORE THE FIRST RECORD ON TAPE (BUT
8699				;NOT AT BOT) CAUSES TAPE STATUS ALERT TERMINATION WITH
8700				;THE REVERSE INTO BOT (RIB) STATUS BIT SET.
8701				;
8702				;
8703				;
8704				;-
8705	045240			BGNSUB
	045240			;***** BEGIN SUBTEST *****
	045240	104402		T2.15:
				TRAP C\$BSUB
8706	045242	005037	046110	CLR T26CNU ;CLEAR OUT COUNTER
8707	045246	004737	050650	JSR PC,T26REST ;SET COMMAND PACKET
8708	045252	004737	050742	JSR PC,T26RT2 ;SET UP OTHER COMMAND PACKET
8709	045256	004737	051004	JSR PC,T26RT3 ;SET UP OTHER COMMAND PACKET
8710				
8711				;*****
8712				;
8713				;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
8714				;
8715				;*****
8716				
8717	045262	004737	016660	JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
8718	045266	103407		BCS 20\$;BR IF INIT WAS OK
8719	045270	004737	020116	JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
8723	045274	010001		MOV RO,R1 ;CONTENTS OF TSSR REGISTER
8724	045276			ERRDF ERNO,\$FIERR,\$FIMSG ;FATAL ERROR TSSR WAS NOT OK
	045276	104455		TRAP C\$ERDF
	045300	000525		.WORD 341
	045302	003550		.WORD \$FIERR
	045304	011676		.WORD \$FIMSG
8725	045306			20\$:
8726				
8727	045306	012704	045730	MOV @T26PACKET,h1 ;SUBROUTINE NEEDS PACKET ADDRESS
8728				
8729				;*****
8730				;
8731				;WRITE CHARACTERISTICS COMMAND (CALL TO WRTHCR)
8732				;
8733				;*****
8734				
8735	045312	004737	010342	JSR PC,WRTHCR ;ISSUE WRITE CHARACTERISTICS
8736	045316	103407		BCS 25\$;BR, IF COMMAND ISSUED OK
8737	045320	004737	020116	JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
8741	045324	010001		MOV RO,R1 ;SAVE CONTENTS OF TSSR
8742	045326			ERRHRD ERNO,WRTMSG,\$FIMSG ;WRITE CHARACTERISTISC FAILED
	045326	104456		TRAP C\$ERHRD
	045330	000526		.WORD 342
	045332	004754		.WORD WRTMSG
	045334	011676		.WORD \$FIMSG
8743	045336			25\$: CKLOOP ;LOOP IF SELECTED

```
045336 104406                                TRAP      C$CLP1
8744
8745      ;*****
8746      ;
8747      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8748      ;
8749      ;*****
8750
8751 045340 004737 010444      26$:      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
8752 045344 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR
8753 045350 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED TSSR
8754 045354 103407      BCS      30$      ;BR, IF NO PROBLEM
8755 045356 010004      MOV      R0,R4      ;PACKET ADDRESS SET UP
8756 045360 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8760 045364      ERRHRD  ERRNO,T26RWN,PKTSSR      ;REWIND NOT ACCEPTED
      045364 104456                                TRAP      C$ERHRD
      045366 000527                                .WORD      343
      045370 047414                                .WORD      T26RWN
      045372 011710                                .WORD      PKTSSR
8761 045374      30$:      CKLOOP                                ;LOOP IF SELECTED
      045374 104406                                TRAP      C$CLP1
8762
8763      ;*****
8764      ;
8765      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8766      ;
8767      ;*****
8768
8769 045376 013701 045756      MOV      T26BFR+6,R1      ;PICK UP XSTO
8770 045402 010102      MOV      R1,R2      ;SET UP EXPECTED
8771 045404 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
8772 045410 020102      CMP      R1,R2      ;DOES EXP = REC'D
8773 045412 001406      BEQ      40$      ;BR, IF EQUAL (OK)
8774 045414 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8778 045420      ERRHRD  ERRNO,T26BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      045420 104456                                TRAP      C$ERHRD
      045422 000530                                .WORD      344
      045424 047125                                .WORD      T26BOT
      045426 016360                                .WORD      EXPREC
8779 045430      40$:
8780
8781      ;*****
8782      ;
8783      ;ISSUE SPACE RECORDS COMMAND      VALUE IN R3 SETS NUMBER OF RECORDS
8784      ;BIT 15 SETS DIRECTION - 0=FORWARD      1=REVERSE
8785      ;
8786      ;*****
8787
8788 045430 012703 000001      MOV      #000001,R3      ;SET UP SPACE FORWARD 1 RECORD
8789 045434 004737 010144      JSR      PC,SPACE      ;ISSUE SPACE COMMAND
8790 045440 103411      BCS      75$      ;BR, IF OK
8791 045442 016501 000000      MOV      TSSR(R5),R1      ;GET STATUS DATA
8792 045446 010004      MOV      R0,R4      ;GET PACKET ADDRESS
8793 045450 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
8797 045454      ERRHRD  ERRNO,T26WDE,PKTSSR      ;TSSR INCORRECT AFTER READ DATA
      045454 104456                                TRAP      C$ERHRD
      045456 000531                                .WORD      345
```

Address	Offset	Value	Label	Comment	Word	Value
8798	045460	047053	75\$:	CKLOOP		
	045462	011710				
	045464	104406				
8799						
8800						
8801						
8802						
8803						
8804						
8805						
8806						
8807	045466	012703	100001	MOV #100001,R3		
8808	045472	004737	010144	JSR PC,SPACE		
8809	045476	103411		BCS 175\$		
8810	045500	016501	000000	MOV TSSR(R5),R1		
8811	045504	010004		MOV R0,R4		
8812	045506	004737	020116	JSR PC,FATCHK		
8816	045512			ERRHRD ERRNO,T26WDE,PKTSSR		
	045512	104456				
	045514	000532				
	045516	047053				
	045520	011710				
8817	045522			175\$:	CKLOOP	
	045522	104406				
8818	045524	013737	003072	MOV FREE,T26RB		
8819	045532	005737	046110	TST T26CNU		
8820	045536	001404		BEQ 176\$		
8821						
8822						
8823						
8824						
8825						
8826						
8827						
8828	045540	012737	161001	MOV #161001,T26PK3		
8829	045546	000403	046060	BR 178\$		
8830						
8831						
8832						
8833						
8834						
8835						
8836						
8837	045550	012737	141001	MOV #141001,T26PK3		
8838	045556		046060	178\$:		
8839	045556	012704	046060	MOV #T26PK3,R4		
8840	045562	010465	177776	MOV R4,TSD8(R5)		
8841	045566	004737	017134	JSR PC,WAITF		
8842	045572	016501	000000	MOV TSSR(R5),R1		
8843	045576	012702	100204	MOV #SSR!SC!BIT2,R2		
8844	045602	020102		CMP R1,R2		
8845	045604	001406		BEQ 180\$		
8846	045606	004737	020116	JSR PC,FATCHK		
8850	045612			ERRHRD ERRNO,T26WDE,PKTSSR		
	045612	104456				
	045614	000533				

Address	Hex	Dec	Label	Op	Opnd	Comment	Trap	Word
8851	045616	047053	1801:	CKLOOP		;LOOP IF SELECTED		
	045620	011710						.WORD T26WDE
	045622	104406						.WORD PKTSSR
8852	045624	013701	045764	MOV	T26BFR.14,R1	;GET XST3 STATUS WORD	TRAP	C1CLP1
8853	045630	010102		MOV	R1,R2	;SET UP EXPECTED		
8854	045632	052702	000001	BIS	#BIT0,R2	;SET THE NEF BIT		
8855	045636	020102		CMP	R1,R2	;ARE THEY EQUAL		
8856	045640	001406		BEQ	1901	;BR, IF EQUAL (GOOD)		
8857	045642	004737	020116	JSR	PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS		
8861	045646			ERRHRD	ERRNO,T26NEF,EXPREC	;NEF SHOULD BE SET		
	045646	104456					TRAP	C1ERHRD
	045650	000534					.WORD	348
	045652	046204					.WORD	T26NEF
	045654	016360					.WORD	EXPREC
8862	045656		1901:					
8863	045656	005137	046110	COM	T26CNU	;SET SWITCH THE OTHER WAY		
8864	045662	001226		BNE	261	;BR, IF FIRST TIME THROUGH		
8865	045664			ENDSUB		;***** END SUBTEST *****		
	045664	104403						
8866	045666	023727	002170	000031	CMP	FATFLG.#25.	TRAP	C1ESUB
8867	045674	002402			BLT	9991		
8868	045676	004737	020170		JSR	PC,CKDROP		
8869	045702		9991:					
8870								
8871								
8872								
8873	045702	004737	017312		JSR	PC,TSTLOOP		
8874	045706	103002			BCC	1631		
8875	045710	000137	031732		JMP	T26LOOP		
8876	045714		1631:					
8877	045714			EXIT	TST	;ALL DONE THIS TEST		
	045714	104432					TRAP	C1EXIT
	045716	003116					.WORD	L10046

8879			; LOCAL STORAGE FOR THIS TEST	
8880				
8881				
8883	045720		.BLKB 10-<.-TUV2A&7>	
8885	045730		T26PACKET:	;COMMAND PACKET FOR TEST
8886	045730	014004	.WORD 14004	;WRITE CHARACTERISTICS COMMAND, WITH CVC=1. ACK
8887	045732	045740	.WORD T26DATA	;ADDRESS OF CHARACTERISTICS BLOCK
8888	045734	000000	.WORD 0	
8889	045736	000012	.WORD 10.	;STARTING VALUE OF BLOCK SIZE
8890	045740		T26DATA:	;CHARACTERISTICS DATA BLOCK
8891	045740	045750	.WORD T26BFR	;ADDRESS OF MESSAGE BUFFER
8892	045742	000000	.WORD 0	
8893	045744	000024	.WORD 20.	;LENGTH OF MESSAGE BUFFER
8894	045746	000000	.WORD 0	
8895	045750		T26BFR: .BLKW 25.	;MESSAGE BUFFER
8896				
8897			; WRITE SUBSYSTEM MEMORY COMMAND PACKET	
8898				
8900	046032		.BLKB 10-<.-TUV2A&7>	
8902	046040		T26PK2:	
8903	046040	100006	.WORD 100006	;WRITE SUB SYS MEM COMMAND, AND ACK
8904	046042	046070	.WORD T26BF2	;ADDRESS OF SELECT BLOCK DATA
8905	046044	000000	.WORD 0	
8906	046046	000006	.WORD 6.	;SIZE OF DATA PACKET
8908	046050		.BLKB 10-<.-TUV2A&7>	
8910	046060		T26PK3:	
8911	046060	140005	.WORD 140005	;REREAD COMMAND, CVC=1 AND ACK
8912	046062		T26RB:	
8913	046062	003072	T26WB: .WORD FREE	;ADDRESS OF WRITE BUFFER
8914	046064	000000	.WORD 0	
8915	046066	000000	T26SZ: .WORD 0	;SIZE OF BUFFER (EXTENT)
8916			.EVEN	
8917				
8918	046070		T26BF2:	
8919	046070	010	T26BS0: .BYTE 10	;BSELO AREA
8920	046071	200	T26BS1: .BYTE 200	;BSEL1 AREA
8921	046072	000000	T26S2: .WORD 0	;SEL 2 AREA
8922	046074	000000	T26S3: .WORD 0	;DATA AREA
8923				
8924			.EVEN	
8925			;TAPE MOTION PACKET COMMAND VALUES	
8926				
8927	046076	140001	T26RN: .WORD 140001	;READ DATA
8928	046100	141401	.WORD 141401	;REREAD NEXT OPP=0
8929	046102	161401	.WORD 161401	;REREAD NEXT OPP=1
8930	046104	177777	.WORD 177777	;END OF DATA
8931				
8932	046106	000000	T26CNT: .WORD 0	;TAPE RECORD COUNTER STORAGE AREA
8933	046110	000000	T26CNU: .WORD 0	;TAPE RECORD COUNTER STORAGE AREA
8934	046112	000000	T26RSZ: .WORD 0	;RECORD STORAGE SIZE AREA
8935	046114	000000	T26DLV: .WORD 0	;DELAY COUNTER AREA

```

8937
8938
8939      ;*
8940      ;LOCAL TEXT MESSAGES FOR TEST
8941      ;*
8942
8943 046116      124      141      160  T26WNG: .ASCIZ 'Tape Position Incorrect After REREAD Previous (OPP=1)'
8944 046204      122      105      122  T26NEF: .ASCIZ 'REREAD PREVIOUS, At BOT, Failed To Set NEF (XST0)'
8945 046266      124      123      123  T26RDF: .ASCIZ 'TSSR Incorrect After READ DATA Command'
8946 046335      122      105      122  T26RRF: .ASCIZ 'REREAD Previous (Space Reverse, Read Forward) Command Failed'
8947 046432      122      105      122  T26RRG: .ASCIZ 'REREAD Previous (Read Reverse, Space Forward) Command Failed'
8948 046527      120      117      123  T26SC: .ASCIZ 'POSITION (Space Command) Failed, TSSR Not Correct'
8949 046611      122      111      102  T26LOR: .ASCIZ 'RIB NOT SET AFTER READ REVERSE INTO BOT'
8950 046661      124      123      123  T26WDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
8951 046736      111      154      154  T26LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XST0'
8952 047017      122      105      122  T26SSR: .ASCIZ 'REREAD COMMAND Not Accepted'
8953 047053      124      123      123  T26WDE: .ASCIZ 'TSSR Not Correct After WRITE DATA Command'
8954 047125      124      141      160  T26BOT: .ASCIZ 'Tape Not At BOT After REWIND Command'
8955 047172      104      141      164  T26DTA: .ASCIZ 'Data Written To Tape Not Equal To Data Read From Tape'
8956 047260      122      105      122  T26EOT: .ASCIZ 'REREAD DATA OVER EOT GAVE NO TAPE STATUS ALERT'
8957 047337      124      123      123  T26TM: .ASCIZ 'TSSR Not Correct After REREAD COMMAND Reject'
8958 047414      122      145      167  T26RWN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
8959 047463      122      101      115  T26RNC: .ASCIZ 'RAM Error, Correct Data Pattern Not In Ram'
8960 047536      124      123      123  T26AM3: .ASCIZ 'TSSR Init. Failed After REREAD COMMAND'
8961 047605      104      162      151  T26OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
8962 047660      124      123      123  T26WDD: .ASCIZ 'TSSR Not Correct After REREAD DATA Command, SWB Bit Set'
8963 047750      124      123      123  T26WDC: .ASCIZ 'TSSR Not Correct After REREAD DATA Command'
8964 050023      103      126      103  T26VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
8965 050076      124      123      102  T26BA: .ASCIZ 'TSBA Not Correct After REREAD DATA Command'
8966 050151      127      122      111  T26WSS: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
8967 050240      122      145      141  T26LON: .ASCIZ 'Reading Long Record Failed To Set RLL Bit In XST0'
8968 050322      122      145      141  T26LOP: .ASCIZ 'Reading Long Record Failed To Set RLS Bit In XST0'
8969 050404      122      145      163  T26PBP: .ASCIZ 'Residual Byte Count Incorrect After Short Record Read'
8970 050472      122      145      141  T26TRL: .ASCIZ 'Reading Long Record Failed To Give Tape Status Alert'
8971 050560      104      141      164  T26NEQ: .ASCIZ 'Data REREAD From Tape Not Correct, After SWB=1'
8972 050637      122      145      162  TST26ID: .ASCIZ 'Rereads'
8973
8974      .EVEN
8975
8976      ;*
8977      ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
8978      ;WRITE SUBSYSTEM MEMORY COMMAND
8979      ;*
8980
8981 050650      T26REST:
8982 050650      SAVREG
8983 050654      012701 045730      MOV      #T26PACKET,R1      ;SAVE THE REGISTERS
8984 050660      012721 140004      MOV      #140004,(R1).      ;START OF THE PACKET
8985 050664      012721 045740      MOV      #T26DATA,(R1).      ;WRITE SUBSYSTEM MEM. WITH ACK, CVC=1
8986 050670      005021      CLR      (R1).      ;ADDRESS OF CHARAISTICS DATA BLOCK
8987 050672      012721 000012      MOV      #10.,(R1).      ;EXTENDED ADDRESS
8988 050676      012721 045750      MOV      #T26BFR,(R1).      ;SIZE OF DATA BLOCK IN BYTES
8989 050702      005021      CLR      (R1).      ;ADDRESS OF MESSAGE BUFFER
8990 050704      012721 000024      MOV      #20.,(R1).      ;LENGTH OF MESSAGE BUFFER
8991 050710      005021      CLR      (R1).
8992 050712      012711 000000      MOV      #0,(R1)      ;SELECT DRIVE ZERO (0)
8993 050716      012702 000030      MOV      #24.,R2      ;NUMBER OF LOCATIONS TO BE CLEARED

```

E2

CZTKGA TK 25 FRI END FUNC 03 MACRO M1200 20-APR 84 08:13 PAGE 113-1

SEQ 224

```

8994 050722 012762 177777 045750 64: MOV #177777,T26BFR(R2) ;ALL ONES TO MESSAGE BUFFER
8995 050730 005742 TST -(R2) ;NEXT LOCATION
8996 050732 020227 000000 CMP R2,#0 ;CHECK FOR END OF LOOP
8997 050736 001371 BNE 64: ;KEEP GOING UNTIL DONE
8998 050740 000207 RTS PC ;RETURN
8999
9000
9001 050742 T26RT2: SAVREG ;SAVE THE REGISTERS
9002 050742 MOV #T26PK2,R1 ;START OF THE PACKET
9003 050746 012701 046040 MOV #140006,(R1); ;WRITE SUBSYSTEM MEM. WITH ACK,CVC=1.
9004 050752 012721 140006 MOV #T26BF2,(R1); ;ADDRESS OF DATA BLOCK
9005 050756 012721 046070 CLR (R1); ;EXTENDED ADDRESS
9006 050762 005021 MOV #6,(R1); ;SIZE OF DATA BLOCK IN BYTES
9007 050764 012721 000006 CLR (R1);
9008 050770 005021 MOV #T26BF2,R1 ;POINT TO DATA SEL AREA
9009 050772 012701 046070 CLR (R1);
9010 050776 005021 CLR (R1);
9011 051000 005011 RTS PC ;RETURN
9012 051002 000207 T26RT3: SAVREG ;SAVE THE REGISTERS
9013 051004 MOV #T26PK3,R1 ;START OF THE PACKET
9014 051004 MOV #0,(R1); ;WRITE SUBSYSTEM MEM. WITH ACK,
9015 051010 012701 046060 MOV #0,(R1); ;ADDRESS OF DATA BLOCK
9016 051014 012721 000000 CLR (R1); ;EXTENDED ADDRESS
9017 051020 012721 000000 MOV #0,(R1) ;SIZE OF DATA BLOCK IN BYTES
9018 051024 005021 RTS PC ;RETURN
9019 051026 012711 000000
9020 051032 000207
9021 051034
051034
051034 104401

```

L10046: TRAP C\$ETST

```
9023          .SBTTL TEST 3: WRITE DATA RETRY  
9024      ;  
9025      ;  
9026      ;THIS TEST VERIFIES PROPER OPERATION OF THE WRITE DATA RETRY  
9027      ;COMMAND (SPACE REVERSE, ERASE, WRITE DATA)  
9028      ;  
9029      ;  
9030      ;THE TEST CONSISTS OF THE FOLLOWING 5 SUBTESTS  
9031      ;  
9032      ;  
9033      ;  
9034      ;-  
9035      BGNTST                                     T3::  
9036      051036   005037   002170       CLR     FATFLG           ;CLEAR FATAL ERROR FLAG  
9037      051042   005037   003100       CLR     KTFLG             ;HOLD OFF KT11  
9038      051046   012737   005771   002146    MOV     @EPRT2,EPRTSW        ;SECONDARY ERROR MESSAGE  
9039      051054   005037   003102       CLR     KTENABLE            ;TURN OFF KT11  
9040      051060   004737   020262       JSR     PC,KTOFF              ;TURN KT11 BACK OFF IF THERE  
9041      051064   012700   060633       MOV     @TST27ID,R0           ;ASCII MESSAGE TO IDENTIFY TEST  
9042      051070   004737   017424       JSR     PC,TSTSETUP          ;DO INITIAL TEST SETUP  
9043      051074   012737   000001   002164    MOV     @I,LOOPCNT         ;PERFORM I ITERATIONS  
9044      051102   005037   055676       CLR     T27CNT               ;CLEAR TAPE RECORD COUNTER  
9045      ;  
9046      ;  
9047      ;TEST 3, SUBTEST 1  
9048      ;  
9049      ;  
9050      ;VERIFIES THAT A WRITE DATA RETRY COMMAND ISSUED WHILE  
9051      ;THE TAPE IS POSITIONED AT BOT CAUSES FUNCTION REJECT  
9052      ;TERMINATION, WITH THE NON-EXECUTABLE FUNCTION (NEF)  
9053      ;ERROR BIT SET.  
9054      ;  
9055      ;  
9056      ;-  
9057      ;  
9058      T27LOOP:  
9059      051106   051106   051106   BGNSUB                               ;>>>>>>>> BEGIN SUBTEST >>>>>>>  
9060      051106   051106   104402                                          T3.1: TRAP C$BSUB  
9061      051110   004737   060654       JSR     PC,T27REST           ;SET COMMAND PACKET  
9062      051114   004737   060746       JSR     PC,T27RT2           ;SET UP OTHER COMMAND PACKET  
9063      051120   004737   061010       JSR     PC,T27RT3           ;SET UP OTHER COMMAND PACKET  
9064      051124   012737   176750   055702    MOV     @65000.,T27DLY        ;SET UP DELAY COUNTER  
9065      051132   005037   055676       CLR     T27CNT               ;CLEAR COUNTER  
9066      ;*****  
9067      ;ISSUE CONTROLLER "SOFT" INITIALIZ - CARRY BIT CLEAR IF ERROR  
9068      ;*****  
9069      10%:      JSR     PC,SOFINIT                ;DO INITIALIZE ON CONTROLLER  
9070                      BCS     20%                  ;BR IF INIT WAS OK  
9071                      DELAY    250                 ;DELAY ABOUT .25 SEC  
9072      051144   012727   000250                          MOV     #250,(PC).  
9073      051150   000000                                         WORD     0
```

```

051152 013727 002116
051156 000000
051160 005367 177772
051164 001375
051166 005367 177756
051172 001367
9079 051174 005337 055702
9080 051200 001356
9081 051202 004737 020116
9085 051206 010001
9086 051210
051210 104455
051212 000455
051214 003550
051216 011676
9087 051220
9088 051220 012704 055520
9089
9090
9091
9092
9093
9094
9095
9096 051224 004737 010342
9097 051230 103407
9098 051232 004737 020116
9102 051236 010001
9103 051240
051240 104456
051242 000456
051244 004754
051246 011676
9104 051250
051250 104406
9105
9106
9107
9108
9109
9110
9111
9112 051252 004737 010444
9113 051256 103407
9114 051260 010004
9115 051262 004737 020116
9119 051266
051266 104456
051270 000457
051272 057055
051274 011710
9120 051276
051276 104406
9121
9122
9123
9124

MOV L$DLY,(PC)+
.WORD 0
DEC -6(PC)
BNE -4
DEC -22(PC)
BNE -20

DEC T27DLY
BNE 10$
JSR PC,FATCHK
MOV R0,R1
ERRDF ERRNO,SFIERR,SFIMSG
;BUMP COUNTER
;BR, IF COUNTER NOT DONE
;INC AND CHECK FOR MORE THAN 25 ERRORS
;CONTENTS OF TSSR REGISTER
;FATAL ERROR TSSR WAS NOT OK
TRAP C$ERDF
.WORD 301
.WORD SFIERR
.WORD SFIMSG

20$: MOV #T27PACKET,R4
;SUBROUTINE NEEDS PACKET ADDRESS
;*****
;
;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
;
;*****
JSR PC,WRTCHR
BCS 25$
JSR PC,FATCHK
MOV R0,R1
ERRHRD ERRNO,WRTMSG,SFIMSG
;ISSUE WRITE CHARACTERISTICS
;BR, IF COMMAND ISSUED OK
;INC AND CHECK FOR MORE THAN 25 ERRORS
;SAVE CONTENTS OF TSSR
;WRITE CHARACTERISTICS FAILED
TRAP C$ERHRD
.WORD 302
.WORD WRTMSG
.WORD SFIMSG

25$: CKLOOP
;LOOP IF SELECTED
TRAP C$CLP1
;*****
;
;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
;
;*****
JSR PC,REWIND
BCS 30$
MOV R0,R4
JSR PC,FATCHK
ERRHRD ERRNO,T27RWN,PKTSSR
;CALL TAPE REWIND COMMAND
;BR, IF NO PROBLEM
;SET UP REWIND PACKET ADDRESS
;INC AND CHECK FOR MORE THAN 25 ERRORS
;REWIND NOT ACCEPTED
TRAP C$ERHRD
.WORD 303
.WORD T27RWN
.WORD PKTSSR

30$: CKLOOP
;LOOP IF SELECTED
TRAP C$CLP1
;*****
;
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)

```

```

9125
9126
9127
9128 051300 013701 055546      MOV      T27BFR+6,R1      ;PICK UP XSTO
9129 051304 010102      MOV      R1,R2      ;SET UP EXPECTED
9130 051306 052702 000002      BIS      @BIT1,R2      ;SET BOT BIT IN EXPECTED
9131 051312 020102      CMP      R1,R2      ;DOES EXP = REC'D
9132 051314 001406      BEQ      *0$      ;BR, IF EQUAL (OK)
9133 051316 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9137 051322      ERRHRD  ERRNO,T27BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
          051322 104456      TRAP      C$ERHRD
          051324 000460      .WORD    304
          051326 056551      .WORD    T27BOT
          051330 016360      .WORD    EXPREC
9138 051332      40$:      CKLOOP      ;LOOP IF SELECTED      TRAP      C$CLP1
          051332 104406      MOV      @256.,T27SZ      ;SET UP RECORD SIZE
9139 051334 012737 000400 055656      MOV      FREE,T27WB      ;ADDRESS OF WRITE BUFFER
9140 051342 013737 003072 055652
9141
9142
9143
9144
9145
9146
9147
9148 051350 012737 141005 055650      MOV      @141005,T27PK3      ;WRITE DATA RETRY,ACK,CVC=1 COMMAND
9149 051356 012704 055650      MOV      @T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
9150 051362 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
9151 051366 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
9152 051372 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
9153 051376 012702 100206      MOV      @SSR!SC!BIT1!BIT2,R2      ;SET UP EXPECTED
9154 051402 020102      CMP      R1,R2      ;ARE THEY EQUAL
9155 051404 001406      BEQ      75$      ;BR, IF OK
9156 051406 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9160 051412      ERRHRD  ERRNO,T27WDE,PKTSSR      ;TSSR INCORRECT AFTER READ DATA
          051412 104456      TRAP      C$ERHRD
          051414 000461      .WORD    305
          051416 056462      .WORD    T27WDE
          051420 011710      .WORD    PKTSSR
9161 051422      75$:      CKLOOP      ;LOOP IF SELECTED      TRAP      C$CLP1
          051422 104406
9162
9163
9164
9165
9166
9167
9168
9169 051424 013701 055546      MOV      T27BFR+6,R1      ;GET XSTO STATUS WORD
9170 051430 010102      MOV      R1,R2      ;SET UP EXPECTED
9171 051432 052702 002000      BIS      @BIT10,R2      ;SET THE NEF BIT
9172 051436 020102      CMP      R1,R2      ;ARE THEY EQUAL
9173 051440 001406      BEQ      170$      ;BR, IF EQUAL (GOOD)
9174 051442 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9178 051446      ERRHRD  ERRNO,T27NEF,EXPREC      ;NEF SHOULD BE SET
          051446 104456      TRAP      C$ERHRD
          051450 000462      .WORD    306

```

CZTKGA TK 25 FRT END FUNC #3 MACRO M1200 20-APR-84 08:13 PAGE 114 3
TEST 3: WRITE DATA RETRY

SEQ 228

```
          051452 060221
          051454 016360
9179 051456
9180 051456          1704:
          051456          ENDSUB
          051456 104403
9181 051460 023727 002170 000031
9182 051466 002402
9183 051470 004737 020170
9184 051474          9994:

          CMP      FATFLG, #25.
          BLT      9994
          JSR      PC, CKDROP
```

```
.WORD  T27NEF
.WORD  EXPREC
```

```
          L10067:
                                TRAP
;IS ERROR COUNT AT 25
;BR, IF LESS THAN 25
;TRY TO DROP THE UNIT
                                C$ESUB
```

[illegible]

```

9238
9239 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9240 ;
9241 ;*****
9242
9243 051570 004737 010444 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
9244 051574 103411 BCS 26$ ;BR, IF NO PROBLEM
9245 051576 010004 MOV R0,R4 ;SET UP REWIND PACKET ADDRESS
9246 051600 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
9247 051604 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
9251 051610 ERRHRD ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
          051610 104456 TRAP C$ERHRD
          051612 000465 .WORD 309
          051614 057055 .WORD T27RWN
          051616 011710 .WORD PKTSSR
9252 051620 26$: CKLOOP ;LOOP IF SELECTED
          051620 104406 TRAP C$CLP1
9253 051622 012703 000400 MOV #256.,R3 ;STARTING RECORD SIZE
9254 051626 013737 0C3072 055652 MOV FREE,T27WB ;STARTING WRITE BUFFER ADDRESS
9255
9256 ;*****
9257 ;
9258 ;WRITE DATA,CVC=1,ACK COMMAND
9259 ;
9260 ;*****
9261
9262 051634 012737 140005 055650 MOV #140005,T27PK3 ;WRITE DATA,CVC=1,ACK COMMAND
9263 051642 012704 055650 MOV #T27PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
9264 051646 010337 055656 MOV R3,T27SZ ;SET UP RECORD SIZE IN PACKET
9265 051652 010465 177776 MOV R4,TSD8(R5) ;ISSUE COMMAND
9266 051656 004737 017134 JSR PC,WAITF ;WAIT FOR SSR TO SET
9267 051662 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
9268 051666 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
9269 051672 020102 CMP R1,R2 ;ARE THEY EQUAL
9270 051674 001406 BEQ 28$ ;BR, IF OK
9271 051676 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
9275 ;SOFT ERROR GENERATED BECAUSE THE
9276 ;WRITE COMMAND IS NOT BEING CHECKED
9277 ;HERE. IT WAS CHECKED IN LEAH2
9278 051702 ERRSOFT ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
          051702 104457 TRAP C$ERSOFT
          051704 000466 .WORD 310
          051706 005011 .WORD WRTErr
          051710 011710 .WORD PKTSSR
9279 051712 28$: CKLOOP ;LOOP IF SELECTED
          051712 104406 TRAP C$CLP1
9280
9281 ;*****
9282 ;
9283 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9284 ;
9285 ;*****
9286
9287 051714 004737 010444 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
9288 051720 103411 BCS 30$ ;BR, IF NO PROBLEM
9289 051722 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
9290 051726 010004 MOV R0,R4 ;SET UP REWIND PACKET ADDRESS

```

```

9291 051730 004737 020116      JSR    PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9295 051734      ERRHRD    ERRNO,T27RWN,PKTSSR      ;REWIND NOT ACCEPTED
      051734 104456      TRAP    C$ERHRD
      051736 000467      .WORD   311
      051740 057055      .WORD   T27RWN
      051742 011710      .WORD   PKTSSR
9296 051744      30$:    CKLOOP      ;LOOP IF SELECTED      TRAP    C$CLP1
      051744 104406
9297
9298      ;*****
9299      ;
9300      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9301      ;
9302      ;*****
9303
9304 051746 013701 055546      MOV     T27BFR+6,R1      ;PICK UP XSTO
9305 051752 010102      MOV     R1,R2      ;SET UP EXPECTED
9306 051754 052702 000002      BIS     @BIT1,R2      ;SET BOT BIT IN EXPECTED
9307 051760 020102      CMP     R1,R2      ;DOES EXP = REC'D
9308 051762 001406      BEQ     40$      ;BR, IF EQUAL (OK)
9309 051764 004737 020116      JSR     PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9313 051770      ERRHRD    ERRNO,T27BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      051770 104456      TRAP    C$ERHRD
      051772 000470      .WORD   312
      051774 056551      .WORD   T27BOT
      051776 016360      .WORD   EXPREC
9314 052000      40$:    CKLOOP      ;LOOP IF SELECTED      TRAP    C$CLP1
      052000 104406
9315
9316      ;*****
9317      ;
9318      ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
9319      ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
9320      ;
9321      ;*****
9322
9323 052002 012703 000001      MOV     @1,R3      ;PARAMETER SPACE FORWARD 1 RECORD
9324 052006 004737 010144      JSR     PC,SPACE      ;CALL SPACE RECORDS ROUTINE
9325 052012 103413      BCS     50$      ;BR, IF NO ERRORS
9326 052014 016501 000000      MOV     TSSR(R5),R1      ;GET TSSR CONTENTS
9327 052020 012702 000200      MOV     @SSR,R2      ;SET UP EXPECTED
9328 052024 010004      MOV     R0,R4      ;SET UP REWIND PACKET ADDRESS
9329 052026 004737 020116      JSR     PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9333 052032      ERRHRD    ERRNO,T27SCF,PKTSSR      ;SPACE RECORDS COMMAND FAILED
      052032 104456      TRAP    C$ERHRD
      052034 000471      .WORD   313
      052036 060317      .WORD   T27SCF
      052040 011710      .WORD   PKTSSR
9334 052042      50$:    CKLOOP      ;LOOP IF SELECTED      TRAP    C$CLP1
      052042 104406

```

```

9336 ;*****
9337 ;
9338 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
9339 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
9340 ;
9341 ;*****
9342
9343 052044 012703 100001      MOV      #100001,R3      ;PARAMETER SPACE REVERSE 1 RECORD
9344 052050 004737 010144      JSR      PC,SPACE      ;CALL SPACE RECORDS ROUTINE
9345 052054 103413              BCS      60$          ;BR, IF NO ERRORS
9346 052056 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
9347 052062 012702 000200      MOV      #SSR,R2          ;SET JP EXPECTED
9348 052066 010004              MOV      R0,R4          ;SET UP REWIND PACKET ADDRESS
9349 052070 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9353 052074              ERRHRD  ERRNO,T27SCF,PKTSSR      ;SPACE RECORDS COMMAND FAILED
9354 052074 104456              TRAP      C$ERHRD
9354 052076 000472              .WORD    314
9354 052100 060317              .WORD    T27SCF
9354 052102 011710              .WORD    PKTSSR
9354 052104              60$:  CKLOOP          ;LOOP IF SELECTED
9354 052104 104406              TRAP      C$CLP1
9355 052106 013737 003072 055652  MOV      FREE,T27RB      ;ADDRESS OF BUFFER

```



```

9447      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9448      ;
9449      ;*****
9450
9451 052342 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
9452 052346 103407      BCS      30$      ;BR, IF NO PROBLEM
9453 052350 010004      MOV      R0,R4      ;SET UP REWIND PACKET ADDRESS
9454 052352 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9455 052356      ERRHRD      ERRNO,T27RWN,PKTSSR      ;REWIND NOT ACCEPTED
9456      052356 104456      TRAP      C$ERHRD
9457      052360 000477      .WORD      319
9458      052362 057055      .WORD      T27RWN
9459 052366 011710      .WORD      PKTSSR
9460      052366 104406      30$:      CKLOOP      ;LOOP IF SELECTED
9461      TRAP      C$CLP1
9462      ;*****
9463      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9464      ;
9465      ;*****
9466
9467 052370 013701 055546      MOV      T27BFR+6,R1      ;PICK UP XSTO
9468 052374 010102      MOV      R1,R2      ;SET UP EXPECTED
9469 052376 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
9470 052402 020102      CMP      R1,R2      ;DOES EXP = REC'D
9471 052404 001406      BEQ      40$      ;BR, IF EQUAL (OK)
9472 052406 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9473 052412      ERRHRD      ERRNO,T27BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
9474      052412 104456      TRAP      C$ERHRD
9475      052414 000500      .WORD      320
9476      052416 056551      .WORD      T27BOT
9477 052422 016360      .WORD      EXPREC
9478      052422 104406      40$:      CKLOOP      ;LOOP IF SELECTED
9479      TRAP      C$CLP1
9480 052424 012703 000024      MOV      #20.,R3      ;STARTING RECORD SIZE
9481 052430 013737 003072 055652      MOV      FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
9482      ;*****
9483      ;WRITE DATA,CVC=1,ACK COMMAND
9484      ;
9485      ;*****
9486
9487 052436 012737 140005 055650 65$:      MOV      #140005,T27PK3      ;WRITE DATA,CVC=1,ACK COMMAND
9488 052444 012704 055650      MOV      #T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
9489 052450 010300      MOV      R3,R0      ;SET PATTERN IN CORRECT REGISTER
9490 052452 004737 020410      JSR      PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
9491 052456 010337 055656      MOV      R3,T27SZ      ;SET UP RECORD SIZE IN PACKET
9492 052462 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
9493 052466 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
9494 052472 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
9495 052476 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
9496 052502 020102      CMP      R1,R2      ;ARE THEY EQUAL
9497 052504 001406      BEQ      80$      ;BR, IF OK
9498 052506 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9502      ;SOFT ERROR GENERATED BECAUSE THE

```

```

9503
9504
9505 052512          ERRSOF ERRNO,WRTErr,PKTSSR      ;WRITE COMMAND IS NOT BEING CHECKED
                                           ;HERE. IT WAS CHECKED IN LEAM2
                                           ;TSSR INCORRECT AFTER WRITE DATA
                                           TRAP      C$ERSOFT
                                           .WORD      321
                                           .WORD      WRTErr
                                           .WORD      PKTSSR
9506 052522          80$: CKLOOP                      ;LOOP IF SELECTED
                                           TRAP      C$CLP1
9507
9508
9509
9510
9511
9512
9513
9514 052524 012737 141005 055650      MOV      #141005,T27PK3      ;WRITE DATA RETRY,CVC=1,ACK COMMAND
9515 052532 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
9516 052536 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
9517 052542 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
9518 052546 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
9519 052552 020102      CMP      R1,R2      ;ARE THEY EQUAL
9520 052554 001406      BEQ      90$      ;BR. IF OK
9521 052556 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9525 052562          ERRMRD ERRNO,T27WRF,PKTSSR      ;TSSR INCORRECT AFTER WRITE DATA RETRY
                                           TRAP      C$ERRMRD
                                           .WORD      322
                                           .WORD      T27WRF
                                           .WORD      PKTSSR
9526 052572          90$: CKLOOP                      ;LOOP IF SELECTED
                                           TRAP      C$CLP1
9527 052574 005723      TST      (R3),      ;BUMP RECORD SIZE COUNTER
9528 052576 020327 000050      CMP      R3,#40.      ;AT 40 SIZE YET
9529 052602 001315      BNE      65$      ;BR. IF MORE RECORDS TO WRITE
9530
9531
9532
9533
9534
9535
9536
9537 052604 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
9538 052610 103407      BCS      230$      ;BR. IF NO PROBLEM
9539 052612 010001      MOV      R0,R1      ;SAVE TSSR
9540 052614 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9544 052620          ERRMRD ERRNO,T27RWN,EXPREC      ;REWIND NOT ACCEPTED
                                           TRAP      C$ERRMRD
                                           .WORD      323
                                           .WORD      T27RWN
                                           .WORD      EXPREC
9545 052630          230$: CKLOOP                      ;LOOP IF SELECTED
                                           TRAP      C$CLP1
9546
9547
9548
9549
9550

```

```

9551
9552 ;*****
9553 052632 013701 055546      MOV      T27BFR+6,R1      ;PICK UP XSTO
9554 052636 010102      MOV      R1,R2      ;SET UP EXPECTED
9555 052640 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
9556 052644 020102      CMP      R1,R2      ;DOES EXP = REC'D
9557 052646 001406      BEQ      240$      ;BR. IF EQUAL (OK)
9558 052650 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9562 052654      ERRMRD      ERRNO,T27BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
          052654 104456      TRAP      C$ERRMRD
          052656 000504      .WORD      324
          052660 056551      .WORD      T27BOT
          052662 016360      .WORD      EXPREC
9563 052664      240$: CKLOOP      ;LOOP IF SELECTED
          052664 104406      TRAP      C$CLP1
9564 052666 012703 000024      MOV      #20.,R3      ;STARTING RECORD SIZE
9565 052672 013737 003072 055652      MOV      FREE,T27RB      ;STARTING READ BUFFER ADDRESS
9566
9567 ;*****
9568 ;
9569 ;READ DATA,ACK COMMAND
9570 ;
9571 ;*****
9572
9573 052700 012737 100001 055650 265$: MOV      #100001,T27PK3      ;READ DATA,ACK COMMAND
9574 052706 012704 055650      MOV      #T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
9575 052712 010337 055656      MOV      R3,T27SZ      ;SET UP RECORD SIZE IN PACKET
9576 052716 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
9577 052722 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
9578 052726 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
9579 052732 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
9580 052736 020102      CMP      R1,R2      ;ARE THEY EQUAL
9581 052740 001406      BEQ      280$      ;BR. IF OK
9582 052742 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9586 052746      ERRMRD      ERRNO,RDERR,PKTSSR      ;TSSR INCORRECT AFTER READ DATA
          052746 104456      TRAP      C$ERRMRD
          052750 000505      .WORD      325
          052752 005104      .WORD      RDERR
          052754 011710      .WORD      PKTSSR
9587 052756      280$: CKLOOP      ;LOOP IF SELECTED
          052756 104406      TRAP      C$CLP1
9588 052760 013702 003072      MOV      FREE,R2      ;GET BUFFER ADDRESS
9589 052764 010304      MOV      R3,R4      ;GET RECORD SIZE
9590 052766 162704 000024      SUB      #20.,R4      ;POINT BACK TO 1ST RECORD
9591 052772 060204      285$: ADD      R2,R4      ;POINT TO 1ST LOC IN BUFFER
9592 052774 021403      CMP      (R4),R3      ;DATA WRITTEN = READ
9593 052776 001410      BEQ      290$      ;BR. IF DATA OK (GOOD)
9594 053000 011401      MOV      (R4),R1      ;PICK UP BAD DATA
9595 053002 010302      MOV      R3,R2      ;SET UP EXPECTED
9596 053004 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9600 053010      ERRMRD      ERRNO,T27DTA,EXPREC      ;DATA IN BUFFER NOT CORRECT
          053010 104456      TRAP      C$ERRMRD
          053012 000506      .WORD      326
          053014 060536      .WORD      T27DTA
          053016 016360      .WORD      EXPREC
9601 053020      290$: CKLOOP      ;LOOP IF SELECTED
          053020 104406      TRAP      C$CLP1

```

9602	053022	005724			TST	(R4),
9603	053024	160204			SUB	R2,R4
9604	053026	020403			CMP	R4,R3
9605	053030	001360			BNE	285\$
9606	053032	005723			TST	(R3),
9607	053034	020327	000050		CMP	R3,#40.
9608	053040	001317			BNE	265\$
9609	053042			300\$:	CKLOOP	
	053042	104406				
9610	053044			330\$:		
9611	053044				ENDSUB	
	053044					
	053044	104403				
9612	053046	023727	002170	000031	CMP	FATFLG,#25.
9613	053054	002402			BLT	999\$
9614	053056	004737	020170		JSR	PC,CKDROP
9615	053062			999\$:		

[illegible]

```

9617          ;*
9618          ;
9619          ;TEST 3, SUBTEST 4
9620          ;
9621          ;VERIFIES THAT A WRITE DATA RETRY COMMAND WITH SWB=1
9622          ;TERMINATES PROPERLY AND WRITES CORRECT DATA ON TAPE
9623          ;(THE WRITTEN RECORD IS READ AND CHECKED). VAR JUS
9624          ;BYTE COUNTS AND DATA PATTERNS ARE USED.
9625          ;
9626          ;
9627          ;
9628          ;-
9629 053062      BGNSUB                                ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
          053062                                         T3.4:
          053062 104402                                     TRAP      C$BSUB
9630 053064 004737 060654                JSR      PC,T27REST        ;SET COMMAND PACKET
9631 053070 004737 060746                JSR      PC,T27RT2         ;SET UP OTHER COMMAND PACKET
9632 053074 004737 061010                JSR      PC,T27RT3         ;SET UP OTHER COMMAND PACKET
9633 053100 012737 176750 055702        MOV      #65000.,T27DLY     ;SET UP DELAY COUNTER
9634
9635          ;*****
9636          ;
9637          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
9638          ;
9639          ;*****
9640
9641 053106 004737 016660        10$:    JSR      PC,SOFINIT           ;DO INITIALIZE ON CONTROLLER
9642 053112 103426                        BCS      20$               ;BR IF INIT WAS OK
9643 053114                        DELAY      250                     ;DELAY ABOUT .25 SEC
          053114 012727 000250                                MOV      #250,(PC)+
          053120 000000                                .WORD      0
          053122 013727 002116                                MOV      L$DLY,(PC)+
          053126 000000                                .WORD      0
          053130 005367 177772                                DEC      -6(PC)
          053134 001375                                BNE      -.4
          053136 005367 177756                                DEC      -22(PC)
          053142 001367                                BNE      -.20
9644 053144 005337 055702                DEC      T27DLY            ;BUMP COUNTER
9645 053150 001356                BNE      10$                    ;BR, IF COUNTER NOT DONE
9646 053152 004737 020116                JSR      PC,FATCHK         ;INC AND CHECK FOR MORE THAN 25 ERRORS
9650 053156 010001                MOV      R0,R1                  ;CONTENTS OF TSSR REGISTER
9651 053160                ERROF      ERRNO,SFIERR,SFIMSG          ;FATAL ERROR TSSR WAS NOT OK
          053160 104455                                TRAP      C$ERDF
          053162 000507                                .WORD      327
          053164 003550                                .WORD      SFIERR
          053166 011676                                .WORD      SFIMSG
9652 053170
9653 053170 012704 055520        20$:    MOV      @T27PACKET,R4       ;SUBROUTINE NEEDS PACKET ADDRESS
9654
9655          ;*****
9656          ;
9657          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
9658          ;
9659          ;*****
9660
9661 053174 004737 010342                JSR      PC,WRTCHR         ;ISSUE WRITE CHARACTERISTICS
9662 053200 103407                BCS      23$                    ;BR, IF COMMAND ISSUED OK

```

H3

CZIKGA TK-25 ERT END FUNC #3
TECT 3: WRITE DATA RETRY

MACRO M1200 20-APR-84 08:13 PAGE 119 1

SEQ 240

```

9663 053202 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9667 053206 010001              MOV      R0,R1      ;SAVE CONTENTS OF TSSR
9668 053210              ERRHRD  ERRNO,WRTMSG,SFMSG    ;WRITE CHARACTERISTICS FAILED
          053210 104456              TRAP      C$ERHRD
          053212 000510              .WORD    328
          053214 004754              .WORD    WRTMSG
          053216 011676              .WORD    SFMSG
9669 053220              23$:      CKLOOP              ;LOOP IF SELECTED
          053220 104406              TRAP      C$CLP1
9670
9671      ;*****
9672      ;
9673      ;ISSUE REWIND COMMAND ,0 SELECTED TAPE DRIVE
9674      ;
9675      ;*****
9676
9677 053222 004737 010444      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
9678 053226 103411              BCS      30$      ;BR, IF NO PROBLEM
9679 053230 016501 000000      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
9680 053234 010004              MOV      R0,R4      ;GET PACKET ADDRESS
9681 053236 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9685 053242              ERRHRD  ERRNO,T27RWN,PKTSSR    ;REWIND NOT ACCEPTED
          053242 104456              TRAP      C$ERHRD
          053244 000511              .WORD    329
          053246 057055              .WORD    T27RWN
          053250 011710              .WORD    PKTSSR
9686 053252              30$:      CKLOOP              ;LOOP IF SELECTED
          053252 104406              TRAP      C$CLP1
9687
9688      ;*****
9689      ;
9690      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9691      ;
9692      ;*****
9693
9694 053254 013701 055546      MOV      T27BFR+6,R1    ;PICK UP XSTO
9695 053260 010102              MOV      R1,R2      ;SET UP EXPECTED
9696 053262 052702 000002      BIS      #BIT1,R2    ;SET BOT BIT IN EXPECTED
9697 053266 020102              CMP      R1,R2      ;DOES EXP = REC'D
9698 053270 001406              BEQ      40$      ;BR, IF EQUAL (OK)
9699 053272 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9703 053276              ERRHRD  ERRNO,T27BOT,EXPREC    ;TAPE NOT AT BOT AFTER REWIND
          053276 104456              TRAP      C$ERHRD
          053300 000512              .WORD    330
          053302 056551              .WORD    T27BOT
          053304 016360              .WORD    EXPREC
9704 053306              40$:      CKLOOP              ;LOOP IF SELECTED
          053306 104406              TRAP      C$CLP1
9705 053310 012703 000024      MOV      #20.,R3      ;STARTING RECORD SIZE
9706 053314 013737 003072 055652  MOV      FREE,T27WB    ;STARTING WRITE BUFFER ADDRESS
9707
9708      ;*****
9709      ;
9710      ;WRITE DATA,CVC=1,ACK COMMAND
9711      ;
9712      ;*****
9713

```

```

9714 053322 012737 140005 055650 65$:  MOV    #140005,T27PK3      ;WRITE DATA,CVC=1,ACK COMMAND
9715 053330 012704 055650      MOV    #T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
9716 053334 010300      MOV    R3,R0      ;SET PATTERN IN CORRECT REGISTER
9717 053336 004737 020410      JSR    PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
9718 053342 010337 055656      MOV    R3,T27SZ      ;SET UP RECORD SIZE IN PACKET
9719 053346 010465 177776      MOV    R4,TSD8(R5)      ;ISSUE COMMAND
9720 053352 004737 017134      JSR    PC,WAITF      ;WAIT FOR SSR TO SET
9721 053356 016501 000000      MOV    TSSR(R5),R1      ;GET TSSR CONTENTS
9722 053362 012702 000200      MOV    #SSR,R2      ;SET UP EXPECTED
9723 053366 020102      CMP    R1,R2      ;ARE THEY EQUAL
9724 053370 001406      BEQ    80$      ;BR, IF OK
9725 053372 004737 020116      JSR    PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9729      ;SOFT ERROR GENERATED BECAUSE THE
9730      ;WRITE COMMAND IS NOT BEING CHECKED
9731      ;HERE. IT WAS CHECKED IN LEAH2
9732      ERRSOFT ERRNO,WRTErr,PKTSSR      ;TSSR INCORRECT AFTER WRITE DATA
          TRAP    C$ERSOFT
          .WORD    331
          .WORD    WRTErr
          .WORD    PKTSSR
          TRAP    C$CLP1
9733 053406 104406      80$:  CKLOOP      ;LOOP IF SELECTED
9734      ;*****
9735      ;WRITE DATA RETRY,ACK,SWB=1 COMMAND
9736      ;*****
9737
9738
9739
9740
9741 053410 012737 111005 055650      MOV    #111005,T27PK3      ;WRITE DATA RETRY,ACK,SWB=1 COMMAND
9742 053416 010465 177776      MOV    R4,TSD8(R5)      ;ISSUE COMMAND
9743 053422 004737 017134      JSR    PC,WAITF      ;WAIT FOR SSR TO SET
9744 053426 016501 000000      MOV    TSSR(R5),R1      ;GET TSSR CONTENTS
9745 053432 012702 000200      MOV    #SSR,R2      ;SET UP EXPECTED
9746 053436 020102      CMP    R1,R2      ;ARE THEY EQUAL
9747 053440 001406      BEQ    90$      ;BR, IF OK
9748 053442 004737 020116      JSR    PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9752 053446      ERRHRD ERRNO,T27WRF,EXPREC      ;TSSR INCORRECT AFTER WRITE DATA RETRY
          TRAP    C$ERHRD
          .WORD    332
          .WORD    T27WRF
          .WORD    EXPREC
          TRAP    C$CLP1
9753 053456 104406      90$:  CKLOOP      ;LOOP IF SELECTED
9754 053460 005723      TST    (R3),      ;BUMP RECORD SIZE COUNTER
9755 053462 020327 000050      CMP    R3,#40      ;AT 40 SIZE YET
9756 053466 001315      BNE    65$      ;BR, IF MORE RECORDS TO WRITE
9757
9758      ;*****
9759      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9760      ;*****
9761
9762
9763
9764 053470 004737 010444      JSR    PC,REWIND      ;CALL TAPE REWIND COMMAND
9765 053474 103411      BCS    230$      ;BR, IF NO PROBLEM
9766 053476 016501 000000      MOV    TSSR(R5),R1      ;GET TSSR CONTENTS

```

```

9767 053502 010004      MOV      R0,R4      ;GET PACKET ADDRESS
9768 053504 004737 020116 JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9772 053510      ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
          053510 104456      TRAP      C$ERHRD
          053512 000515      .WORD    333
          053514 057055      .WORD    T27RWN
          053516 011710      .WORD    PKTSSR
9773 053520      230$: CKLOOP      ;LOOP IF SELECTED
          053520 104406      TRAP      C$CLP1
9774
9775      ;*****
9776      ;
9777      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9778      ;
9779      ;*****
9780
9781 053522 013701 055546      MOV      T27BFR+6,R1      ;PICK UP XSTO
9782 053526 010102      MOV      R1,R2      ;SET UP EXPECTED
9783 053530 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
9784 053534 020102      CMP      R1,R2      ;DOES EXP = REC'D
9785 053536 001406      BEQ      240$      ;BR, IF EQUAL (OK)
9786 053540 004737 020116 JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9790 053544      ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
          053544 104456      TRAP      C$ERHRD
          053546 000516      .WORD    334
          053550 056551      .WORD    T27BOT
          053552 016360      .WORD    EXPREC
9791 053554      240$: CKLOOP      ;LOOP IF SELECTED
          053554 104406      TRAP      C$CLP1
9792 053556 012703 000024      MOV      #20.,R3      ;STARTING RECORD SIZE
9793 053562 013737 003072 055652 MOV      FREE,T27RB      ;STARTING READ BUFFER ADDRESS
9794
9795      ;*****
9796      ;
9797      ;READ DATA,ACK COMMAND
9798      ;
9799      ;*****
9800
9801 053570 012737 100001 055650 265$: MOV      #100001,T27PK3      ;READ DATA,ACK COMMAND
9802 053576 012704 055650      MOV      #T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
9803 053602 010337 055656      MOV      R3,T27SZ      ;SET UP RECORD SIZE IN PACKET
9804 053606 010465 177776      MOV      R4,T27S2      ;ISSUE COMMAND
9805 053612 004737 017134      JSR      PC,WAITE      ;WAIT FOR SSR TO SET
9806 053616 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
9807 053622 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
9808 053626 020102      CMP      R1,R2      ;ARE THEY EQUAL
9809 053630 001406      BEQ      280$      ;BR, IF OK
9810 053632 004737 020116 JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
9814 053636      ERRHRD  ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
          053636 104456      TRAP      C$ERHRD
          053640 000517      .WORD    335
          053642 005104      .WORD    RDERR
          053644 011710      .WORD    PKTSSR
9815 053646      280$: CKLOOP      ;LOOP IF SELECTED
          053646 104406      TRAP      C$CLP1
9816 053650 013702 003072      MOV      FREE,R2      ;GET BUFFER ADDRESS
9817 053654 010304      MOV      R3,R4      ;GET RECORD SIZE

```


L3

```

9847 ;
9848 : TEST 3, SUBTEST 5
9849 ;
9850 ; VERIFIES THAT A WRITE DATA RETRY COMMAND IS
9851 ; PERFORMING THE "ERASE" PART OF THE OPERATION BY
9852 ; PERFORMING THE FOLLOWING SERIES OF STEPS.
9853 ;
9854 ; 1. THE TAPE IS REWOUND AND A SERIES OF RECORDS ARE
9855 ; WRITTEN WITH THE NORMAL WRITE DATA COMMAND. THIS
9856 ; SHOULD RESULT IN RECORDS SEPARATED BY THE
9857 ; STANDARD INTERRECORD GAP.
9858 ;
9859 ; 2. A PROGRAM TIMING VALUE IS CALIBRATED BY REWINDING
9860 ; THE TAPE AND THEN COUNTING THE NUMBER OF CYCLES
9861 ; THROUGH A PROGRAMMED LOOP REQUIRED TO SPACE OVER
9862 ; THE SERIES OF RECORDS WRITTEN IN PREVIOUS
9863 ; STEP
9864 ;
9865 ; 3. THE TAPE IS AGAIN REWOUND AND THE SAME SERIES OF
9866 ; RECORDS WRITTEN AGAIN, THIS TIME USING THE WRITE
9867 ; DATA RETRY COMMAND. THIS SHOULD RESULT IN
9868 ; RECORDS SEPARATED BY A LONG INTERRECORD GAP.
9869 ;
9870 ; 4. THE TAPE IS AGAIN REWOUND, THE SPACING COMMAND
9871 ; ISSUED, AND THE NUMBER OF TIMING LOOP CYCLES
9872 ; COUNTED TO COMPLETE THE OPERATION.
9873 ;
9874 ; 5. THE TWO LOOP COUNTS ARE COMPARED, CHECKING TO SEE
9875 ; THAT THEY DIFFER BY A SIGNIFICANT AMOUNT.
9876 ;
9877 ;
9878 ;
9879 ;
9880 ;
9881 ; -
9882 053756 BGNSUB ; >>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
      053756 T3.5: TRAP C$BSUB
      053756 104402
9883 053760 005037 002172 CLR INTRECV ; INTERRUPT INDICATOR
9884 053764 005037 055676 CLR T27CNT ; TIMER FOR WRITE DATA SPACING
9885 053770 005037 055700 CLR T27CNJ ; TIMER FOR WRITE DATA RETRY SPACING
9886 053774 004737 060654 JSR PC,T27REST ; SET COMMAND PACKET
9887 054000 004737 060746 JSR PC,T27RT2 ; SET UP OTHER COMMAND PACKET
9888 054004 004737 061010 JSR PC,T27RT3 ; SET UP OTHER COMMAND PACKET
9889 054010 012737 176750 055702 MOV #65000.,T27DLY ; SET UP DELAY COUNTER
9890
9891 ; *****
9892 ;
9893 ; ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
9894 ;
9895 ; *****
9896
9897 054016 004737 016660 10$: JSR PC,SOFINIT ; DO INITIALIZE ON CONTROLLER
9898 054022 103426 BCS 20$ ; BR IF INIT WAS OK
9899 054024 012727 000250 DELAY 250 ; DELAY ABOUT .25 SEC
      054024 012727 000250
      054030 000000 MOV #250,(PC).
                      WORD 0

```

```

054032 013727 002116                                MOV     L$DLY,(PC)+
054036 000000                                .WORD   0
054040 005367 177772                                DEC     -6(PC)
054044 001375                                BNE     .-4
054046 005367 177756                                DEC     -22(PC)
054052 001367                                BNE     .-20
9900 054054 005337 055702                DEC     T27DLY                ;BUMP COUNTER
9901 054060 001356                BNE     10$                ;BR, IF COUNTER NOT DONE
9902 054062 004737 020116                JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
9906 054066 010001                MOV     R0,R1              ;CONTENTS OF TSSR REGISTER
9907 054070                ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
054070 104455                                TRAP    C$ERDF
054072 000521                                .WORD   337
054074 003550                                .WORD   SFIERR
054076 011676                                .WORD   SFIMSG
9908 054100                20$:
9909
9910 054100 012704 055520                MOV     @T27PACKET,R4        ;SUBROUTINE NEEDS PACKET ADDRESS
9911
9912                ;*****
9913                ;
9914                ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTHR)
9915                ;
9916                ;*****
9917
9918 054104 004737 010342                JSR     PC,WRTHR            ;ISSUE WRITE CHARACTERISTICS
9919 054110 103407                BCS     23$                ;BR, IF COMMAND ISSUED OK
9920 054112 004737 020116                JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
9924 054116 010001                MOV     R0,R1              ;SAVE CONTENTS OF TSSR
9925 054120                ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
054120 104456                                TRAP    C$ERHRD
054122 000522                                .WORD   338
054124 004754                                .WORD   WRTMSG
054126 011676                                .WORD   SFIMSG
9926 054130                23$:  CKLOOP                ;LOOP IF SELECTED
054130 104406                                TRAP    C$CLP1
9927
9928                ;*****
9929                ;
9930                ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9931                ;
9932                ;*****
9933
9934 054132 004737 010444                JSR     PC,REWIND          ;CALL TAPE REWIND COMMAND
9935 054136 103411                BCS     30$                ;BR, IF NO PROBLEM
9936 054140 016501 000000                MOV     TSSR(R5),R1        ;GET TSSR CONTENTS
9937 054144 010004                MOV     R0,R4              ;GET PACKET ADDRESS
9938 054146 004737 020116                JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
9942 054152                ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
054152 104456                                TRAP    C$ERHRD
054154 000523                                .WORD   339
054156 057055                                .WORD   T27RWN
054160 011710                                .WORD   PKTSSR
9943 054162                30$:  CKLOOP                ;LOOP IF SELECTED
054162 104406                                TRAP    C$CLP1
9944
9945                ;*****

```

```

9946
9947      ; READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9948      ;
9949      ; *****
9950
9951 054164 013701 055546      MOV      T27BFR+6,R1      ; PICK UP XSTO
9952 054170 010102      MOV      R1,R2      ; SET UP EXPECTED
9953 054172 052702 000002      BIS      @BIT1,R2      ; SET BOT BIT IN EXPECTED
9954 054176 020102      CMP      R1,R2      ; DOES EXP = REC'D
9955 054200 001406      BEQ      40$      ; BR, IF EQUAL (OK)
9956 054202 004737 020116      JSR      PC,FATCHK      ; INC AND CHECK FOR MORE THAN 25 ERRORS
9960 054206      ERRHRD      ERRNO,T27BOT,EXPREC      ; TAPE NOT AT BOT AFTER REWIND
          054206 104456      TRAP      C$ERHRD
          054210 000524      .WORD      340
          054212 056551      .WORD      T27BOT
          054214 016360      .WORD      EXPREC
9961 054216      40$:      CKLOOP      ; LOOP IF SELECTED
          054216 104406      TRAP      C$CLP1
9962 054220 012703 000144      MOV      @100.,R3      ; NUMBER OF RECORDS TO BE WRITTEN
9963 054224 013737 003072 055652      MOV      FREE,T27WB      ; STARTING WRITE BUFFER ADDRESS
9964
9965      ; *****
9966      ;
9967      ; WRITE DATA,ACK,CVC=1 COMMAND
9968      ;
9969      ; *****
9970
9971 054232 012737 140005 055650 65$:      MOV      @140005,T27PK3      ; WRITE DATA,ACK,CVC=1 COMMAND
9972 054240 012704 055650      MOV      @T27PK3,R4      ; SET UP R4 WITH PACKET ADDRESS
9973 054244 012737 000024 055656      MOV      @20.,T27SZ      ; SET UP RECORD SIZE IN PACKET
9974 054252 010465 177776      MOV      R4,TSD8(R5)      ; ISSUE COMMAND
9975 054256 004737 017134      JSR      PC,WAITF      ; WAIT FOR SSR TO SET
9976 054262 016501 000000      MOV      TSSR(R5),R1      ; GET TSSR CONTENTS
9977 054266 012702 000200      MOV      @SSR,R2      ; SET UP EXPECTED
9978 054272 020102      CMP      R1,R2      ; ARE THEY EQUAL
9979 054274 001406      BEQ      70$      ; BR, IF OK
9980 054276 004737 020116      JSR      PC,FATCHK      ; INC AND CHECK FOR MORE THAN 25 ERRORS
9984      ; SOFT ERROR GENERATED BECAUSE THE
9985      ; WRITE COMMAND IS NOT BEING CHECKED
9986      ; HERE. IT WAS CHECKED IN LEAH2
9987 054302      ERRSOFT      ERRNO,WRTErr,PKTSSR      ; TSSR INCORRECT AFTER WRITE DATA
          054302 104457      TRAP      C$ERSOFT
          054304 000525      .WORD      341
          054306 005011      .WORD      WRTErr
          054310 017110      .WORD      PKTSSR
9988 054312      70$:      CKLOOP      ; LOOP IF SELECTED
          054312 104406      TRAP      C$CLP1
9989 054314 005303      DEC      R3      ; DEC RECORD COUNTER
9990 054316 001345      BNE      65$      ; BR, IF MORE RECORDS TO WRITE
9991
9992      ; *****
9993      ;
9994      ; ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9995      ;
9996      ; *****
9997
9998 054320 004737 010444      JSR      PC,REWIND      ; CALL TAPE REWIND CO ND

```

```

9999 054324 103411          BCS      1301          ;BR, IF NO PROBLEM
10000 054326 016501 000000   MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
10001 054332 010004          MOV      R0,R4          ;GET PACKET ADDRESS
10002 054334 004737 020116   JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
10006 054340          ERRMRD  ERRNO,T27RWN,PKTSSR      ;REWIND NOT ACCEPTED
          054340 104456          TRAP      C1ERMRD
          054342 000526          .WORD    342
          054344 057055          .WORD    T27RWN
          054346 011710          .WORD    PKTSSR
10007 054350          1301:  CKLOOP          ;LOOP IF SELECTED
          054350 104406          TRAP      C1CLP1
10008
10009          ;*****
10010          ;
10011          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
10012          ;
10013          ;*****
10014
10015 054352 013701 055546   MOV      T27BFR+6,R1      ;PICK UP XST0
10016 054356 010102          MOV      R1,R2          ;SET UP EXPECTED
10017 054360 052702 000002   BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
10018 054364 020102          CMP      R1,R2          ;DOES EXP = REC'D
10019 054366 001406          BEQ      1401          ;BR, IF EQUAL (OK)
10020 054370 004737 020116   JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
10024 054374          ERRMRD  ERRNO,T27BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
          054374 104456          TRAP      C1ERMRD
          054376 000527          .WORD    343
          054400 056551          .WORD    T27BOT
          054402 016360          .WORD    EXPREC
10025 054404          1401:  CKLOOP          ;LOOP IF SELECTED
          054404 104406          TRAP      C1CLP1
10026 054406 012704 055650   MOV      #T27PK3,R4      ;SET UP PACKET ADDRESS
10027 054412 012737 000010 055652   MOV      #10,T27RB      ;SET UP RECORDS TO SPACE OVER
10028
10029          ;*****
10030          ;
10031          ;ACK,CVC=1,SPACE FORWARD COMMAND
10032          ;
10033          ;*****
10034
10035 054420 012737 140010 055650   MOV      #140010,T27PK3      ;ACK,CVC=1,SPACE FORWARD COMMAND
10036 054426 010465 177776   1501:  MOV      R4,TSD8(R5)      ;ISSUE COMMAND
10037 054432 005237 055676   1521:  INC      T27CNT      ;BUMP TIMER
10038 054436          DELAY      1          ;DELAY ABOUT 100US
          054436 012727 000001          MOV      #1,(PC)+
          054442 000000          .WORD    0
          054444 013727 002116          MOV      L1DLY,(PC)+
          054450 000000          .WORD    0
          054452 005367 177772          DEC      -6(PC)
          054456 001375          BNE      .4
          054460 005367 177756          DEC      -22(PC)
          054464 001367          BNE      .20
10039 054466 016501 000000   MOV      TSSR(R5),R1      ;GET TSSR
10040 054472 032701 000200   BIT      #BIT7,R1      ;CHECK FOR TSSR'S SSR SET
10041 054476 001755          BEQ      1521          ;KEEP COUNTING UNTIL SET
10042 054500 016501 000000   MOV      TSSR(R5),R1      ;GET STATUS FROM TSSR
10043 054504 012702 000200   MOV      #SSR,R2      ;SET UP EXPECTED

```

```

10044 054510 020201      CMP      R2,R1      ;WAS EVERYTHING OK
10045 054512 001406      BEQ      1601      ;BR, IF ALL IS WELL
10046 054514 004737 020116 JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
10050 054520      ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE FORWARD DIDN'T WORK OUT
      054520 104456      TRAP      C1ERHRD
      054522 000530      .WORD    344
      054524 060317      .WORD    T27SCF
      054526 011710      .WORD    PKTSSR
10051 054530      1601:   CKLOOP      ;LOOP IF SELECTED
      054530 104406      TRAP      C1CLP1
10052
10053      ;*****
10054      ;
10055      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
10056      ;
10057      ;*****
10058
10059 054532 004737 010444 JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
10060 054536 004737 017252 JSR      PC,CHKTSSR     ;SEE HOW TSSR IS
10061 054542 103407      BCS      1701      ;BR, IF NO PROBLEM
10062 054544 010001      MOV      R0,R1      ;SAVE TSSR
10063 054546 004737 020116 JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
10067 054552      ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
      054552 104456      TRAP      C1ERHRD
      054554 000531      .WORD    345
      054556 057055      .WORD    T27RWN
      054560 011710      .WORD    PKTSSR
10068 054562      1701:   CKLOOP      ;LOOP IF SELECTED
      054562 104406      TRAP      C1CLP1
10069
10070      ;*****
10071      ;
10072      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
10073      ;
10074      ;*****
10075
10076 054564 013701 055546 MOV      T27BFR+6,R1    ;PICK UP XSTO
10077 054570 010102      MOV      R1,R2      ;SET UP EXPECTED
10078 054572 052702 000002 BIS      @BIT1,R2      ;SET BOT BIT IN EXPECTED
10079 054576 020102      CMP      R1,R2      ;DOES EXP = REC'D
10080 054600 001406      BEQ      1751      ;BR, IF EQUAL (OK)
10081 054602 004737 020116 JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
10085 054606      ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      054606 104456      TRAP      C1ERHRD
      054610 000532      .WORD    346
      054612 056551      .WORD    T27BOT
      054614 016360      .WORD    EXPREC
10086 054616      1751:   CKLOOP      ;LOOP IF SELECTED
      054616 104406      TRAP      C1CLP1
10087 054620 012703 000144 MOV      @100.,R3      ;STARTING RECORD SIZE
10088 054624 013737 003072 055652 1771:  MOV      FREE,T27WB ;STARTING WRITE BUFFER ADDRESS
10089
10090      ;*****
10091      ;
10092      ;WRITE DATA,CVC=1,ACK COMMAND
10093      ;
10094      ;*****

```

```

10095
10096 054632 012737 140005 055650      MOV      #140005,T27PK3      ;WRITE DATA,CVC=1,ACK COMMAND
10097 054640 012704 055650      MOV      #T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
10098 054644 012737 000024 055656      MOV      #20.,T27SZ      ;SET UP RECORD SIZE IN PACKET
10099 054652 010465 177776      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
10100 054656 004737 017134      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
10101 054662 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
10102 054666 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
10103 054672 020102      CMP      R1,R2      ;ARE THEY EQUAL
10104 054674 001406      BEQ      180$      ;BR, IF OK
10105 054676 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
10109
10110
10111
10112 054702      ERRSOF T ERRNO,WRTErr,PKTSSR      ;SOFT ERROR GENERATED BECAUSE THE
                                ;WRITE COMMAND IS NOT BEING CHECKED
                                ;HERE. IT WAS CHECKED IN LEAH2
                                ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C$ERSOFT
                                .WORD      347
                                .WORD      WRTErr
                                .WORD      PKTSSR
10113 054712      180$: CKLOOP      ;LOOP IF SELECTED
                                TRAP      C$CLP1
10114 054714 005303      DEC      R3      ;COUNT NUMBER OF RECORDS
10115 054716 001342      BNE      177$      ;BR, IF MORE RECORDS TO WRITE
10116
10117
10118
10119
10120
10121
10122
10123 054720 004737 010444      JSR      PC,REWIND      ;ISSUE REWIND
10124 054724 103411      BCS      182$      ;BR, IF ALL IS WELL
10125 054726 010004      MOV      R0,R4      ;GET PACKET ADDRESS
10126 054730 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
10127 054734 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
10131 054740      ERRHRD ERRNO,T27RWN,PKTSSR      ;REWIND FAILED
                                TRAP      C$ERHRD
                                .WORD      348
                                .WORD      T27RWN
                                .WORD      PKTSSR
10132 054750      182$: CKLOOP      ;SELECT LOOP MAYBE
                                TRAP      C$CLP1
10133
10134
10135
10136
10137
10138
10139
10140
10141 054752 012703 000001      MOV      #1.,R3      ;SPACE 1 RECORD FORWARD
10142 054756 004737 010144      JSR      PC,SPACE      ;ISSUE SPACE COMMAND
10143 054762 103411      BCS      185$      ;BR, IF COMMAND OK
10144 054764 010004      MOV      R0,R4      ;GET PACKET ADDRESS
10145 054766 016501 000000      MOV      TSSR(R5),R1      ;GET TSSR STATUS
10146 054772 004737 020116      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
10150 054776      ERRHRD ERRNO,T27SCF,PKTSSR      ;SPACE FORWARD COMMAND FAILED

```

```

054776 104456
055000 000535
055002 060317
055004 011710
10151 055006 185: CKLOOP ;LOOP IF SELECTED
055006 104406 TRAP C$ERMRD
10152 055010 012703 000144 MOV #100.,R3 ;NUMBER OF RECORDS TO BE WRITTEN
10153 055014 013737 003072 055652 MOV FREE,T27WB ;STARTING WRITE BUFFER ADDRESS
10154
10155 ;*****
10156 ;
10157 ;WRITE DATA RETRY,ACK COMMAND
10158 ;
10159 ;*****
10160
10161 055022 012737 101005 055650 190: MOV #101005,T27PK3 ;WRITE DATA RETRY,ACK COMMAND
10162 055030 012704 055650 MOV #T27PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
10163 055034 012737 000024 055656 MOV #20.,T27SZ ;SET UP RECORD SIZE IN PACKET
10164 055042 010465 177776 MOV R4,TSD8(R5) ;ISSUE COMMAND
10165 055046 004737 017134 JSR PC,WAITF ;WAIT FOR SSR TO SET
10166 055052 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
10167 055056 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
10168 055062 020102 CMP R1,R2 ;ARE THEY EQUAL
10169 055064 001406 BEQ 200: ;BR, IF OK
10170 055066 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
10174 055072 ERRMRD ERRNO,T27WDC,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
055072 104456 TRAP C$ERMRD
055074 000536 .WORD 350
055076 057411 .WORD T27WDC
055100 011710 .WORD PKTSSR
10175 055102 200: CKLOOP ;LOOP IF SELECTED
055102 104406 TRAP C$CLP1
10176 055104 013737 003072 055652 MOV FREE,T27WB ;STARTING WRITE BUFFER ADDRESS
10177
10178 ;*****
10179 ;
10180 ;WRITE DATA,CVC=1,ACK COMMAND
10181 ;
10182 ;*****
10183
10184 055112 012737 140005 055650 MOV #140005,T27PK3 ;WRITE DATA,CVC=1,ACK COMMAND
10185 055120 012704 055650 MOV #T27PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
10186 055124 012737 000024 055656 MOV #20.,T27SZ ;SET UP RECORD SIZE IN PACKET
10187 055132 010465 177776 MOV R4,TSD8(R5) ;ISSUE COMMAND
10188 055136 004737 017134 JSR PC,WAITF ;WAIT FOR SSR TO SET
10189 055142 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
10190 055146 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
10191 055152 020102 CMP R1,R2 ;ARE THEY EQUAL
10192 055154 001406 BEQ 210: ;BR, IF OK
10193 055156 004737 020116 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
10197 ;SOFT ERROR GENERATED BECAUSE THE
10198 ;WRITE COMMAND IS NOT BEING CHECKED
10199 ;HERE. IT WAS CHECKED IN LEAM2
10200 055162 ERRSOF ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
055162 104457 TRAP C$ERSOFT
055164 000537 .WORD 351
055166 005011 .WORD WRERR

```

Address	Op Code	Register	Value	Comment	Trap	Word	PKTSSR
10201	055170	011710		210\$: CKLOOP			PKTSSR
	055172	104406					
10202	055174	005303		DEC R3			C\$CLP1
10203	055176	001311		BNE 190\$			
10204							
10205							
10206							
10207							
10208							
10209							
10210							
10211	055200	004737	010444	JSR PC,REWIND			
10212	055204	103411		BCS 230\$			
10213	055206	016501	000000	MOV TSSR(R5),R1			
10214	055212	010004		MOV R0,R4			
10215	055214	004737	020116	JSR PC,FATCHK			
10219	055220			ERRHRD ERRNO,T27RWN,PKTSSR			
	055220	104456					
	055222	000540					
	055224	057055					
	055226	011710					
10220	055230			230\$: CKLOOP			
	055230	104406					
10221							
10222							
10223							
10224							
10225							
10226							
10227							
10228	055232	013701	055546	MOV T27BFR+6,R1			
10229	055236	010102		MOV R1,R2			
10230	055240	052702	000002	BIS #BIT1,R2			
10231	055244	020102		CMR R1,R2			
10232	055246	001406		BEQ 240\$			
10233	055250	004737	020116	JSR PC,FATCHK			
10237	055254			ERRHRD ERRNO,T27BOT,EXPREC			
	055254	104456					
	055256	000541					
	055260	056551					
	055262	016360					
10238	055264			240\$: CKLOOP			
	055264	104406					
10239	055266	012704	055650	MOV #T27PK3,R4			
10240	055272	012737	000010	MOV #10,T27RB			
10241							
10242							
10243							
10244							
10245							
10246							
10247							
10248	055300	012737	140010	MOV #140010,T27PK3			
10249	055306	010465	177776	MOV R4,TSDR(R5)			
10250	055312	005237	055700	INC T27CNU			
10251	055316			DELAY 1			

[illegible]

10290			;		
10291			;LOCAL STORAGE FOR THIS TEST		
10292			;		
10294	055512		.BLKB 10-<.-TUV2A&7>		
10296	055520		T27PACKET:		;COMMAND PACKET FOR TEST
10297	055520	100004	.WORD 100004		;WRITE CHARACTERISTICS COMMAND, WITH . ACK
10298	055522	055530	.WORD T27DATA		;ADDRESS OF CHARACTERISTICS BLOCK
10299	055524	000000	.WORD 0		
10300	055526	000012	.WORD 10.		;STARTING VALUE OF BLOCK SIZE
10301	055530		T27DATA:		;CHARACTERISTICS DATA BLOCK
10302	055530	055540	.WORD T27BFR		;ADDRESS OF MESSAGE BUFFER
10303	055532	000000	.WORD 0		
10304	055534	000024	.WORD 20.		;LENGTH OF MESSAGE BUFFER
10305	055536	000000	.WORD 0		
10306	055540		T27BFR: .BLKW 25.		;MESSAGE BUFFER
10307			;		
10308			;WRITE SUBSYSTEM MEMORY COMMAND PACKET		
10309			;		
10311	055622		.BLKB 10-<.-TUV2A&7>		
10313	055630		T27PK2:		
10314	055630	100006	.WORD 100006		;WRITE SUB SYS MEM COMMAND, AND ACK
10315	055632	055660	.WORD T27BF2		;ADDRESS OF SELECT BLOCK DATA
10316	055634	000000	.WORD 0		
10317	055636	000006	.WORD 6.		;SIZE OF DATA PACKET
10318					
10320	055640		.BLKB 10-<.-TUV2A&7>		
10322	055650		T27PK3:		
10323	055650	100005	.WORD 100005		;REREAD COMMAND, AND ACK
10324	055652		T27RB:		
10325	055652	003072	T27WB: .WORD FREE		;ADDRESS OF WRITE BUFFER
10326	055654	000000	.WORD 0		
10327	055656	000000	T27SZ: .WORD 0		;SIZE OF BUFFER (EXTENT)
10328			.EVEN		
10329	055660		T27BF2:		
10330	055660	010	T27BS0: .BYTE 10		;BSELO AREA
10331	055661	200	T27BS1: .BYTE 200		;BSEL1 AREA
10332	055662	000000	T27S2: .WORD 0		;SEL 2 AREA
10333	055664	000000	T27S3: .WORD 0		;DATA AREA
10334			.EVEN		
10335			;TAPE MOTION PACKET COMMAND VALUES		
10336					
10337	055666	100205	T27RN: .WORD 100205		;REREAD DATA (NEXT)
10338	055670	100605	T27WR: .WORD 100605		;REREAD DATA RETRY
10339	055672	102205	T27CON: .WORD 102205		;WRITE CONTINOUS
10340	055674	177777	.WORD 177777		;END OF DATA
10341	055676	000000	T27CNT: .WORD 0		;TAPE TIMER COUNTER STORAGE AREA
10342	055700	000000	T27CNU: .WORD 0		;TAPE TIMER COUNTER STORAGE AREA
10343	055702	000000	T27DLY: .WORD 0		;DELAY COUNTER

```

10345
10346
10347      ;*
10348      ;LOCAL TEXT MESSAGES FOR TEST
10349      ;*
10350
10351 055704      124      141      160  T27WNG: .ASCIZ 'Tape Position Incorrect After REREAD Previous (OPP=1)'
10352 055772      124      123      123  T27RDF: .ASCIZ 'TSSR Incorrect After READ DATA Command'
10353 056041      122      105      122  T27RRF: .ASCIZ 'REREAD Previous (Space Reverse, Read Forward) Command Failed'
10354 056136      120      117      123  T27SC: .ASCIZ 'POSITION (Space Command) Failed, TSSR Not Correct'
10355 056220      122      111      102  T27LOR: .ASCIZ 'RIB NOT SET AFTER READ REVERSE INTO BOT'
10356 056270      124      123      123  T27WDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
10357 056345      111      154      154  T27LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XST0'
10358 056426      122      105      122  T27SSR: .ASCIZ 'REREAD COMMAND Not Accepted'
10359 056462      124      123      123  T27WDE: .ASCIZ 'TSSR Not Correct After WRITE DATA RETRY Command, At BOT'
10360 056551      124      141      160  T27BOT: .ASCIZ 'Tape Not At BOT After REWIND Command (BOT Not Set In XST0)'
10361 056644      127      122      111  T27TIM: .ASCIZ 'WRITE DATA RETRY'S Erase Tape Not Long Enough'
10362 056721      122      105      122  T27EOT: .ASCIZ 'REREAD DATA OVER EOT GAVE NO TAPE STATUS ALERT'
10363 057000      124      123      123  T27TM: .ASCIZ 'TSSR Not Correct After REREAD COMMAND Reject'
10364 057055      122      145      167  T27RWN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
10365 057124      122      101      115  T27RNC: .ASCIZ 'RAM Error, Correct Data Pattern Not In Ram'
10366 057177      124      123      123  T27AM3: .ASCIZ 'TSSR Init. Failed After REREAD COMMAND'
10367 057246      104      162      151  T27OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
10368 057321      124      123      123  T27WDD: .ASCIZ 'TSSR Not Correct After REREAD DATA Command, SWB Bit Set'
10369 057411      124      123      123  T27WDC: .ASCIZ 'TSSR Not Correct After REREAD DATA Command'
10370 057464      103      126      103  T27VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
10371 057537      124      123      102  T27BA: .ASCIZ 'TSBA Not Correct After REREAD DATA Command'
10372 057612      127      122      111  T27WSS: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
10373 057701      122      145      141  T27LON: .ASCIZ 'Reading Long Record Failed To Set RLL Bit In XST0'
10374 057763      122      145      141  T27LOP: .ASCIZ 'Reading Long Record Failed To Set RLS Bit In XST0'
10375 060045      122      145      163  T27PBP: .ASCIZ 'Residual Byte Count Incorrect After Short Record Read'
10376 060133      122      145      141  T27TRL: .ASCIZ 'Reading Long Record Failed To Give Tape Status Alert'
10377 060221      127      122      111  T27NEF: .ASCIZ 'WRITE DATA RETRY, At First Record, Failed To Set RIB Bit XST3'
10378 060317      124      123      123  T27SCF: .ASCIZ 'TSSR Not Correct After SPACE RECORDS Command'
10379 060374      124      123      123  T27TSA: .ASCIZ 'TSSR Not Correct After WRITE DATA RETRY, Into BOT'
10380 060456      124      123      123  T27WRF: .ASCIZ 'TSSR Not Correct After WRITE DATA RETRY Command'
10381 060536      104      141      164  T27DTA: .ASCIZ 'Data Compare Error, Data Read From Tape Not Equal To Written'
10382 060633      127      162      151  T27ID: .ASCIZ 'Write Data Retry'
10383      .EVEN
10384
10385      ;*
10386      ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
10387      ;WRITE SUBSYSTEM MEMORY COMMAND
10388      ;
10389      ;*
10390
10391 060654      012701      055520      T27REST:
10392 060654      012721      100004      SAVREG
10393 060660      012721      055530      MOV      #T27PACKET,R1
10394 060664      012721      055530      MOV      #100004,(R1).
10395 060670      005021      000012      MOV      #T27DATA,(R1).
10396 060674      012721      055540      CLR      (R1).
10397 060676      012721      000012      MOV      #10.,(R1).
10398 060702      012721      055540      MOV      #T27BFR,(R1).
10399 060706      005021      000024      CLR      (R1).
10400 060710      012721      000024      MOV      #20.,(R1).
10401 060714      005021      000024      CLR      (R1).

;SAVE THE REGISTERS
;START OF THE PACKET
;WRITE SUBSYSTEM MEM. WITH ACK,
;ADDRESS OF CHARAISTICS DATA BLOCK
;EXTENDED ADDRESS
;SIZE OF DATA BLOCK IN BYTES
;ADDRESS OF MESSAGE BUFFER
;LENGTH OF MESSAGE BUFFER

```

10402	060716	012711	000000		MOV	#0,(R1)	;SELECT DRIVE ZERO
10403	060722	012702	000030		MOV	#24.,R2	;NUMBER OF LOCATIONS TO BE CLEARED
10404	060726	012762	177777	055540 64#:	MOV	#177777,T27BFR(R2)	;ALL ONES TO MESSAGE BUFFER
10405	060734	005742			TST	-(R2)	;NEXT LOCATION
10406	060736	022702	000000		CMP	#0,R2	;AT END OF LOOP YET
10407	060742	001371			BNE	64#	;KEEP GOING UNTIL DONE
10408	060744	000207			RTS	PC	;RETURN
10409							
10410							
10411	060746			T27RT2:	SAVREG		;SAVE THE REGISTERS
10412	060746				MOV	#T27PK2,R1	;START OF THE PACKET
10413	060752	012701	055630		MOV	#100006,(R1)+	;WRITE SUBSYSTEM MEM. WITH ACK.
10414	060756	012721	100006		MOV	#T27BF2,(R1)+	;ADDRESS OF DATA BLOCK
10415	060762	012721	055660		CLR	(R1)+	;EXTENDED ADDRESS
10416	060766	005021			MOV	#6.,(R1)+	;SIZE OF DATA BLOCK IN BYTES
10417	060770	012721	000006		CLR	(R1)+	
10418	060774	005021			MOV	#T27BF2,R1	;POINT TO DATA SEL AREA
10419	060776	012701	055660		CLR	(R1)+	
10420	061002	005021			CLR	(R1)	
10421	061004	005011			RTS	PC	;RETURN
10422	061006	000207					
10423	061010			T27RT3:	SAVREG		;SAVE REGISTERS
10424	061010				MOV	#T27PK3,R1	;SET UP POINTER ADDRESS
10425	061014	012701	055650		CLR	(R1)+	;COMMAND SPACE
10426	061020	005021			CLR	(R1)+	;ADDRESS OF DATA BLOCK
10427	061022	005021			CLR	(R1)+	;EXTENDED ADDRESS
10428	061024	005021			CLR	(R1)	;SIZE OF DATA TRANSFER BLOCK
10429	061026	005011			RTS	PC	;RETURN
10430	061030	000207			ENDTST		
10431	061032						
	061032						
	061032	104401					

L10066: TRAP C\$ETST

```

10433          .SBTTL TEST 4: WRITE/READ TAPE MARK
10434          ;*
10435          ;
10436          ;THIS TEST VERIFIES THAT THE WRITE TAPE MARK COMMAND OPERATES
10437          ;PROPERLY. IT IS VERIFIED THAT THE TAPE MARK IS WRITTEN ONTO TAPE
10438          ;BY CHECKING THAT THE READ AND SPACE RECORDS COMMANDS DETECT THE
10439          ;TAPE MARK. IN ADDITION, SINCE WRITE TAPE MARK IS THE FIRST
10440          ;SUBCOMMAND UNDER THE FORMAT COMMAND BEING TESTED, IT IS VERIFIED
10441          ;THAT THE CLEAR VOLUME CHECK (CVC) BIT OPERATES PROPERLY AND THAT
10442          ;FORMAT COMMANDS WITH ILLEGAL MODE CODES ARE REJECTED.
10443          ;
10444          ;
10445          ;THE TEST CONSISTS OF THE FOLLOWING 3 SUBTESTS
10446          ;
10447          ;
10448          ;
10449          ;-
10450          BGNTST
10451          061034
10452          061034
10453          061034 005037 002170          CLR      FATFLG          ;CLEAR FATAL ERROR FLAG
10454          061040 005037 003100          CLR      KTFLG          ;HOLD OFF KT11
10455          061044 012737 005771 002146  MOV      @EPRT2,EPRTSW    ;SECONDARY ERROR MESSAGE
10456          061052 005037 003102          CLR      KTENABLE        ;TURN KT11 OFF
10457          061056 004737 020262          JSR      PC,KTOFF          ;TURN KT11 BACK OFF IF THERE
10458          061062 012700 065251          MOV      @TST28ID,R0        ;ASCII MESSAGE TO IDENTIFY TEST
10459          061066 004737 017424          JSR      PC,TSTSETUP      ;DO INITIAL TEST SETUP
10460          061072 012737 000001 002164  MOV      @1,LOOPCNT      ;PERFORM 1 ITERATIONS
10461          061100
10462          T28LOOP:

```

```

10465      ;*
10466      ;
10467      ;TEST 4, SUBTEST 1
10468      ;
10469      ;VERIFIES THAT WRITE TAPE MARK COMMANDS OPERATE
10470      ;PROPERLY, AND THAT READ COMMANDS SUBSEQUENTLY ISSUED
10471      ;TO DETECT THE WRITTEN TAPE MARKS TERMINATE WITH TAPE
10472      ;STATUS ALERT WITH THE TAPE MARK DETECTED (TMK) STATUS
10473      ;BIT SET. THE FOLLOWING SEQUENCE IS EXECUTED.
10474      ;
10475      ;1.   THE CONTROLLER IS INITIALIZED AND TAPE REWOUND.
10476      ;      THIS SETS THE VOLUME CHECK (VCK) STATUS BIT.
10477      ;
10478      ;2.   A WRITE TAPE MARK COMMAND WITH CVC=1 IS ISSUED
10479      ;      AND PROPER TERMINATION AND STATUS IS VERIFIED
10480      ;      (I.E. VCK=0 AND TMK=1).
10481      ;
10482      ;3.   SEVERAL MORE WRITE TAPE MARK COMMANDS, THESE WITH
10483      ;      CVC=0 ARE ISSUED AND PROPER TERMINATION (NORMAL)
10484      ;      AND STATUS (TMK) VERIFIED.
10485      ;
10486      ;4.   A READ REVERSE COMMAND IS ISSUED AND PROPER
10487      ;      TERMINATION (TAPE STATUS ALERT) AND STATUS (TMK)
10488      ;      VERIFIED. IT IS ALSO VERIFIED THAT NO DATA IS
10489      ;      TRANSFERRED INTO MEMORY.
10490      ;
10491      ;5.   A SPACE RECORDS REVERSE COMMAND IS ISSUED AND
10492      ;      PROPER TERMINATION (TAPE STATUS ALERT) AND STATUS
10493      ;      (TMK) VERIFIED.
10494      ;
10495      ;6.   THE TAPE IS REWOUND AND A READ FORWARD COMMAND IS
10496      ;      ISSUED AND PROPER TERMINATION (TAPE STATUS ALERT)
10497      ;      AND STATUS (TMK) VERIFIED. IT IS ALSO VERIFIED
10498      ;      THAT NO DATA IS TRANSFERRED INTO MEMORY.
10499      ;
10500      ;7.   A SPACE RECORDS REVERSE COMMAND THAT CONTAINS A
10501      ;      RECORD COUNT GREATER THAN 1 IS ISSUED AND IT IS
10502      ;      VERIFIED THAT TAPE STATUS ALERT TERMINATION
10503      ;      OCCURED, TMK=1 AND THAT RBPCR (RESIDUAL
10504      ;      BYTE/RECORD COUNTER) CONTAINS THE PROPER NONZERO
10505      ;      VALUE. THIS OPERATION VERIFIES THAT DETECTION OF
10506      ;      THE TAPE MARK CAUSES THE SPACE RECORDS OPERATION
10507      ;      TO BE PREMATURELY TERMINATED. THIS SHOULD LEAVE
10508      ;      THE POSITION JUST BEFORE THE FIRST RECORD ON
10509      ;      TAPE.
10510      ;
10511      ;8.   TAPE POSITION IS VERIFIED BY ISSUING ANOTHER
10512      ;      SPACE RECORDS REVERSE COMMAND AND VERIFYING THAT
10513      ;      TAPE STATUS ALERT TERMINATION OCCURS, WITH THE
10514      ;      REVERSE INTO BOT (RIB) STATUS ERROR BIT SET.
10515      ;
10516      ;9.   A SPACE RECORDS FORWARD COMMAND THAT CONTAINS A
10517      ;      RECORD COUNT GREATER THAN 1 IS ISSUED AND IT IS
10518      ;      VERIFIED THAT TAPE STATUS ALERT TERMINATION
10519      ;      OCCURED, TMK=1, AND THAT RBPCR (RESIDUAL
10520      ;      BYTE/RECORD COUNTER) CONTAINS THE PROPER NONZERO
10521      ;      VALUE. THIS OPERATION VERIFIES THAT DETECTION OF

```

10522		:				THE TAPE MARK CAUSES THE SPACE RECORDS OPERATION TO BE PREMATURELY TERMINATED.
10523		:				
10524		:				
10525		:				
10526		:				
10527		:				
10528		:				
10529		:				
10530		:				
10531	061100	:	BGNSUB		; >>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>	
	061100				T4.1:	
	061100				TRAP	C\$BSUB
10532	061102	004737	065276		JSR PC,T28REST	;SET COMMAND PACKET
10533	061106	004737	065370		JSR PC,T28RT2	;SET UP OTHER COMMAND PACKET
10534	061112	004737	065432		JSR PC,T28RT3	;SET UP OTHER COMMAND PACKET
10535	061116	012737	023420	063402	MOV #10000.,T28DLY	;SET UP DELAY ROUTINE
10536	061124	004737	016660	10\$:	JSR PC,SOFINIT	;DO INITIALIZE ON CONTROLLER
10537	061130	103426			BCS 20\$	;BR IF INIT WAS OK
10538	061132				DELAY 250	;DELAY ABOUT .25 SECONDS
	061132	012727	000250			MOV #250,(PC)+
	061136	000000				.WORD 0
	061140	013727	002116			MOV L\$DLY,(PC)+
	061144	000000				.WORD 0
	061146	005367	177772			DEC -6(PC)
	061152	001375				BNE -.4
	061154	005367	177756			DEC -22(PC)
	061160	001367				BNE -.20
10539	061162	005337	063402		DEC T28DLY	;BUMP DELAY ROUTINE DOWN
10540	061166	001356			BNE 10\$	;BR, IF MORE DELAY TIME LEFT
10541	061170	004737	020116		JSR PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS
10545	061174	010001			MOV R0,R1	;CONTENTS OF TSSR REGISTER
10546	061176				ERRDF ERRNO,SFIERR,SFIMSG	;FATAL ERROR TSSR WAS NOT OK
	061176	104455				TRAP C\$ERDF
	061200	000621				.WORD 401
	061202	003550				.WORD SFIERR
	061204	011676				.WORD SFIMSG
10547	061206			20\$:		
10548	061206	012704	063210		MOV #T28PACKET,R4	;SUBROUTINE NEEDS PACKET ADDRESS
10549	061212	004737	010342		JSR PC,WRTCHR	;ISSUE WRITE CHARACTERISTICS
10550	061216	103407			BCS 23\$	;BR, IF COMMAND ISSUED OK
10551	061220	004737	020116		JSR PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS
10555	061224	010001			MOV R0,R1	;SAVE CONTENTS OF TSSR
10556	061226				ERRHRD ERRNO,WRTMSG,SFIMSG	;WRITE CHARACTERISTICSC FAILED
	061226	104456				TRAP C\$ERHRD
	061230	000622				.WORD 402
	061232	004754				.WORD WRTMSG
	061234	011676				.WORD SFIMSG
10557	061236			23\$:	CKLOOP	;LOOP IF SELECTED
	061236	104406				TRAP C\$CLP1
10558	061240	004737	010444		JSR PC,REWIND	;CALL TAPE REWIND COMMAND
10559	061244	103411			BCS 30\$	;BR, IF NO PROBLEM
10560	061246	016501	000000		MOV TSSR(R5),R1	;GET TSSR
10561	061252	010004			MOV R0,R4	;SAVE PACKET ADDRESS
10562	061254	004737	020116		JSR PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS
10566	061260				ERRHRD ERRNO,T28RWN,PKTSSR	;REWIND NOT ACCEPTED
	061260	104456				TRAP C\$ERHRD
	061262	000623				.WORD 403

Address	Offset	Label	Instruction	Comment	Register	Value	Register	Value	Register	Value
061264	064331									
061266	011710									
10567	061270	30\$:	CKLOOP	;LOOP IF SELECTED						
061270	104406									
10568	061272	013701	063236	MOV	T28BFR+6,R1					
10569	061276	010102		MOV	R1,R2					
10570	061300	052702	000002	BIS	#BIT1,R2					
10571	061304	020102		CMP	R1,R2					
10572	061306	001406		BEQ	40\$					
10573	061310	004737	020116	JSR	PC,FATCHK					
10577	061314			ERRHRD	ERRNO,T28BOT,EXPREC					
061314	104456									
061316	000624									
061320	064207									
061322	016360									
10578	061324	40\$:	CKLOOP	;LOOP IF SELECTED						
061324	104406									
10579	061326	012704	063210	MOV	#T28PACKET,R4					
10580	061332	004737	010342	JSR	PC,WRTCHR					
10581	061336	103407		BCS	68\$					
10582	061340	004737	020116	JSR	PC,FATCHK					
10586	061344	010001		MOV	R0,R1					
10587	061346			ERRHRD	ERRNO,WRTMSG,SFMSG					
061346	104456									
061350	000625									
061352	004754									
061354	011676									
10588	061356	68\$:	CKLOOP	;LOOP IF SELECTED						
061356	104406									
10589	061360	012737	140011	MOV	#140011,T28PK3					
10590	061366	012704	063340	MOV	#T28PK3,R4					
10591	061372	010465	177776	MOV	R4,TSD8(R5)					
10592	061376	004737	017134	JSR	PC,WAITF					
10593	061402	016501	000000	MOV	TSSR(R5),R1					
10594	061406	012702	000200	MOV	#SSR,R2					
10595	061412	020102		CMP	R1,R2					
10596	061414	001406		BEQ	70\$					
10597	061416	004737	020116	JSR	PC,FATCHK					
10601	061422			ERRHRD	ERRNO,T28WDC,PKTSSR					
061422	104456									
061424	000626									
061426	064453									
061430	011710									
10602	061432	70\$:	CKLOOP	;LOOP IF SELECTED						
061432	104406									
10603	061434	013701	063236	MOV	T28BFR+6,R1					
10604	0									

Address	Offset	Hex	Label	Instruction	Comment	Trap	Mark
061466		104406					
10614	061470	013701	063236	MOV	T28BFR+6,R1	PICK UP XSTO (CHECK TMK)	C:CLP1
10615	061474	010102		MOV	R1,R2	SET UP EXPECTED	
10616	061476	052702	100000	BIS	#BIT15,R2	TMK SHOULD BE SET	
10617	061502	020102		CMP	R1,R2	WAS TMK SET	
10618	061504	001406		BEQ	901	BR, IF TMK WAS SET	
10619	061506	004737	020116	JSR	PC,FATCHK	INC AND CHECK FOR MORE THAN 25 ERRORS	
10623	061512			ERRHRD	ERRNO,T28TMK,EXPREC	TMK WAS NOT SET AFTER WRT TAPE	MARK
	061512	104456				TRAP	C:ERHRD
	061514	000630				.WORD	408
	061516	064605				.WORD	T28TMK
	061520	016360				.WORD	EXPREC
10624	061522			901:	CKLOOP	LOOP IF SELECTED	
	061522	104406				TRAP	C:CLP1
10625	061524	004737	010444	JSR	PC,REWIND	CALL TAPE REWIND COMMAND	
10626	061530	103411		BCS	1301	BR, IF NO PROBLEM	
10627	061532	010004		MOV	R0,R4	SAVE PACKET ADDRESS	
10628	061534	016501	000000	MOV	TSSR(R5),R1	GET TSSR STATUS	
10629	061540	004737	020116	JSR	PC,FATCHK	INC AND CHECK FOR MORE THAN 25 ERRORS	
10633	061544			ERRHRD	ERRNO,T28RWN,PKTSSR	REWIND NOT ACCEPTED	
	061544	104456				TRAP	C:ERHRD
	061546	000631				.WORD	409
	061550	064331				.WORD	T28RWN
	061552	011710				.WORD	PKTSSR
10634	061554			1301:	CKLOOP	LOOP IF SELECTED	
	061554	104406				TRAP	C:CLP1
10635	061556	013701	063236	MOV	T28BFR+6,R1	PICK UP XSTO	
10636	061562	010102		MOV	R1,R2	SET UP EXPECTED	
10637	061564	052702	000002	BIS	#BIT1,R2	SET BOT BIT IN EXPECTED	
10638	061570	020102		CMP	R1,R2	DOES EXP = REC'D	
10639	061572	001406		BEQ	1401	BR, IF EQUAL (OK)	
10640	061574	004737	020116	JSR	PC,FATCHK	INC AND CHECK FOR MORE THAN 25 ERRORS	
10644	061600			ERRHRD	ERRNO,T28BOT,EXPREC	TAPE NOT AT BOT AFTER REWIND	
	061600	104456				TRAP	C:ERHRD
	061602	000632				.WORD	410
	061604	064207				.WORD	T28BOT
	061606	016360				.WORD	EXPREC
10645	061610			1401:	CKLOOP	LOOP IF SELECTED	
	061610	104406				TRAP	C:CLP1
10646	061612	012703	000012	MOV	#10.,R3	NUMBER OF RECORDS TO WRITE TM	
10647	061616	012737	140011	MOV	#140011,T28PK3	WRITE TAPE MARK,ACK,CVC+1 COMMAND	
10648	061624	012704	063340	MOV	#T28PK3,R4	SET UP R4 WITH PACKET ADDRESS	
10649	061630	010465		MOV	R4,TSD8(R5)	ISSUE COMMAND	
10650	061634	004737	017134	JSR	PC,WAITF	WAIT FOR SSR TO SET	
10651	061640	016501	000000	MOV	TSSR(R5),R1	PICK UP TSSR	
10652	061644	012702	000200	MOV	#SSR,R2	SET UP EXPECTED (SSR ONLY)	
10653	061650	020102		CMP	R1,R2	WAS STATUS GOOD	
10654	061652	001406		BEQ	1651	BR, IF TERMINATION WAS GOOD	
10655	061654	004737	020116	JSR	PC,FATCHK	INC AND CHECK FOR MORE THAN 25 ERRORS	
10659	061660			ERRHRD	ERRNO,T28WDC,PKTSSR	TSSR NOT CORRECT AFTER WRT TAPE M.	
	061660	10					

10662	061676	010102		MOV	R1,R2		;SET UP EXPECTED
10663	061700	052702	100000	BIS	#BIT15,R2		;SET TMK BIT IN EXPECTED
10664	061704	020102		CMP	R1,R2		;DOES EXP = REC'D
10665	061706	001406		BEQ	180:		;BR, IF EQUAL (OK)
10666	061710	004737	020116	JSR	PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS
10670	061714			ERRHRD	ERRNO,T28TMK,EXPREC		;TMK NOT SET AFTER WRT TAPE MARK
	061714	104456					TRAP C:ERHRD
	061716	000634					.WORD 412
	061720	064605					.WORD T28TMK
	061722	016360					.WORD EXPREC
10671	061724			180:	CKLOOP		;LOOP IF SELECTED
	061724	104406					TRAP C:CLP1
10672	061726	005303		DEC	R3		;BUMP COUNTER DOWN
10673	061730	001337		BNE	155:		;BR, IF LESS THAN 10 TAPE MARKS
10674	061732	012700	177777	MOV	#177777,R0		;VALUE TO WRITTEN TO MEMORY
10675	061736	004737	020410	JSR	PC,FILLMEM		;FILL MEM WITH ALL ONES
10676	061742	013737	003072	MOV	FREE,T28WB	063342	;STARTING READ BUFFER ADDRESS
10677	061750	012737	140401	MOV	#140401,T28PK3	063340	;READ REVERSE,ACK, COMMAND
10678	061756	012704	063340	MOV	#T28PK3,R4		;SET UP R4 WITH PACKET ADDRESS
10679	061762	013737	000024	MOV	20.,T28SZ	063346	;SET UP RECORD SIZE IN PACKET
10680	061770	010465	177776	MOV	R4,TSD8(R5)		;ISSUE COMMAND
10681	061774	004737	017134	JSR	PC,WAITF		;WAIT FOR SSR TO SET
10682	062000	016501	000000	MOV	TSSR(R5),R1		;GET TSSR CONTENTS
10683	062004	012702	100204	MOV	#SSR!SC!BIT2,R2		;SET UP EXPECTED
10684	062010	020102		CMP	R1,R2		;ARE THEY EQUAL
10685	062012	001406		BEQ	200:		;BR, IF OK
10686	062014	004737	020116	JSR	PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS
10690	062020			ERRHRD	ERRNO,T28RDF,PKTSSR		;TSSR INCORRECT AFTER WRITE DATA
	062020	104456					TRAP C:ERHRD
	062022	000635					.WORD 413
	062024	063544					.WORD T28RDF
	062026	011710					.WORD PKTSSR
10691	062030			200:	CKLOOP		;LOOP IF SELECTED
	062030	104406					TRAP C:CLP1
10692	062032	013701	063236	MOV	T28FR+6,R1		;PICK UP XSTO
10693	062036	010102		MOV	R1,R2		;SET UP EXPECTED
10694	062040	052702	100000	BIS	#BIT15,R2		;TMK SHOULD BE SET
10695	062044	020102		CMP	R1,R2		;IS TMK SET
10696	062046	001406		BEQ	210:		;BR, IF TMK WAS SET (GOOD)
10697	062050	004737	020116	JSR	PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS
10701	062054			ERRHRD	ERRNO,T28RRM,EXPREC		;TMK NOT SET AFTER READ REV
	062054	104456					TRAP C:ERHRD
	062056	000636					.WORD 414
	062060	064657					.WORD T28RRM
	062062	016360					.WORD EXPREC
10702	062064			210:	CKLOOP		;LOOP IF SELECTED
	062064	104406					TRAP C:CLP1
10703	062066	017701	121000	MOV	#FREE,R1		;FIRST LOC IN READ BUFFER
10704	062072	012702	177777	MOV	#177777,R2		;EXPECTED IF NO DATA TRANS.
10705	062076	020102		CMP	R1,R2		;DID ANY DATA GET TRANSFERRED
10706	062100	001406		BEQ	220:		;BR, IF NO DATA TRANS (GOOD)
10707	062102	004737	020116	JSR	PC,FATCHK		;INC AND CHECK FOR MORE THAN 25 ERRORS
10711	062106			ERRHRD	ERRNO,T28DTR,EXPREC		;DATA TRANSFERRED ON READ TAPE MARK
	062106	104456					TRAP C:ERHRD
	062110	000637					.WORD 415
	062112	065072					.WORD T28DTR
	062114	016360					.WORD EXPREC

10712	062116			220:	CKLOOP		;LOOP IF SELECTED	
	062116	104406						TRAP C\$CLP1
10713	062120	012737	100410	063340	MOV	#100410,T28PK3	;SPACE REVERSE,ACK, COMMAND	
10714	062126	012737	000001	063342	MOV	#1,T28RB	;NUMBER OF RECORDS TO SPACE BACK	
10715	062134	012704	063340		MOV	#T28PK3,R4	;SET UP R4 WITH PACKET ADDRESS	
10716	062140	010465	177776		MOV	R4,TSD8(R5)	;ISSUE COMMAND	
10717	062144	004737	017134		JSR	PC,WAITF	;WAIT FOR SSR TO SET	
10718	062150	016501	000000		MOV	TSSR(R5),R1	;GET TSSR CONTENTS	
10719	062154	012702	100204		MOV	#SSR!SC!BIT2,R2	;SET UP EXPECTED	
10720	062160	020102			CMP	R1,R2	;ARE THEY EQUAL	
10721	062162	001406			BEQ	222:	;BR, IF OK	
10722	062164	004737	020116		JSR	PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS	
10726	062170				ERRHRD	ERRNO,T28RDG,PKTSSR	;TSSR INCORRECT AFTER SPACE CMD.	
	062170	104456						TRAP C\$ERHRD
	062172	000640						.WORD 416
	062174	063625						.WORD T28RDG
	062176	011710						.WORD PKTSSR
10727	062200			222:	CKLOOP		;LOOP IF SELECTED	
	062200	104406						TRAP C\$CLP1
10728	062202	013701	063236		MOV	T288FR+6,R1	;PICK UP XST0	
10729	062206	010102			MOV	R1,R2	;SET UP EXPECTED	
10730	062210	052702	100000		BIS	#BIT15,R2	;TMK SHOULD BE SET	
10731	062214	020102			CMP	R1,R2	;IS TMK SET	
10732	062216	001406			BEQ	226:	;BR, IF TMK WAS SET (GOOD)	
10733	062220	004737	020116		JSR	PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS	
10737	062224				ERRHRD	ERRNO,T28RRN,EXPREC	;TMK NOT SET AFTER SPACE REV	
	062224	104456						TRAP C\$ERHRD
	062226	000641						.WORD 417
	062230	064735						.WORD T28RRN
	062232	016360						.WORD EXPREC
10738	062234			226:	CKLOOP		;LOOP IF SELECTED	
	062234	104406						TRAP C\$CLP1
10739	062236	004737	010444		JSR	PC,REWIND	;CALL TAPE REWIND COMMAND	
10740	062242	103411			BCS	230:	;BR, IF NO PROBLEM	
10741	062244	010004			MOV	R0,R4	;SAVE PACKET ADDRESS	
10742	062246	016501	000000		MOV	TSSR(R5),R1	;GET TSSR	
10743	062252	004737	020116		JSR	PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS	
10747	062256				ERRHRD	ERRNO,T28RWN,PKTSSR	;REWIND NOT ACCEPTED	
	062256	104456						TRAP C\$ERHRD
	062260	000642						.WORD 418
	062262	064331						.WORD T28RWN
	062264	011710						.WORD PKTSSR
10748	062266			230:	CKLOOP		;LOOP IF SELECTED	
	062266	104406						TRAP C\$CLP1
10749	062270	013701	063236		MOV	T288FR+6,R1	;PICK UP XST0	
10750	062274	010102			MOV	R1,R2	;SET UP EXPECTED	
10751	062276	052702	000002		BIS	#BIT1,R2	;SET BOT BIT IN EXPECTED	
10752	062302	020102			CMP	R1,R2	;DOES EXP = REC'D	
10753	062304	001406			BEQ	240:	;BR, IF EQUAL (OK)	
10754	062306	004737	020116		JSR	PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS	
10758	062312				ERRHRD	ERRNO,T28BOT,EXPREC	;TAPE NOT AT BOT AFTER REWIND	
	062312	104456						TRAP C\$ERHRD
	062314	000643						.WORD 419
	062316	064207						.WORD T28BOT
	062320	016360						.WORD EXPREC
10759	062322			240:	CKLOOP		;LOOP IF SELECTED	
	062322	104406						TRAP C\$CLP1

10760	062324	012700	177777		MOV	#177777,R0	;VALUE TO WRITTEN TO MEMORY
10761	062330	004737	020410		JSR	PC,FILLMEM	;FILL MEM WITH ALL ONES
10762	062334	013737	003072	063342	MOV	FREE,T28R8	;STARTING READ BUFFER ADDRESS
10763	062342	012737	100001	063340	MOV	#100001,T28PK3	;READ FORWARD,ACK, COMMAND
10764	062350	012704	063340		MOV	#T28PK3,R4	;SET UP R4 WITH PACKET ADDRESS
10765	062354	013737	000024	063346	MOV	20.,T28SZ	;SET UP RECORD SIZE IN PACKET
10766	062362	010465	177776		MOV	R4,TSD8(R5)	;ISSUE COMMAND
10767	062366	004737	017134		JSR	PC,WAITF	;WAIT FOR SSR TO SET
10768	062372	016501	000000		MOV	TSSR(R5),R1	;GET TSSR CONTENTS
10769	062376	012702	100204		MOV	#SSR!SC!BIT2,R2	;SET UP EXPECTED
10770	062402	020102			CMP	R1,R2	;ARE THEY EQUAL
10771	062404	001406			BEQ	245:	;BR, IF OK
10772	062406	004737	020116		JSR	PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS
10776	062412				ERRHRD	ERRNO,T28WDE,PKTSSR	;TSSR INCORRECT AFTER WRITE DATA
	062412	104456					TRAP C:ERHRD
	062414	000644					.WORD 420
	062416	064116					.WORD T28WDE
	062420	011710					.WORD PKTSSR
10777	062422			245:	CKLOOP		;LOOP IF SELECTED
	062422	104406					TRAP C:CLP1
10774	062424	013701	063236		MOV	T28BFR+6,R1	;PICK UP XSTO
10779	062430	010102			MOV	R1,R2	;SET UP EXPECTED
10780	062432	052702	100000		BIS	#BIT15,R2	;TMK SHOULD BE SET
10781	062436	020102			CMP	R1,R2	;IS TMK SET
10782	062440	001406			BEQ	247:	;BR, IF TMK WAS SET (GOOD)
10783	062442	004737	020116		JSR	PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS
10787	062446				ERRHRD	ERRNO,T28RRP,EXPREC	;TMK NOT SET AFTER READ REV
	062446	104456					TRAP C:ERHRD
	062450	000645					.WORD 421
	062452	065014					.WORD T28RRP
	062454	016360					.WORD EXPREC
10788	062456			247:	CKLOOP		;LOOP IF SELECTED
	062456	104406					TRAP C:CLP1
10789	062460	017701	120406		MOV	#FREE,R1	;FIRST LOC IN READ BUFFER
10790	062464	012702	177777		MOV	#177777,R2	;EXPECTED IF NO DATA TRANS.
10791	062470	020102			CMP	R1,R2	;DID ANY DATA GET TRANSFERRED
10792	062472	001406			BEQ	250:	;BR, IF NO DATA TRANS (GOOD)
10793	062474	004737	020116		JSR	PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS
10797	062500				ERRHRD	ERRNO,T28DTR,EXPREC	;DATA TRANSFERRED ON READ TAPE MARK
	062500	104456					TRAP C:ERHRD
	062502	000646					.WORD 422
	062504	065072					.WORD T28DTR
	062506	016360					.WORD EXPREC
10798	062510			250:	CKLOOP		;LOOP IF SELECTED
	062510	104406					TRAP C:CLP1
10799	062512	012737	100410	063340	MOV	#100410,T28PK3	;SPACE REVERSE,ACK, COMMAND
10800	062520	012737	000005	063342	MOV	#5,T28R8	;NUMBER OF RECORDS TO SPACE BACK
10801	062526	012704	063340		MOV	#T28PK3,R4	;SET UP R4 WITH PACKET ADDRESS
10802	062532	010465	177776		MOV	R4,TSD8(R5)	;ISSUE COMMAND
10803	062536	004737	017134		JSR	PC,WAITF	;WAIT FOR SSR TO SET
10804	062542	016501	000000		MOV	TSSR(R5),R1	;GET TSSR CONTENTS
10805	062546	012702	100204		MOV	#SSR!SC!BIT2,R2	;SET UP EXPECTED
10806	062552	020102			CMP	R1,R2	;ARE THEY EQUAL
10807	062554	001406			BEQ	260:	;BR, IF OK
10808	062556	004737	020116		JSR	PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS
10812	062562				ERRHRD	ERRNO,T28RDG,PKTSSR	;TSSR INCORRECT AFTER SPACE REV CMD.
	062562	104456					TRAP C:ERHRD

Address	Offset	Hex	Label	Operation	Comment	Register	Value	Word	Value
062564	000647							.WORD	423
062566	063625							.WORD	T28RDG
062570	011710							.WORD	PKTSSR
10813	062572		260:	CKLOOP					
	062572	104406							
10814	062574	013701	063236	MOV	T28BFR+6,R1			TRAP	C\$CLP1
10815	062600	010102		MOV	R1,R2				
10816	062602	052702	100000	BIS	#BIT15,R2				
10817	062606	020102		CMP	R1,R2				
10818	062610	001406		BEQ	270:				
10819	062612	004737	020116	JSR	PC,FATCHK				
10823	062616			ERRHRD	ERRNO,T28RRN,EXPREC				
	062616	104456							
	062620	000650						TRAP	C\$ERHRD
	062622	064735						.WORD	424
	062624	016360						.WORD	T28RRN
10824	062626		270:	CKLOOP				.WORD	EXPREC
	062626	104406							
10825	062630	013701	063234	MOV	T28BFR+4,R1			TRAP	C\$CLP1
10826	062634	012702	000004	MOV	#4,R2				
10827	062640	020102		CMP	R1,R2				
10828	062642	001406		BEQ	280:				
10829	062644	004737	020116	JSR	PC,FATCHK				
10833	062650			ERRHRD	ERRNO,T28PBP,EXPREC				
	062650	104456							
	062652	000651						TRAP	C\$ERHRD
	062654	063461						.WORD	425
	062656	016360						.WORD	T28PBP
10834	062660		280:	CKLOOP				.WORD	EXPREC
	062660	104406							
10835	062662	012737	100410	MOV	#100410,T28PK3			TRAP	C\$CLP1
10836	062670	012737	000001	MOV	#1,T28RB				
10837	062676	012704	063340	MOV	#T28PK3,R4				
10838	062702	010465	177776	MOV	R4,TSD8(R5)				
10839	062706	004737	017134	JSR	PC,WAITF				
10840	062712	016501	000000	MOV	TSSR(R5),R1				
10841	062716	012702	100204	MOV	#SSR!SC!BIT2,R2				
10842	062722	020102		CMP	R1,R2				
10843	062724	001406		BEQ	290:				
10844	062726	004737	020116	JSR	PC,FATCHK				
10848	062732			ERRHRD	ERRNO,T28RDG,PKTSSR				
	062732	104456							
	062734	000652						TRAP	C\$ERHRD
	062736	063625						.WORD	426
	062740	011710						.WORD	T28RDG
10849	062742		290:	CKLOOP				.WORD	PKTSSR
	062742	104406							
10850	062744	013701	063244	MOV	T28BFR+14,R1	</			

Line	Address	Offset	Label	Operation	Comments	Register/Value	Condition	Trap	Word	ExpRec
10860	062776	016360	300\$:	CKLOOP						
	062776	104406								
10861	063000	012737	100010	063340	MOV	#100010,T28PK3				
10862	063006	012737	000005	063342	MOV	#5,T28RB				
10863	063014	012704	063340		MOV	#T28PK3,R4				
10864	063020	010465	177776		MOV	R4,TSD8(R5)				
10865	063024	004737	017134		JSR	PC,WAITF				
10866	063030	016501	000000		MOV	TSSR(R5),R1				
10867	063034	012702	100204		MOV	#SSR!SC!BIT2,R2				
10868	063040	020102			CMP	R1,R2				
10869	063042	001406			BEQ	310\$				
10870	063044	004737	020116		JSR	PC,FATCHK				
10874	063050				ERRHRD	ERRNO,T28RDF,EXPREC				
	063050	104456								
	063052	000654								
	063054	063544								
	063056	016360								
10875	063060				310\$:	CKLOOP				
	063060	104406								
10876	063062	013701	063236		MOV	T28BFR+6,R1				
10877	063066	010102			MOV	R1,R2				
10878	063070	052702	100000		BIS	#BIT15,R2				
10879	063074	020102			CMP	R1,R2				
10880	063076	001406			BEQ	320\$				
10881	063100	004737	020116		JSR	PC,FATCHK				
10885	063104				ERRHRD	ERRNO,T28RRP,EXPREC				
	063104	104456								
	063106	000655								
	063110	065014								
	063112	016360								
10886	063114				320\$:	CKLOOP				
	063114	104406								
10887	063116	013701	063234		MOV	T28BFR+4,R1				
10888	063122	012702	000004		MOV	#4.,R2				
10889	063126	020102			CMP	R1,R2				
10890	063130	001406			BEQ	330\$				
10891	063132	004737	020116		JSR	PC,FATCHK				
10895	063136				ERRHRD	ERRNO,T28PBP,EXPREC				
	063136	104456								
	063140	000656								
	063142	063461								
	063144	016360								
10896	063146				330\$:	CKLOOP				
	063146	104406								
10897	063150				ENDSUB					
	063150									
	063150	104403								
10898	063152	023727	002170	000031	CMP	FATFLG,#25.				
10899	063160	002402			BLT	999\$				
10900	063162	004737	020170		JSR	PC,CKDROP				
10901	063166				999\$:					
10902					:					
10903					:					
10904					:					
10905										

H5

CZTKGA TK-25 FRI END FUNC 03
TEST 4: WRITE/READ TAPE MARK

MACRO M1200 20-APR 84 08:13 PAGE 124-9

SEQ 266

10907 063174 000137 061100
10908 063200
10909 063200
063200 104432
063202 002252

1634: JMP T28LOOP
EXIT TST

;EXECUTE AGAIN

;ALL DONE THIS TEST

TRAP C\$EXIT
.WORD L10074-

```
10911
10912
10913
10915 063204
10917 063210
10918 063210 100004
10919 063212 063220
10920 063214 000000
10921 063216 000012
10922 063220
10923 063220 063230
10924 063222 000000
10925 063224 000024
10926 063226 000000
10927 063230
10928
10929
10930
10932 063312
10934 063320
10935 063320 100006
10936 063322 063350
10937 063324 000000
10938 063326 000006
10939
10941 063330
10943 063340
10944 063340 100005
10945 063342
10946 063342 003072
10947 063344 000000
10948 063346 000000
10949
10950
10951
10952
10953 063350
10954 063350 010
10955 063351 200
10956 063352 000000
10957 063354 000000
10958
10959
10960
10961
10962
10963 063356
10964 063356 101411
10965 063360 102011
10966 063362 103411
10967 063364 177777
10968 063366 100011
10969 063370 100411
10970 063372 101011
10971 063374 177777
10972
10973
```

```

;
;LOCAL STORAGE FOR THIS TEST
;
;
      .BLKB    10-<.-TUV2A&7>
T28PACKET:
      .WORD    100004
      .WORD    T28DATA
      .WORD    0
      .WORD    10.
T28DATA:
      .WORD    T28BFR
      .WORD    0
      .WORD    20.
      .WORD    0
T28BFR: .BLKW  25.
;
;WRITE SUBSYSTEM MEMORY COMMAND PACKET
;
      .BLKB    10-<.-TUV2A&7>
T28PK2:
      .WORD    100006
      .WORD    T28BF2
      .WORD    0
      .WORD    6.
;
      .BLKB    10-<.-TUV2A&7>
T28PK3:
      .WORD    100005
T28RB:
T28WB: .WORD   FREE
      .WORD    0
T28SZ: .WORD    0
      .EVEN
;
;
T28BF2:
T28BS0: .BYTE   10
T28BS1: .BYTE   200
T28S2: .WORD    0
T28S3: .WORD    0
;
;
      .EVEN
;TAPE MOTION PACKET COMMAND VALUES
T28IMV:
      .WORD    101411
      .WORD    102011
      .WORD    103411
      .WORD    177777
T28RN: .WORD    100011
T28WDR: .WORD    100411
T28CON: .WORD    101011
      .WORD    177777
;
;COMMAND PACKET FOR TEST
;WRITE CHARACTERISTICS COMMAND, WITH IE, ACK
;ADDRESS OF CHARACTERISTICS BLOCK
;
;STARTING VALUE OF BLOCK SIZE
;CHARACTERISTICS DATA BLOCK
;ADDRESS OF MESSAGE BUFFER
;LENGTH OF MESSAGE BUFFER
;MESSAGE BUFFER
;
;WRITE SUB SYS MEM COMMAND, IE AND ACK
;ADDRESS OF SELECT BLOCK DATA
;SIZE OF DATA PACKET
;
;REREAD COMMAND, AND ACK
;ADDRESS OF WRITE BUFFER
;SIZE OF BUFFER (EXTENT)
;
;BSELO AREA
;BSEL1 AREA
;SEL 2 AREA
;DATA AREA
;
;ILLEGAL MODE BITS TEST DATA
;
;WRITE TAPE MARK COMMAND
;ERASE COMMAND
;WRITE TAPE MARK RETRY
;END OF DATA
```

CZTKGA TK-25 FRT END FUNC #3
TEST 4: WRITE/READ TAPE MARK

MACRO M1200 20-APR-84 08:13 PAGE 125-1

SEQ 268

10974 063376 000000
10975 063400 000000
10976 063402 000000
10977
10978

T28CNT: .WORD 0
T28CNU: .WORD 0
T28DLY: .WORD 0
.EVEN

;TAPE TIMER COUNTER STORAGE AREA
;TAPE TIMER COUNTER STORAGE AREA
;DELAY COUNTER

```

10980
10981      ;*
10982      ;LOCAL TEXT MESSAGES FOR TEST
10983      ;-
10984
10985 063404      124      141      160  T28RIB: .ASCIZ 'Tape Position Not Correct, RIB Should Be Set'
10986 063461      122      145      163  T28PBP: .ASCIZ 'Residual Byte Counter Register (RBPCR) Not Correct'
10987 063544      124      123      123  T28RDF: .ASCIZ 'TSSR Incorrect After READ REVERSE Into TAPE MARK'
10988 063625      124      123      123  T28RDG: .ASCIZ 'TSSR Incorrect After SPACE Command Into TAPE MARK'
10989 063707      124      123      123  T28WDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
10990 063764      111      154      154  T28LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XSTO'
10991 064045      127      122      111  T28SSR: .ASCIZ 'WRITE MISCELLANEOUS Command Not Accepted'
10992 064116      124      123      123  T28WDE: .ASCIZ 'TSSR Not Correct After READ DATA Command, Into TAPE MARK'
10993 064207      124      141      160  T28BOT: .ASCIZ 'Tape Not At BOT After REWIND Command'
10994 064254      124      123      123  T28TM: .ASCIZ 'TSSR Not Correct After FORMAT Command Reject'
10995 064331      122      145      167  T28RWN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
10996 064400      104      162      151  T28OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
10997 064453      124      123      123  T28WDC: .ASCIZ 'TSSR Not Correct After WRITE TAPE MARK Command'
10998 064532      103      126      103  T28VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
10999 064605      124      115      113  T28TMK: .ASCIZ 'TMK Not Set After WRITE TAPE MARK Command'
11000 064657      124      115      113  T28RRM: .ASCIZ 'TMK Not Set After READ REVERSE Into TAPE MARK'
11001 064735      124      115      113  T28RRN: .ASCIZ 'TMK Not Set After SPACE REVERSE Into TAPE MARK'
11002 065014      124      115      113  T28RRP: .ASCIZ 'TMK Not Set After READ FORWARD Into TAPE MARK'
11003 065072      104      141      164  T28DTR: .ASCIZ 'Data Transferred On READ REVERSE Into A TAPE MARK'
11004 065154      104      141      164  T28DTA: .ASCIZ 'Data Compare Error, Data Read From Tape Not Equal To Written'
11005 065251      127      162      151  T28TID: .ASCIZ 'Write/Read Tape Mark'
11006      .EVEN
11007
11008      ;*
11009      ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
11010      ;WRITE SUBSYSTEM MEMORY COMMAND
11011      ;
11012      ;-
11013
11014 065276      T28REST:
11015 065276      SAVREG
11016 065302      012701 063210      MOV      #T28PACKET,R1      ;SAVE THE REGISTERS
11017 065306      012721 100004      MOV      #100004,(R1)      ;START OF THE PACKET
11018 065312      012721 063220      MOV      #T28DATA,(R1)      ;WRITE SUBSYSTEM MEM. WITH ACK.
11019 065316      005021      CLR      (R1)      ;ADDRESS OF CHARAISTICS DATA BLOCK
11020 065320      012721 000012      MOV      #10,(R1)      ;EXTENDED ADDRESS
11021 065324      012721 063230      MOV      #T28BFR,(R1)      ;SIZE OF DATA BLOCK IN BYTES
11022 065330      005021      CLR      (R1)      ;ADDRESS OF MESSAGE BUFFER
11023 065332      012721 000024      MOV      #20,(R1)      ;LENGTH OF MESSAGE BUFFER
11024 065336      005021      CLR      (R1)
11025 065340      012711 000000      MOV      #0,(R1)      ;SELECT DRIVE ZERO
11026 065344      012702 000030      MOV      #24,R2      ;NUMBER OF LOCATIONS TO BE CLEARED
11027 065350      012762 177777 063230 64: MOV      #177777,T28BFR(R2) ;ALL ONES TO MESSAGE BUFFER
11028 065356      005742      TST      -(R2)      ;NEXT LOCATION
11029 065360      020227 000000      CMP      R2,#0      ;CHECK FOR END
11030 065364      001371      BNE      64:      ;KEEP GOING UNTIL DONE
11031 065366      000207      RTS      PC      ;RETURN
11032
11033
11034 065370      T28RT2:
11035 065370      SAVREG
11036 065374      012701 063320      MOV      #T28PK2,R1      ;SAVE THE REGISTERS

```

```

11037 065400 012721 100006      MOV      #100006,(R1)+      ;WRITE SUBSYSTEM MEM. WITH ACK,
11038 065404 012721 063350      MOV      #T28BF2,(R1)+      ;ADDRESS OF DATA BLOCK
11039 065410 005021              CLR      (R1)+          ;EXTENDED ADDRESS
11040 065412 012721 000006      MOV      #6,(R1)+          ;SIZE OF DATA BLOCK IN BYTES
11041 065416 005021              CLR      (R1)+          ;
11042 065420 012701 063350      MOV      #T28BF2,R1        ;POINT TO DATA SEL AREA
11043 065424 005021              CLR      (R1)+          ;
11044 065426 005011              CLR      (R1)+          ;
11045 065430 000207              RTS      PC              ;RETURN
11046 065432
11047 065432
11048 065436 012701 063340      SAVREG
11049 065442 005021      MOV      #T28PK3,R1        ;GET PACKET ADDRESS
11050 065444 005021      CLR      (R1)+          ;CLEAR COMMAND AREA
11051 065446 005021      CLR      (R1)+          ;CLEAR ADDRESS AREA
11052 065450 005011      CLR      (R1)+          ;CLEAR EXTENDED ADDRESS AREA
11053 065452 000207      RTS      PC              ;SIZE OF DATA TRANSFER
11054 065454              ENDTST                          ;RETURN
        065454
        065454 104401
11060
11065
11071
11072
11073
11074
11075
11076
11077
11078
11079
11080
11081
11082
11083
11084 065456
        065456 000015
        065460
11085
11086 065460      GPRMA      HPM1,0,0,160000,177776,YES      ;GET TSBA/TSDB REGISTER ADDRESS.
        065460 000031      .WORD      T$CODE
        065462 065512      .WORD      HPM1
        065464 160000      .WORD      T$LOLIM
        065466 177776      .WORD      T$HILIM
11087 065470      GPRMA      HPM2,2,0,0,776,YES              ;GET VECTOR ADDRESS.
        065470 001031      .WORD      T$CODE
        065472 065541      .WORD      HPM2
        065474 000000      .WORD      T$LOLIM
        065476 000776      .WORD      T$HILIM
11088 065500      GPRMD      HPM3,4,0,340,0,7,YES            ;GET INTERRUPT PRIORITY.
        065500 002032      .WORD      T$CODE
        065502 065565      .WORD      HPM3
        065504 000340      .WORD      340
        065506 000000      .WORD      T$LOLIM
        065510 000007      .WORD      T$HILIM
11089 065512      ENDMRD
        .EVEN

```

.SBTTL HARDWARE PARAMETER CODING SECTION

```

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

```

```

BGNMRD
.WORD L10076-L$HARD/2
L$HARD::

```

```

L10074:
        TRAP      C$ETST

```

	065512				L10076:		
11090	065512	104	105	126	HPM1:	.ASCIZ	'DEVICE ADDRESS (TSSR) '
11091	065541	111	116	124	HPM2:	.ASCIZ	'INTERRUPT VECTOR '
11092	065565	111	116	124	HPM3:	.ASCIZ	'INTERRUPT PRIORITY '
11093						.EVEN	
11094							

```

11096                                     .SBTTL  SOFTWARE PARAMETER CODING SECTION
11097
11098                                     ;**
11099                                     ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
11100                                     ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
11101                                     ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
11102                                     ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
11103                                     ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
11104                                     ; WITH THE OPERATOR.
11105                                     ;--
11106 065616                                     BGNSFT
11107 065616 000006                                     .WORD L10077-L$SOFT/2
11108 065620                                     L$SOFT::
11109 065620 000130                                     GPRML  SPM1,0,-1,YES ;GET RAM DUMP FLAG
11110 065622 065634                                     .WORD  T$CODE
11111 065624 177777                                     .WORD  SPM1
11112                                     .WORD  -1
11113 065626 001130                                     GPRML  SPM4,2,-1,YES ; GET ITERATION CONTROL.
11114 065626 001130                                     .WORD  T$CODE
11115 065630 065700                                     .WORD  SPM4
11116 065632 177777                                     .WORD  -1
11117                                     : GPRMD  SPM6,4,D,7777,0,7777,YES ; GET LOCAL ERROR LIMIT
11118                                     : GPRMD  SPM7,6,D,7777,0,7777,YES ; GET GLOBAL ERROR LIMIT
11119 065634                                     ENDSFT
11120                                     .EVEN
11121                                     L10077:
11122
11123 11112 065634 105 116 101 SPM1: .ASCIZ 'ENABLE CONTROLLER RAM DUMP ON ERROR'
11124 11113 065700 111 116 110 SPM4: .ASCIZ 'INHIBIT ITERATIONS'
11125 11114 065730 120 105 122 SPM6: .ASCIZ 'PER TEST ERROR LIMIT'
11126 11115 065760 120 105 122 SPM7: .ASCIZ 'PER UNIT ERROR LIMIT'
11127 11118                                     .EVEN
11128 11119                                     .SBTTL  PATCH AREA
11129
11130                                     ;*
11131                                     ;DISPATCH TABLE
11132                                     ;
11133                                     ; *** MOVE TO FRONT OF PROGRAM FOR RELEASE ***
11134                                     ;--
11135 066010                                     DISPATCH  TESTNO
11136 066010 000004                                     .WORD  4
11137 066012                                     L$DISPATCH::
11138 066012 023650                                     .WORD  T1
11139 066014 031666                                     .WORD  T2
11140 066016 051036                                     .WORD  T3
11141 066020 061034                                     .WORD  T4
11142
11143                                     ;
11144                                     ; FINALLY A GENEROUS PATCH AREA.
11145                                     ;
11146                                     ; AND AN ADJUSTMENT TO ACCOUNT FOR THE "LASTAD BIT7" HACK
11147                                     ; DESCRIBED IN "SUPPRG.MEM" (FOR REV C).
11148                                     ;
11149                                     PATCH::
11150
11151
11152
11153
11154
11155
11156

```

```

11137      ;      .IF      NZ..6377
11138      ;      .-..!377.1
11139      ;      ENDC
11140 066022      LASTAD      ;SET LAST USED ADDRESS.
                .EVEN
                .WORD T$FREE
                .WORD T$SIZE
11141      L$LAST::      .SBTTL  HARD CODED P-TABLE
11142      ;**
11143      ;      DIAGNOSTIC IS PRE-PARAMETERIZED PER THIS TABLE
11144      ;--
11145 066026      BGNSETUP      1
11146 066026      BGNPTAB
                .WORD 0
                .WORD L10102-./2-1
11147 066032      L10100:      .WORD      172522
11148 066034      .WORD      224
11149 066036      .WORD      PRI05
11150 066040      ENDP TAB
11151 066040      L10102:      ENDSETUP
11152      .END
11153      000001

```

ADDSR	012002	G	C#AU	000052	DEBUG	011474	FATFLG	002170	G	HIMEM	007776		
ADR	000020	G	C#AUTO	000061	DEVcnt	002166	G	FERCH	011564	HOE	100000	G	
AMBTSS	006343		C#BRK	000022	DEVDR0	023600		FIFEXP	012032	G	HPM1	065512	
ASSEMB	000010		C#BSEG	000004	DEVNRD	023517		FIF1MS	012104		HPM2	065541	
A1716	000003		C#BSUB	000002	DEVNRR	023435		FIF2MS	012153		HPM3	065565	
BADDAT	003110	G	C#CEFG	000045	DEVONL	023365		FILLME	020410		IBE	010000	G
BADSSR	016564	G	C#CLCK	000062	DEVSUM	023330		FNOINT	004113		IDU	000040	G
BAR	174402		C#CLEA	000012	DFPTBL	002124	G	FORCER	002144	G	IER	020000	G
BENBSW	002174	G	C#CLOS	000035	DIAGMC	000000		FREE	003072	G	IFault	004154	
BIE	040000		C#CLP1	000006	DLCYL	000177		FREEM	003076		INCERK	017760	
BIT0	000001	G	C#CVEC	000036	OLDNER	100200		FRESIZ	003074	G	INTCPC	017034	
BIT00	000001	G	C#DCLN	000044	DLERR	177730		FUSI	004015		INTFLA	017031	
BIT01	000002	G	C#DODU	000051	DLGETS	000004		F#AU	000015		INTMAS	017030	
BIT02	000004	G	C#DRPT	000024	DLRDMD	000010		F#AUTO	000020		INTR	017102	G
BIT03	000010	G	C#DU	000053	DLRDNH	000016		F#BGN	000040		INTREC	002172	G
BIT04	000020	G	C#EDIT	000003	DLSR	000013		F#CLEA	000007		INTVEC	017032	
BIT05	000040	G	C#ERDF	000055	DLUN	000006		F#DU	000016		INTX	004176	
BIT06	000100	G	C#ERMR	000056	DSBINT	017070		F#END	000041		IOCKKI	000200	
BIT07	000200	G	C#ERRO	000060	DUAD12	004541		F#HARD	000004		IOKSTP	000001	
BIT08	000400	G	C#ERSF	000054	DUFLG	003060	G	F#HW	000013		IPRI	002160	G
BIT09	001000	G	C#ERSO	000057	DUMMY	003030		F#INIT	000006		ISR	000100	G
BIT1	000002	G	C#ESCA	000010	EF.CON	000036	G	F#JMP	000050		IVEC	002156	G
BIT10	002000	G	C#ESEG	000005	EF.NEW	000035	G	F#MOD	000000		IXE	004000	G
BIT11	004000	G	C#ESUB	000003	EF.PWR	000034	G	F#MSG	000011		I#AU	000041	
BIT12	010000	G	C#ETST	000001	EF.RES	000037	G	F#PROT	000021		I#AUTO	000041	
BIT13	020000	G	C#EXIT	000032	EF.STA	000040	G	F#PWR	000017		I#CLN	000041	
BIT14	040000	G	C#GETB	000026	EMAXDU	017713		F#RPT	000012		I#DU	000041	
BIT15	100000	G	C#GETW	000027	EN	000000		F#SEG	000003		I#HRD	000041	
BIT2	000004	G	C#GMAN	000043	ENAIN	017036		F#SOFT	000005		I#INIT	000041	
BIT3	000010	G	C#GPHR	000042	ENVIRN	021550		F#SRV	000010		I#MOD	000040	
BIT4	000020	G	C#GPLO	000030	EPRTSW	002146	G	F#SUB	000002		I#MSG	000041	
BIT5	000040	G	C#GPRI	000040	EPRT1	005672		F#SW	000014		I#PROT	000040	
BIT6	000100	G	C#INIT	000011	EPRT2	005771		F#TEST	000001		I#PTAB	000041	
BIT7	000200	G	C#INLP	000020	EPRT3	006032		GDDAT	003112	G	I#PWR	000041	
BIT8	000400	G	C#MANI	000050	ERCM	011575		GERRMA	002142	G	I#RPT	000041	
BIT9	001000	G	C#MEM	000031	ERRMI	002202	G	GETPAT	021114	G	I#SEG	000041	
BOE	000400	G	C#MSG	000023	ERRK	017672		GETSEL	021176	G	I#SETU	000041	
BRINIT	004355		C#OPEN	000034	ERRLO	002204	G	G#CNT0	000200		I#SFT	000041	
BSELO	000000		C#PNTB	000014	ERRNO	000656		G#DELM	000372		I#SRV	000041	
BSEL1	000001		C#PNTF	000017	ERRVEC	000004	G	G#DISP	000003		I#SUB	000041	
CHKAMB	016730		C#PNTS	000016	ERTABE	003330		G#EXCP	000400		I#TST	000041	
CHKMAN	021420	G	C#PNTX	000015	ERTABL	003130		G#HILI	000002		J#JMP	000167	
CHKTSS	017252		C#QIO	000377	ESUM	017674		G#LOLI	000001		KIPAR0	172340	
CKDROP	020170		C#RDBU	000007	EVL	000004	G	G#NO	000000		KIPAR1	172342	
CKEMAX	020016		C#REFG	000047	EXBCNT	000010		G#OFFS	000400		KIPAR2	172344	
CKMSG	011222	G	C#RESE	000033	EXPBRE	016366	G	G#OF SI	000376		KIPAR3	172346	
CKMSG2	011342	G	C#REVI	000003	EXPD	002176	G	G#PRMA	000001		KIPAR4	172350	
CKRAM	010544	G	C#RFLA	000021	EXPGOT	004431		G#PRMD	000002		KIPAR5	172352	
CKRAM2	011120	G	C#RPT	000025	EXPGT2	004465		G#PRML	000000		KIPAR6	172354	
CMEMEM	020574		C#SEFG	000046	EXPMMSG	002266	G	G#RADA	000140		KIPAR7	172356	
CONF IG	020236		C#SPRI	000041	EXPREC	016360	G	G#RADB	000000		KIPDR0	172300	
COUNT	002254	G	C#SVEC	000037	EXTA	005232		G#RAD0	000040		KIPDR1	172302	
CSR	174400		C#TPRI	000013	EXTEND	005230		G#RADL	000120		KIPDR2	172304	
CSRADD	002154	G	DAR	174404	E#END	002100		G#RADO	000020		KIPDR3	172306	
CTAB	003116	G	DATA	002256	E#LOAD	000035		G#XFER	000004		KIPDR4	172310	
CTABE	003130	G	DATAFL	015100	FATCHK	020116		G#YES	000010		KIPDR5	172312	
CTABM	003116	G	DATASC	021152	FATERR	000060		HIADDR	001400		KIPDR6	172314	

KIPDR7= 172316	L\$SOFT 065620 G	L10061 042734	NXRX 003675	PRMESS 014072
KTENAB 003102 G	L\$SPC 002056 G	L10062 043654	NXTU 022224	PRMNO 002264 G
KTFLG 003100 G	L\$SPCP 002020 G	L10063 044652	OFL = 000100	PRMSG 015436 G
KTINIT 021636	L\$SPTP 002024 G	L10064 045222	ONEFIL= 000000	PRMSG0 015616
KTOFF 020262	L\$STA 002030 G	L10065 045664	O\$APTS= 000000	PRMSG1 015663
KTON 020244	L\$SW 002134 G	L10066 061032	O\$AU = 000001	PRMSG2 015721
LERRMA 002140 G	L\$TEST 002114 G	L10067 051456	O\$BGNR= 000001	PROASC 014730
LISTAL= 000001	L\$TIML 002014 G	L10070 052230	O\$BGNS= 000001	PR1ASC 014775
LOE = 040000 G	L\$UNIT 002012 G	L10071 053044	O\$DU = 000001	PST32W 003104 G
LOOPCN 002164 G	L10000 002132	L10072 053740	O\$ERRT= 000000	PUNIT 022512
LOOPCO 012770	L10001 002144	L10073 055456	O\$GNSW= 000001	PW.D11= 000021
LOOPFL 003114 G	L10002 005226	L10074 065454	O\$POIN= 000001	PW.D13= 000022
LOT = 000010 G	L10003 011706	L10075 063150	O\$SETU= 000001	PW.D22= 000020
L\$ACP 002110 G	L10004 011736	L10076 063512	PASRPT 022256	PW.NOP= 000000
L\$APT 002036 G	L10005 011754	L10077 065634	PATLM 066022 G	PW.NO1= 000023
L\$AU 022560 G	L10006 011762	L10100 066032	PATDAT 021150	PW.RDE= 000024
L\$AUT 002070 G	L10007 012000	L10102 066040	PC.ERA= 002400	PW.RDR= 000001
L\$AUTO 022764 G	L10010 012016	MEMADD 013616 G	PC.IER= 002000	PW.RDS= 000005
L\$CCP 002106 G	L10011 012030	MENASC 021367	PC.NOO= 001000	PW.RFI= 000003
L\$CLEA 023040 G	L10012 012102	MENERR 021314	PC.REL= 000000	PW.WCT= 000006
L\$CO 002032 G	L10013 012252	MENRES 021416	PC.REW= 000400	PW.WFI= 000004
L\$DEPO 002011 G	L10014 012766	MESBFA 002716 G	PKBCNT= 000006	PW.WFM= 000007
L\$DESC 003342 G	L10015 013614	MESBFN 014650	PKHI = 000004	PW.WMI= 000010
L\$DESP 002076 G	L10016 013636	MESHEA 015033	PKLOW = 000002	PW.WNP= 000011
L\$DEVP 002060 G	L10017 016364	MMVEC = 000250	PKTADD 007302	PW.WTR= 000002
L\$DISP 066012 G	L10020 016372	MPR = 174406	PKTFRM 007244	P.ACK = 100000
L\$DLY 002116 G	L10021 016400	MSA.FR= 000006	PKTGET 011740 G	P.CMD = 000037
L\$DTP 002040 G	L10022 016412	MSA.NO= 000000	PKTMES 011764 G	P.CONT= 000012
L\$DTYP 002034 G	L10023 016434	MSA.NR= 000004	PKTNEW 007337	P.CVC = 040000
L\$DU 022656 G	L10024 016462	MSA.VD= 000002	PKTRAM 004643 G	P.FMT = 000140
L\$DUT 002072 G	L10025 016622	MSGEXP 012020 G	PKTSSR 011710 G	P.FORM= 000011
L\$DVTY 003334 G	L10026 017132	MSGLOO 012726 G	PNT = 001000 G	P.GETS= 000017
L\$EF 002052 G	L10030 022510	MSGSTA 012212 G	PRAMPK 013640	P.IE = 000200
L\$ENVI 002044 G	L10031 022654	MSGSUB 013604 G	PRBEXP 016354	P.INIT= 000013
L\$ETP 002102 G	L10032 022762	MS.ATT= 000006	PRBMSG 016222	P.MODE= 007400
L\$EXP1 002046 G	L10033 023036	MS.EXT= 000200	PRBREC 016356	P.OPP = 020000
L\$EXP4 002064 G	L10034 023064	MS.RSD= 000001	PRBTOT 016307	P.POSI= 000010
L\$EXP5 002066 G	L10035 023326	MS.RSF= 000020	PRBYTE 016006 G	P.READ= 000001
L\$HARD 065460 G	L10036 031664	MS.RST= 000010	PRI = 002000 G	P.SWB = 010000
L\$HIME 002120 G	L10037 024372	NBA = 002000	PRIADD 007716	P.WRT= 000005
L\$HPCP 002016 G	L10040 025032	NEWPAS 022212	PRIAO 007766	P.WRTC= 000004
L\$HPTP 002022 G	L10041 025466	NODEV 003062 G	PRIBXO 007350 G	P.WRTS= 000006
L\$HW 002124 G	L10042 026216	NOINIT 004233	PRIEQU 007616	QVP 002152 G
L\$ICP 002104 G	L10043 027144	NOINTR 004117	PRIPKT 007076 G	RAMASC 014006
L\$INIT 022000 G	L10044 027454	NOITS 002136 G	PRIRAM 007624	RANDAT 002206 G
L\$LADP 002026 G	L10045 030044	NOHAN 021454	PRITAD 010032	RAMER 010646 G
L\$LAST 066026 G	L10046 051034	NP.IR = 000200	PRITSS 005264	RAMERR 016374 G
L\$LOAD 002100 G	L10047 032616	NP.LOO= 000040	PRITO 010102	RAMEXP 016414 G
L\$LUN 002074 G	L10050 033454	NP.OUT= 000100	PRI XOR 007500 G	RAMFMR 014552
L\$PREV 002050 G	L10051 034340	NP.WRP= 000020	PRI00 = 000000 G	RAMFOR 007654
L\$NAME 002000 G	L10052 035260	NSI 004050	PRI01 = 000040 G	RAMILD 011030
L\$PRIO 002042 G	L10053 036026	NSINIT 004305	PRI02 = 000100 G	RAMIOP 011034
L\$PROT 021770 G	L10054 036656	NUL 004425	PRI03 = 000140 G	RAMPD 011105
L\$PRT 002112 G	L10055 037520	NULCR 004426	PRI04 = 000200 G	RAMRSH 011032
L\$REPP 002062 G	L10056 040362	NXM = 004000	PRI05 = 000240 G	RAMSIZ 002246 G
L\$REV 002010 G	L10057 041230	NXR 003636	PRI06 = 000300 G	RAMTAD 016402 G
L\$RPT 023066 G	L10060 042076	NXRERR 005176 G	PRI07 = 000340 G	RBPCRA 015145

RCVHIA	002250	G	SO.ION=	000040	TSTPTR	002262	G	T##SUB=	010075	T25TH	030524	
RCVLOA	002252	G	SO.IRD=	000100	TSTSET	017424	G	T##SW =	010001	T25WB	030242	
RDERR	005104		SO.IRW=	000004	TST25I	031462		T##TES=	010074	T25WDC	031411	
READ	000014		SO.ISP=	000200	TST26I	050637		T1	023650	G	T25WDE	030377
READY	000001		S1.ICE=	002000	TST27I	060633		T1.1	023710		T25WDR	030260
RECMG	002432	G	S1.IEO=	010000	TST28I	065251		T1.2	024410		T25WNG	030667
RECV	002200	G	S1.IFM=	001000	TTIBFR=	177562	G	T1.3	025050		T25WNH	031042
REGSAV	021054		S1.IHE=	000400	TTICSR=	177560	G	T1.4	025504		T26AM3	047536
REWIND	010444	G	S1.IID=	004000	TTIVEC=	000060	G	T1.5	026234		T26BA	050076
RMCHBE=	000167		S1.IIR=	020000	TTIBFR=	177566		T1.6	027162		T26BFR	045750
RMCHEN=	000200		S1.I2R=	040000	TTICSR=	177564		T1.7	027472		T26BF2	046070
RMMSGB=	000104		S1.PAR=	100000	TUV2A	002000	G	T2	031666	G	T26BOT	047125
RMMSGC=	000117		S2.ATI=	000010	T#ARGC=	000003		T2.1	031732		T26BSO	046070
RMPKTB=	000020		S2.BTI=	000004	T#CODE=	001130		T2.10	041246		T26BS1	046071
RMPKTE=	000027		S2.DIM=	000200	T#ERRN=	000656		T2.11	042114		T26CNT	046106
RMR	010000		S2.ILW=	000100	T#EXCP=	000000		T2.12	042752		T26CNU	046110
RMPACK	010540		S2.INR=	000020	T#FLAG=	000040		T2.13	043672		T26DAT	045740
SC	100000		S2.OUT=	000040	T#FREE=	066040		T2.14	044670		T26DLY	046114
SCE	020000		S2.UND=	000003	T#GMAN=	000000		T2.15	045240		T26DTA	047172
SCME	004711		TBLEND=	003030	G	T#HILI=	000007	T2.2	032634		T26EOT	047260
SDELAY	010340		TCOASC	006204		T#LAST=	000001	T2.3	033472		T26LON	050240
SEEK	000006		TCOCOD	006404		T#LOLI=	000000	T2.4	034356		T26LOO	031732
SELASC	021362		TEMP1	003064	G	T#LSYM=	010000	T2.5	035276		T26LOP	050322
SELDAT=	000004		TEMP2	003066	G	T#LTNO=	000004	T2.6	036044		T26LOQ	046736
SEL2	000002		TERCLS=	000016		T#NEST=	000000	T2.7	036674		T26LOR	046611
SETMAP	020304		TESTNO=	000004		T#NSO =	000000	T2.8	037536		T26NEF	046204
SETU	022310		TEXASC	006143		T#NS1 =	000005	T2.9	040400		T26NEQ	050560
SFFMSG	011756	G	TFCASC	006245		T#NS2 =	000002	T210FL	030274		T26OFL	047605
SFHERR	003603		TIMEXP	016436	(	T#PCNT=	000000	T25BFR	030130		T26PAC	045730
SFIERR	003550		TIMSGO	016464		T#PTAB=	010101	T25BF2	030250		T26PBP	050404
SFIMSG	011676	G	TINERR	011663		T#PTHV=	000001	T25BNC	030752		T26PK2	046040
SFPTBL	002134	G	TKB	177562		T#PTNU=	000001	T25BOT	030457		T26PK3	046060
SIFLAG	003106	G	TKS	177560		T#SAVL=	177777	T25BSO	030250		T26RB	046062
SIMSG	011630		TMPBFR	002576	G	T#SEGL=	177777	T25BS1	030251		T26RDF	046266
SKIPT	003332		TNAM	017620		T#SIZE=	000005	T25CNT	030270		T26RES	050650
SOFINI	016660	G	TPB	177566		T#SUBN=	000001	T25CN2	030266		T26RN	046076
SPACE	010144	G	TPS	177564		T#TAGL=	177777	T25CON	030262		T26RNC	047463
SPM1	065634		TRANST	002134	G	T#TAGN=	010103	T25DAT	030120		T26RRF	046335
SPM4	065700		TSBA	177776	G	T#TEMP=	000005	T25DLY	030272		T26RRG	046432
SPM6	065730		TSBAH	177777	G	T#TEST=	000004	T25L00	023710		T26RSZ	046112
SPM7	065760		TSBAL	177776	G	T#TSTM=	177777	T25NEF	031125		T26RT2	050742
SR0	177572		TSDB	177776	G	T#TSTS=	000001	T25NET	030613		T26RT3	051004
SR1	177574		TSDBH	177777	G	T##AU =	010031	T25OFL	031336		T26RWN	047414
SR2	177576		TSDBL	177776	G	T##AUT=	010033	T25PAC	030110		T26SC	046527
SR3	172516		TSFCOD	006744		T##CLE=	010034	T25PK2	030220		T26SSR	047017
SSR	000200		TSREJ	000006		T##DAT=	010102	T25PK3	030240		T26SZ	046066
STATCO	012254		TSSDEF	006314		T##DU =	010032	T25RB	030242		T26S2	046072
SVCGBL=	000000		TSSR	000000	G	T##HAR=	010076	T25RES	031500		T26S3	046074
SVCINS=	000000		TSSRBI	003400	G	T##HW =	010000	T25RI8	031205		T26TM	047337
SVCSUB=	000001		TSSRFO	006123		T##INI=	010030	T25RN	030256		T26TRL	050472
SVCTAG=	000000		TSSRH	000001	G	T##MSG=	010025	T25RT2	031572		T26VCK	050023
SVCTST=	000001		TSSX	003716		T##PC =	000001	T25RT3	031634		T26WB	046062
S#LSYM=	010000		TSTBLK	002720	G	T##PRO=	010027	T25RWN	031267		T26WDC	047750
SO.IDB=	000010		TSTCNT	002162	G	T##PTA=	010101	T25SSR	030316		T26WDD	047660
SO.IFB=	000002		TSTEND	017634		T##RPT=	010035	T25SZ	030246		T26WDE	047053
SO.IFP=	000001		TSTFLA	002260	G	T##SOF=	010077	T25S2	030252		T26WDF	046661
SO.ILD=	000020		TSTL00	017372	G	T##SRV=	010026	T25S3	030254		T26WNG	046116

T26WSS	050151	T27S2	055662	T28RES	065276	WC.IRW=	000004	XSOVCK=	000020
T27AM3	057177	T27S3	055664	T28RIB	063404	WC.IOT=	000100	XSOVLE=	004000
T27BA	057537	T27TIM	056644	T28RN	063366	WC.IIT=	000040	XSOVLK=	000004
T27BFR	055540	T27TM	057000	T28RRM	064657	WC.ISR=	000020	XS1CON	015257
T27BF2	055660	T27TRL	060133	T28RRN	064735	WF.IED=	000010	XS2CON	015324
T27BOT	056551	T27TSA	060374	T28RRP	065014	WF.IER=	000004	XS3CON	015371
T27BSO	055660	T27VCK	057464	T28RT2	065370	WF.IHI=	000200	XXCOMM	003070 G
T27BS1	055661	T27WB	055652	T28RT3	065432	WF.IRE=	000040	X1ALWA=	000000
T27CNT	055676	T27WDC	057411	T28RWN	064331	WF.IWF=	000020	X1FALS=	000040
T27CNU	055700	T27WDD	057321	T28SSR	064045	WF.IWR=	000100	X1OFFS=	000400
T27CON	055672	T27WDE	056462	T28SZ	063346	WF.ISR=	000002	X1TRUE=	000020
T27DAT	055530	T27WDF	056270	T28S2	063352	WF.IAR=	000001	X1COR=	020000
T27DLY	055702	T27WDR	055670	T28S3	063354	WRTCHR	010342 G	X1.DLT=	100000
T27DTA	060536	T27WNG	055704	T28TM	064254	WRTERR	005011	X1.MBZ=	017375
T27EOT	056721	T27WRF	060456	T28TMK	064605	WRTMSG	004754	X1.RBP=	000400
T27LON	057701	T27WSS	057612	T28VCK	064532	XFERAS	016624	X1.SPA=	040000
T27LOO	051106	T28BFR	063230	T28WB	063342	XNXM	017312	X1.UNC=	000002
T27LOP	057763	T28BF2	063350	T28WDC	064453	XORBFO	007432	X2.BUF=	000100
T27LOQ	056345	T28BOT	064207	T28WDE	064116	XORFOR	007550	X2.EXT=	000200
T27LOR	056220	T28BSO	063350	T28WDF	063707	XST0 =	000006 G	X2.OPM=	100000
T27NEF	060221	T28BS1	063351	T28WDR	063370	XST1 =	000010 G	X2.RCE=	040000
T27OFL	057246	T28CNT	063376	T3	051036 G	XST2 =	000012 G	X2.REV=	000077
T27PAC	055520	T28CNU	063400	T3.1	051106	XST3 =	000014 G	X2.SPA=	035400
T27PBP	060045	T28CON	063372	T3.2	051474	XST4 =	000016 G	X2.UNI=	000007
T27PK2	055630	T28DAT	063220	T3.3	052246	XS0BOT=	000002	X2.WCF=	002000
T27PK3	055650	T28DLY	063402	T3.4	053062	XS0CON	015212	X3.DCK=	000010
T27RB	055652	T28DTA	065154	T3.5	053756	XS0EOT=	000001	X3.MBZ=	000006
T27RDF	055772	T28DTR	065072	T4	061034 G	XS0IE =	000040	X3.MDE=	177400
T27RES	060654	T28IMV	063356	T4.1	061100	XS0ILA=	000400	X3.OPI=	000100
T27RN	055666	T28LOO	061100	UAM =	000200 G	XS0ILC=	001000	X3.REV=	000040
T27RNC	057124	T28LOQ	063764	UNITN =	002150 G	XS0LET=	020000	X3.RIB=	000001
T27RRF	056041	T28OFL	064400	UNREC =	000006	XS0MOT=	000200	X3.SPA=	000200
T27RT2	060746	T28PAC	063210	USI	004021	XS0NEF=	002000	X3.TRF=	000020
T27RT3	061010	T28PBP	063461	WAITF	017134 G	XS0ONL=	000100	X4.HSP=	100000
T27RWN	057055	T28PK2	063320	WC.IFA=	000200	XS0PED=	000010	X4.MBZ=	017400
T27SC	056136	T28PK3	063340	WC.IFE=	000002	XS0RLI=	010000	X4.RCE=	040000
T27SCF	060317	T28RB	063342	WC.IGO=	000001	XS0RLS=	040000	X4.TSM=	020000
T27SSR	056426	T28RDF	063544	WC.IRE=	000010	XS0TMK=	100000	X4.WRC=	000377
T27SZ	055656	T28RDG	063625						

. ABS. 066040 000
000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 29424 WORDS (115 PAGES)

DYNAMIC MEMORY: 20060 WORDS (77 PAGES)

ELAPSED TIME: 00:37:17

CZTKGA.BIC,CZTKGA/-SP=SVC/ML,CZTKGA