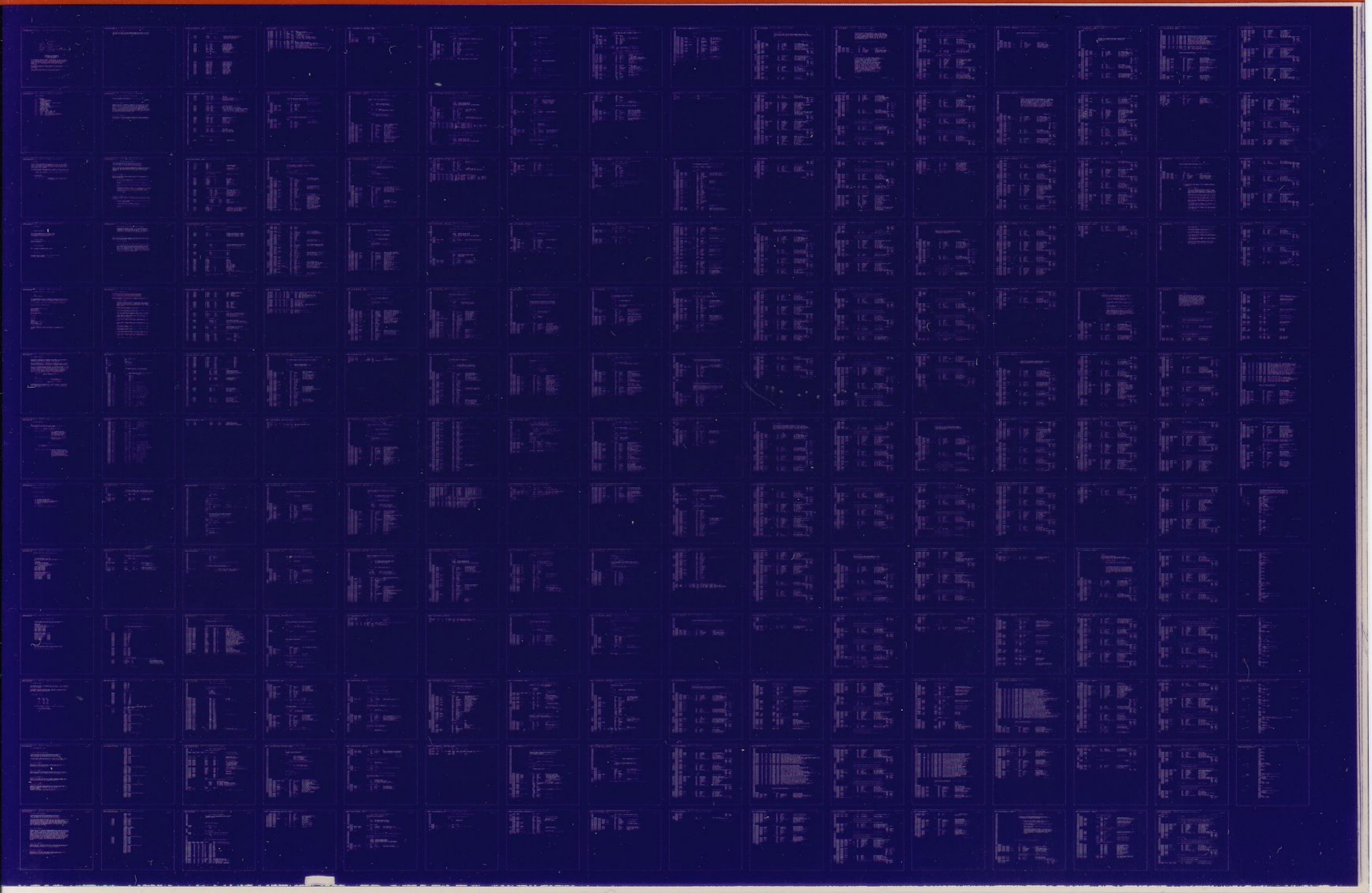
TK25

TK25 FRT END FUNC #4
CZTKHB0

AH-T782B-MC
1 OF 2 OCT 1985

digital
MADE IN USA



TK25

TK25 FRT END FUNC #4
CZTKHB0

AH-T782B-MC
2 OF 2 OCT 1985

digital
MADE IN USA

.REM\

IDENTIFICATION

PRODUCT ID: AC-T781B-MC

PRODUCT TITLE: CZTKHB TK25 FRT END FUNC #4

PRODUCT DATE: JUNE 1985

DEPARTMENT: TAPE AND OPTICAL DIAGNOSTIC ENGINEERING

AUTHOR: RAYMOND CHANG

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 - CZTKH
6.0 TEST DESCRIPTIONS - CZTKH
6.1 TEST 1 - WRITE TAPE MARK RETRY
6.2 TEST 2 - SKIP TAPE MARKS
6.3 TEST 3 - NO-OP ("CLEAN TAPE") AND INITIALIZE
6.4 TEST 4 - ERASE AND OPERATIONS INCOMPLETE
6.5 TEST 5 - OPERATIONS AT EOT

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

1.1 REVISION HISTORY

NEW RELEASE APRIL 1984

REV B	NOV 1985	INCREASED DELAY LOOPS IN WAITF ROUTINE SO SUPERVISOR COULD FIELD BREAK CALLS AT ITS LIMIT.

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 (XXDP+ VERSION 2.1)
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)

```
BOOTING UP XXDP-XM EXTENDED MONITOR


XXDP-XM EXTENDED MONITOR VERSION 2.1
BOOTED FROM DL0
28KW OF MEMORY
NON-UNIBUS SYSTEM

RESTART ADDR: 065570
THIS IS XXDP-XM.  TYPE "H" OR "H/L" FOR HELP.

.R CZTKHB
 - ------

CZTKHB.BIN

DRSXM-X0
CZTKH-B-0
CZTKHB TK-25 FRT END FUNC #4
UNIT IS TK-25
RESTART ADDRESS 141656
DR>START/FLAG:PNT:HOE
   ------------------
```

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) RESPONCE 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 FOUR 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.>

NOTE

THE FOLLOWING QUESTION IS ONLY ASKED FOR THE
CZTKH DIAGNOSTIC.

INHIBIT EOT CHECKING (REDUCES RUN TIME BY 22 MINUTES) (L) N?
<THIS WILL SIGNAL THE DIAGNOSTIC
SKIP END OF TAPE CHECKING. IF THE
OPERATOR IS CONVINCED THAT THERE
IS NO PROBLEM WITH EITHER THE
TRACK SWITCHING CAPABILITY OR
THE EOT DETECTION MECHANISM, THIS
TEST MAY BE SKIPPED TO REDUCE RUN
TIME.>

4.0 OPERATING INSTRUCTIONS - SAMPLE PRINTOUTS

4.1 SUCCESSUL RUN EXAMPLES

4.1.1 SUCCESSFUL RUN EXAMPLE - CZTKH -

```
TST: 001 WRITE TAPE MARK RETRY TEST
TST: 002 SKIP TAPE MARKS TEST
TST: 003 NO-OP ("CLEAN TAPE") AND INITIALIZE TEST
TST: 004 ERASE AND OPERATION INCOMPLETE TEST
TST: 005 TEST OF OPERATIONS AT EOT TEST
CZTKH 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: 005 OPERATIONS AT EOT TEST
CZTKH HRD ERR 00517 ON UNIT 00 TST 005 SUB 001 PC:054200
UNABLE TO CLEAR EOT INIDICATION (XSTO) BIT 0

TSSR=000311
TSSR CONTENTS ARE AMBIGUOUS
TSSR BITS SET: SSR,OFL,BITO
TERMINATION CLASS CODE=RECOVERABLE ERROR - TAPE
        POSITION ONE RECORD DOWN
*****CHECK TRANSPORT*****
PACKET ADDRESS=055510
PACKET WORD #0=140410
PACKET WORD #1=000003
PACKET WORD #2=000000
PACKET WORD #3=006654

MESSAGE BUFFER ADDRESS=055400
MESSAGE BUFFER CONTENTS:
MESSAGE BUFFER HEADER   =100020
DATA FIELD LENGTH       =000012
RESIDUAL BYTE COUNTER   =000000
XSTAT0 CONTENTS         =000311
XSTAT1 CONTENTS         =000000
XSTAT2 CONTENTS         =100000
XSTAT3 CONTENTS         =000040
```

ERROR MESSAGE EXAMPLE 2

```
CZTKH HRD ERR 00106 ON UNIT 00 TST 001 SUB 001 PC:024240
TSSR NOT CORRECT AFTER SPACE REVERSE DATA COMMAND

TSSR=100214
TSSR BITS SET: SC, SSR
TERMINATION CODE = UNRECOVERABLE ERROR
*****CHECK TRANSPORT*****
PACKET ADDRESS =026510
PACKET WORD #0 =141011
PACKET WORD #1 =065152
PACKET WORD #2 =000000
PACKET WORD #3 =000000

MESSAGE BUFFER ADDRESS  =026400
MESSAGE BUFFER CONTENTS:
MESSAGE BUFFER HEADER   =100022
DATA FIELD LENGTH       =000012
RESIDUAL BYTE COUNTER   =000000
XSTAT0 CONTENTS         =000312
XSTAT1 CONTENTS         =000000
XSTAT2 CONTENTS         =100000
XSTAT3 CONTENTS         =000141
```

ERROR MESSAGE EXAMPLE 3

```
CZTKH HRD ERR 00107 ON UNIT 00 TST 001 SUB 001 PC:024274
WRITE TAPE MARK RETRY AT BOT, FAILED TO SET NEF (XST0)

EXPD: 002312 RECV: 000312 XOR: 002000
```

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 - CZTKH

TEST NUMBER	N/I SECS.	DEF SECS.
1	180	180
2	113	113
3	11	11
4	120	120
5	1320	1320

THE TIMES REQUIRED TO RUN TESTS 1 THROUGH 5 IN ONE COMMAND:

Q.V. 29 MINS 4 SECONDS
DEFAULT 29 MINS 4 SECONDS

9.0 TEST DESCRIPTIONS - CZTKH

6.1 TEST 1 - WRITE TAPE MARK RETRY

```
**********************************************************************
* 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 PROPER OPERATION OF THE WRITE TAPE MARK RETRY COMMAND (SPACE REVERSE, ERASE, WRITE TAPE MARK). SUBTESTS ARE AS FOLLOWS:

6.1.1 TEST 1, SUBTEST 1: -

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

6.1.2 TEST 1, SUBTEST 2: -

VERIFIES THAT A WRITE TAPE MARK RETRY COMMAND ISSUED WHILE THE TAPE IS POSITIONED BEFORE THE FIRST RECORD, BUT NOT AT BOT, RESULTS IN TAPE STATUS ALERT TERMINATION, WITH THE REVERSE INTO BOT (RIB) STATUS BIT SET.

6.1.3 TEST 1, SUBTEST 3: -

VERIFIES THAT A WRITE TAPE MARK RETRY COMMAND TERMINATES PROPERLY AND WRITES THE TAPE MARK ONTO TAPE (BY ISSUING A READ REVERSE COMMAND AND CHECKING FOR TAPE STATUS ALERT TERMINATION AND TMK=1).

6.1.4 TEST 1, SUBTEST 4: -

VERIFIES THAT THE SPACE-REVERSE PORTION OF THE WRITE TAPE MARK RETRY OPERATION IS PERFORMED BY REWINDING THE TAPE, ISSUING SEVERAL WRITE TAPE MARK RETRY COMMANDS IN SUCCESSION, THEN ISSUING TWO SP
ACE RECORDS REVERSE
COMMANDS IN SUCCESSION. THE SECOND SPACE RECORDS REVERSE COMMAND SHOULD TERMINATE WITH REVERSE INTO BOT (RIB) STATUS SET.

6.2 TEST 2 - SKIP TAPE MARKS

```
*****************************************************************
* NOTE: THIS TEST 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 SKIP TAPE MARKS FORWARD AND SKIP TAPE MARKS REVERSE COMMANDS. PROPER OPERATION UNDER CONTROL OF ALL COMBINATIONS OF THE ENABLE SKIP TAPE MARKS STOP (ESS) AND ENABLE TAPE MARKS STOP OFF BOT (ENB) BITS SPECIFIED BY THE WRITE CHARACTERISTICS COMMAND. THE TEST CONSISTS OF THE FOLLOWING SUBTESTS (FOR EACH SUBTEST, THE TAPE IS FIRST WRITTEN WITH AN APPROPRIATE SERIES OF DATA RECORDS, AND/OR TAPE MARKS, AND/OR DOUBLE TAPE MARKS).

6.2.1 TEST 2, SUBTEST 1: -

VERIFIES THAT A SKIP TAPE MARKS FORWARD COMMAND WITH A TAPE MARK COUNT OF 1 OPERATES PROPERLY. THE TAPE IS FIRST REWOUND AND THEN WRITTEN WITH SEVERAL "FILES"; EACH FILE CONSISTS OF A NUMBER OF DATA RECORDS FOLLOWED BY A TAPE MARK. EACH DATA RECORD CONTAINS THE FILE NUMBER AND THE RECORD NUMBER WITHIN THE FILE SO THAT TAPE POSITION CAN BE SUBSEQUENTLY VERIFIED BY READING THE DATA. THE TAPE IS AGAIN REWOUND AND A SERIES OF TAPE SKIP MARK COMMANDS ARE ISSUED AND THE RESULTS (TAPE STATUS ALERT TERMINATION, TMK=1, STATUS, TAPE POSITION VIA READ COMMAND) IS CHECKED. PRIOR TO ISSUANCE OF EACH SKIP COMMAND, A WRITE CHARACTERISTICS COMMAND IS ISSUED TO SET UP THE ESS AND ENB CONTROL BITS. ALL COMBINATIONS OF ESS AND ENB ARE USED (00,01,10,11) ; OPERATION SHOULD BE THE SAME IN EACH CASE FOR THIS SUBTEST.

6.2.2 TEST 2, SUBTEST 2: -

VERIFIES THAT SKIP TAPE MARKS COMMAND WITH A TAPE MARK COUNT GREATER THAN 1 OPERATES PROPERLY. COUNTS OF 2, 3, 8, 32, 64, 256, AND 512 ARE TESTED. THE TESTING SEQUENCE IS SIMILAR TO THAT USED IN SUBTEST 1.

6.2.3 TEST 2, SUBTEST 3: -

VERIFIES THAT A SKIP TAPE MARKS REVERSE COMMAND ISSUED WHILE THE TAPE IS POSITIONED AT BOT CAUSES FUNCTION REJECT TERMINATION WITH THE NON-EXECUTABLE FUNCTION (NEF) ERROR BIT SET.

SEQ 0014

6.2.4 TEST 2, SUBTEST 4: -

VERIFIES THAT A SKIP TAPE MARKS REVERSE COMMAND ISSUED WHILE THE TAPE IS 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 - NO-OP ("CLEAN TAPE") AND INITIALIZE

THIS TEST VERIFIES PROPER OPERATION OF THE NO-OP ("CLEAN TAPE") AND INITIALIZE COMMAND. SUBTESTS ARE:

6.3.1 TEST 3, SUBTEST 1: -

VERIFIES THAT THE NO-OP COMMAND (CORRESPONDS TO THE CLEAN TAPE COMMAND) TERMINATES PROPERLY (NORMAL TERMINATION), STORES PROPER STATUS IN THE MESSAGE BUFFER (LIKE THE GET STATUS COMMAND), AND INDEED DOES NOT MOVE TAPE. THE TAPE IS FIRST REWOUND AND WRITTEN WITH THE SEQUENCED TEST RECORDS. IT IS THEN REWOUND AGAIN AND THE NO-OP COMMAND IS ISSUED. IT IS VERIFIED THAT THE TAPE IS STILL AT BOT AND THAT PROPER STATUS IS STORED. THE FIRST RECORD ON TAPE IS READ AND VERIFIED (TO CHECK THAT TAPE POSITION AND VERIFYING DATA WERE NOT CHANGED), THEN THE NO-OP COMMAND IS ISSUED AGAIN AND STATUS AND POSITION ARE VERIFIED.

6.3.2 TEST 3, SUBTEST 2: -

VERIFIES THAT THE INITIALIZE COMMAND OPERATES AS A NO-OP, ASSUMING NO MICRODIAGNOSTIC ERRORS ARE PRESENT (THEY WOULD HAVE ALREADY BEEN DETECTED IN OTHER TESTS). THE TEST SEQUENCE IS SIMILAR TO THAT USED IN SUBTEST 1.

6.4 TEST 4 - ERASE AND OPERATION INCOMPLETE

```
**********************************************************************
* 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 ERASE COMMAND OPERATES PROPERLY AND THAT THE VARIOUS OTHER TAPE MOTION COMMANDS TERMINATE WITH UNRECOVERABLE ERROR (TAPE POSITION LOST) AND OPERATION INCOMPLETE (OPI) STATUS WHEN THEY DO NOT ENCOUNTER ANY DATA ON THE TAPE. THE TEST CONSISTS OF THE FOLLOWING SUBTESTS:

6.4.1 TEST 4, SUBTEST 1: -

VERIFIES THAT AN ERASE COMMAND ISSUED WHEN THE TAPE IS POSITIONED AT BOT OPERATES PROPERLY AND ACTUALLY ERASES THE TAPE. THE FOLLOWING TEST SEQUENCE IS PERFORMED:

1. THE TAPE IS FIRST REWOUND, THEN SEVERAL TEST RECORDS ARE WRITTEN AND THE TAPE IS REWOUND AGAIN.

2. AN ERASE COMMAND IS ISSUED, WHICH SHOULD ERASE A NUMBER OF TEST RECORDS.

3. NORMAL TERMINATION IS VERIFIED AND POSITION IS CHECKED (BOT SHOULD BE 0).

4. A READ REVERSE COMMAND IS ISSUED. IT IS VERIFIED THAT THE COMMAND TERMINATES WITH TAPE STATUS ALERT, THAT THE REVERSE INTO BOT (RIB) STATUS BIT IS SET, AND THAT NO DATA IS TRANSFERRED. THIS DEMONSTRATES THAT NO DATA WAS ENCOUNTERED IN THE AREA ERASED BY THE ERASE COMMAND.

6.4.2 TEST 4, SUBTEST 2: -

VERIFIES THAT AN ERASE COMMAND, EXECUTED WHEN THE TAPE IS NOT POSITIONED AT BOT OPERATES PROPERLY AND DOES NOT CORRUPT PREVIOUS TAPE RECORDS. THE TEST SEQUENCE IS:

1. THE TAPE IS FIRST REWOUND, SEVERAL TEST RECORDS ARE WRITTEN, AND THE TAPE IS REWOUND AGAIN.

2. A SPACE RECORDS FORWARD COMMAND IS ISSUED TO MOVE THE TAPE OFF OF BOT AND SKIP OVER THE FIRST SEVERAL RECORDS.

3. AN ERASE COMMAND IS ISSUED, WHICH SHOULD ERASE A NUMBER OF TEST RECORDS.

4. NORMAL TERMINATION IS VERIFIED AND STATUS IS CHECKED.

5. A READ REVERSE COMMAND IS ISSUED. IT IS VERIFIED THAT NORMAL TERMINATION IS ACCOMPLISHED AND THAT THE DATA TRANSFERRED CORRESPONDS TO THAT FOR THE EXPECTED RECORD. THIS DEMONSTRATES THAT NO DATA WAS ENCOUNTERED IN THE AREA ERASED BY THE ERASE COMMAND, AND THAT THE PREVIOUS RECORD WAS NOT CORRUPTED.

6.4.3 TEST 4, SUBTEST 3: -

VERIFIES THAT THE TAPE MOTION COMMANDS, EXECUTED WHEN THE TAPE IS BLANK, RESULT IN UNRECOVERABLE ERROR TERMINATION AND OPERATION INCOMPLETE STATUS. THE FOLLOWING TEST SEQUENCE IS EXECUTED:

1. THE TAPE IS REWOUND.

2. 300 ERASE COMMANDS ARE ISSUED (ABOUT HALF-WAY DOWN FIRST TRACK).

3. IT IS VERIFIED THAT EACH OF THE FOLLOWING COMMANDS (ISSUED IN THE ORDER GIVEN) RESULTS IN UNRECOVERABLE ERROR TERMINATION WITH OPI=1; SPACE RECORDS REVERSE, SKIP TAPE MARKS REVERSE, READ REVERSE, REREAD PREVIOUS (OPP=0), REREAD PREVIOUS (OPP=1), REREAD NEXT (OPP=1), REREAD NEXT (OPP=0), READ NEXT, SKIP TAPE MARKS REVERSE, SKIP TAPE MARKS FORWARD, REVERSE SKIP TAPE MARKS FORWARD, SPACE RECORDS FORWARD, WRITE DATA RETRY.

6.5 TEST 5 - OPERATIONS AT EOT

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

THIS TEST VERIFIES THAT THE EOT STATUS IS HANDLED PROPERLY BY THE VARIOUS TAPE MOTION COMMANDS. THE FOLLOWING TEST SEQUENCE IS PERFORMED:

1. THE TAPE IS REWOUND.

2. WRITE DATA COMMANDS ARE REPEATEDLY ISSUED UNTIL TAPE STATUS ALERT TERMINATION IS SEEN WITH EOT=1. ERRORS OTHER THAN OCCASIONAL CORRECTABLE, OR UNCORRECTABLE DATA ERRORS CAUSE A FATAL ERROR REPORT. RECORDS WITH DATA ERRORS ARE RETRIED, SO THE TAPE ENDS UP WITH GOOD DATA.

3. ANOTHER WRITE DATA COMMAND IS ISSUED AND IT IS CHECKED THAT TAPE STATUS ALERT TERMINATION OCCURS, WITH EOT=1.

4. A WRITE TAPE MARK COMMAND IS ISSUED, AND IT IS CHECKED THAT TAPE STATUS ALERT TERMINATION OCCURS, WITH EOT=1.

5. A SKIP TAPE MARKS REVERSE COMMAND IS ISSUED, AND IT IS CHECKED THAT TAPE STATUS ALERT TERMINATION OCCURS WITH EOT=1, AND TMK=1.

6. A SPACE RECORDS REVERSE COMMAND, WITH A RECORD COUNT OF 1, IS ISSUED, AND IT IS CHECKED THAT TAPE STATUS ALERT TERMINATION OCCURS, WITH EOT=1, AND TMK=1.

7. A SPACE RECORDS REVERSE COMMAND, WITH A RECORD COUNT OF 1, IS ISSUED, AND IT IS CHECKED THAT NORMAL TERMINATION OCCURS WITH EOT=1.

8. A SPACE RECORDS FORWARD COMMAND, WITH A RECORD COUNT OF 1, IS ISSUED, AND IT IS CHECKED THAT NORMAL TERMINATION OCCURS, WITH EOT=1.

9. A READ REVERSE COMMAND IS ISSUED, AND IT IS CHECKED THAT NORMAL TERMINATION OCCURS WITH EOT=1.

10. A READ FORWARD COMMAND IS ISSUED, AND IT IS CHECKED THAT NORMAL TERMIANTION OCCURS, WITH EOT=1.

11. A SPACE RECORDS REVERSE COMMAND, WITH A RECORD COUNT OF 3, IS ISSUED, AND IT CHECKS THAT NORMAL TERMINATION OCCURS WITH EOT=0.

12. A SPACE RECORDS FORWARD COMMAND WITH A RECORD COUNT OF 3 IS ISSUED, AND IT IS CHECKED THAT NORMAL TERMINATION OCCURS WITH EOT=1.

13. A REWIND COMMAND IS ISSUED TO RETURN TO BOT.

```
                \

                                        .SBTTL  PROGRAM HEADER
                                        .MCALL  SVC
000000                                  SVC                     ; INITIALIZE SUPERVISOR MACROS
                                        .ENABLE LC
                                        .NLIST  BEX,CND
000000                                  .ENABL  AMA,ABS
        002000                          . = 2000
002000                                          BGNMOD  TUV2A
002000                                  TUV2A::

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


002000                                          POINTER BGNSW,BGNSFT,BGNAU,BGNDU,BGNRPT,BGNSETUP
002000                                          HEADER  CZTKH,B,0,655.,0
002000                                  L$NAME::                ;DIAGNOSTIC NAME
002000  103                                     .ASCII /C/
002001  132                                     .ASCII /Z/
002002  124                                     .ASCII /T/
002003  113                                     .ASCII /K/
002004  110                                     .ASCII /H/
002005  000                                     .BYTE   0
002006  000                                     .BYTE   0
002007  000                                     .BYTE   0
002010                                  L$REV::                 ;REVISION LEVEL
002010  102                             .ASCII  /B/
002011                                  L$DEPO::                ;0
002011  060                             .ASCII  /0/
002012                                  L$UNIT::                ;NUMBER OF UNITS
002012  000001                          .WORD   T$PTHV
002014                                  L$TIML::                ;LONGEST TEST TIME
002014  001217                          .WORD   655.
002016                                  L$HPCP::                ;POINTER TO H.W. QUES.
002016  060214                          .WORD   L$HARD
002020                                  L$SPCP::                ;POINTER TO S.W. QUES.
002020  060354                          .WORD   L$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$LADP::                ;DIAG. END ADDRESS
002026  060600                          .WORD   L$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$DTP::                 ;PTR. TO DISPATCH TABLE
002040  060562                          .WORD   L$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$DEVP::                ; POINTER TO DEVICE TYPE LIST
002060  003340                          .WORD   L$DVTYP
002062                          L$REPP::                ;PTR. TO REPORT CODE
002062  023060                          .WORD   L$RPT
002064                          L$EXP4::
002064  000000                          .WORD   0
002066                          L$EXP5::
002066  000000                          .WORD   0
002070                          L$AUT::                 ;PTR. TO ADD UNIT CODE
002070  022552                          .WORD   L$AU
002072                          L$DUT::                 ;PTR. TO DROP UNIT CODE
002072  022650                          .WORD   L$DU
002074                          L$LUN::                 ;LUN FOR EXERCISERS TO FILL
002074  000000                          .WORD   0
002076                          L$DESP::                ;POINTER TO DIAG. DESCRIPTION
002076  003346                          .WORD   L$DESC
002100                          L$LOAD::                ;GENERATE SPECIAL AUTOLOAD EMT
002100  104035                        EMT     E$LOAD
002102                          L$ETP::                 ;POINTER TO ERRTBL
002102  000000                          .WORD   0
002104                          L$ICP::                 ;PTR. TO INIT CODE
002104  021766                          .WORD   L$INIT
002106                          L$CCP::                 ;PTR. TO CLEAN-UP CODE
002106  023032                          .WORD   L$CLEAN
002110                          L$ACP::                 ;PTR. TO AUTO CODE
002110  022756                          .WORD   L$AUTO
002112                          L$PRT::                 ;PTR. TO PROTECT TABLE
002112  021756                          .WORD   L$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
```

```
                                .SBTTL  DEFAULT HARDWARE P-TABLE

                              ;++
                              ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
                              ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
                              ; IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
                              ;--
002122                                BGNHW   DFPTBL          ;DEFAULT HARD-P-TABLE
002122  000003                        .WORD   L10000-L$HW/2
002124                        L$HW::
002124                        DFPTBL::

002124  172522                        .WORD   172522          ; 2ND (OF 2) REGISTERS.
002126  000224                        .WORD   224             ; INTERRUPT VECTOR
002130  000240                        .WORD   PRI05           ; INTERRUPT PRIORITY.
002132                                ENDHW
002132                        L10000:
```

```
                                .SBTTL  SOFTWARE P-TABLE

                        ;++
                        ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
                        ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
                        ;--
002132                          BGNSW   SFPTBL
002132  000005                 .WORD   L10001-L$SW/2
002134                  L$SW::
002134                  SFPTBL::

002134  000000          TRANSTST::    .WORD   0       ;ENABLE RAM DUMP IF =1
002136  000000          NOITS::       .WORD   0       ; INHIBIT ITERATION OPTION.
                                                      ; ... 0 = ITERATE.
                                                      ; ...NZ = INHIBIT ITERATE.

002140  000000          EOTSEL::      .WORD   0       ;"INHIBIT EOT CHECKING (REDUCES TEST TIME
                                                      ;BY ABOUT 22 MINUTES"
002142  000031          LERRMAX::     .WORD   25.     ; LOCAL (PER TEST) ERROR LIMIT
002144  000310          GERRMAX::     .WORD   200.    ; GLOBAL (PER UNIT) ERROR LIMIT
002146                          ENDSW
002146                  L10001:
```

```
                                .SBTTL  GLOBAL EQUATES SECTION


                                .SBTTL  GLOBAL EQUATES SECTION

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

002146                          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                  HOE==  100000
768
769 002146                      KT11    ,,      ;DEFINE MEMORY MANAGEMENT REGISTERS
                        .SBTTL  MEMORY MANAGEMENT DEFINITIONS
                        ;*KT11 VECTOR ADDRESS
000250                  MMVEC=  250
                        ;*KT11 STATUS REGISTER ADDRESSES
177572                  SR0=    177572
177574                  SR1=    177574
177576                  SR2=    177576
172516                  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 REGISTORS
                        UDPDR0= 177620
                        UDPDR1= 177622
                        UDPDR2= 177624
                        UDPDR3= 177626
                        UDPDR4= 177630
                        UDPDR5= 177632
                        UDPDR6= 177634
                        UDPDR7= 177636
                        .ENDC
                        ;*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
                        ;*KERNFL "D" PAGE ADDRESS REGISTERS
                        KDPAR0= 172360
                        KDPAR1= 172362
                        KDPAR2= 172364
                        KDPAR3= 172366
                        KDPAR4= 172370
                        KDPAR5= 172372
                        KDPAR6= 172374
                        KDPAR7= 172376
                        .ENDC
```

```
                                .SBTTL  TK-25 REGISTER AND PACKET DEFINITIONS

                        ;
                        ; SOME GENERAL EQUATES.
                        ;

000004                  ERRVEC==        4               ; POINTER TO ERROR VECTOR FOR BUS TIME OUT.
000060                  TTIVEC==        60              ; INTERRUPT VECTOR FOR CONSOLE INPUT
177560                  TTICSR==        177560          ; BUS ADDRESS OF CONSOLE INPUT
177562                  TTIBFR==        177562          ; CONSOLE INPUT DATA BUFFER

                        ;+
                        ;BIT DEFINITIONS FOR TSSR REGISTER
                        ;-

100000                  SC=     BIT15                   ;SPECIAL CONDITION
040000                  BIE=    BIT14                   ;BUS INTERFACE ERROR
020000                  SCE=    BIT13                   ;SANITY CHECK ERROR
010000                  RMR=    BIT12                   ;MODIFICATION REFUSED
004000                  NXM=    BIT11                   ;NONEXISTANT MEMORY ERROR
002000                  NBA=    BIT10                   ;NEED BUFFER ADDRESS
001400                  HIADDR= BIT9!BIT8               ;EXTENDED ADDRESS BITS
000200                  SSR=    BIT7                    ;SUB SYSTEM READY
000100                  OFL=    BIT6                    ;OFF LINE BIT
000060                  FATERR= BIT4!BIT5               ;FATAL TERMINATION ERROR CODES
000016                  TERCLS= BIT3!BIT2!BIT1          ;TERMINATION CODES


                        ;+
                        ;
                        ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 0
                        ;(XST0)
                        ;
                        ;-

100000                  XSOTMK= BIT15                   ;TAPE MARK DETECTED
040000                  XSORLS= BIT14                   ;RECORD LENGTH SHORT
020000                  XSOLET= BIT13                   ;LOGICAL END OF TAPE
010000                  XSORLL= BIT12                   ;RECORD LENGTH LONG
004000                  XSOWLE= BIT11                   ;WRITE LOCK ERROR
002000                  XSONEF= BIT10                   ;NON EXECUTABLE FUNCTION
001000                  XSOILC= BIT9                    ;ILLEGAL COMMAND
000400                  XSOILA= BIT8                    ;ILLEGAL ADDRESS
000200                  XSOMOT= BIT7                    ;TAPE IN MOTION
000100                  XSOONL= BIT6                    ;TRANSPORT ON LINE
000040                  XSOIE=  BIT5                    ;INTERRUPT ENABLE
000020                  XSOVCK= BIT4                    ;VOLUME CHECK BIT
000010                  XSOPED= BIT3                    ;PHASE ENCODED DRIVE
000004                  XSOWLK= BIT2                    ;WRITE LOCKED
000002                  XSOBOT= BIT1                    ;BEGINNING OF TAPE
000001                  XSOEOT= BIT0                    ;END OF TAPE


                        ;+
                        ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 1
                        ;(XST1)
                        ;-
```

```
                        X1.DLT  = BIT15                 ;DATA LATE
                        X1.SPARE= BIT14                 ;NOT USED
                        X1.COR  = BIT13                 ;CORRECTABLE DATA ERROR
                        X1.MBZ  = BIT12+BIT11+BIT10+BIT9+BIT7+BIT6+BIT5+BIT4+BIT3+BIT2+BIT0 ;ALWAYS 0
                        X1.RBP  = BIT8                  ;READ BUS PARITY ERROR
                        X1.UNC  = BIT1                  ;UNCORRECTABLE DATA OR HARD ERROR

                        ;+
                        ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 2
                        ;(XST2)
                        ;-
                        X2.OPM  = BIT15                 ;OPERATION IN PROGRESS (TAPE MOVING)
                        X2.RCE  = BIT14                 ;RAM CHECKSUM ERROR
                        X2.SPARE= BIT13+BIT12+BIT11+BIT9+BIT8   ;NOT USED BY TK-25 (ALWAYS=0)
                        X2.WCF  = BIT10                 ;WRITE CLOCK FAILURE (FIFO NOT EMPTIED BY TRANSPORT)
                        X2.EXTF = BIT7                  ;IF WRITE CHAR CMD THEN = EXTENDED FEATURES ENABLED
                        X2.BUFE = BIT6                  ;IF WRITE CHAR CMD THEN = BUFFERING ENABLED
                        X2.REV  = 000077                ;IF WRITE CHAR CMD THEN = MICROCODE REVISION LEVEL
                        X2.UNIT = BIT2+BIT1+BIT0        ;IF GET STATUS THEN = CURRENTLY SELECTED UNIT NO.

                        ;+
                        ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 3
                        ;(XST3)
                        ;-
                        X3.MDE  = 177400                ;MICRO-DIAGNOSTIC ERROR CODE
                        X3.SPARE= BIT7                  ;NOT USED BY TK-25
                        X3.OPI  = BIT6                  ;OPERATION INCOMPLETE
                        X3.REV  = BIT5                  ;REVERSE
                        X3.TRF  = BIT4                  ;TRANSPORT RESPONSE FAILURE
                        X3.DCK  = BIT3                  ;DENSITY CHECK
                        X3.MBZ  =BIT2+BIT1              ;NOT USED ALWAYS 0
                        X3.RIB  = BIT0                  ;REVERSE INTO BOT

                        ;+
                        ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 4
                        ;(XST4)
                        ;-
                        X4.HSP  = BIT15                 ;HIGH SPEED
                        X4.RCE  = BIT14                 ;RETRY COUNT EXCEEDED
                        X4.TSM  = BIT13                 ;TRANSPORT SPECIAL MODE
                        X4.MBZ  = BIT12+BIT11+BIT10+BIT9+BIT8   ;NOT USED ALWAYS 0
                        X4.WRC  = 000377                ;WRITE RETRY COUNT FIELD


                        ;+
                        ;
                        ;TSSR TERMINATION CODES (BIT 0-2)
                        ;
                        ;-

                                TSREJ=  3*2                     ;COMMAND REJECTED
                                UNREC=  6                       ;UNRECOVERABLE ERROR

                        ;+
                        ;
                        ;DEVICE REGISTER OFFSETS
                        ;
```

```
                        ;-
                        
177776                  TSBA==  -2
177776                  TSBAL== -2
177776                  TSDB==  -2                      ;TSDB/TSBA REGISTER
177776                  TSDBL== -2                      ;TSDB/TSBA REGISTER
177777                  TSBAH== -1
177777                  TSDBH== -1                      ;TSDB/TSBA REGISTER HIGH BYTE
000000                  TSSR==  0                       ;TSSR REGISTER
000001                  TSSRH== 1                       ;TSSR REGISTER HIGH BYTE

                        ;+
                        ; TSDB ADDRESS BIT DEFINITIONS
                        ;-
000003                  A1716   = BIT1+BIT0             ;ADDRESS BITS 17;16 ARE IN 1;0

                        ;+
                        ; COMMAND DEFINITIONS
                        ;-
000017                  P.GETSTAT       = 17            ;GET STATUS
000013                  P.INIT          = 13            ;INITIALIZE
000012                  P.CONTROL       = 12            ;CONTROL COMMANDS
000011                  P.FORMAT        = 11            ;FORMAT
000010                  P.POSITION      = 10            ;POSITION
000006                  P.WRTSUB        = 6             ;SUBSYSTEM WRITE
000005                  P.WRITE         = 5             ;WRITE
000004                  P.WRTCHAR       = 4             ;WRITE CHARACTERISTICS
000001                  P.READ          = 1             ;READ

                        ;+
                        ; COMMAND PACKET HEADER WORD BIT DEFINITIONS
                        ;-
100000                          P.ACK   = BIT15                 ;BUFFER AVAIL FOR CONTROLLER
040000                          P.CVC   = BIT14                 ;CLEAR VOLUME CHECK
020000                          P.OPP   = BIT13                 ;REVERSE SEQUENCE OF DATA BITS
010000                          P.SWB   = BIT12                 ;SWAP BYTES IN MEMORY
007400                          P.MODE  = BIT11!BIT10!BIT9!BIT8 ;EXTENDED COMMAND MODE FIELD
000200                          P.IE    = BIT7                  ;INTERRUPT ENABLE
000140                          P.FMT= BIT6!BIT5                ;PACKET HEADER TYPE (ALWAYS=0)
000037                          P.CMD   = 37                    ;MAJOR COMMAND FIELD
                        ;+
                        ; CONTROL COMMAND MODE CODES
                        ;-
000000                          PC.RELEASE      = 0*256.        ;RELEASE BUFFER
000400                          PC.REWIND       = 1*256.        ;REWIND
001000                          PC.NOOP         = 2*256.        ;NO-OP
002000                          PC.IEREW        = 4*256.        ;REWIND IMMEDIATE INTERRUPT
002400                          PC.ERASE        = 5*256.        ;SECURITY ERASE

                        ;+
                        ; CONTROLLER RAM DEFINITIONS
                        ;-
000167                  RMCHBEG = 167                   ;CHARACTERISTICS IO DATA BEGIN RAM ADDRESS
000200                  RMCHEND = 200                   ;CHARACTERISTICS IO DATA END RAM ADDRESS
000020                  RMPKTBEG= 20                    ;COMMAND PACKET BEGIN RAM ADDRESS
000027                  RMPKTEND= 27                    ;COMMAND PACKET END RAM ADDRESS
000104                  RMMSGBEG= 104                   ;MESSAGE BUFFER BEGIN RAM ADDRESS
```

```
                        RMMSGEND= 117                   ;MESSAGE BUFFER END RAM ADDRESS
                        ;+
                        ;
                        ;REGISTER DEFINITIONS IN THE MESSAGE BUFFER
                        ;
                        ;-

000006                  XST0==  6                       ;EXTENDED STATUS REGISTER 0 (WORD 4)
000010                  XST1==  8.                      ;EXTENDED STATUS REGISTER 1 (WORD 5)
000012                  XST2==  10.                     ;EXTENDED STATUS REGISTER 2 (WORD 6)
000014                  XST3==  12.                     ;EXTENDED STATUS REGISTER 3 (WORD 7)
000016                  XST4==  14.                     ;EXTENDED STATUS REGISTER 4 (WORD 8)


                        ;+
                        ;
                        ;OFFSETS TO WORD LOCATIONS IN PACKET DEFINITIONS
                        ;
                        ;-

000002                          PKLOW   = 2             ;LOW ORDER CHARACTERISTIC DATA POINTER
000004                          PKHI    = 4             ;HIGH ORDER CHARACTERISTIC DATA POINTER
000006                          PKBCNT  = 6             ;NUMBER OF BYTES IN DATA PACKET

000010                          EXBCNT=10               ;NUMBER OF BYTES IN EXTENDED DATA PACKET

                        ;+
                        ;DATA PACKET OFFSETS FOR WRITE SUBSYTEM COMMAND
                        ;-
000000                          BSEL0   = 0             ;BYTE 0
000001                          BSEL1   = 1             ;BYTE 1
000002                          SEL2    = 2             ;WORD 2
000004                          SELDATA = 4             ;WORD 3

                        ;+
                        ;BSEL0 SELECT CODES FOR WRITE SUBSYSTEM COMMAND
                        ;-
000000                  PW.NOP          = 0             ;NO-OP
000001                  PW.RDRAM        = 1             ;READ RAM
000002                  PW.WTRAM        = 2             ;WRITE RAM
000003                  PW.RFIFO        = 3             ;READ FIFO
000004                  PW.WFIFO        = 4             ;WRITE FIFO
000005                  PW.RDSTAT       = 5             ;READ STATUS
000006                  PW.WCTL         = 6             ;WRITE TAPE CONTROL
000007                  PW.WFMT         = 7             ;WRITE TAPE FORMAT
000010                  PW.WMISC        = 10            ;WRITE MISCELLANEOUS
000011                  PW.WNPR         = 11            ;WRITE NPR CONTROL
000020                  PW.D22          = 20            ;DO MICROTEST 22
000021                  PW.D11          = 21            ;DO MICROTEST 11
000022                  PW.D13          = 22            ;DO MICROTEST 13
000023                  PW.NO1311       = 23            ;DISABLE MICROTEST 11 AND 13
000024                  PW.RDEXT        = 24            ;READ EXT. TAPE STATUS (NOT SUPPORTED BY ALL TRANSPO

                        ;+
                        ;BSEL1 CODES FOR WRITE TAPE CONTROL
                        ;-
000200                  WC.IFAD         = BIT7          ;IFAD - FORMATTER ADDRESS
```

```
1002    000100          WC.IOTAD        = BIT6          ;ITAD0  - TRANSPORT ADDRESS BIT 0
1003    000040          WC.I1TAD        = BIT5          ;ITAD1  - TRANSPORT ADDRESS BIT 1
1004    000020          WC.I5RESV       = BIT4          ;IRESV5 - RESERVED #5
1005    000010          WC.IREW         = BIT3          ;IREW   - REWIND
1006    000004          WC.IRWU         = BIT2          ;IRWU   - REWIND AND UNLOAD
1007    000002          WC.IFEN         = BIT1          ;IFEN   - FORMATTER ENABLE
1008    000001          WC.IGO          = BIT0          ;GO
1009
1010                    ;+
1011                    ;BSEL1 CODES FOR WRITE FORMAT
1012                    ;-
1013    000200          WF.IHISP        = BIT7          ;IHISP  - HIGH SPEED
1014    000100          WF.IWRT         = BIT6          ;IWRT   - WRITE
1015    000040          WF.IREV         = BIT5          ;IREV   - REVERSE
1016    000020          WF.IWFM         = BIT4          ;IWFM   - WRITE FILE MARK
1017    000010          WF.IEDIT        = BIT3          ;IEDIT  - EDIT
1018    000004          WF.IERASE       = BIT2          ;IERASE - ERASE
1019    000002          WF.I3RESV       = BIT1          ;IRESV3 - RESERVED #3
1020    000001          WF.I4RESV       = BIT0          ;IRESV4 - RESERVED #4
1021
1022
1023                    ;+
1024                    ;BSEL1 CODES FOR WRITE MISCELLANEOUS SUBCOMMAND
1025                    ;-
1026    000200          MS.EXT          = BIT7          ;INVERT SENSE OF EXTENDED FEATURES SWITCH
1027    000020          MS.RSFIFO       = BIT4          ;RESET FIFO AND INPUT PARITY ERRORR
1028    000010          MS.RSTAPE       = BIT3          ;RESET TAPE STATUS IN 2 FLIP-FLOPS
1029    000006          MS.ATTN         = BIT2!BIT1     ;ATTENTION TRIGGER FIELD
1030    000001          MS.RSD          = BIT0          ;RESET TIMER A,B THEN DELAY TIMES IN SEL2
1031                    ;+
1032                    ; MS.ATTN SUBCODES
1033                    ;-
1034    000000                  MSA.NOP = 0*2           ;NO-OP (NOTHING TRIGGERED)
1035    000002                  MSA.VOL = 1*2           ;SIMULATE ON-LINE/OFF-LINE TRANSISTION
1036    000004                  MSA.NRAM= 2*2           ;FORCE NON-FATAL RAM ERROR (FORCES ERRCODE 54)
1037    000006                  MSA.FRAM= 3*2           ;FORCE FATAL RAM ERROR (CAUSES SCE TO SET)
1038                    ;+
1039                    ; WRITE SUBSYSTEM WRITE NPR BSEL1 BIT DEFINITIONS
1040                    ;-
1041    000200          NP.IR           = BIT7          ;INTERRUPT REQUEST (0-1 TRANSITION)
1042    000100          NP.OUT          = BIT6          ;TAPE DATA DIRECTION OUT (0= IN)
1043    000040          NP.LOOP         = BIT5          ;ENABLE TRANSPORT LOOPBACK
1044    000020          NP.WRP          = BIT4          ;WRITE CORRECT PARITY (SET=0 TO WRITE WRONG)
1045                    ;+
1046                    ; READ STATUS MESSAGE BUFFER BIT DEFINITIONS
1047                    ;-
1048
1049    000200          S2.DIM          = BIT7          ;WORD #9 BYTE 2 DATA IN MISS
1050    000100          S2.ILW          = BIT6          ;               ILW H
1051    000040          S2.OUTRDY       = BIT5          ;               OUT RDY H
1052    000020          S2.INRDY        = BIT4          ;               IN RDY H
1053    000010          S2.ATIMR        = BIT3          ;               TIMER A FLAG H
1054    000004          S2.BTIMR        = BIT2          ;               TIMER B FLAG H
1055    000003          S2.UNDEF        = BIT1+BIT0     ;(UNDEFINED)
1056    100000          S1.PARIN        = BIT15         ;WORD #8 BYTE 1 PARIN H
1057    040000          S1.I2RESV       = BIT14         ;               IRESV2
1058    020000          S1.I1RESV       = BIT13         ;               IRESV1
```

```
1059    010000          S1.IEOT     = BIT12         ;               IEOT L
1060    004000          S1.IIDENT   = BIT11         ;               IIDENT H
1061    002000          S1.ICER     = BIT10         ;               ICER H
1062    001000          S1.IFMK     = BIT9          ;               IFMK H
1063    000400          S1.IHER     = BIT8          ;               IHER H
1064    000200          S0.ISPEED   = BIT7          ;WORD #8 BYTE 0 ISPEED H
1065    000100          S0.IRDY     = BIT6          ;               IRDY L
1066    000040          S0.IONL     = BIT5          ;               IONL L
1067    000020          S0.ILDP     = BIT4          ;               ILDP L
1068    000010          S0.IDBY     = BIT3          ;               IDBY L
1069    000004          S0.IRWD     = BIT2          ;               IRWD L
1070    000002          S0.IFBY     = BIT1          ;               IFBY L
1071    000001          S0.IFPT     = BIT0          ;               IFPT L
1072                    ;       SPECIAL KEYBORD STUFF FOR MOVER PROGRAM
1073
1074    177560              TKS     =177560         ;KEYBOARD STATUS REGISTER
1075    177562              TKB     =177562         ;KEYBOARD DATA REGISTER
1076    177564              TPS     =177564         ;CONSOLE PRINTER STATUS REGISTER
1077    177566              TPB     =177566         ;CONSOLE PRINTER DATA REGISTER
1078    007776              HIMEM   =007776         ;HIGH MEMORY MASK VALUE
1079                    ;       CONTROLLER DEFINITIONS
1080                    ;
1081                    ;
1082    174400              CSR     =174400         ;STATUS AND CONTROL REGISTER
1083    174402              BAR     =174402         ;DL ADDRESS REGISTER
1084    174404              DAR     =174404         ;PLATTER ADDRESS
1085    174406              MPR     =174406         ;MULTIPURPOSE REGISTER
1086                    ;
1087                    ;
1088                    ;
1089                    ;
1090                    ;
1091                    ;       CONTROLLER COMMANDS
1092                    ;
1093                    ;
1094    000004              DLGETS  =4              ;GET STATUS COMMAND
1095    000006              SEEK    =6              ;SEEK TRACK AND HEAD SELECT
1096    000010              DLRDHD  =10             ;READ SECTOR HEADER
1097    000014              READ    =14             ;READ COMMAND
1098    000016              DLRDNH  =16             ;READ SECTOR NO HEADER CHECK
1099                    ;
1100                    ;
1101                    ;
1102                    ;
1103                    ;
1104                    ;
1105    000001              READY   =1              ;DRIVE READY BIT IN STATUS REG.
1106    000013              DLSR    =13             ;STATUS AND RESET
1107    177730              DLERR   =177730         ;MASK FOR COVER OPEN
1108    000006              DLUN    =6              ;HEADS UNLOADED
1109    000177              DLCYL   =000177         ;MASK FOR CYLINDER ADDRESS
1110    100200              DLDNER  =100200         ;DONE SET OR ERROR SET BITS
1111                    ;
1112                    ;
1113                    ;
1114                    ;
1115                    ;       ROMBASE =       MOVER           ;START OF THE BOOT ROM 00000
```

```
1116        177560                  TTICSR  =       177560          ;KEYBOARD INPUT STATUS
1117        177562                  TTIBFR  =       177562          ;KEYBOARD DATA REGISTER
1118        177564                  TTOCSR  =       177564          ;CONSOLE PRINTER STATUS REGISTER
1119        177566                  TTOBFR  =       177566          ;CONSOLE PRINTER DATA REGISTER
1120
```

```
1122                            .SBTTL  SPECIAL MACROS AND OPDEFS.
1123
1124
1125                    ;+
1126                    ;SAVE GENERAL REGS 1 TO 5
1127                    ;-
1128
1129                            .MACRO  SAVREG
1130                            JSR     R5,REGSAV
1131                            .ENDM
1132
1133                    ;+
1134                    ; MACRO TO FORCE AN ERROR
1135                    ;-
1136                    .MACRO  FORCERROR       TAG,NOTSSR
1137                    .NLIST
1138                    .IIF NDF LISTALL, .NLIST
1139                    .LIST
1140                    .IF B NOTSSR
1141                            MOV     TSSR(R5),R1             ;READ TSSR
1142                    .ENDC
1143                            MOV     FORCER,FORCER           ;IS FORCER SET? (LEAVE C BIT ALONE)
1144                            BNE     TAG                     ;BR IF YES
1145                    .NLIST
1146                    .IIF NDF LISTALL, .LIST
1147                    .LIST
1148                    .ENDM
1149
1150                    ;+
1151                    ; MACRO TO FORCE AN EXIT TO AVOID SECTION ITERATIONS
1152                    ;       WILL EXIT TO A LABEL IF FORCER IS NEGATIVE
1153                    ;       SO TO FORCE ERRORS AND EXIT ON 1 ERROR SET
1154                    ;       FORCER TO 177777
1155                    ;       TO FORCE ERRORS AND ITERATIONS SET FORCER TO 1.
1156                    ;-
1157                    .MACRO  FORCEXIT        TAG
1158                    .NLIST
1159                    .IIF NDF LISTALL, .NLIST
1160                    .LIST
1161                            MOV     FORCER,FORCER           ;IS FORCER NEGATIVE?
1162                            BMI     TAG                     ;BR IF YES
1163                    .NLIST
1164                    .IIF NDF LISTALL, .LIST
1165                    .LIST
1166                    .ENDM
1167                    ;+
1168                    ; MACRO TO INCREMENT ERROR COUNTS
1169                    ;-
1170                    .MACRO  NEXT.ERRNO
1171                    .NLIST
1172                    ;;;.IIF NDF LISTALL, .NLIST
1173                            ERRNO=ERRNO+1
1174                    ;;;.IIF NDF LISTALL, .LIST
1175                    .LIST
1176                    .ENDM
1177
1178                    ;+
```

```
                        ;MACRO TO PERFORM XOR
                        ;-

                                .MACRO  XOR     A,B
                                MOV     A,-(SP)
                                BIC     B,(SP)
                                BIC     A,B
                                BIS     (SP)+,B
                                .ENDM

        000000                  EN=0                            ; INITIALIZE ERROR NUMBER
                                .SBTTL  FORCER  - FORCE ERROR FLAG

                        ;
                        ; THE FOLLOWING LOCATIONS MAY BE PATCHED BY THE USER
                        ; TO OBTAIN THE RESULTS DESCRIBED FOR EACH.
                        ;

002146  000000          FORCER::        0       ; FORCE TYPE ALL HARD ERRORS (THE ONES CALLED -
                                        ; - BY THE MACRO "IFERROR").  AN ERROR NEED NOT -
                                        ; - EXIST, JUST ASSUME AND TYPE THE MESSAGE.
```

```
 1204                                   .SBTTL  GLOBAL DATA SECTION
 1205
 1206                                   ;++
 1207                                   ;THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
 1208                                   ;IN MORE THAN ONE TEST.
 1209                                   ;--
 1210
 1211                                   ;
 1212                                   ;THE FOLLOWING DATA ARE SET FOR EACH UNIT AT INIT TIME.
 1213                                   ;SINGLE UNIT DEFAULTS (LISTED) ARE IN THE DEFAULT P-TABLE.
 1214                                   ;
 1215 002150  000000                   EPRTSW::        .WORD   0               ;PRINT SWITCH
 1216 002152  000000                   UNITN::         .WORD   0               ;UNIT # UNDER TEST.
 1217 002154  000000                   QVP::           .WORD   0               ;QUICK VERIFY FLAG.
 1218 002156  000000                   CSRADDR::       .WORD   0               ;ADDRESS OF CSR FOR CURRENT DEVICE
 1219 002160  000224                   IVEC::          .WORD   224             ;INTERRUPT VECTOR
 1220 002162  000200                   IPRI::          .WORD   PRI04           ;INTERRUPT PRIORITY.
 1221 002164  000000                   TSTCNT::        .WORD   0               ;NUMBER OF TESTS RUN IN THIS PASS
 1222 002166  000000                   LOOPCNT::       .WORD   0               ;REMAINING ITERATION COUNT FOR TEST
 1223 002170  000000                   DEVCNT::        .WORD   0               ;NUMBER OF DEVICE UNDER TEST
 1224 002172  000000                   FATFLG::        .WORD   0               ;SET IF FATAL ERROR IS DETECTED IN TEST
 1225 002174  000000                   INTRECV::       .WORD   0               ;SET IF TAPE INTERRUPT WAS RECEIVED
 1226 002176  000000                   BENBSW::        .WORD   0               ;BUFFER ENABLE SWITCH SW 0=OFF;1=ON
 1227 002200  000000                   EXPD::          .WORD   0               ;EXPECTED RAM DATA FOR PRAMPKT ROUTINE
 1228 002202  000000                   RECV::          .WORD   0               ;RECEIVED RAM DATA FOR PRAMPKT ROUTINE
 1229 002204  000000                   ERRHI::         .WORD   0               ;HIGH ADDRESS MEMORY ERROR
 1230 002206  000000                   ERRLO::         .WORD   0               ;LOW ADDRESS MEMORY ERROR
 1231 002210                           RAMDATA::       .BLKW   16.             ;DATA READ FROM RAM PACKET OR MESSAGE BUF AREA
 1232 002250  000000                   RAMSIZ::        .WORD   0               ;RAM DATA SIZE FOR PRAMPKT ROUTINE
 1233 002252  000000                   RCVHIADD::      .WORD   0               ;RECEIVED BUFFER HIGH ADDRESS
 1234 002254  000000                   RCVLOADD::      .WORD   0               ;RECEIVED BUFFER LOW ADDRESS
 1235 002256  000000                   COUNT::         .WORD   0               ;TEST COUNT PATTERN
 1236 002260  000000                   DATA::          .WORD   0               ;TEST DATA
 1237 002262  000000                   TSTFLAG::       .WORD   0               ;TEST FLAG WORD
 1238 002264  000000                   TSTPTR::        .WORD   0               ;TSTBLK POINTER
 1239 002266  000000                   PRMNO::         .WORD   0               ;PRINT ROUTINE TEMP
 1240 002270                           EXPMSG::        .BLKB   100.            ;EXPECTED MESSAGE BUFFER DATA
 1241 002434                           RECMSG::        .BLKB   100.            ;RECEIVED MESSAGE BUFFER DATA
 1242 002600                           TMPBFR::        .BLKB   80.             ;TEMPORARY STORAGE FOR PRINT
 1243 002720  000000                   MESBFA::        .WORD   0               ;STORES ADDRESS OF MESSAGE BUFFER FOR ERR PRT
 1244
 1245 002722  000000                   FLLTSW::        .WORD   0               ;0=1ST PASS, NON-ZERO= OTHER (FAULT MES)
```

```
1247                                    .SBTTL  TSTBLK  - TEST DATA TABLE
1248
1249                                ;+
1250                                ;
1251                                ;THIS TABLE CONTAINS TEST DATA USED IN SEVERAL TESTS
1252                                ;
1253                                ;IN SEQUENCE THE DATA IS:
1254                                ;
1255                                ;       ALL ZEROS
1256                                ;       ALL ONES
1257                                ;       WALKING ONES
1258                                ;       WALKING ZEROS
1259                                ;       ALTERNATING ONES AND ZEROS
1260                                ;
1261                                ;-
1262
1263 002724                        TSTBLK::
1264 002724  000000                     .WORD   0                       ;ALL ZEROS
1265 002726  177777                     .WORD   177777                  ;ALL ONES
1266 002730  000001                     .WORD   BIT0                    ;DATA FOR WALKING ONES
1267 002732  000002                     .WORD   BIT1
1268 002734  000004                     .WORD   BIT2
1269 002736  000010                     .WORD   BIT3
1270 002740  000020                     .WORD   BIT4
1271 002742  000040                     .WORD   BIT5
1272 002744  000100                     .WORD   BIT6
1273 002746  000200                     .WORD   BIT7
1274 002750  000400                     .WORD   BIT8
1275 002752  001000                     .WORD   BIT9
1276 002754  002000                     .WORD   BIT10
1277 002756  004000                     .WORD   BIT11
1278 002760  010000                     .WORD   BIT12
1279 002762  020000                     .WORD   BIT13
1280 002764  040000                     .WORD   BIT14
1281 002766  100000                     .WORD   BIT15
1282 002770  177776                     .WORD   ^CBIT0                  ;DATA FOR WALKING ZEROS
1283 002772  177775                     .WORD   ^CBIT1
1284 002774  177773                     .WORD   ^CBIT2
1285 002776  177767                     .WORD   ^CBIT3
1286 003000  177757                     .WORD   ^CBIT4
1287 003002  177737                     .WORD   ^CBIT5
1288 003004  177677                     .WORD   ^CBIT6
1289 003006  177577                     .WORD   ^CBIT7
1290 003010  177377                     .WORD   ^CBIT8
1291 003012  176777                     .WORD   ^CBIT9
1292 003014  175777                     .WORD   ^CBIT10
1293 003016  173777                     .WORD   ^CBIT11
1294 003020  167777                     .WORD   ^CBIT12
1295 003022  157777                     .WORD   ^CBIT13
1296 003024  137777                     .WORD   ^CBIT14
1297 003026  077777                     .WORD   ^CBIT15
1298 003030  125252                     .WORD   125252                  ;ALTERNATING ONES, ZEROS
1299 003032  052525                     .WORD   052525                  ;ALTERNATING ONES, ZERO OPPOSITE FROM ABOVE
1300         003034             TBLEND==.
```

```
                                        .SBTTL  GLOBAL ENVIRONMENT STORAGE
                                ;
                                ;STORAGE FOR DEVICE REGISTERS
                                ;
003034  000000  100000  000000  DUMMY:  0,100000,0,0                  ;DUMMY DEVICE REGISTERS...
003044  000000  000000  000000          0,0,0,0,0,0,0,0
                                                                      ;...FOR MULTI-UNIT CHECKOUT.


003064  000000                  DUFLG::         .WORD   0             ;"DROPPED UNIT" FLAG.
                                                                      ;INHIBITS CODE IN "CLEAN-UP".
003066  000000                  NODEV::         .WORD   0             ;FLAG TO SAY NO DEVICE.

003070  000000                  TEMP1::         .WORD   0             ;SOME TEMP LOCATIONS.
003072  000000                  TEMP2::         .WORD   0
003074  000000                  XXCOMM::        .WORD   0             ;XXDP+ COMM BLOCK POINTER.
003076  000000                  FREE::          .WORD   0             ;1ST FREE MEMORY ADDRESS...
003100  000000                  FRESIZ::        .WORD   0             ;...AND SIZE (IN WORDS).
003102  000000                  FREEHI:         .WORD   0             ;LAST WORD IN FREE SPACE
003104  000000                  KTFLG::         .WORD   0             ;KT11, MEM AVAIL FLAG -
                                                                      ;- .WORD        0 = <24K OR NO KT -
                                                                      ;- NZ = >24K AND KT.
003106  000000                  KTENABLE::      .WORD   0             ;SET BY TEST ROUTINES TO FLAG >28K UNDER TEST
003110  002000                  PST32W::        .WORD   2000          ;32W BLOCK ADDRESS FOR 32K START
003112  000000                  SIFLAG::        .WORD   0             ;
003114  000000                  BADDAT::        .WORD   0             ;ACTUAL DATA
003116  000000                  GDDAT::         .WORD   0             ;EXPECTED DATA
003120  000000                  LOOPFL::        .WORD   0
003122                          CTAB::                                ;CONFIGURATION TABLES.
003122  000000                  CTABM::         .WORD   0             ;CONFIG WORK.
003124  000000                                  .WORD   0
003126  000000                                  .WORD   0
003130  000000                                  .WORD   0
003132  177777                                  .WORD   -1            ;END OF MEM TABLE.
003134                          CTABE::
                                ;ERROR STATISTICS TABLE (1 WORD PER UNIT), 64 UNITS MAX:
                                ;
                                ;       0       =       UNIT NOT TESTED
                                ;       100000  =       UNIT ONLINE, NO ERRORS
                                ;       10XXXX  =       UNIT ONLINE, ENCOUNTERED XXXX ERRORS
                                ;       160000  =       UNIT DROPPED, NON-EXISTENT DEVICE REGISTER
                                ;       160001  =       UNIT DROPPED, NOT IDLE AT START
                                ;       14XXXX  =       UNIT DROPPED, ENCOUNTERED XXXX ERRORS
                                ;
003134                          ERTABL:         .BLKW   64.
003334  000000                  ERTABE:         .WORD   0

003336  000000  SKIPT:  .WORD   0                                     ;1=SKIP SUBTEST 0=NO SKIP OF SUBTEST
```

```
                                        .SBTTL  GLOBAL TEXT MESSAGES
                                ;++
                                ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
                                ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
                                ; MORE THAN ONE TEST.
                                ;--



                                ;+
                                ;NAMES OF DEVICES SUPPORTED
                                ;-

003340                                  DEVTYP  <TK-25>
003340                          L$DVTYP::
003340  124  113  055                    .ASCIZ  /TK-25/
                                        .EVEN

                                ;+
                                ;TEST DESCRIPTION
                                ;-
003346                                  DESCRIPT <CZTKHB TK-25 FRT END FUNC #4>
003346                          L$DESC::
003346  103  132  124                    .ASCIZ  /CZTKHB TK-25 FRT END FUNC #4/
                                        .EVEN



                                ;+
                                ;BIT TO ASCII CONVERSION FOR TSSR REGISTER
                                ;-

003404  003444  003447  003453  TSSRBIT::       .WORD   1$,2$,3$,4$,5$,6$,7$,8$
003424  003505  003511  003515                  .WORD   9$,10$,11$,12$,13$,14$,15$,16$
003444    123    103    000    1$:     .ASCIZ  'SC'
003447    102    111    10?    2$:     .ASCIZ  'BIE'
003453    123    103    1?     3$:     .ASCIZ  'SCE'
003457    122    115    122    4$:     .ASCIZ  'RMR'
003463    116    130    115    5$:     .ASCIZ  'NXM'
003467    116    102    101    6$:     .ASCIZ  'NBA'
003473    102    111    124    7$:     .ASCIZ  'BIT9'
003500    102    111    124    8$:     .ASCIZ  'BIT8'
003505    123    123    122    9$:     .ASCIZ  'SSR'
003511    117    106    114    10$:    .ASCIZ  'OFL'
003515    102    111    124    11$:    .ASCIZ  'BIT5'
003522    102    111    124    12$:    .ASCIZ  'BIT4'
003527    102    111    124    13$:    .ASCIZ  'BIT3'
003534    102    111    124    14$:    .ASCIZ  'BIT2'
003541    102    111    124    15$:    .ASCIZ  'BIT1'
003546    102    111    124    16$:    .ASCIZ  'BIT0'
                                        .EVEN
003554    124    123    123    SFIERR: .ASCIZ  'TSSR ERROR AFTER SOFT INIT'
003607    124    123    123    SFHERR: .ASCIZ  'TSSR ERROR AFTER BUS RESET'
003642    040    040    116    NXR:    .ASCIZ  /  NON-EXISTANT DEVICE REGISTER/
003701    045    101    040    NXRX:   .ASCIZ  /%A  ADDRESS: %O6/
003722    045    101    040    TSSX:   .ASCII  /%A  TSBA,TSSR EXP'D: %O6%A,%O6%N/
003762    045    101    040            .ASCIZ  /%A  TSBA,TSSR REC'D: %O6%A,%O6/
```

```
1402 004021  045  116  045  FUSI:    .ASCII  /%N%A/
1403 004025  040  040  125  USI:     .ASCIZ  /  UNEXPECTED INTERRUPT/
1404 004054  040  040  111  NSI:     .ASCIZ  /  INTERRUPT EXPECTED, NOT RECEIVED/
1405 004117  045  116  045  FNOINTR:         .ASCII  /%N%A/
1406 004123  040  040  116  NOINTR:  .ASCIZ  /  NO INTERRUPT WAS GENERATED/
1407 004160  040  040  111  IFAULT:  .ASCIZ  /  INTERRUPT FAULT/
1408 004202  045  101  040  INTX:    .ASCIZ  /%A  CPU PC: %O6%A  TSBA: %O6/
1409 004237  040  040  042  NOINIT:  .ASCIZ  /  "BUS-INIT" DIDN'T INITIALIZE CONTROLLER/
1410 004311  040  040  042  NSINIT:  .ASCIZ  /  "SOFT-INIT" DIDN'T INITIALIZE THE DPU/
1411 004361  040  040  042  BRINIT:  .ASCIZ  /  "BUS-RESET" DIDN'T INITIALIZE THE DPU/
1412
1413 004431  000            NUL:     .ASCIZ  //
1414 004432  045  116  000  NULCR:   .ASCIZ  /%N/
1415 004435  045  101  040  EXPGOT:  .ASCIZ  /%A EXP'D: %O6%A, REC'D: %O6/
1416 004471  045  116  045  EXPGT2:  .ASCIZ  /%N%A EXP'D: %O6%A, %O6%N%A REC'D: %O%A, %O6/
1417 004545  045  101  040  DUAD12:  .ASCIZ  /%A REG(W) WRITTEN TO: %O6%A REG(R) READ; EXP'D: %O6%A, REC'D: %O6/
1418 004647  122  101  115  PKTRAM::         .ASCIZ  'RAM Contents Do Not Match Packet Sent'
1419 004715  040  040  103  SCME:    .ASCIZ  /  CONFIG DOESN'T MATCH MFG. MASTER/
1420 004760  127  122  111  WRTMSG:  .ASCIZ  'WRITE CHARACTERISTICS Failed'
1421 005015  124  123  123  WRTERR:  .ASCIZ  'TSSR Incorrect After WRITE Command, More Bits Set Than SSR'
1422 005110  124  123  123  RDERR:   .ASCIZ  'TSSR Incorrect After READ Command, More Bits Set Than SSR'
1423                                  .EVEN
1424
1425
1426
```

```
                        .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.
                        ;--

005202                          BGNMSG  NXRERR                  ;NON-EXISTANT DEVICE REGISTER.
005202                  NXRERR::
005202                          PRINTX  #NXRX,NODEV             ;NODEV = NEXM ADDRESS.
005202  013746  003066          MOV     NODEV,-(SP)
005206  012746  003701          MOV     #NXRX,-(SP)
005212  012746  000002          MOV     #2,-(SP)
005216  010600                  MOV     SP,R0
005220  104415                  TRAP    C$PNTX
005222  062706  000006          ADD     #6,SP
005226  004737  005234          JSR     PC,EXTEND               ; PRINT EXTENSION IF REQUIRED.
005232                          ENDMSG
005232                  L10002:
005232  104423                  TRAP    C$MSG

                        ;
                        ; THIS ROUTINE APPENDS A UNIQUE EXTENSION (IF REQUIRED)
                        ; TO ANY OF THE ABOVE ERROR SIGNATURES.
                        ;
005234  005727          EXTEND: TST     (PC)+
005236  000000          EXTA:   0                               ; 0 = NO EXTENSION.
005240  001402                  BEQ     1$
005242  004777  177770          JSR     PC,@EXTA                ; APPEND EXTENSION TEXT.
005246                  1$:     PRINTX  #NULCR                  ; PRINT A BLANK LINE
005246  012746  004432          MOV     #NULCR,-(SP)
005252  012746  000001          MOV     #1,-(SP)
005256  010600                  MOV     SP,R0
005260  104415                  TRAP    C$PNTX
005262  062706  000004          ADD     #4,SP
005266  000207                  RTS     PC
```

```
                        .SBTTL  PRITSSR - PRINT TSSR CONTENTS

                ;+
                ;
                ;ROUTINE TO DISPLAY THE CONTENTS, AND BIT DEFINITIONS, OF
                ;THE TSSR REGISTER. THIS ROUTINE IS NORMALLY CALLED ONLY
                ;BY A MESSAGE PRINTING ROUTINE
                ;
                ;INPUTS:
                ;
                ;       R1      CONTENTS OF TSSR
                ;
                ;SUBORDINATE ROUTINES:
                ;
                ;       CHKAMB  CHECK FOR AMBIGUOUS CONTENTS
                ;
                ;-

005270          PRITSSR:
005270                  SAVREG                          ;SAVE GENERAL REGISTERS
005274  010104          MOV     R1,R4                   ;SAVE THE TSSR CONTENTS
005276                  PRINTB  #TSSRFOR,R4             ;PRINT THE CONTENTS OF TSSR
005276  010446          MOV     R4,-(SP)
005300  012746  006112  MOV     #TSSRFOR,-(SP)
005304  012746  000002  MOV     #2,-(SP)
005310  010600          MOV     SP,R0
005312  104414          TRAP    C$PNTB
005314  062706  000006  ADD     #6,SP
005320  010400          MOV     R4,R0                   ;GET TSSR BACK FOR CHKAMB
005322  004737  016720  JSR     PC,CHKAMB               ;ARE CONTENTS AMBIGUOUS ?
005326  103410          BCS     5$                      ;BRANCH IF NOT
005330                  PRINTX  #AMBTSSR                ;SHOW CONTENTS ARE AMBIGUOUS
005330  012746  006332  MOV     #AMBTSSR,-(SP)
005334  012746  000001  MOV     #1,-(SP)
005340  010600          MOV     SP,R0
005342  104415          TRAP    C$PNTX
005344  062706  000004  ADD     #4,SP
005350  010403  5$:     MOV     R4,R3                   ;CONTENTS OF TSSR
005352  042703  001476  BIC     #HIADDR!FATERR!TERCLS,R3        ;CLEAR ALL MULTIPLE BIT FIELDS
005356  001434          BEQ     20$                     ;NO BITS ARE SET
005360  012702  002600  MOV     #TMPBFR,R2              ;TEMPORARY ASCII BUFFER
005364  012701  003404  MOV     #TSSRBIT,R1             ;ASCII EQUIVALENT OF BITS
005370  005703  10$:     TST     R3                      ;REMAINING BITS TO CONVERT
005372  001413          BEQ     15$                     ;BRANCH WHEN ALL ARE DONE
005374  000241          CLC                             ;CLEAR CARRY FOR SHIFT
005376  006103          ROL     R3                      ;SHIFT NEXT BIT TO CARRY
005400  103006          BCC     13$                     ;BRANCH IF BIT NOT SET
005402  011100          MOV     (R1),R0                 ;POINTER TO BIT DEFINITION
005404  112022  11$:     MOVB    (R0)+,(R2)+             ;MOVE ASCIZ TO BUFFER
005406  001376          BNE     11$                     ;MOVE ALL BITS
005410  112762  000054  177777  MOVB    #',,-1(R2)      ;INSERT A COMMA TO TERMINATE
005416  005721  13$:     TST     (R1)+                   ;POINT TO NEXT DESCRIPTION
005420  000763          BR      10$                     ;GET THE REMAINING BITS
005422  105042  15$:     CLRB    -(R2)                   ;TERMINATE THE LINE
005424                  PRINTX  #TSSDEF,#TMPBFR         ;PRINT THE BIT DEFINITIONS
005424  012746  002600  MOV     #TMPBFR,-(SP)
005430  012746  006303  MOV     #TSSDEF,-(SP)
```

```
                005434  012746  000002                  MOV     #2,-(SP)
                005440  010600                          MOV     SP,R0
                005442  104415                          TRAP    C$PNTX
                005444  062706  000006                  ADD     #6,SP
     1498
     1499       005450  010403                  20$:    MOV     R4,R3           ;GET THE TSSR CONTENTS
     1500       005452  042703  177761                  BIC     #^CTERCLS,R3    ;CLEAR ALL BUT TERMINATION
     1501       005456  016303  006374                  MOV     TCOCOD(R3),R3   ;GET THE TERMINATION CODE MEANING
     1502       005462                                  PRINTX  #TCOASC,R3      ;PRINT THE TERMINATION CODE
                005462  010346                          MOV     R3,-(SP)
                005464  012746  006173                  MOV     #TCOASC,-(SP)
                005470  012746  000002                  MOV     #2,-(SP)
                005474  010600                          MOV     SP,R0
                005476  104415                          TRAP    C$PNTX
                005500  062706  000006                  ADD     #6,SP
     1503       005504  010403                          MOV     R4,R3           ;TSSR CONTENTS AGAIN
     1504       005506  042703  177717                  BIC     #^CFATERR,R3    ;CLEAR ALL BUT FATAL TERMINATION
     1505       005512  001421                          BEQ     25$             ;DON'T PRINT IF ZERO
     1506       005514  006203                          ASR     R3
     1507       005516  006203                          ASR     R3
     1508       005520  006203                          ASR     R3              ;ALINE TERMINATION CODE FOR INDEX
     1509       005522  016303  006734                  MOV     TSFCOD(R3),R3   ;GET THE FATAL TERMINATION CODE
     1510       005526                                  PRINTX  #TFCASC,R3      ;PRINT THE FATAL TERMINATION CODE
                005526  010346                          MOV     R3,-(SP)
                005530  012746  006234                  MOV     #TFCASC,-(SP)
                005534  012746  000002                  MOV     #2,-(SP)
                005540  010600                          MOV     SP,R0
                005542  104415                          TRAP    C$PNTX
                005544  062706  000006                  ADD     #6,SP
     1511       005550  012737  000031  002172          MOV     #25.,FATFLG     ;DROP THIS UNIT AFTER ERROR MESSAGE
     1512       005556  010403                  25$:    MOV     R4,R3           ;GET TSSR CONTENTS
     1513       005560  042703  176377                  BIC     #^CHIADDR,R3    ;CLEAR ALL BUT EXTENDED ADDRESS
     1514       005564  001411                          BEQ     30$             ;DON'T PRINT IF ZERO
     1515       005566                                  PRINTX  #TEXASC,R3      ;PRINT THE EXTENDED ADDRESS BITS
                005566  010346                          MOV     R3,-(SP)
                005570  012746  006132                  MOV     #TEXASC,-(SP)
                005574  012746  000002                  MOV     #2,-(SP)
                005600  010600                          MOV     SP,R0
                005602  104415                          TRAP    C$PNTX
                005604  062706  000006                  ADD     #6,SP
     1516       005610  022704  100210          30$:    CMP     #100210,R4      ;CHECK FOR MEDIA ERROR
     1517       005614  001003                          BNE     31$             ;BR, IF PROBABLY NOT TAPE ERROR
     1518       005616  012737  006021  002150          MOV     #EPRT3,EPRTSW   ;"PROBABLY MEDIA RELETED ERROR - BAD TAPE"
     1519       005624  005737  002150          31$:    TST     EPRTSW          ;CHECK FOR THE SWITCH EMPTY
     1520       005630  001003                          BNE     310$            ;BR, IF SWITCH IS NOT EMPTY
     1521       005632  012737  005676  002150          MOV     #EPRT1,EPRTSW   ;SET SWITCH TO DEFAULT
     1522       005640  013737  002150  005650  310$:   MOV     EPRTSW,32$+2    ;PUT REAL SWITCHABLE MESSAGE IN PLACE
     1523       005646                          32$:    PRINTB  #EPRT1          ;PRINT THE ERROR MESSAGE
                005646  012746  005676                  MOV     #EPRT1,-(SP)
                005652  012746  000001                  MOV     #1,-(SP)
                005656  010600                          MOV     SP,R0
                005660  104414                          TRAP    C$PNTB
                005662  062706  000004                  ADD     #4,SP
     1524       005666  012737  005676  002150          MOV     #EPRT1,EPRTSW   ;RESET TO NORMAL ERROR POINTER
     1525       005674  000207                          RTS     PC              ;RETURN TO CALLER
     1526
     1527       005676      045     116     045  EPRT1:  .ASCIZ  '%N%A *****CHECK TRANSPORT******%S'
```

```
1528 005737     045     116     045  EPRT2:  .ASCIZ  '%N%A *****CHECK PARITY SWITCH IN TRANSPORT*****%S'
1529 006021     045     116     045  EPRT3:  .ASCIZ  '%N%A *****POSSIBLE MEDIA RELATED ERROR - BAD TAPE*****%S'
1530 006112     045     116     045  TSSRFOR:        .ASCIZ  '%N%A TSSR = %O6'
1531 006132     045     116     045  TEXASC:         .ASCIZ  '%N%A Extended Address Bits = %O6'
1532 006173     045     116     045  TCOASC:         .ASCIZ  '%N%A Termination Class Code = %T'
1533 006234     045     116     045  TFCASC:         .ASCIZ  '%N%A Fatal Termination Class Code = %T'
1534 006303     045     116     045  TSSDEF:         .ASCIZ  '%N%A TSSR Bits Set: %T'
1535 006332     045     116     045  AMBTSSR:        .ASCIZ  '%N%A TSSR Contents Are Ambiguous'
1536                                          .EVEN
1537 006374  006414  006437  006465  TCOCOD: .WORD   1$,2$,3$,4$,5$,6$,7$,8$
1538 006414     116     157     162  1$:     .ASCIZ  'Normal Termination'
1539 006437     124     145     162  2$:     .ASCIZ  'Termination Condition'
1540 006465     124     141     160  3$:     .ASCIZ  'Tape Status Alert'
1541 006507     106     165     156  4$:     .ASCIZ  'Function Reject'
1542 006527     122     145     143  5$:     .ASCIZ  'Recoverable Error - Tape Position One Record Down'
1543 006611     122     145     143  6$:     .ASCIZ  'Recoverable Error - Tape Was Not Moved'
1544 006660     125     156     162  7$:     .ASCIZ  'Unrecoverable Error'
1545 006704     106     141     164  8$:     .ASCIZ  'Fatal Controller Error'
1546                                          .EVEN
1547
1548 006734  006744  007000  007011  TSFCOD: .WORD   1$,2$,3$,4$
1549 006744     111     156     164  1$:     .ASCIZ  'Internal Diagnostic Failure'
1550 007000     122     145     163  2$:     .ASCIZ  'Reserved'
1551 007011     102     165     163  3$:     .ASCIZ  'Bus Interface or Sanity Check Error'
1552 007055     122     145     163  4$:     .ASCIZ  'Reserved'
1553                                          .EVEN
```

```
                                        .SBTTL  PRIPKT  - PRINT THE ADDRESS/CONTENTS OF COMMAND PACKET

                                ;+
                                ;THIS ROUTINE PRINTS THE ADDRESS AND CONTENTS OF A COMMAND PACKET.
                                ;THIS ROUTINE IS NORMALLY ONLY CALLED FROM A PRINT ROUTINE.
                                ;
                                ;INPUT:
                                ;
                                ;       R0      NUMBER OF WORDS IN PACKET
                                ;       R3      HIGH ORDER COMMAND PACKET ADDRESS
                                ;       R4      ADDRESS OF COMMAND PACKET
                                ;
                                ;       NOTE:   R3 IS IGNORED IF THE KTENABLE FLAG IS CLEAR.
                                ;-

007066                          PRIPKT::
007066                                  SAVREG                  ;SAVE THE REGISTERS
007072  010005                          MOV     R0,R5           ;SAVE NO. OF WORDS IN PACKET
007074  005737  003106                  TST     KTENABLE        ;ABOVE 28K UNDER TEST?
007100  001001                          BNE     10$             ;BR IF YES
007102  005003                          CLR     R3              ;SET HIGH ORDER ADDRESS TO 0
007104  010301                  10$:    MOV     R3,R1           ;COPY HIGH ORDER ADDRESS
007106  010400                          MOV     R4,R0           ;GET LOWER ADDRESS
007110  006100                          ROL     R0              ;SHIFT BIT 15 INTO C BIT
007112  006101                          ROL     R1              ;AND INTO HIGH ORDER.
007114                                  PRINTB  #PKTADD,R1,R4   ;PRINT PACKET ADDRESS
007114  010446                          MOV     R4,-(SP)
007116  010146                          MOV     R1,-(SP)
007120  012746  007272                  MOV     #PKTADD,-(SP)
007124  012746  000003                  MOV     #3,-(SP)
007130  010600                          MOV     SP,R0
007132  104414                          TRAP    C$PNTB
007134  062706  000010                  ADD     #10,SP
007140  010300                  15$:    MOV     R3,R0           ;GET HIGH ORDER ADDRESS
007142  001404                          BEQ     20$             ;BR IF NOT ABOVE 28K.
007144  010401                          MOV     R4,R1           ;GET LOW ORDER ADDRESS
007146  004737  020272                  JSR     PC,SETMAP       ;SETUP PAR6 MAPPING FOR 18 BIT ADDRESS
007152  010004                          MOV     R0,R4           ;GET RETURNED PAR6 ADDRESS BIAS
007154  005001                  20$:    CLR     R1              ;SAVE WORD NUMBER
007156  012402                  25$:    MOV     (R4)+,R2        ;GET PACKET CONTENTS
007160                                  PRINTB  #PKTFRM,R1,R2   ;PRINT THE DATA
007160  010246                          MOV     R2,-(SP)
007162  010146                          MOV     R1,-(SP)
007164  012746  007234                  MOV     #PKTFRM,-(SP)
007170  012746  000003                  MOV     #3,-(SP)
007174  010600                          MOV     SP,R0
007176  104414                          TRAP    C$PNTB
007200  062706  000010                  ADD     #10,SP
007204  005201                          INC     R1              ;NEXT WORD NUMBER
007206  020105                          CMP     R1,R5           ;DONE ALL PACKET WORDS?
007210  002762                          BLT     25$             ;LOOP TILL ALL DONE
007212                                  PRINTB  #PKTNEW         ;JUST A COUPLE NEW LINES
007212  012746  007327                  MOV     #PKTNEW,-(SP)
007216  012746  000001                  MOV     #1,-(SP)
007222  010600                          MOV     SP,R0
007224  104414                          TRAP    C$PNTB
007226  062706  000004                  ADD     #4,SP
```

```
 1593 007232  000207                          RTS     PC              ;RETURN
 1594
 1595 007234     045     116     045  PKTFRM: .ASCIZ  '%N%A Packet Word #%D1%A = %06'
 1596 007272     045     116     045  PKTADD: .ASCIZ  '%N%A Packet Address = %01%05'
 1597
 1598 007327     045     116     045  PKTNEW: .ASCIZ  '%N%N%A  '
 1599                                  .EVEN
 1600
```

```
                        .SBTTL  PRIBXOR - PRINT EXPD, RECV AND XOR BYTE

                ;+
                ;
                ;PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE DATA BYTE
                ;THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
                ;
                ;INPUTS:
                ;
                ;       R1      RECEIVED DATA
                ;       R2      EXPECTED DATA
                ;
                ;OUTPUT:
                ;
                ;       R0      XOR OF EXPECTED/RECEIVED DATA
                ;
                ;-

007340          PRIBXOR::
007340                  SAVREG                  ;SAVE THE REGISTERS
007344  010203          MOV     R2,R3           ;EXPECTED DATA
007346                  XOR     R1,R3           ;FORM THE EXCLUSIVE OR
007356  012700  177400  MOV     #^C<377>,R0     ;BYTE MASK
007362  040001          BIC     R0,R1           ;SAVE LOW BYTE RECV
007364  040002          BIC     R0,R2           ;SAVE LOW BYTE EXPD
007366  040003          BIC     R0,R3           ;SAVE LOW BYTE XOR
007370                  PRINTB  #XORBFOR,R2,R1,R3 ;PRINT THE MESSAGE
007370  010346          MOV     R3,-(SP)
007372  010146          MOV     R1,-(SP)
007374  010246          MOV     R2,-(SP)
007376  012746  007422  MOV     #XORBFOR,-(SP)
007402  012746  000004  MOV     #4,-(SP)
007406  010600          MOV     SP,R0
007410  104414          TRAP    C$PNTB
007412  062706  000012  ADD     #12,SP
007416  010300          MOV     R3,R0           ;R0 HAS XOR ON RETURN
007420  000207          RTS     PC              ;RETURN TO CALLER

007422      045     116     045  XORBFOR:       .ASCIZ  '%N%A EXPD: %03%A RECV: %03%A XOR: %03'
                                .EVEN
```

```
                                .SBTTL  PRIXOR  - PRINT EXPD, RECV AND XOR

                        ;+
                        ;
                        ;PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE TWO
                        ;THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
                        ;
                        ;INPUTS:
                        ;
                        ;       R1      RECEIVED DATA
                        ;       R2      EXPECTED DATA
                        ;
                        ;OUTPUT:
                        ;
                        ;       R0      XOR OF EXPECTED/RECEIVED DATA
                        ;
                        ;-

007470                  PRIXOR::
007470                          SAVREG                          ;SAVE THE REGISTERS
007474  010203                  MOV     R2,R3                   ;EXPECTED DATA
007476                          XOR     R1,R3                   ;FORM THE EXCLUSIVE OR
007506                          PRINTB  #XORFOR,R2,R1,R3 ;PRINT THE MESSAGE
007506  010346                  MOV     R3,-(SP)
007510  010146                  MOV     R1,-(SP)
007512  010246                  MOV     R2,-(SP)
007514  012746  007540          MOV     #XORFOR,-(SP)
007520  012746  000004          MOV     #4,-(SP)
007524  010600                  MOV     SP,R0
007526  104414                  TRAP    C$PNTB
007530  062706  000012          ADD     #12,SP
007534  010300                  MOV     R3,R0                   ;R0 HAS XOR ON RETURN
007536  000207                  RTS     PC                      ;RETURN TO CALLER

007540     045     116     045  XORFOR: .ASCIZ  '%N%A EXPD: %06%A RECV: %06%A XOR: %06'
                                .EVEN
```

```
                                        .SBTTL  PRIEQU  - PRINT BIT NUMBERS AS ASCII EQUIVALENT

                                ;+
                                ;
                                ;ROUTINE TO CONVERT BIT VALUES TO ASCII AND PRINT THE STRING
                                ;THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
                                ;
                                ;INPUTS:
                                ;
                                ;       R0      OCTAL VALUE TO CONVERT
                                ;       R1      TABLE OF POINTERS TO ASCII EQUIVALENT
                                ;
                                ;-

007606                          PRIEQU:
007606                                  SAVREG                          ;SAVE THE REGISTERS
007612  000207                          RTS     PC                      ;RETURN TO CALLER


                                        .SBTTL  PRIRAM  - PRINT RAM ADDRESS
                                ;+
                                ;
                                ;PRINT CONTROLLER RAM ADDRESS.
                                ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
                                ;
                                ;INPUTS:
                                ;
                                ;       R4      RAM ADDRESS
                                ;
                                ;-
007614                          PRIRAM:
007614                                  SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
007620                                  PRINTB  #RAMFOR,R4              ;PRINT RAM ADDRESS IN ERROR
007620  010446                          MOV     R4,-(SP)
007622  012746  007644                  MOV     #RAMFOR,-(SP)
007626  012746  000002                  MOV     #2,-(SP)
007632  010600                          MOV     SP,R0
007634  104414                          TRAP    C$PNTB
007636  062706  000006                  ADD     #6,SP
007642  000207                          RTS     PC                      ;RETURN

007644     045     116     045  RAMFOR: .ASCIZ  '%N%A CONTROLLER RAM ADDRESS = %O6'
                                        .EVEN


                                        .SBTTL  PRIADD  - PRINT MEMORY ERROR ADDRESS
                                ;+
                                ;
                                ;PRINT MEMORY ADDRESS
                                ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
                                ;
                                ; IMPLICIT INPUTS
                                ;
                                ;       ERRHI   - HIGH ORDER ADDRESS
                                ;       ERRLO   - LOW ORDER ADDRESS
```

```
                                        ;
                                        ;-
007706                                  PRIADD:
007706                                          SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
007712  013700  002204                          MOV     ERRHI,R0                ;GET HIGH ADDRESSS
007716  013701  002206                          MOV     ERRLO,R1                ;GET LOW ADDRESS
007722  010102                                  MOV     R1,R2                   ;COPY LOW ADDRESS
007724  006101                                  ROL     R1                      ;SHIFT BIT 15 TO C BIT
007726  006100                                  ROL     R0                      ;SHIFT INTO HIGH ORDER
007730                                          PRINTB  #PRIA0,R0,R2            ;PRINT MEMORY ADDRESS IN ERROR
007730  010246                                  MOV     R2,-(SP)
007732  010046                                  MOV     R0,-(SP)
007734  012746  007756                          MOV     #PRIA0,-(SP)
007740  012746  000003                          MOV     #3,-(SP)
007744  010600                                  MOV     SP,R0
007746  104414                                  TRAP    C$PNTB
007750  062706  000010                          ADD     #10,SP
007754  000207                                  RTS     PC                      ;RETURN

007756    045    116    045  PRIA0:  .ASCIZ  '%N%A MEMORY ERROR ADDRESS = %01%05'
                                    .EVEN


                                    .SBTTL  PRITADD - PRINT MEMORY TEST ADDRESS
                                ;+
                                ;
                                ;PRINT MEMORY ADDRESS
                                ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
                                ;
                                ; IMPLICIT INPUTS
                                ;
                                ;       ERRHI   - HIGH ORDER ADDRESS
                                ;       ERRLO   - LOW ORDER ADDRESS
                                ;
                                ;-
010022                          PRITADD:
010022                                  SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
010026  013700  002204                  MOV     ERRHI,R0                ;GET HIGH ADDRESSS
010032  013701  002206                  MOV     ERRLO,R1                ;GET LOW ADDRESS
010036  010102                          MOV     R1,R2                   ;COPY LOW ADDRESS
010040  006101                          ROL     R1                      ;SHIFT BIT 15 TO C BIT
010042  006100                          ROL     R0                      ;SHIFT INTO HIGH ORDER
010044                                  PRINTB  #PRIT0,R0,R2            ;PRINT MEMORY ADDRESS IN ERROR
010044  010246                          MOV     R2,-(SP)
010046  010046                          MOV     R0,-(SP)
010050  012746  010072                  MOV     #PRIT0,-(SP)
010054  012746  000003                  MOV     #3,-(SP)
010060  010600                          MOV     SP,R0
010062  104414                          TRAP    C$PNTB
010064  062706  000010                  ADD     #10,SP
010070  000207                          RTS     PC                      ;RETURN

010072    045    116    045  PRIT0:  .ASCIZ  '%N%A MEMORY TEST ADDRESS = %01%05'
                                    .EVEN
```

```
                        .SBTTL  SPACE   - SPACE RECORDS (FORWARD AND REVERSE) COMMAND

                ;+
                ;
                ;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:
                ;
                ;
                ;-

010134                  SPACE::
010134                          SAVREG                  ;SAVE THE GENERAL REGISTERS
010140  012737  000764  010330          MOV     #500.,SDELAY    ;SET UP DELAY
010146  012737  140010  010320          MOV     #140010,80$     ;SET UP COMMAND, SPACE FORWARD
010154  005703                          TST     R3              ;CHECK FOR DIRECTION
010156  100403                          BMI     5$              ;BR, IF REVERSE INDICATED
010160  010337  010322                  MOV     R3,90$          ;LOAD UP NUMBER OF RECORDS TO SPACE
010164  000407                          BR      10$             ;GO DO COMMAND
010166  042703  100000          5$:     BIC     #BIT15,R3       ;CLEAR DIRECTION BIT
010172  010337  010322                  MOV     R3,90$          ;LOAD UP NUMBER OF RECORDS TO SPACE
010176  052737  000400  010320          BIS     #BIT8,80$       ;SET REVERSE BIT IN COMMAND PACKET
010204  012704  010320          10$:    MOV     #80$,R4         ;SET UP R4 WITH PACKET ADDRESS
010210  010465  177776                  MOV     R4,TSDB(R5)     ;SEND OUT COMMAND
010214  004737  017124          15$:    JSR     PC,WAITF        ;WAIT FOR SSR
010220  103420                          BCS     20$             ;BR, IF SSR IS SET AND OK
010222                                  DELAY   250             ;DELAY ABOUT .25 SECONDS
010222  012727  000250                  MOV     #250,(PC)+
010226  000000                          .WORD   0
010230  013727  002116                  MOV     L$DLY,(PC)+
010234  000000                          .WORD   0
010236  005367  177772                  DEC     -6(PC)
010242  001375                          BNE     .-4
```

```
        010244  005367  177756                DEC     -22(PC)
        010250  001367                        BNE     .-20
 1811   010252  005337  010330                DEC     SDELAY          ;BUMP DELAY COUNTER DOWN
 1812   010256  001356                        BNE     15$             ;BR, IF MORE DELAY
 1813   010260  000411                        BR      60$             ;BR IF TROUBLE CARRY = CLEAR
 1814   010262  016501  000000        20$:    MOV     TSSR(R5),R1     ;READ TSSR
 1815   010266  012702  000200                MOV     #SSR,R2         ;SET UP EXPECTED
 1816   010272  020201                25$:    CMP     R2,R1           ;ARE THEY OK
 1817   010274  001401                        BEQ     40$             ;BR, IF EQUAL = OK
 1818   010276  000402                        BR      60$             ;TROUBLE EXIT
 1819   010300  000261                40$:    SEC                     ;SET CARRY NO TROUBLE
 1820   010302  000401                        BR      70$             ;EXIT
 1821   010304  000241                60$:    CLC                     ;CARRY CLEAR = ERROR
 1822   010306                        70$:
 1823   010306  010400                        MOV     R4,R0           ;PASS PACKET ADDRESS
 1824   010310  000207                        RTS     PC              ;RETURN
```

```
1826                                    ;
1827                                    ;
1828                                    ;
1829                                    ;PACKET FOR SPACE COMMAND
1830                                    ;
1832 010312                             .BLKB   10-<.-TUV2A&7>
1834                                    ;
1835                                    ;COMMAND WORD
1836 010320  000000                     80$:    .WORD
1837                                    ;NUMBER OF RECORDS TO BE SPACED OVER WORD
1838 010322  000000                     90$:    .WORD
1839 010324  000000                             .WORD
1840 010326  000000                             .WORD
1841 010330  000000                     SDELAY: .WORD   0               ;DELAY COUNTER
1842                                            .EVEN
```

```
                                .SBTTL  WRTCHR  - WRITE CHARACTERISTICS COMMAND

                        ;*
                        ;
                        ;ROUTINE TO ISSUE A WRITE CHARACTERISTICS
                        ;COMMAND SO THAT OTHER COMMANDS WILL BE ACCEPTED
                        ;
                        ;INPUT:
                        ;
                        ;       R4      ADDRESS OF PACKET FROM TEST
                        ;       R5      FIRST DEVICE UNIBUS ADDRESS
                        ;       REQUIRES A CALL TO SOFINIT BE DONE PREVIOUSLY
                        ;
                        ;OUTPUT:
                        ;
                        ;       R0      TSSR CONTENTS
                        ;       CARRY   SET - WRITE CHARACTERISTICS COMMAND OK
                        ;               CLR - WRITE CHARACTERISTICS FAILED
                        ;
                        ;IMPLICIT OUTPUT:
                        ;
                        ;       MESSAGE BUFFER AND OTHER BUFFERS ALL SET UP
                        ;       SOFTWARE SWITCHES SET AS FOLLOWS:
                        ;               BENBSW = BUFFER ENABLE SWITCH ON OR OFF
                        ;
                        ;
                        ;SIDE EFFECTS:
                        ;
                        ;
                        ;-

010332                  WRTCHR::
010332                          SAVREG                  ;SAVE THE GENERAL REGISTERS
010336  005037  002176          CLR     BENBSW          ;CLEAR BUFFER ENABLE SWITCH
010342  010465  177776  10$:    MOV     R4,TSDB(R5)     ;SEND OUT COMMAND
010346  004737  017240          JSR     PC,CHKTSSR      ;WAIT FOR SSR
010352  103401                  BCS     20$             ;BR, IF SSR IS SET AND OK
010354  000423                  BR      60$             ;BR IF TROUBLE CARRY = CLEAR
010356  016501  000000  20$:    MOV     TSSR(R5),R1     ;READ TSSR
010362  012702  000200          MOV     #SSR,R2         ;SET UP EXPECTED
010366  032701  000100          BIT     #OFL,R1         ;WAS OFF LINE SET IN TSSR
010372  001402                  BEQ     25$             ;BR, IF NO OFL SET
010374  052702  000100          BIS     #OFL,R2         ;MAKE THEM LOOK ALIKE
010400  020201          25$:    CMP     R2,R1           ;ARE THEY OK
010402  001401                  BEQ     40$             ;BR, IF EQUAL = OK
010404  000407                  BR      60$             ;TROUBLE EXIT
010406  062704  000010  40$:    ADD     #8.,R4          ;POINT TO WRT CHARA DATA PACKET
010412  011403                  MOV     (R4),R3         ;GET ADDRESS OF MESSAGE BUFFER
010414  010337  002720          MOV     R3,MESBFA       ;STORE FOR PRINT ROUTINES
010420  000261                  SEC                     ;SET CARRY NO TROUBLE
010422  000401                  BR      70$             ;EXIT
010424  000241          60$:    CLC                     ;CARRY CLEAR = ERROR
010426  016500  000000  70$:    MOV     TSSR(R5),R0     ;RETURN TSSR CONTENTS
010432  000207                  RTS     PC              ;RETURN
```

```
                        .SBTTL  REWIND  - POSITION TAPE (REWIND) COMMAND

                ;+
                ;
                ;THIS ROUTINE WILL REWIND THE SELECTED TAPE.
                ;
                ;       CAUTION: THE ROUTINE DOES NOT WAIT FOR BOT
                ;                TO ARRIVE. ALSO THE CALLER MUST CHECK FOR
                ;                SSR TO SET IN THE TSSR
                ;
                ;
                ;CALLING SEQUENCE:
                ;
                ;       DO A SOFT INIT
                ;       DO A WRITE CHARACTERISTICS
                ;       JSR     PC,REWIND
                ;
                ;INPUT:
                ;
                ;       R5      FIRST DEVICE UNIBUS ADDRESS
                ;
                ;
                ;OUTPUT
                ;
                ;       R0      THE CONTENTS OF R4 IS PASSED TO R0
                ;
                ;
                ;-
010434          REWIND::
010434                  SAVREG                  ;SAVE R1-R5 UNTIL NEXT RETURN
010440  012704  010530          MOV     #RWPACK,R4      ;GET PACKET ADDRESS
010444  010465  177776          MOV     R4,TSDB(R5)     ;SEND PACKET ADDRESS TO EXECUTE
010450  012703  000550          MOV     #360.,R3        ;ENOUGH TIME FOR 2400' REEL TO REWIND
010454  004737  017124  10$:    JSR     PC,WAITF        ;WAIT FOR SSR TO SET
010460  103417                  BCS     20$             ;LEAVE WHEN SSR IS SET
010462                          DELAY   250.            ;WAIT FOR .25 SECONDS
010462  012727  000372          MOV     #250.,(PC)+
010466  000000                  .WORD   0
010470  013727  002116          MOV     L$DLY,(PC)+
010474  000000                  .WORD   0
010476  005367  177772          DEC     -6(PC)
010502  001375                  BNE     .-4
010504  005367  177756          DEC     -22(PC)
010510  001367                  BNE     .-20
010512  005303                  DEC     R3              ;BUMP COUNTER DOWN
010514  001357                  BNE     10$             ;KEEP GOING
010516  000241                  CLC                     ;CLEAR CARRY TO SET ERROR
010520  010400          20$:    MOV     R4,R0           ;PASS THE PACKET ADDRESS
010522  000207                  RTS     PC              ;RETURN
010524                          .BLKB   10-<.-TUV2A&7>
010530          RWPACK:
010530  102010                  .WORD   102010          ;POSTION COMMAND (REWIND)
010532  000000                  .WORD   0               ;NOT USED
```

SEQ 0056

```
                        .SBTTL  CKRAM   - COMPARE RAM TO I/O PACKET

                ;+
                ;
                ;ROUTINE TO READ THE FIRST 8 BYTES FROM RAM
                ;MEMORY AND COMPARE THIS DATA TO A COMMAND PACKET.
                ;
                ;INPUT:
                ;
                ;       R4      ADDRESS OF THE COMMAND PACKET
                ;       R5      FIRST DEVICE UNIBUS ADDRESS
                ;
                ;OUTPUT:
                ;
                ;       CARRY   SET - RAM MATCHES PACKET
                ;               CLR - RAM DOES NOT MATCH PACKET
                ;
                ;IMPLICIT OUTPUT:
                ;
                ;       THE TABLE RAMDATA IS FILLED WITH THE
                ;       DATA HELD IN RAM.
                ;       RAMSIZ IS SET TO 8. FOR PRAMPKT ROUTINE
                ;
                ;SIDE EFFECTS:
                ;
                ;
                ;-

010534          CKRAM::
010534                  SAVREG                          ;SAVE THE GENERAL REGISTERS
010540  012701  002210          MOV     #RAMDATA,R1     ;ADDRESS TO SAVE THE RAM DATA
010544  012702  000020          MOV     #RMPKTBEG,R2    ;BYTE ADDRESS OF FIRST RAM DATA
010550  005003                  CLR     R3              ;CLEAR THE ERROR FLAG
010552  004737  017240          JSR     PC,CHKTSSR      ;WAIT FOR SSR
010556  004737  017240  10$:    JSR     PC,CHKTSSR      ;WAIT FOR SSR TO SET
010562  110265  177777          MOVB    R2,TSDBH(R5)    ;SELECT NEXT RAM ADDRESS
010566  004737  017240          JSR     PC,CHKTSSR      ;WAIT FOR SSR TO SET
010572  116511  177776          MOVB    TSBAL(R5),(R1)  ;READ THE RAM DATA
010576  122124                  CMPB    (R1)+,(R4)+     ;COMPARE TO EXPECTED
010600  001401                  BEQ     20$             ;BRANCH IF OK
010602  005203                  INC     R3              ;SET ERROR FLAG
010604  005202          20$:    INC     R2              ;ADDRESS OF NEXT RAM LOCATION
010606  020227  000027          CMP     R2,#RMPKTEND    ;REACHED END YET ?
010612  003761                  BLE     10$             ;BRANCH TILL ALL READ
010614  005703                  TST     R3              ;WAS AN ERROR FOUND ?
010616  001402                  BEQ     30$             ;BRANCH IF NOT
010620  000241                  CLC                     ;CLEAR CARRY TO SHOW ERROR
010622  000401                  BR      50$             ;AND EXIT
010624  000261          30$:    SEC                     ;SHOW GOOD COMPARE
010626  012737  000010  002250  50$:    MOV     #8.,RAMSIZ      ;SETUP RAMSIZ FOR PRAMPKT ROUTINE
010634  000207                  RTS     PC              ;RETURN
```

SEQ 0057

```
                                        .SBTTL RAMER - READ AND DISPLAY SELECTED RAM
                                ;+
                                ;
                                ;ROUTINE TO READ THE SELECTED RAM LOCATIONS
                                ;
                                ;INPUT:
                                ;
                                ;       R5      FIRST DEVICE UNIBUS ADDRESS
                                ;       CONSOLE WILL ALSO BE PRINTED TO
                                ;
                                ;IMPLICIT OUTPUT:
                                ;
                                ;       THE TABLE RAMDATA IS FILLED WITH THE
                                ;       DATA HELD IN RAM.
                                ;
                                ;SIDE EFFECTS:
                                ;
                                ;
                                ;-

010636                          RAMER::
010636                                  SAVREG                          ;SAVE THE GENERAL REGISTERS
010642  013705  011022                  MOV     RAMR5H,R5               ;RESET R5 TO FIRST DEVICE REGISTER
010646  012701  002210                  MOV     #RAMDATA,R1             ;ADDRESS TO SAVE THE RAM DATA
010652  013702  011020                  MOV     RAMHLD,R2               ;BYTE ADDRESS OF THE FIRST RAM DATA
010656  013703  002250                  MOV     RAMSIZ,R3               ;SET THE SIZE OF THE READ UP
010662  004737  017240          10$:    JSR     PC,CHKTSSR              ;WAIT FOR THE SSR TO SET
010666  110265  177777                  MOVB    R2,TSDBH(R5)            ;SELECT NEXT RAM ADDRESS
010672  004737  017240                  JSR     PC,CHKTSSR              ;WAIT FOR SSR TO SET
010676  116521  177776                  MOVB    TSBAL(R5),(R1)+         ;READ THE RAM DATA
010702  062702  000001          20$:    ADD     #1,R2                   ;ADDRESS OF THE NEXT RAM LOCATION
010706  077313                          SOB     R3,10$                  ;NUMBER OF LOCATIONS COUNTER
010710  013704  002250                  MOV     RAMSIZ,R4               ;GET THE RAM SIZE
010714  013702  011020                  MOV     RAMHLD,R2               ;GET THE STARTING RAM ADDRESS
010720  060204                          ADD     R2,R4                   ;CALCULATE THE END ADDRESS
010722  162704  000001                  SUB     #1,R4                   ;CORRECT VALUE OF PRINTOUT
010726                                  PRINTX  #RAMIOP,R2,R4           ;RAM ADDRESS = 10 - 17, ETC.
010726  010446                          MOV     R4,-(SP)
010730  010246                          MOV     R2,-(SP)
010732  012746  011024                  MOV     #RAMIOP,-(SP)
010736  012746  000003                  MOV     #3,-(SP)
010742  010600                          MOV     SP,R0
010744  104415                          TRAP    C$PNTX
010746  062706  000010                  ADD     #10,SP
010752  012701  002210                  MOV     #RAMDATA,R1             ;ADDRESS OF WHERE RAM DATA IS
010756  013703  002250                  MOV     RAMSIZ,R3               ;THE SIZE OF THE RAM FIELD READ
010762  005004                  30$:    CLR     R4                      ;NO EXTRA DATA LEFT OVER
010764  112104                          MOVB    (R1)+,R4                ;PICK UP BYTE OF RAM DATA
010766  042704  177400                  BIC     #177400,R4              ;GET RID OF SIGN EXTEND
010772                                  PRINTX  #RAMPD,R4               ;"010 211 111 222 377 000 123 134 ETC."
010772  010446                          MOV     R4,-(SP)
010774  012746  011075                  MOV     #RAMPD,-(SP)
011000  012746  000002                  MOV     #2,-(SP)
011004  010600                          MOV     SP,R0
011006  104415                          TRAP    C$PNTX
011010  062706  000006                  ADD     #6,SP
011014  077316                          SOB     R3,30$                  ;LOOP UNTIL ALL PRINTED
```

```
2046 011016  000207                  50$:    RTS     PC              ;RETURN
2047
2048 011020  000000                  RAMHLD: .WORD   0               ;RAM ADDR HOLDER 1ST ADDRESS
2049 011022  000000                  RAMR5H: .WORD   0               ;HOLDS R5 FOR LATER
2050 011024     045     116     045  RAMIOP: .ASCIZ  '%N%A Ram Address (Octal) = %03%A - %03%N'
2051 011075     045     101     040  RAMPD:  .ASCIZ  '%A %03%A '
2052                                         .EVEN
2053
```

```
                                .SBTTL  CKRAM2  - COMPARE RAM TO I/O CHARACTERISTICS DATA
                        ;+
                        ;
                        ;ROUTINE TO READ THE FIRST 8 OR 10 BYTES FROM RAM
                        ;MEMORY AND COMPARE THIS DATA TO A CHARACTERISTICS DATA BLOCK.
                        ;
                        ;INPUT:
                        ;
                        ;       R4      ADDRESS OF THE CHARACTERISTICS DATA
                        ;       R5      FIRST DEVICE UNIBUS ADDRESS
                        ;
                        ;OUTPUT:
                        ;
                        ;       CARRY   SET - RAM MATCHES PACKET
                        ;               CLR - RAM DOES NOT MATCH PACKET
                        ;
                        ;IMPLICIT OUTPUT:
                        ;
                        ;       THE TABLE RAMDATA IS FILLED WITH THE
                        ;       DATA HELD IN RAM.
                        ;       RAMSIZ IS SET TO 8. OR 10. FOR PRAMPKT ROUTINE
                        ;
                        ;SIDE EFFECTS:
                        ;
                        ;
                        ;-

011110                  CKRAM2::
011110                          SAVREG                          ;SAVE THE GENERAL REGISTERS
011114  012701  002210          MOV     #RAMDATA,R1             ;ADDRESS TO SAVE THE RAM DATA
011120  012702  000167          MOV     #RMCHBEG,R2             ;BYTE ADDRESS OF FIRST RAM DATA
011124  005003                  CLR     R3                      ;CLEAR THE ERROR FLAG
011126  004737  017240          JSR     PC,CHKTSSR              ;WAIT FOR SSR
011132  004737  017240  10$:    JSR     PC,CHKTSSR              ;WAIT FOR SSR TO SET
011136  110265  177777          MOVB    R2,TSDBH(R5)            ;SELECT NEXT RAM ADDRESS
011142  004737  017240          JSR     PC,CHKTSSR              ;WAIT FOR SSR TO SET
011146  116511  177776          MOVB    TSBAL(R5),(R1)          ;READ THE RAM DATA
011152  122124                  CMPB    (R1)+,(R4)+             ;COMPARE TO EXPECTED
011154  001401                  BEQ     20$                     ;BRANCH IF OK
011156  005203                  INC     R3                      ;SET ERROR FLAG
011160  005202          20$:    INC     R2                      ;ADDRESS OF NEXT RAM LOCATION
011162  012737  000010  002250  MOV     #8.,RAMSIZ              ;ASSUME NORMAL NOT SET
011170  020227  000176          CMP     R2,#RMCHEND-2           ;REACHED END YET ?
011174  003756                  BLE     10$                     ;BRANCH TILL ALL READ
011176  005703          27$:    TST     R3                      ;WAS AN ERROR FOUND ?
011200  001402                  BEQ     30$                     ;BRANCH IF NOT
011202  000241                  CLC                             ;CLEAR CARRY TO SHOW ERROR
011204  000401                  BR      50$                     ;AND EXIT
011206  000261          30$:    SEC                             ;SHOW GOOD COMPARE
011210  000207          50$:    RTS     PC                      ;RETURN
```

```
                                .SBTTL  CKMSG   - COMPARE WRITE CHAR. MESSAGE BUFFERS
                        ;+
                        ;
                        ;ROUTINE TO COMPARE A WRITE CHARACTERISTICS EXPD AND RECV
                        ;BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
                        ;ERROR PRINT ROUTINES.
                        ;
                        ;INPUT:
                        ;
                        ;       R0      RECV MESSAGE BUFFER HIGH ORDER ADDRESS
                        ;       R1      RECV MESSAGE BUFFER LOW ORDER ADDRESS
                        ;       R2      EXPD MESSAGE BUFFER ADDRESS
                        ;OUTPUT:
                        ;
                        ;       CARRY   SET - MESSAGE BUFFERS MATCH
                        ;               CLR -MESSAGE BUFFERS DON'T MATCH
                        ;
                        ;IMPLICIT OUTPUT:
                        ;
                        ;       EXPMSG          BUFFER IS SET TO EXPD DATA
                        ;       RECMSG          BUFFER IS SET TO RECV DATA
                        ;       RCVHIADD        SET TO HIGH ORDER ADDRESS OF RECV
                        ;       RCVLOADD        SET TO LOW ORDER ADDRESS OF RECV
                        ;
                        ;-
011212                  CKMSG::
011212                          SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
011216  010037  002252          MOV     R0,RCVHIADD             ;SAVE RECV HIGH ADDRESS
011222  010137  002254          MOV     R1,RCVLOAD              ;SAVE RECV LOW ADDRESS
011226  005737  003106          TST     KTENABLE                ;TESTING ABOVE 28K?
011232  001403                  BEQ     10$                     ;BR IF NO
011234  004737  020272          JSR     PC,SETMAP               ;RETURN ADDRESS BIASED TO PAR6 IN R0
011240  010001                  MOV     R0,R1                   ;GET RETURNED ADDRESS BIASED TO PAR6
011242  005004          10$:    CLR     R4                      ;WORD IN BUFFER
011244  005003                  CLR     R3                      ;CLEAR ERROR SEEN FLAG
011246  010205                  MOV     R2,R5                   ;GET EXPD BUFFER ADDRESS
011250  011264  002270  15$:    MOV     (R2),EXPMSG(R4)         ;SAVE EXPD FOR ERROR REPORT
011254  011164  002434          MOV     (R1),RECMSG(R4)         ;SAVE RECV FOR ERROR REPORT
011260  022221                  CMP     (R2)+,(R1)+             ;EXPD EQUAL RECV?
011262  001401                  BEQ     25$                     ;BR IF YES
011264  005203                  INC     R3                      ;SET ERROR SEEN FLAG
011266  062704  000002  25$:    ADD     #2,R4                   ;POINT TO NEXT WORD ADDRESS
011272  020427  000014          CMP     R4,#14                  ;DONE FIRST 7 WORDS?
011276  003764                  BLE     15$                     ;BR IF NO
011300  032765  000200  000012  BIT     #X2.EXTF,XST2(R5);IS EXTENDED FEATURES SET IN EXPD?
011306  001403                  BEQ     50$                     ;BR IF NO
011310  020427  000016          CMP     R4,#16                  ;DONE EXTENDED FEATURES WORD?
011314  003755                  BLE     15$                     ;BR IF NO
011316  005703          50$:    TST     R3                      ;ANY ERRORS SEEN?
011320  001402                  BEQ     55$                     ;BR IF NO
011322  000241                  CLC                             ;SET FAILURE
011324  000401                  BR      60$                     ;
011326  000261          55$:    SEC                             ;SET SUCCESS
011330  000207          60$:    RTS     PC                      ;RETURN
```

```
                                .SBTTL  CKMSG2  - COMPARE EXPD RECV MESSAGE BUFFERS
                        ;+
                        ;
                        ;ROUTINE TO COMPARE AN EXPECTED AND RECEIVED MESSAGE
                        ;BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
                        ;ERROR PRINT ROUTINES.
                        ;
                        ;INPUT:
                        ;
                        ;       R0      RECV MESSAGE BUFFER HIGH ORDER ADDRESS
                        ;       R1      RECV MESSAGE BUFFER LOW ORDER ADDRESS
                        ;       R2      EXPD MESSAGE BUFFER ADDRESS
                        ;       R3      NUMBER OF BYTES TO COMPARE
                        ;
                        ;OUTPUT:
                        ;
                        ;       CARRY   SET - MESSAGE BUFFERS MATCH
                        ;               CLR - MESSAGE BUFFERS DON'T MATCH
                        ;
                        ;IMPLICIT OUTPUT:
                        ;
                        ;       EXPMSG          BUFFER IS SET TO EXPD DATA
                        ;       RECMSG          BUFFER IS SET TO RECV DATA
                        ;       RCVHIADD        SET TO HIGH ORDER ADDRESS OF RECV
                        ;       RCVLOADD        SET TO LOW ORDER ADDRESS OF RECV
                        ;
                        ;-
011332                  CKMSG2::
011332                          SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
011336  020327  000144          CMP     R3,#RECMSG-EXPMSG;@@D IS COUNT ABOVE MAX ALLOWED?
011342  003412                  BLE     5$                      ;@@D BR IF NO
011344  012703  000144          MOV     #RECMSG-EXPMSG,R3;@@D
011350                          PRINTF  #DEBUGMSG               ;@@D
011350  012746  011464          MOV     #DEBUGMSG,-(SP)
011354  012746  000001          MOV     #1,-(SP)
011360  010600                  MOV     SP,R0
011362  104417                  TRAP    C$PNTF
011364  062706  000004          ADD     #4,SP
011370  010037  002252  5$:     MOV     R0,RCVHIADD             ;SAVE RECV HIGH ADDRESS
011374  010137  002254          MOV     R1,RCVLOAD              ;SAVE RECV LOW ADDRESS
011400  005737  003106          TST     KTENABLE                ;TESTING ABOVE 28K?
011404  001403                  BEQ     10$                     ;BR IF NO
011406  004737  020272          JSR     PC,SETMAP               ;RETURN ADDRESS BIASED TO PAR6 IN R0
011412  010001                  MOV     R0,R1                   ;GET RETURNED ADDRESS BIASED TO PAR6
011414  005004          10$:    CLR     R4                      ;WORD IN BUFFER
011416  005005                  CLR     R5                      ;CLEAR ERROR SEEN FLAG
011420  111264  002270  15$:    MOVB    (R2),EXPMSG(R4) ;SAVE EXPD FOR ERROR REPORT
011424  111164  002434          MOVB    (R1),RECMSG(R4) ;SAVE RECV FOR ERROR REPORT
011430  122221                  CMPB    (R2)+,(R1)+             ;EXPD EQUAL RECV?
011432  001401                  BEQ     25$                     ;BR IF YES
011434  005205                  INC     R5                      ;SET ERROR SEEN FLAG
011436  062704  000001  25$:    ADD     #1,R4                   ;POINT TO NEXT BYTE
011442  020403                  CMP     R4,R3                   ;DONE ALL BYTES?
011444  002001                  BGE     50$                     ;BR IF YES
011446  000764                  BR      15$                     ;DO NEXT BYTE
011450  005705          50$:    TST     R5                      ;ANY ERRORS SEEN?
011452  001402                  BEQ     55$                     ;BR IF NO
```

```
2215 011454  000241                    CLC                     ;SET FAILURE
2216 011456  000401                    BR      60$             ;
2217 011460  000261              55$:  SEC                     ;SET SUCCESS
2218 011462  000207              60$:  RTS     PC              ;RETURN
2219
2220 011464     120     122     117  DEBUGMSG:       .ASCIZ  'PROGRAM INTERNAL ERROR -CKMSG2 MESSAGE BUFFER EXCEEDED-';@@D
2221 011554     045     116     045  FERCM:  .ASCII  /%N%A  ***/
2222 011565     040     040     124  ERCM:   .ASCIZ  /  TSSR ERROR CODE REC'D = /
2223 011620     056     056     056  SIMSG:  .ASCIZ  /.... AFTER DOING SOFT INIT/
2224 011653     124     105     123  TINERR: .ASCIZ  /TEST: .../
2225                                   .EVEN
```

```
                                ;+
                                ;
                                ;PRINT ROUTINE TO FATAL SOFT INIT ERRORS
                                ;
                                ;INPUT:
                                ;
                                ;       R1      CONTENTS OF TSSR AT ERROR
                                ;
                                ;SIDE EFFECTS:
                                ;
                                ;       EXECUTES DROP UNIT TO CEASE TESTING
                                ;
                                ;-

011666                                  BGNMSG  SFIMSG
011666                          SFIMSG::
011666  004737  005270                  JSR     PC,PRITSSR      ;PRINT CONTENTS OF TSSR REGISTER
011672  004737  020156                  JSR     PC,CKDROP       ;DROP UNIT, IF ALLOWED
011676                                  ENDMSG
011676                          L10003:
011676  104423                          TRAP    C$MSG

                                ;+
                                ;PRINT ROUTINE TO PRINT THE CONTENTS OF
                                ;TSSR AND A COMMAND PACKET OTHER THAN GET STATUS COMMAND PACKET.
                                ;
                                ;INPUTS:
                                ;
                                ;       R1      TSSR CONTENTS
                                ;       R4      ADDRESS OF COMMAND PACKET
                                ;
                                ;-

011700                                  BGNMSG  PKTSSR
011700                          PKTSSR::
011700  004737  005270                  JSR     PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
011704  012700  000004                  MOV     #4,R0           ;NO. OF WORDS IN PACKET
011710  004737  007066                  JSR     PC,PRIPKT       ;PRINT THE CONTENTS OF COMMAND PACKET
011714  013700  002720                  MOV     MESBFA,R0       ;ADDRESS OF MESSAGE BUFFER
011720  005001                          CLR     R1              ;ASSUME NO HIGH MEMORY
011722  004737  014062                  JSR     PC,PRMESS       ;PRINT THE MESSAGE BUFFER ALSO
011726                                  ENDMSG
011726                          L10004:
011726  104423                          TRAP    C$MSG

                                ;+
                                ;PRINT ROUTINE TO PRINT THE CONTENTS OF
                                ;TSSR AND A GET STATUS COMMAND PACKET.
                                ;
                                ;INPUTS:
                                ;
                                ;       R1      TSSR CONTENTS
                                ;       R4      ADDRESS OF COMMAND PACKET
                                ;
                                ;-
```

```
2278
2279 011730                            BGNMSG  PKTGETS
     011730                       PKTGETS::
2280 011730  004737  005270              JSR     PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
2281 011734  012700  000002              MOV     #2,R0           ;NO. OF WORDS IN GET STATUS PACKET
2282 011740  004737  007066              JSR     PC,PRIPKT       ;PRINT THE CONTENTS OF COMMAND PACKET
2283 011744                              ENDMSG
     011744                       L10005:
     011744  104423                      TRAP    C$MSG
2284
2285
2286                                 ;+
2287                                 ;PRINT TSSR ERRORS FOR INITIALIZATION TESTS
2288                                 ;
2289                                 ;INPUTS:
2290                                 ;
2291                                 ;       R1      TSSR CONTENTS
2292                                 ;       R4      ADDRESS OF COMMAND PACKET
2293                                 ;-
2294
2295 011746                            BGNMSG  SFFMSG
     011746                       SFFMSG::
2296 011746  004737  005270              JSR     PC,PRITSSR      ;PRINT CONTENTS OF TSSR REGISTER
2297 011752                              ENDMSG
     011752                       L10006:
     011752  104423                      TRAP    C$MSG
2298
2299
2300                                         .SBTTL  PKTMES  - PRINT TSSR AND MESSAGE BUFFER
2301                                 ;+
2302                                 ;
2303                                 ;PRINT ROUTINE TO PRINT THE CONTENTS OF TSSR AND MESSAGE
2304                                 ;BUFFER FOR ERROR REPORTS
2305                                 ;
2306                                 ;INPUTS:
2307                                 ;
2308                                 ;       R1      CONTENTS OF TSSR
2309                                 ;       R2      LOW ORDER MESSAGE BUFFER
2310                                 ;       R3      HIGH ORDER MESSAGE BUFFER ADDRESS
2311                                 ;       NOTE: R3 IS IGNORED IF KTENABLE FLAG IS CLEAR
2312                                 ;-
2313 011754                              BGNMSG  PKTMES
     011754                       PKTMES::
2314 011754  004737  005270              JSR     PC,PRITSSR      ;PRINT CONTENTS OF TSSR
2315 011760  010200                      MOV     R2,R0           ;LOW ORDER ADDRESS
2316 011762  010301                      MOV     R3,R1           ;HIGH ORDER ADDRESS
2317 011764  004737  014062              JSR     PC,PRMESS       ;PRINT THE MESSAGE BUFFER
2318 011770                              ENDMSG
     011770                       L10007:
     011770  104423                      TRAP    C$MSG
2319
```

```
                                        .SBTTL  ADDSSR  - PRINT TEST ADDRESS AND TSSR
                                ;+
                                ;PRINT ROUTINE TO PRINT THE CONTENTS OF
                                ;TSSR AND A MEMORY TEST ADDRESS
                                ;
                                ;INPUTS:
                                ;
                                ;       R5      FIRST DEVICE UNIBUS ADDRESS
                                ;       ERRHI   HIGH ORDER MEMORY TEST ADDRESS
                                ;       ERRLO   LOW ORDER MEMORY TEST ADDRESS
                                ;-
                                
011772                                  BGNMSG  ADDSSR
011772                          ADDSSR::
011772  004737  010022                  JSR     PC,PRITADD      ;PRINT MEMORY TEST ADDRESS
011776  016501  000000                  MOV     TSSR(R5),R1     ;GET CURRENT TSSR
012002  004737  005270                  JSR     PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
012006                                  ENDMSG
012006                          L10010:
012006  104423                          TRAP    C$MSG

                                        .SBTTL  MSGEXP - PRINT WRITE CHAR. EXPD-RECV MESSAGE BUFFERS
                                ;+
                                ;
                                ;PRINT ROUTINE TO PRINT WRITE CHARACTERISTIC MESSAGE BUFFER
                                ;
                                ;IMPLICIT INPUTS:
                                ;
                                ;       EXPMSG  - EXPECTED MESSAGE BUFFER
                                ;       RECMSG  - RECEIVED MESSAGE BUFFER
                                ;       RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
                                ;       RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
                                ;-
012010                                  BGNMSG  MSGEXP
012010                          MSGEXP::
012010  012700  000007                  MOV     #7,R0           ;ASSUME NO EXT FEATURES
012014  004737  015426          5$:     JSR     PC,PRMSGEXP     ;PRINT EXPD/RECV MESSAGE BUFFERS
012020                                  ENDMSG
012020                          L10011:
012020  104423                          TRAP    C$MSG
```

```
                                .SBTTL  FIFEXP  - PRINT FIFO EXP/RECV DATA
                              ;*
                              ;
                              ;PRINT ROUTINE TO PRINT FIFO EXP/RECV DATA
                              ;
                              ;       R1      - BYTE COUNT
                              ;
                              ;IMPLICIT INPUTS:
                              ;
                              ;       EXPMSG  - EXPECTED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY
                              ;       RECMSG  - RECEIVED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY)
                              ;-
                                        BGNMSG  FIFEXP
                              FIFEXP::
                                        PRINTX  @FIF1MSG,R1     ;PRINT BYTES TRANSFERRED
                                        MOV     R1,-(SP)
                                        MOV     @FIF1MSG,-(SP)
                                        MOV     #2,-(SP)
                                        MOV     SP,R0
                                        TRAP    C$PNTX
                                        ADD     #6,SP
                                        PRINTX  @FIF2MSG        ;PRINT HEADER MSG
                                        MOV     @FIF2MSG,-(SP)
                                        MOV     #1,-(SP)
                                        MOV     SP,R0
                                        TRAP    C$PNTX
                                        ADD     #4,SP
                                        MOV     R1,R0           ;GET BYTE COUNT
                                        JSR     PC,PRBYTEXP     ;PRINT FIFO BYTES IN ERROR
                                        ENDMSG
                              L10012:
                                        TRAP    C$MSG
                              FIF1MSG:          .ASCIZ  '%N%A NUMBER OF BYTES TRANSFERRED = %D2'
                              FIF2MSG:          .ASCIZ  '%N%A FIFO DATA BYTES IN ERROR:'
                                        .EVEN
```

```
                                        .SBTTL  MSGSTAT - PRINT STATUS HEADER AND MESSAGE BUFFERS
                                ;*
                                ;
                                ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV
                                ;
                                ;
                                ;IMPLICIT INPUTS:
                                ;
                                ;       EXPMSG  - EXPECTED MESSAGE BUFFER
                                ;       RECMSG  - RECEIVED MESSAGE BUFFER
                                ;       RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
                                ;       RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
                                ;-
012202                                  BGNMSG  MSGSTAT
012202                          MSGSTAT::
012202  012701  012244                   MOV     #STATCOD,R1     ;ASCII ADDRESS TABLE
012206  012100                  10$:     MOV     (R1)+,R0        ;DONE ALL MSG LINES?
012210  001410                           BEQ     20$             ;BR IF YES
012212                                   PRINTX  R0              ;PRINT STATUS BIT NAMES
012212  010046                           MOV     R0,-(SP)
012214  012746  000001                   MOV     #1,-(SP)
012220  010600                           MOV     SP,R0
012222  104415                           TRAP    C$PNTX
012224  062706  000004                   ADD     #4,SP
012230  000766                           BR      10$             ;DO ANOTHER MSG LINE
012232  012700  000012          20$:     MOV     #10.,R0         ;NUMBER OF WORDS IN A READ STATUS BUFFER
012236  004737  015426                   JSR     PC,PRMSGEXP     ;PRINT EXPD/RECV MESSAGE BUFFERS
012242                                   ENDMSG
012242                          L10013:
012242  104423                           TRAP    C$MSG

012244  012262  012324  012415  STATCOD:        .WORD   1$,2$,3$,4$,5$,6$,0
012262    045     116    045    1$:.ASCIZ  '<N><A Tape Bus Signals in Word #8:'
012324    045     116    045    2$:.ASCIZ  '<N><A       PARERR<15>  IEOT  <12>  IFMK  <9>  IRDY<6>  IRWD<2>'
012415    045     116    045    3$:.ASCIZ  '<N><A       IRESV2<14>  IIDENT<11>  IHER  <8>  IONL<5>  IFBY<1>'
012506    045     116    045    4$:.ASCIZ  '<N><A       IRESV1<13>  ICER  <10>  ISPEED<7>  ILDP<4>  IFPT<0>'
012577    045     116    045    5$:.ASCIZ  '<N><A Tape Bus Signals in Word #9:'
012641    045     116    045    6$:.ASCIZ  '<N><A       DATMIS<7>  ILW<6>  OUTRDY<5>  INRDY<4>'
                                        .EVEN


                                        .SBTTL  MSGLOOP - PRINT LOOPBACK HEADER AND MESSAGE BUFFERS
                                ;*
                                ;
                                ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV
                                ;
                                ;IMPLICIT INPUTS:
                                ;
                                ;       EXPMSG  - EXPECTED MESSAGE BUFFER
                                ;       RECMSG  - RECEIVED MESSAGE BUFFER
                                ;       RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
                                ;       RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
                                ;-
012716                                  BGNMSG  MSGLOOP
012716                          MSGLOOP::
012716  012701  012760                   MOV     #LOOPCOD,R1     ;ASCII ADDRESS TABLE
```

```
2430 012722  012100                  10$:   MOV     (R1)+,R0        ;DONE ALL MSG LINES?
2431 012724  001410                         BEQ     20$             ;BR IF YES
2432 012726                                 PRINTX  R0              ;PRINT STATUS BIT NAMES
     012726  010046                         MOV     R0,-(SP)
     012730  012746  000001                 MOV     #1,-(SP)
     012734  010600                         MOV     SP,R0
     012736  104415                         TRAP    C$PNTX
     012740  062706  000004                 ADD     #4,SP
2433 012744  000766                         BR      10$             ;DO ANOTHER MSG LINE
2434 012746  012700  000012          20$:   MOV     #10.,R0         ;NUMBER OF WORDS IN A READ STATUS BUFFER
2435 012752  004737  015426                 JSR     PC,PRMSGEXP     ;PRINT EXPD/RECV MESSAGE BUFFERS
2436 012756                                 ENDMSG
     012756                          L10014:
     012756  104423                         TRAP    C$MSG
2437
2438 012760  013000  013053  013152  LOOPCOD:        .WORD   1$,2$,3$,4$,5$,6$,7$,0
2439 013000    045     116    045   1$:.ASCIZ  '<N><A Tape Bus Loopback Signals in Word #8:'
2440 013053    045     116    045   2$:.ASCIZ  '<N><A      PARERR<15>          IRESV2<14>          IRESV1<13>'
2441 013152    045     116    045   3$:.ASCIZ  '<N><A IHISP=>IEOT<12>   IWRT=>IIDENT<11>   IREV  =>ICER  <10>'
2442 013251    045     116    045   4$:.ASCIZ  '<N><A IWFM =>IFMK<09>   IEDIT=>IHER <08>   IFAD  =>ISPEED<07>'
2443 013350    045     116    045   5$:.ASCIZ  '<N><A ITAD0=>IRDY<06>   ITAD1=>IONL <05>   IERASE=>ILDP  <04>'
2444 013447    045     116    045   6$:.ASCIZ  '<N><A IREW =>IDBY<03>   IRWU =>IRWD <02>   IFEN  =>IFBY  <01>'
2445 013546    045     116    045   7$:.ASCIZ  '<N><A IGO  =>IFPT<00>'
2446                                        .EVEN
2447
```

```
2449                                            .SBTTL  MSGSUB - PRINT WRITE SUBSYSTEM MESSAGE BUFFER
2450                                    ;+
2451                                    ;
2452                                    ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV
2453                                    ;
2454                                    ;
2455                                    ;IMPLICIT INPUTS:
2456                                    ;
2457                                    ;       EXPMSG  - EXPECTED MESSAGE BUFFER
2458                                    ;       RECMSG  - RECEIVED MESSAGE BUFFER
2459                                    ;       RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2460                                    ;       RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2461                                    ;-
2462 013574                                            BGNMSG  MSGSUB
     013574                             MSGSUB::
2463 013574  012700  000012                           MOV     #10.,R0         ;SIZE OF WRITE SUBSYSTEM BUFFER
2464 013600  004737  015426                           JSR     PC,PRMSGEXP     ;PRINT EXPD/RECV MESSAGE BUFFERS
2465 013604                                           ENDMSG
     013604                             L10015:
     013604  104423                               TRAP    C$MSG
2466
2467
2468
2469
2470
2471                                            .SBTTL  MEMADD - PRINT MEMORY ADDRESS DATA ERROR
2472                                    ;+
2473                                    ;
2474                                    ;PRINT ROUTINE TO PRINT MEMORY ADDRESS DATA COMPARE ERROR
2475                                    ;
2476                                    ;IMPLICIT INPUTS:
2477                                    ;
2478                                    ;       ERRHI   - MEMORY ERROR HIGH ORDER ADDRESS
2479                                    ;       ERRLO   - MEMORY ERROR LOW ORDER ADDRESS
2480                                    ;       EXP     - EXPECTED DATA
2481                                    ;       RECV    - RECEIVED DATA
2482                                    ;-
2483 013606                                            BGNMSG  MEMADD
     013606                             MEMADD::
2484 013606  004737  007706                           JSR     PC,PRIADD       ;PRINT MEMORY ADDRESS IN ERROR
2485 013612  013701  002200                           MOV     EXPD,R1         ;GET EXPD DATA
2486 013616  013702  002202                           MOV     RECV,R2         ;GET RECEIVED DATA
2487 013622  004737  007470                           JSR     PC,PRIXOR       ;PRINT EXPD/RECV
2488 013626                                           ENDMSG
     013626                             L10016:
     013626  104423                               TRAP    C$MSG
2489
```

```
                                .SBTTL  PRAMPKT - PRINT RAM AND PACKET DATA
                            ;+
                            ;
                            ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
                            ;WHEN THE RAM DATA DOES NOT MATCH.
                            ;
                            ;INPUTS:
                            ;
                            ;       R4      POINTER TO COMMAND PACKET
                            ;
                            ;IMPLICIT INPUTS:
                            ;
                            ;       RAMDATA         DATA AS READ FROM THE RAM
                            ;       RAMSIZ          NUMBER OF BYTES IN PACKET
                            ;                       IF RAMSIZ=0 THEN DEFAULT TO 8.
                            ;
                            ;IMPLICIT OUTPUTS:
                            ;
                            ;       RAMSIZ  SET TO 0
                            ;-
                            PRAMPKT:
013630                              SAVREG                  ;SAVE R1-R5 UNTIL NEXT RETURN
013634  012701  002210              MOV     #RAMDATA,R1     ;DATA FROM THE RAM
013640  005002                      CLR     R2              ;INIT BYTE NUMBER
013642  122124              5$:     CMPB    (R1)+,(R4)+     ;COMPARE EXPECTED, RECEIVED
013644  001000                      BNE     7$              ;BR IF NO MATCH
013646  116105  177777      7$:     MOVB    -1(R1),R5       ;GET RECV RAM DATA
013652  116403  177777              MOVB    -1(R4),R3       ;GET EXPD PACKET DATA
013656                              XOR     R5,R3           ;XOR EXPD/RECV
013666  042703  177400              BIC     #177400,R3      ;LOW BYTE ONLY
013672  116137  177777  002202      MOVB    -1(R1),RECV     ;GET RECEIVED RAM DATA
013700  116437  177777  002200      MOVB    -1(R4),EXPD     ;GET EXPECTED RAM DATA
013706                              PRINTB  #RAMASC,R2,RECV,EXPD,R3
013706  010346                      MOV     R3,-(SP)
013710  013746  002200              MOV     EXPD,-(SP)
013714  013746  002202              MOV     RECV,-(SP)
013720  010246                      MOV     R2,-(SP)
013722  012746  013776              MOV     #RAMASC,-(SP)
013726  012746  000005              MOV     #5,-(SP)
013732  010600                      MOV     SP,R0
013734  104414                      TRAP    C$PNTB
013736  062706  000014              ADD     #14,SP
013742  005202              10$:    INC     R2              ;UPDATE BYTE COUNT
013744  005737  002250              TST     RAMSIZ          ;DEFAULT TO 8.?
013750  001404                      BEQ     15$             ;BR IF YES
013752  020237  002250              CMP     R2,RAMSIZ       ;DONE ALL BYTES?
013756  003731                      BLE     5$              ;BR IF NO
013760  000403                      BR      25$             ;
013762  020227  000010      15$:    CMP     R2,#8.          ;DONE DEFAULT NUMBER OF BYTES?
013766  002725              20$:    BLT     5$              ;BR IF NO
013770  005037  002250      25$:    CLR     RAMSIZ          ;SET DEFAULT RAMSIZ
013774  000207                      RTS     PC              ;RETURN
013776    045     116     045 RAMASC: .ASCIZ  '%N%A  BYTE: %D2%A RAM: %03%A Packet: %03%A XOR:%03'
                                    .EVEN
```

```
                                .SBTTL  PRMESS  - PRINT CONTENTS OF MESSAGE BUFFER
                        ;+
                        ;
                        ;THIS ROUTINE PRINTS THE CONTENTS OF
                        ;THE 7 WORD MESSAGE BUFFER RETURNED BY THE
                        ;TK-25.
                        ;
                        ;INPUT:
                        ;
                        ;       R0      LOW ORDER ADDRESS OF MESSAGE BUFFER
                        ;       R1      HIGH ORDER ADDRESS OF MESSAGE BUFFER
                        ;       NOTE: R1 IS IGNORED IF KTENABLE FLAG IS CLEAR
                        ;
                        ;THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
                        ;
                        ;-

014062                  PRMESS:
014062                          SAVREG                  ;SAVE THE REGISTERS
014066  010537  011022          MOV     R5,RAMR5H       ;SAVE DEVICE REGISTER POINTER
014072  010005                  MOV     R0,R5           ;SAVE LOW ORDER ADDRESS
014074  005737  003106          TST     KTENABLE        ;ADDRESS ABOVE 28K?
014100  001001                  BNE     10$             ;BR IF YES
014102  005001                  CLR     R1              ;SET HIGH ORDER ADDRESS TO 0
014104  010103          10$:    MOV     R1,R3           ;SAVE HIGH ORDER ADDRESS
014106  006100                  ROL     R0              ;SHIFT BIT15 TO C BIT
014110  006101                  ROL     R1              ;SHIFT TO HIGH ORDER FOR PRINTOUT
014112                          PRINTX  #PROASC,R1,R5   ;PRINT MESSAGE BUFFER ADDRESS
014112  010546                  MOV     R5,-(SP)
014114  010146                  MOV     R1,-(SP)
014116  012746  014720          MOV     #PROASC,-(SP)
014122  012746  000003          MOV     #3,-(SP)
014126  010600                  MOV     SP,R0
014130  104415                  TRAP    C$PNTX
014132  062706  000010          ADD     #10,SP
014136  022715  177777          CMP     #177777,(R5)    ;MESSAGE BUFFER FULL OF ONES
014142  001010                  BNE     15$             ;BR IF BUFFER IS PROBABLY OKAY
014144                          PRINTX  #MESBFN         ;"MESSAGE BUFFER PROBABLY NOT VALID"
014144  012746  014640          MOV     #MESBFN,-(SP)
014150  012746  000001          MOV     #1,-(SP)
014154  010600                  MOV     SP,R0
014156  104415                  TRAP    C$PNTX
014160  062706  000004          ADD     #4,SP
014164                  15$:    PRINTX  #PR1ASC         ;PRINT HEADER FOR CONTENTS
014164  012746  014765          MOV     #PR1ASC,-(SP)
014170  012746  000001          MOV     #1,-(SP)
014174  010600                  MOV     SP,R0
014176  104415                  TRAP    C$PNTX
014200  062706  000004          ADD     #4,SP
014204  005004                  CLR     R4              ;NUMBER OF THE NEXT WORD
014206  010501                  MOV     R5,R1           ;COPY LOW ORDER ADDRESS
014210  010300                  MOV     R3,R0           ;COPY HIGH ORDER ADDRESS
014212  001403                  BEQ     20$             ;BR IF NOT ABOVE 28K
014214  004737  020272          JSR     PC,SETMAP       ;SETUP PAR ADDRESS IN R0
014220  010005                  MOV     R0,R5           ;GET PAR FORMAT ADDRESS ABOVE 28K
014222                  20$:
014222                          PRINTX  #MESHEA,(R5)+   ;PRINT "MESSAGE BUFFER HEADER ="
```

```
                014222  012546                  MOV     (R5)+,-(SP)
                014224  012746  015023          MOV     #MESHEA,-(SP)
                014230  012746  000002          MOV     #2,-(SP)
                014234  010600                  MOV     SP,R0
                014236  104415                  TRAP    C$PNTX
                014240  062706  000006          ADD     #6,SP
2578            014244                          PRINTX  #DATAFL,(R5)+   ;PRINT "DATA FIELD LENGTH     ="
                014244  012546                  MOV     (R5)+,-(SP)
                014246  012746  015070          MOV     #DATAFL,-(SP)
                014252  012746  000002          MOV     #2,-(SP)
                014256  010600                  MOV     SP,R0
                014260  104415                  TRAP    C$PNTX
                014262  062706  000006          ADD     #6,SP
2579            014266                          PRINTX  #RBPCRA,(R5)+   ;PRINT "RESIDUAL BYTE COUNTER ="
                014266  012546                  MOV     (R5)+,-(SP)
                014270  012746  015135          MOV     #RBPCRA,-(SP)
                014274  012746  000002          MOV     #2,-(SP)
                014300  010600                  MOV     SP,R0
                014302  104415                  TRAP    C$PNTX
                014304  062706  000006          ADD     #6,SP
2580            014310                          PRINTX  #XSOCON,(R5)+   ;PRINT "XSTAT0 CONTENTS       ="
                014310  012546                  MOV     (R5)+,-(SP)
                014312  012746  015202          MOV     #XSOCON,-(SP)
                014316  012746  000002          MOV     #2,-(SP)
                014322  010600                  MOV     SP,R0
                014324  104415                  TRAP    C$PNTX
                014326  062706  000006          ADD     #6,SP
2581            014332                          PRINTX  #XS1CON,(R5)+   ;PRINT "XSTAT1 CONTENTS       ="
                014332  012546                  MOV     (R5)+,-(SP)
                014334  012746  015247          MOV     #XS1CON,-(SP)
                014340  012746  000002          MOV     #2,-(SP)
                014344  010600                  MOV     SP,R0
                014346  104415                  TRAP    C$PNTX
                014350  062706  000006          ADD     #6,SP
2582            014354                          PRINTX  #XS2CON,(R5)+   ;PRINT "XSTAT2 CONTENTS       ="
                014354  012546                  MOV     (R5)+,-(SP)
                014356  012746  015314          MOV     #XS2CON,-(SP)
                014362  012746  000002          MOV     #2,-(SP)
                014366  010600                  MOV     SP,R0
                014370  104415                  TRAP    C$PNTX
                014372  062706  000006          ADD     #6,SP
2583            014376                          PRINTX  #XS3CON,(R5)+   ;PRINT "XSTAT3 CONTENTS       ="
                014376  012546                  MOV     (R5)+,-(SP)
                014400  012746  015361          MOV     #XS3CON,-(SP)
                014404  012746  000002          MOV     #2,-(SP)
                014410  010600                  MOV     SP,R0
                014412  104415                  TRAP    C$PNTX
                014414  062706  000006          ADD     #6,SP
2584            014420  022737  000001  002134  CMP     #1,TRANSTST     ;CHECK FOR RAM DUMP REQUIRED
2585            014426  001402                  BEQ     40$             ;BR, IF REQUIRED
2586            014430  000137  014540          JMP     50$             ;JMP IF NO DUMP
2587            014434                  40$:    PRINTX  #RAMFHR
                014434  012746  014542          MOV     #RAMFHR,-(SP)
                014440  012746  000001          MOV     #1,-(SP)
                014444  010600                  MOV     SP,R0
                014446  104415                  TRAP    C$PNTX
                014450  062706  000004          ADD     #4,SP
```

```
 2588 014454  012737  000010  002250          MOV     #8.,RAMSIZ      ;RAM FIELD IS 8 BYTES LONG
 2589 014462  012737  000020  011020          MOV     #20,RAMHLD      ;FIELD STARTS AT 20 OCTAL (10 HEX)
 2590 014470  004737  010636                  JSR     PC,RAMER        ;READ AND PRINT THEM
 2591 014474  012737  000040  011020          MOV     #40,RAMHLD      ;FIELD STARTS AT 40 OCTAL (20 HEX)
 2592 014502  004737  010636                  JSR     PC,RAMER        ;READ AND PRINT THEM
 2593 014506  012737  000060  011020          MOV     #60,RAMHLD      ;FIELD STARTS AT 60 OCTAL (30 HEX)
 2594 014514  004737  010636                  JSR     PC,RAMER        ;READ AND PRINT THEM
 2595 014520  012737  000020  002250          MOV     #16.,RAMSIZ     ;RAM FIELD IS SIXTEEN BYTES LONG
 2596 014526  012737  000100  011020          MOV     #100,RAMHLD     ;FIELD STARTS AT 100 OCTAL (40 HEX)
 2597 014534  004737  010636                  JSR     PC,RAMER        ;READ AND PRINT THEM
 2598 014540  000207                  50$:    RTS     PC              ;RETURN
 2599 014542     045     116     045  RAMFHR: .ASCIZ  '%N%A ********** SPECIAL CONTROLLER RAM MEMORY DUMP **********'
 2600 014640     045     116     045  MESBFN: .ASCIZ  '%N%A MESSAGE BUFFER CONTENTS PROBABLY NOT VALID'
 2601 014720     045     116     045  PROASC: .ASCIZ  '%N%A Message Buffer Address = %01%05'
 2602 014765     045     116     045  PR1ASC: .ASCIZ  '%N%A Message Buffer Contents:'
 2603
 2604 015023     045     116     045  MESHEA: .ASCIZ  '%N%A Message Buffer Header     = %06'
 2605 015070     045     116     045  DATAFL: .ASCIZ  '%N%A Data Field Length         = %06'
 2606 015135     045     116     045  RBPCRA: .ASCIZ  '%N%A Residual Byte Counter     = %06'
 2607 015202     045     116     045  XSOCON: .ASCIZ  '%N%A XSTAT0 Contents           = %06'
 2608 015247     045     116     045  XS1CON: .ASCIZ  '%N%A XSTAT1 Contents           = %06'
 2609 015314     045     116     045  XS2CON: .ASCIZ  '%N%A XSTAT2 Contents           = %06'
 2610 015361     045     116     045  XS3CON: .ASCIZ  '%N%A XSTAT3 Contents           = %06'
 2611                                  .EVEN
```

```
                                        .SBTTL  PRMSGEXP - PRINT EXPD/RECV MESSAGE BUFFERS
                                ;+[B
                                ;
                                ;ROUTINE TO PRINT EXPECTED AND RECEIVED MESSAGE BUFFERS
                                ;
                                ;       R0      - NUMBER OF WORDS IN BUFFER
                                ;
                                ;IMPLICIT INPUTS:
                                ;
                                ;       EXPMSG  - EXPECTED MESSAGE BUFFER
                                ;       RECMSG  - RECEIVED MESSAGE BUFFER
                                ;       RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
                                ;       RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
                                ;-
2627 015426                     PRMSGEXP::
2628 015426                             SAVREG                  ;SAVE R1-R5 UNTIL NEXT RETURN
2629 015432  010005                     MOV     R0,R5           ;SAVE NUMBER OF WORDS
2630 015434  013700  002254             MOV     RCVLOADD,R0     ;GET RECV LOW ADDRESS
2631 015440  010004                     MOV     R0,R4           ;COPY LOW ADDRESS
2632 015442  013701  002252             MOV     RCVHIADD,R1     ;GET RECV HIGH ADDRESS
2633 015446  006100                     ROL     R0              ;SHIFT BIT15 TO C BIT
2634 015450  006101                     ROL     R1              ;SHIFT TO HIGH ORDER FOR PRINTOUT
2635 015452                             PRINTX  #PRMSG0,R1,R4   ;PRINT MESSAGE BUFFER ADDRESS
     015452  010446                     MOV     R4,-(SP)
     015454  010146                     MOV     R1,-(SP)
     015456  012746  015606             MOV     #PRMSG0,-(SP)
     015462  012746  000003             MOV     #3,-(SP)
     015466  010600                     MOV     SP,R0
     015470  104415                     TRAP    C$PNTX
     015472  062706  000010             ADD     #10,SP
2636 015476                             PRINTX  #PRMSG1         ;PRINT HEADER FOR CONTENTS
     015476  012746  015653             MOV     #PRMSG1,-(SP)
     015502  012746  000001             MOV     #1,-(SP)
     015506  010600                     MOV     SP,R0
     015510  104415                     TRAP    C$PNTX
     015512  062706  000004             ADD     #4,SP
2637 015516  005004                     CLR     R4              ;NUMBER OF THE CURRENT WORD
2638 015520  012701  002270             MOV     #EXPMSG,R1      ;GET EXPD BUFFER ADDRESS
2639 015524  012702  002434             MOV     #RECMSG,R2      ;GET RECV BUFFER ADDRESS
2640 015530  011100             20$:    MOV     (R1),R0         ;GET EXPD
2641 015532  011203                     MOV     (R2),R3         ;GET RECV
2642 015534                             XOR     R0,R3           ;XOR EXPD/RECV
2643 015544                             PRINTX  #PRMSG2,R4,(R1)+,(R2)+,R3
     015544  010346                     MOV     R3,-(SP)
     015546  012246                     MOV     (R2)+,-(SP)
     015550  012146                     MOV     (R1)+,-(SP)
     015552  010446                     MOV     R4,-(SP)
     015554  012746  015711             MOV     #PRMSG2,-(SP)
     015560  012746  000005             MOV     #5,-(SP)
     015564  010600                     MOV     SP,R0
     015566  104415                     TRAP    C$PNTX
     015570  062706  000014             ADD     #14,SP
2644 015574  005204                     INC     R4              ;NUMBER OF THE NEXT
2645 015576  020405                     CMP     R4,R5           ;DONE ALL YET?
2646 015600  002001                     BGE     50$             ;BR IF YES
2647 015602  000752                     BR      20$             ;DO ANOTHER
2648 015604  000207             50$:    RTS     PC              ;RETURN
```

```
2649
2650 015606     045     116     045  PRMSG0: .ASCIZ  '%N%A Message Buffer Address = %01%05'
2651 015653     045     116     045  PRMSG1: .ASCIZ  '%N%A Message Buffer Contents:'
2652 015711     045     116     045  PRMSG2: .ASCIZ  '%N%A   WORD #%D2%A  EXPD: %06%A  RECV: %06%A  XOR: %06'
2653                                          .EVEN
2654
```

```
                                        .SBTTL  PRBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER
                                ;+
                                ;
                                ;ROUTINE TO PRINT ERROR BYTES IN MESSAGE BUFFERS
                                ;       ONLY THE FIRST 8 ERRORS ENCOUNTERED ARE PRINTED DUE TO SCREEN SPACE
                                ;
                                ;       R0      - NUMBER OF BYTES IN BUFFER
                                ;
                                ;IMPLICIT INPUTS:
                                ;
                                ;       EXPMSG  - EXPECTED MESSAGE BUFFER
                                ;       RECMSG  - RECEIVED MESSAGE BUFFER
                                ;-
016776                          PRBYTEXP::
016776                                  SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
016002  010005                         MOV     R0,R5                   ;SAVE NUMBER OF BYTES
016004  005037  002266                  CLR     PRMNO                   ;INIT ERROR COUNT
016010  005004                         CLR     R4                      ;NUMBER OF THE CURRENT BYTE
016012  012701  002270                  MOV     #EXPMSG,R1              ;GET EXPD BUFFER ADDRESS
016016  012702  002434                  MOV     #RECMSG,R2              ;GET RECV BUFFER ADDRESS
016022  111100                  20$:    MOVB    (R1),R0                 ;GET EXPD BYTE
016024  042700  177400                  BIC     #^C<377>,R0             ;CLEAR UPPER BYTE
016030  110037  016344                  MOVB    R0,PRBEXP               ;SAVE FOR ERROR REPORT
016034  111203                         MOVB    (R2),R3                 ;GET RECV BYTE
016036  042703  177400                  BIC     #^C<377>,R3             ;CLEAR UPPER BYTE
016042  110337  016346                  MOVB    R3,PRBREC               ;FOR ERROR REPORT
016046                                  XOR     R0,R3                   ;XOR EXPD/RECV
016056  122122                         CMPB    (R1)+,(R2)+             ;EXPD = RECV?
016060  001431                         BEQ     30$                     ;BR IF YES
016062  005237  002266                  INC     PRMNO                   ;UPDATE ERROR COUNT
016066  023727  002266  000010         CMP     PRMNO,#8.               ;PRINTED 8?
016074  101023                         BHI     30$                     ;BR IF YES
016076                          27$:    PRINTX  #PRBMSG,R4,PRBEXP,PRBREC,R3
016076  010346                         MOV     R3,-(SP)
016100  013746  016346                  MOV     PRBREC,-(SP)
016104  013746  016344                  MOV     PRBEXP,-(SP)
016110  010446                         MOV     R4,-(SP)
016112  012746  016212                  MOV     #PRBMSG,-(SP)
016116  012746  000005                  MOV     #5,-(SP)
016122  010600                         MOV     SP,R0
016124  104415                         TRAP    C$PNTX
016126  062706  000014                  ADD     #14,SP
016132                                  FORCEXIT        50$     ;@@D
016142  000404                         BR      35$             ;@@D
016144                          30$:
016144                                  FORCERROR  27$,NOTSSR   ;@@D
016154                          35$:                            ;@@D
016154  005204                         INC     R4              ;NUMBER OF THE NEXT
016156  020405                         CMP     R4,R5           ;DONE ALL YET?
016160  002001                         BGE     50$             ;BR IF YES
016162  000717                         BR      20$             ;DO ANOTHER
016164                          50$:    PRINTX  #PRBTOT,PRMNO   ;PRINT TOTAL ERROR COUNT
016164  013746  002266                  MOV     PRMNO,-(SP)
016170  012746  016277                  MOV     #PRBTOT,-(SP)
016174  012746  000002                  MOV     #2,-(SP)
016200  010600                         MOV     SP,R0
016202  104415                         TRAP    C$PNTX
```

```
      016204  062706  000006                  ADD     #6,SP
 2699 016210  000207                          RTS     PC              ;RETURN
 2700
 2701 016212     045     116     045  PRBMSG: .ASCIZ  '%N%A   BYTE #%D2%A  EXPD: %03%A  RECV: %03%A  XOR: %03'
 2702 016277     045     116     045  PRBTOT: .ASCIZ  '%N%A NUMBER OF BYTES IN ERROR = %D2'
 2703                                  .EVEN
 2704 016344  000000                  PRBEXP: .WORD   0               ;EXPD
 2705 016346  000000                  PRBREC: .WORD   0               ;RECV
 2706
```

```
                                        .SBTTL  EXPREC  - PRINT EXPD/RECV WORD DATA
                                ;+
                                ;
                                ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
                                ;
                                ;INPUTS:
                                ;
                                ;       R1      RECEIVED DATA
                                ;       R2      EXPECTED DATA
                                ;
                                ;-

016350                                  BGNMSG  EXPREC
016350                          EXPREC::
016350  004737  007470                  JSR     PC,PRIXOR       ;PRINT THE DATA
016354                                  ENDMSG
016354                          L10017:
016354  104423                          TRAP    C$MSG
```

```
 2726                                           .SBTTL  EXPBREC - PRINT EXPD/RECV BYTE DATA
 2727                                   ;+
 2728                                   ;
 2729                                   ;PRINT ROUTINE TO DISPLAY BYTE EXPD/RECV DATA
 2730                                   ;
 2731                                   ;
 2732                                   ;INPUTS:
 2733                                   ;
 2734                                   ;       R1      RECEIVED DATA BYTE
 2735                                   ;       R2      EXPECTED DATA BYTE
 2736                                   ;
 2737                                   ;-
 2738
 2739 016356                                    BGNMSG  EXPBREC
      016356                            EXPBREC::
 2740 016356  004737  007340                    JSR     PC,PRIBXOR      ;PRINT THE DATA
 2741 016362                                    ENDMSG
      016362                            L10020:
      016362  104423                            TRAP    C$MSG
 2742
 2743
 2744
 2745
 2746                                           .SBTTL  RAMERR  - PRINT RAM AND PACKET DATA
 2747                                   ;+
 2748                                   ;
 2749                                   ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
 2750                                   ;
 2751                                   ;INPUTS:
 2752                                   ;
 2753                                   ;       R4      POINTER TO COMMAND PACKET
 2754                                   ;
 2755                                   ;IMPLICIT INPUTS:
 2756                                   ;
 2757                                   ;       RAMDATA         DATA AS READ FROM THE RAM
 2758                                   ;       RAMSIZ          NUMBER OF BYTES IN PACKET
 2759                                   ;                       IF RAMSIZ=0 THEN DEFAULT TO 8.
 2760                                   ;
 2761                                   ;IMPLICIT OUTPUTS:
 2762                                   ;
 2763                                   ;       RAMSIZ  SET TO 0
 2764                                   ;-
 2765
 2766 016364                                    BGNMSG  RAMERR
      016364                            RAMERR::
 2767 016364  004737  013630                    JSR     PC,PRAMPKT      ;PRINT RAM/PACKET DATA
 2768 016370                                    ENDMSG
      016370                            L10021:
      016370  104423                            TRAP    C$MSG
 2769
 2770
 2771                                           .SBTTL  RAMTADD - PRINT TEST ADDRESS, RAM AND PACKET DATA
 2772                                   ;+
 2773                                   ;
 2774                                   ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
 2775                                   ;
 2776                                   ;INPUTS:
```

```
                        ;
                        ;       R4      POINTER TO COMMAND PACKET
                        ;
                        ;IMPLICIT INPUTS:
                        ;
                        ;       RAMDATA         DATA AS READ FROM THE RAM
                        ;       RAMSIZ          NUMBER OF BYTES IN PACKET
                        ;                       IF RAMSIZ=0 THEN DEFAULT TO 8.
                        ;       ERRHI           HIGH ORDER TEST ADDRESS
                        ;       ERRLO           LOW ORDER TEST ADDRESS
                        ;
                        ;IMPLICIT OUTPUTS:
                        ;
                        ;       RAMSIZ  SET TO 0
                        ;-

                                BGNMSG  RAMTADD
016372                  RAMTADD::
016372 004737 010022            JSR     PC,PRITADD      ;PRINT TEST ADDRESS
016376 004737 013630            JSR     PC,PRAMPKT      ;PRINT RAM/PACKET DATA
016402                          ENDMSG
016402                  L10022:
016402 104423                   TRAP    C$MSG

                                .SBTTL  RAMEXP  - PRINT RAM EXPD/RECV DATA
                        ;*
                        ;
                        ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
                        ;
                        ;INPUTS:
                        ;
                        ;       R1      RECEIVED DATA
                        ;       R2      EXPECTED DATA
                        ;       R4      CONTROLLER RAM ADDRESS
                        ;-

                                BGNMSG  RAMEXP
016404                  RAMEXP::
016404 042701 177400            BIC     #^C<377>,R1     ;SAVE EXPD RAM DATA BYTE
016410 042702 177400            BIC     #^C<377>,R2     ;SAVE EXPD RAM DATA BYTE
016414 004737 007614            JSR     PC,PRIRAM       ;PRINT THE RAM ADDRESS
016420 004737 007470            JSR     PC,PRIXOR       ;PRINT THE DATA
016424                          ENDMSG
016424                  L10023:
016424 104423                   TRAP    C$MSG

                                .SBTTL  TIMEXP  - PRINT TIMER A,B AND EXP/REC
                        ;*
                        ;
                        ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
                        ;AND TIMER A,B HEADER MESSAGE
                        ;
                        ;INPUTS:
                        ;
                        ;       R1      RECEIVED DATA
                        ;       R2      EXPECTED DATA
```

```
2828                                    ;-
2829
2830 016426                                     BGNMSG  TIMEXP
     016426                             TIMEXP::
2831 016426                                     PRINTX  #TIMSGO         ;PRINT HEADER
     016426  012746  016454                     MOV     #TIMSGO,-(SP)
     016432  012746  000001                     MOV     #1,-(SP)
     016436  010600                             MOV     SP,R0
     016440  104415                             TRAP    C$PNTX
     016442  062706  000004                     ADD     #4,SP
2832 016446  004737  007470                     JSR     PC,PRIXOR       ;PRINT THE DATA
2833 016452                                     ENDMSG
     016452                             L10024:
     016452  104423                             TRAP    C$MSG
2834
2835
2836 016454     045     116     045    TIMSGO: .ASCIZ  '<N><A TIMER A STATUS IS IN BIT 3<N><A TIMER B STATUS IS IN BIT 2'
2837                                    .EVEN
```

```
                                        .SBTTL  BADSSR - PRINT TSSR ERRORS ON DATA TRANSFERS

                                ;*
                                ;
                                ;PRINT ROUTINE FOR TSSR ERRORS ON DATA TRANSFERS
                                ;
                                ;INPUTS:
                                ;
                                ;       R1      CONTENTS OF TSSR
                                ;       R2      DATA WRITTEN (8 BITS)
                                ;
                                ;-

016554                                  BGNMSG  BADSSR
016554                          BADSSR::
016554  010246                          MOV     R2,-(SP)        ;SAVE DATA TRANSFERRED
016556  042702  177400                  BIC     #177400,R2      ;GET JUST ONE BYTE
016562                                  PRINTB  #XFERASC,R2
016562  010246                          MOV     R2,-(SP)
016564  012746  016614                  MOV     #XFERASC,-(SP)
016570  012746  000002                  MOV     #2,-(SP)
016574  010600                          MOV     SP,R0
016576  104414                          TRAP    C$PNTB
016600  062706  000006                  ADD     #6,SP
016604  012602                          MOV     (SP)+,R2        ;RESTORE R2
016606  004737  005270                  JSR     PC,PRITSSR      ;DECODE TSSR CONTENTS
016612                                  ENDMSG
016612                          L10025:
016612  104423                          TRAP    C$MSG
016614    045     116     045   XFERASC:        .ASCIZ  '%N%A Data Transferred = %03'
```

```
                                .SBTTL  GLOBAL SUBROUTINES SECTION

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

                                .SBTTL  SOFINIT - SOFT INITIALIZE OF CONTROLLER

                        ;+
                        ;
                        ;ROUTINE TO DO A SOFT INITIALIZE OF THE CONTROLLER
                        ;BY WRITING INTO THE TSSR REGISTER. AFTER THE INIT,
                        ;THE TSSR REGISTER IS TESTED FOR ERRORS. ANY ERRORS
                        ;DETECTED SHOULD BE TREATED AS DEVICE FATAL ERRORS.
                        ;
                        ;INPUTS:
                        ;
                        ;       R5      ADDRESS OF FIRST REGISTER
                        ;
                        ;OUTPUTS:
                        ;
                        ;       R0      CONTENTS OF TSSR, IF ERROR
                        ;       CARRY   SET IF INIT WAS OKAY
                        ;               CLEAR IF FATAL ERROR
                        ;
                        ;CALLING SEQUENCE:
                        ;
                        ;       MOV     #ADDRESS,R5
                        ;       JSR     PC,SOFINIT
                        ;       BCS     CONTINUE
                        ;       ERRDF                   ;REPORT FATAL ERROR
                        ;
                        ;-

016650                  SOFINIT::
016650                          SAVREG                  ; SAVE THE REGISTERS
016654  012765  000000  000000  MOV     #0,TSSR(R5)     ; DO THE INIT.
016662  004737  017124          JSR     PC,WAITF        ; WAIT FOR SSR
016666  016500  000000          MOV     TSSR(R5),R0     ;GET THE TSSR REGISTER
016672  010004                  MOV     R0,R4           ;START SETUP OF EXPECTED TSSR
016674  042704  176277          BIC     #^C<HIADDR!OFL>,R4 ;CLEAR OUT UNUSED BITS
016700  052704  002200          BIS     #SSR!NBA,R4     ;R4 HAS EXPECTED CONTENTS
016704  020400                  CMP     R4,R0           ;ONLY EXPECTED BITS SET ?
016706  001402                  BEQ     5$              ;BRANCH IF OKAY
016710  000241                  CLC                     ;CLEAR THE CARRY FOR ERROR
016712  000401                  BR      10$             ;GO TO EXIT
016714  000261          5$:     SEC                     ;SET THE CARRY BIT
016716  000207          10$:    RTS     PC              ;RETURN TO CALLER
```

```
                                .SBTTL  CHKAMB  - CHECK TSSR FOR AMBIGUITY.

                        ;+
                        ;
                        ;THIS ROUTINE TESTS THE CONTENTS OF THE TSSR REGISTER
                        ;FOR AMBIGUITY
                        ;
                        ;INPUT:
                        ;
                        ;       R0      CONTENTS OF TSSR
                        ;
                        ;OUTPUT:
                        ;
                        ;       R0      CONTENTS OF TSSR
                        ;
                        ;       CARRY   SET - NO AMBIGUITY
                        ;               CLR - AMBIGUOUS CONTENTS
                        ;
                        ;-

016720                  CHKAMB:
016720                          SAVREG                  ;SAVE THE GENERAL REGISTERS
016724  010004                  MOV     R0,R4           ;CONTENTS OF TSSR
016726  032700  100000          BIT     #SC,R0          ;IS BIT 15 SET ?
016732  001004                  BNE     5$              ;BRANCH IF YES
016734  032700  174077          BIT     #†C<NBA!OFL!SSR!HIADDR>,R0      ;ANY OTHER BITS SET ?
016740  001023                  BNE     40$             ;MUST BE AN ERROR
016742  000424                  BR      45$             ;RETURN WITH SUCCESS
016744  032700  000200  5$:     BIT     #SSR,R0         ;IS READY BIT SET ?
016750  001011                  BNE     10$             ;BRANCH IF READY BIT IS SET.
016752  032700  000040          BIT     #BIT5,R0        ;IS FATAL ERROR BIT SET ?
016756  001414                  BEQ     40$             ;ERROR IF NOT
016760  042704  177761          BIC     #†CTERCLS,R4    ;CLEAR ALL BUT TERMINATION CODE
016764  020427  000016          CMP     R4,#16          ;ALL THREE BITS MUST BE SET
016770  001007                  BNE     40$             ;ERROR IF NOT SET
016772  000410                  BR      45$             ;OK IF ALL ARE SET
016774  032700  000040  10$:    BIT     #BIT5,R0        ;IS FATAL ERROR BIT SET ?
017000  001405                  BEQ     45$             ;ERROR IF BIT IS SET WITH SSR
017002  032700  000006          BIT     #BIT2!BIT1,R0   ;IS THIS A FUNCTION REJECT
017006  001002                  BNE     45$             ;BR, IF TSSR IS OK
017010  000241          40$:    CLC                     ;AMBIGUOUS CONTENTS
017012  000401                  BR      50$
017014  000261          45$:    SEC                     ;SHOW SUCCESS - NO AMBIGUITY
017016  000207          50$:    RTS     PC              ;RETURN TO CALLER
```

```
                                .SBTTL  ENAINT,DSBINT - ENABLE/DISABLE INTERRUPTS
                        ;
                        ; DEFAULT DISPLAY INTERRUPT HANDLERS.
                        ; IF DISPLAY TIME-OUT, REPORT DEV FATAL, AND ABORT PASS.
                        ; OTHERWISE, SAVE DPU REGISTERS AND DISMISS.
                        ;

                        ; BIT DEFINITIONS FOR "INTMASK" AND "INTFLAG" BYTES:
                        ;
  000200                        IOKCKIN=BIT7            ; DON'T CHECK FOR BAD INTERRUPTS -- TEST WILL.
  000001                        IOKSTP=BIT0             ; EXPECT "STOP" INTERRUPT.
                        ;
                        ;INTERRUPT MASK -- SAYS EXPECTING INTERRUPTS
017020  000             INTMASK:        .BYTE   0
                        ;INTERRUPT FLAG -- SAYS WE GOT ONE (IF POSITIVE)
017021  000             INTFLAG:        .BYTE   0

                        ;SAVED INTERRUPT VECTOR:
017022  000000          INTVEC: .WORD   0
                        ;SAVE CPU PC
017024  000000          INTCPC: .WORD   0

                        ;SUBROUTINE TO ENABLE INTERRUPTS:
017026  010046          ENAINT: MOV     R0,-(SP)        ;SAVE R0
017030  013700  002160          MOV     IVEC,R0         ;GET POINTER TO VECTORS
017034  012720  017072          MOV     #INTR,(R0)+     ;SET UP INTERRUPT VECTOR
017040  012720  000340          MOV     #PRI07,(R0)+
017044  012600                  MOV     (SP)+,R0        ;RESTORE R0
017046  011646                  MOV     (SP),-(SP)
017050  012766  000000  000002  MOV     #0,2(SP)        ;SET CPU TO LEVEL 0
017056  000002                  RTI

                        ;SUBROUTINE TO DISABLE INTERRUPTS (RAISE PRIORITY TO LEVEL 7)
017060  011646          DSBINT: MOV     (SP),-(SP)
017062  012766  000340  000002  MOV     #PRI07,2(SP)
017070  000002                  RTI
```

```
                                        .SBTTL  INTR    - INTERRUPT HANDLERS

017072                                  BGNSRV  INTR            ;DEFINE INTERRUPT ENTRY
017072                          INTR::
017072  012737  000001  002174          MOV     #1,INTRECV      ;SET FLAG TO SHOW INTERRUPT RECEIVED
017100  105037  017021                  CLRB    INTFLAG         ;CLEAR FLAG TO SAY WE GOT INTERRUPT
017104  132737  000001  017020          BITB    #IOKSTP,INTMASK ;EXPECTING STOP INTERRUPT?
017112  001003                          BNE     1$              ;BR IF YES
017114  152737  000001  017021          BISB    #IOKSTP,INTFLAG ;NO. SET THE ERROR FLAG.

                        ;SAVE REGISTERS, MSG BUFFER, ETC.
017122                  1$:
017122                                  ENDSRV
017122                  L10026:
017122  000002                          RTI
```

```
                                        .SBTTL  WAITF   - WAIT FOR SUBSYSTEM READY
                                ;
                                ; SUBROUTINE TO WAIT FOR THE SUBSYSTEM READY FLAG
                                ;
                                ;INPUTS:
                                ;
                                ;       R5      ADDRESS OF FIRST DEVICE REGISTER
                                ;
                                ;OUTPUTS:
                                ;
                                ;       R0      CONTENTS OF LAST TSSR READ
                                ;       CARRY   SET - READY BIT SET
                                ;               CLR - TIMEOUT WAITING FOR READY
                                ;
017124  012746  177776          WAITF:: MOV     #177776,-(SP)   ;BIG MSEC TIMER
017130                                  DELAY   250             ;DELAY 100MS
017130  012727  000250                  MOV     #250,(PC)+
017134  000000                          .WORD   0
017136  013727  002116                  MOV     L$DLY,(PC)+
017142  000000                          .WORD   0
017144  005367  177772                  DEC     -6(PC)
017150  001375                          BNE     .-4
017152  005367  177756                  DEC     -22(PC)
017156  001367                          BNE     .-20
017160  016500  000000          2$:     MOV     TSSR(R5),R0     ;READ THE TSSR REGISTER
017164  105700                          TSTB    R0              ;TEST FOR READY BIT SET
                                ;
017166  100421                          BMI     3$              ; EXIT ON STOP FLAG.
017170                                  DELAY   250             ; WAIT 100 MSEC
017170  012727  000250                  MOV     #250,(PC)+
017174  000000                          .WORD   0
017176  013727  002116                  MOV     L$DLY,(PC)+
017202  000000                          .WORD   0
017204  005367  177772                  DEC     -6(PC)
017210  001375                          BNE     .-4
017212  005367  177756                  DEC     -22(PC)
017216  001367                          BNE     .-20
017220                                  BREAK                   ; DO A SUPVSR BREAK FIRST.
017220  104422                          TRAP    C$BRK
017222  005316                          DEC     (SP)            ;REDUCE DELAY COUNT
017224  001355                          BNE     2$              ;RETRY UNTIL TIMER EXPIRES
017226  000241                          CLC                     ; C = 0, CONTROLLER STILL RUNNING...
017230  000401                          BR      4$              ;...OR HUNG-UP.
017232  000261                  3$:     SEC                     ; C = 1, CONTROLLER IS STOPPED.
017234  005326                  4$:     DEC     (SP)+           ;RESTORE STACK WITHOUT CHANGING CARRY BIT
017236  000207                          RTS     PC
```

```
                                .SBTTL  CHKTSSR - CHECK TSSR FOR READY

                        ;+
                        ;
                        ;THIS ROUTINE WAITS FOR READY IN THE TSSR
                        ;AND TESTS FOR AMBIGUOUS BIT SETTINGS IN TSSR.
                        ;
                        ;INPUT:
                        ;
                        ;       R5      ADDRESS OF CSR REGISTERS
                        ;
                        ;OUTPUT:
                        ;
                        ;       R0      CONTENTS OF TSSR
                        ;       CARRY   SET - OKAY
                        ;               CLR - NOT READY AMBIGUOUS, OR SC SET
                        ;
                        ;-

017240                  CHKTSSR:
017240  004737  017124          JSR     PC,WAITF        ;WAIT FOR READY
017244  103014                  BCC     20$             ;BRANCH IF TIME OUT
017246  004737  016720          JSR     PC,CHKAMB       ;TSSR AMBIGOUS?
017252  103006                  BCC     10$             ;BR IF YES
017254  032700  100000          BIT     #SC,R0          ;SPECIAL CONDITION SET?
017260  001405                  BEQ     15$             ;BR IF NO
017262  032700  074000          BIT     #<SCE!BIE!RMR!NXM>,R0   ;ANY ERROR BITS SET?
017266  001402                  BEQ     15$             ;BR IF NO
017270  000241          10$:    CLC                     ;SET FAILURE
017272  000401                  BR      20$             ;
017274  000261          15$:    SEC                     ;SET SUCCESS
017276  000207          20$:    RTS     PC              ;RETURN TO CALLER
```

```
                                        .SBTTL  XNXM    - CHECK FOR NONEXISTENT MEMORY
                                ;+
                                ; ROUTINE TO TEST FOR A NEXM IN THE RANGE (R1) THRU (R2).
                                ; ON RETURN, IF "C" = 1, (R1) = NEXM ADDRESS.
                                ;               "C" = 0, ALL ADDRESSES OK.
                                ;
                                ;CALL:  MOV ADR1,R1
                                ;       MOV ADR2,R2
                                ;       JSR PC,NXM
                                ;       RETURN                  ;TEST "C" AND PROCEED.
                                ;
017300  012737  017332  000004  XNXM:   MOV     #2$,@#4         ; SET BUSERR VECTOR.
017306  012737  000200  000006          MOV     #PRI04,@#6
017314  005003                          CLR     R3              ;FLAG.
017316  005711                  1$:     TST     (R1)            ;TEST THE ADDRESS(ES).
                                                                ;IF ANY TRAP, CONTINUE AT 2$.
017320  020102                          CMP     R1,R2           ;OTHERWISE, CONTINUE HERE.
017322  001407                          BEQ     3$              ;BR IF FINISHED (NO NEXM'S).
017324  062701  000002                  ADD     #2,R1           ;SET NEXT ADDRESS...
017330  000772                          BR      1$              ;...AND CONTINUE.

017332  005103                  2$:     COM     R3              ;GOT ONE, SET FLAG...
017334  012716  017342                  MOV     #3$,(SP)
017340  000002                          RTI                     ;...AND DISMISS INTERRUPT...
017342                          3$:     CLRVEC  #4              ;...AND GIVE BACK THE VECTOR.
017342  012700  000004                  MOV     #4,R0
017346  104436                          TRAP    C$CVEC
017350  005703                          TST     R3              ;DID WE CATCH ONE ??
017352  001401                          BEQ     .+4             ;NO, "C" = 0, SKIP NEXT.
017354  000261                          SEC                     ;YES, "C" = 1, (R1) = NEXM ADDR.
017356  000207                          RTS     PC


                                        .SBTTL  TSTLOOP - CHECK ITERATION COUNT
                                ;+
                                ; SUBROUTINE TO EXECUTE TEST ITERATIONS.
                                ; EXIT WITH "C" SET IF LOOPS ALLOWED AND LOOP COUNT NON-ZERO.
                                ; LOOP COUNTER IS SET BY "BEGIN.TEST" MACRO.
                                ;
                                ; CALL: LOOPTO  ARG
                                ;
017360                          TSTLOOP::
017360  005737  002136                  TST     NOITS           ; ITERATIONS INHIBITED?
017364  001006                          BNE     1$              ; YES.
017366  005737  002154                  TST     QVP             ; NO.
017372  100403                          BMI     1$              ;LOOPS DISALLOWED IN QUICK PASS.
017374  005337  002166                  DEC     LOOPCNT         ; BUMP LOOP COUNTER.
017400  001002                          BNE     2$
017402  000241                  1$:     CLC                     ;LOOP DISALLOWED, OR DONE.
017404  000401                          BR      3$
017406  000261                  2$:     SEC                     ;LOOP ENABLED.
017410  000207                  3$:     RTS     PC
```

```
                                        .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 RASIED TO LEVEL OF
                                ;       THE DEVICE UNDER TEST
                                ;
                                ;-

017412                          TSTSETUP::
017412  010046                          MOV     R0,-(SP)        ;SAVE THE TEST ID MESSAGE
017414  005037  003112                   CLR     SIFLAG          ; CLEAR "SOFT INIT" FLAG
017420  005037  017660                   CLR     ERRK            ; CLEAR LOCAL ERROR COUNTER.
017424  005037  005236                   CLR     EXTA            ; CLEAR ERROR EXTENSION FLAG.
017430  105037  017020                   CLRB    INTMASK         ; CLEAR INTERRUPT MASK (CHECK ERROR)
017434  013700  002152                   MOV     UNITN,R0        ; GET THE UNIT NUMBER,
017440  006300                           ASL     R0              ; ... AND MAKE IT A WORD OFFSET.
017442  005737  003066                   TST     NODEV           ; DID STARTUP FIND THE DEVICE?
017446  001430                           BEQ     4$              ; BR IF YES
017450  100010                           BPL     3$              ; BR IF NOT IDLE
017452  052760  160000  003134           BIS     #160000,ERTABL(R0) ; FLAG ERROR IN THE ERROR TABLE
017460                                   ERRDF   1,NXR,NXRERR    ; NO DEVICE HERE -- PRINT IT
017460  104455                           TRAP    C$ERDF
017462  000001                           .WORD   1
017464  003642                           .WORD   NXR
017466  005202                           .WORD   NXRERR
017470  000407                           BR      2$
017472  052760  160001  003134  3$:      BIS     #160001,ERTABL(R0) ; FLAG ERROR IN THE ERROR TABLE
017500                                   ERRDF   2,NOINIT        ; DEVICE NOT IDLE
017500  104455                           TRAP    C$ERDF
017502  000002                           .WORD   2
017504  004237                           .WORD   NOINIT
017506  000000                           .WORD   0
017510  012737  177777  003064  2$:      MOV     #-1,DUFLG       ; DROP THE UNIT
017516                                   DODU    UNITN
017516  013700  002152                   MOV     UNITN,R0
017522  104451                           TRAP    C$DODU
017524                                   DOCLN                   ; ABORT THE PASS
```

```
      017524  104444                          TRAP    C$DCLN
3174  017526  000423                          BR      5$
3175
3176  017530                          4$:     RFLAGS  R0              ; GET THE OPERATOR FLAGS.
      017530  104421                          TRAP    C$RFLA
3177  017532  032700  001000                  BIT     #PNT,R0         ; PRINT THE TEST NUMBERS?
3178  017536  001412                          BEQ     1$              ; BR IF NO
3179  017540  011600                          MOV     (SP),R0         ;GET THE ID MESSAGE
3180  017542                                  PRINTF  #TNAM,R0        ;DISPLAY THE TEST ID
      017542  010046                          MOV     R0,-(SP)
      017544  012746  017606                  MOV     #TNAM,-(SP)
      017550  012746  000002                  MOV     #2,-(SP)
      017554  010600                          MOV     SP,R0
      017556  104417                          TRAP    C$PNTF
      017560  062706  000006                  ADD     #6,SP
3181  017564  005237  002164          1$:     INC     TSTCNT          ; BUMP TEST COUNTER.
3182  017570                                  SETPRI  IPRI            ;PRIORITY THAT OF DEVICE
      017570  013700  002162                  MOV     IPRI,R0
      017574  104441                          TRAP    C$SPRI
3183  017576  005726                  5$:     TST     (SP)+           ;FIX UP THE STACK
3184  017600  013705  002156                  MOV     CSRADDR,R5      ; ADDRESS OF TSV REGISTERS ON UNIBUS
3185  017604  000207                          RTS     PC
3186  017606     045     123     045  TNAM:   .ASCIZ  '%S%T%A Test'
3187                                          .EVEN
```

```
3189                                            .SBTTL  TSTEND  - PRINT ERRORS RECEIVED
3190                                    ;
3191                                    ; AT END OF EACH TEST, PRINT THE NUMBER OF ERRORS RECEIVED
3192                                    ; IF NORMAL ERROR REPORTING IS DISABLED (FLA:IER).
3193                                    ;
3194 017622                            TSTEND: RFLAGS  R0
     017622  104421                            TRAP    C$RFLA
3195 017624  030027  020000                    BIT     R0,#IER
3196 017630  001412                            BEQ     1$                      ; BR IF "IER" NOT SET.
3197 017632                                    PRINTF  #ESUM,ERRK              ; PRINT ERROR COUNT.
     017632  013746  017660                    MOV     ERRK,-(SP)
     017636  012746  017662                    MOV     #ESUM,-(SP)
     017642  012746  000002                    MOV     #2,-(SP)
     017646  010600                            MOV     SP,R0
     017650  104417                            TRAP    C$PNTF
     017652  062706  000006                    ADD     #6,SP
3198 017656  000207                    1$:     RTS     PC
3199
3200 017660  000000                    ERRK:   0                               ; LOCAL ERROR COUNT.
3201 017662     045     101     040    ESUM:   .ASCIZ  /%A %D%A ERRORS/
3202 017701     105     122     122    EMAXDU: .ASCIZ  /ERROR LIMIT REACHED -- DROPPING UNIT/
3203                                            .EVEN
3204
3205                                            .SBTTL  INCERK  - INCREMENT LOCAL ERROR COUNT
3206                                    ;+
3207                                    ; ROUTINES TO INCREMENT LOCAL ERROR COUNT AND CHECK FOR LIMIT:
3208                                    ;-
3209 017746  005237  017660            INCERK: INC     ERRK                    ; INCREMENT LOCAL ERROR COUNT
3210 017752  010046                            MOV     R0,-(SP)                ; SAVE R0
3211 017754  013700  002152                    MOV     UNITN,R0                ; GET UNIT NUMBER,
3212 017760  006300                            ASL     R0                      ; ... AND MAKE IT A WORD OFFSET.
3213 017762  062700  003134                    ADD     #ERTABL,R0              ; R0 GETS ADDRESS OF ERROR TABLE ENTRY.
3214 017766  005210                            INC     (R0)                    ; INCREMENT THE DEVICE ERROR COUNT
3215 017770  032710  007777                    BIT     #7777,(R0)              ; DID WE OVERFLOW THE FIELD?
3216 017774  001001                            BNE     1$                      ; BR IF NO.
3217 017776  005310                            DEC     (R0)                    ; YES -- BACK IT UP TO 7777.
3218 020000  012600                    1$:     MOV     (SP)+,R0                ; RESTORE R0
3219 020002  000207                            RTS     PC                      ; RETURN TO CALLER.
3220
3221 020004  010046                    CKEMAX: MOV     R0,-(SP)                ; SAVE R0
3222 020006  013700  002152                    MOV     UNITN,R0                ; GET UNIT NUMBER
3223 020012  006300                            ASL     R0                      ; ... AND MAKE IT A WORD OFFSET
3224 020014  016000  003134                    MOV     ERTABL(R0),R0           ; GET ERROR TABLE ENTRY
3225 020020  042700  170000                    BIC     #170000,R0              ; EXTRACT ERROR COUNT FIELD
3226 020024  020037  002144                    CMP     R0,GERRMAX              ; IS GLOBAL LIMIT EXCEEDED FOR THIS UNIT?
3227 020030  103004                            BHIS    1$                      ; BR IF YES
3228 020032  023737  017660  002142            CMP     ERRK,LERRMAX            ; IS LOCAL LIMIT EXCEEDED FOR THIS TEST?
3229 020040  103417                            BLO     2$                      ; BR IF NO
3230 020042                            1$:     RFLAGS  R0                      ; GET OPERATOR FLAGS
     020042  104421                            TRAP    C$RFLA
3231 020044  032700  000040                    BIT     #IDU,R0                 ; IS DROPPING INHIBITED?
3232 020050  001013                            BNE     2$                      ; BR IF YES.
3233 020052  012737  177777  003064            MOV     #-1,DUFLG               ; NO -- DROP THE UNIT
3234 020060                                    ERRDF   4,EMAXDU
     020060  104455                            TRAP    C$ERDF
     020062  000004                            .WORD   4
     020064  017701                            .WORD   EMAXDU
```

```
      020066  000000                          .WORD   0
 3235 020070                                  DODU    UNITN
      020070  013700  002152                  MOV     UNITN,R0
      020074  104451                          TRAP    C$DODU
 3236 020076                                  DOCLN
      020076  104444                          TRAP    C$DCLN
 3237 020100  012600                  2$:     MOV     (SP)+,R0        ; RESTORE R0
 3238 020102  000207                          RTS     PC              ; RETURN TO CALLER
 3239                                         .SBTTL  FATCHK - INC FATAL ERRORS AND CHECK FOR LIMIT
 3240                                 ;+
 3241                                 ;
 3242                                 ;       CHECK FATAL COUNTER, AFTER INC, FOR MORE THAN 25
 3243                                 ;       ERRORS AND IF OVER CALL UNIT DROP ROUTINE
 3244                                 ;
 3245                                 ;-
 3246 020104                          FATCHK:
 3247 020104                                  SAVREG                  ;BETTER SAVE THE REGISTERS
 3248 020110  013701  002152                  MOV     UNITN,R1        ;PICK UP THE UNIT NUMBER
 3249 020114  006301                          ASL     R1              ;MAKE IT INTO A BYTE OFFSET
 3250 020116  062761  000001  003134          ADD     #1,ERTABL(R1)   ;ADD 1 TO THE PROPER UNIT'S ERROR COUNTER
 3251 020124  005237  002172                  INC     FATFLG          ;BUMP FATAL ERROR COUNTER
 3252 020130  023727  002172  000031          CMP     FATFLG,#25.     ;CHECK AGAINST 25
 3253 020136  002406                          BLT     9$              ;BR, IF LESS THAN 25 ERRORS
 3254 020140                                  RFLAGS  R0              ;READ THE FLAGS INTO R0
      020140  104421                          TRAP    C$RFLA
 3255 020142  032700  040000                  BIT     #BIT14,R0       ;BR, IF LOOP ON ERROR IS SET
 3256 020146  001002                          BNE     9$              ;OTHERWISE NEVER BE ABLE TO SCOPE ETC.
 3257 020150  004737  020156                  JSR     PC,CKDROP       ;DROP UNIT IF ALLOWED
 3258 020154  000207                  9$:     RTS     PC              ;RETURN ETC.
 3259                                 ;
 3260                                 ;
 3261                                 ;
```

```
                                        .SBTTL  CKDROP  - CHECK IF UNIT SHOULD BE DROPPED
                                ;+
                                ; CHECK IF UNIT SHOULD BE DROPPED
                                ;-
020156  010046                  CKDROP: MOV     R0,-(SP)
020160                                  FORCERROR       1$,NOTSSR
020170                                  RFLAGS  R0
020170  104421                          TRAP    C$RFLA
020172  032700  000040                  BIT     #IDU,R0
020176  001010                          BNE     1$
020200  011600                          MOV     (SP),R0
020202  012737  177777  003064          MOV     #-1,DUFLG
020210                                  DODU    UNITN
020210  013700  002152                  MOV     UNITN,R0
020214  104451                          TRAP    C$DODU
020216                                  DOCLN                   ;ABORT THE PASS
020216  104444                          TRAP    C$DCLN
020220  012600                  1$:     MOV     (SP)+,R0
020222  000207                          RTS     PC


                                        .SBTTL  CONFIG  - DETERMINE CONFIGURATION OF SYSTEM
                                ;
                                ; SUBROUTINE - DETERMINE CONFIGURATION OF TK-25 SYSTEM.
                                ;
020224                          CONFIG:
020224  004737  016650                  JSR     PC,SOFINIT
020230  000207                          RTS     PC
```

```
                                        .SBTTL  KTON,KTOFF      - ENABLE/DISABLE MEMORY MANAGEMENT
                                ;
                                ; SUBROUTINE - ENABLE MEM MGT.
                                ;
020232  005737  003104        KTON:   TST     KTFLG           ; GOT KT?
020236  001403                        BEQ     1$              ; NO.
020240  012737  000001  177572        MOV     #1,SR0          ; YES. ENABLE KT11.
020246  000207                1$:     RTS     PC


                                ;
                                ; SUBROUTINE - DISABLE MEM MGT.
                                ;
020250  005737  003104        KTOFF:  TST     KTFLG           ; GOT KT11?
020254  001405                        BEQ     1$              ; NO.
020256  000240                        NOP
020260  000240                        NOP
020262  012737  000000  177572        MOV     #0,SR0          ; DISABLE KT.
020270  000207                1$:     RTS     PC


```

```
                        .SBTTL  SETMAP  - SETUP PAR6 MAPPING

                ;+
                ;
                ;THIS ROUTINE SETS UP KERNEL PAR6 TP HANDLE
                ;AN 18 BIT ADDRESS. THE OFFSET INTO THE PAGE
                ;IS RETURNED BIASED TO PAR6.
                ;
                ;INPUTS:
                ;
                ;       R0      HIGH ORDER ADDRESS BITS
                ;       R1      LOW ORDER ADDRESS BITS
                ;
                ;OUTPUTS:
                ;
                ;       R0      OFFSET INTO BLOCK WITH PAR6 BIAS (I.E. THE ADDRESS)
                ;       CARRY   SET IF SUCCESS
                ;               CLR IF ERROR
                ;-
020272          SETMAP:
020272                  SAVREG                          ;SAVE R1-R4 UNTIL NEXT RETURN
020276  005737  003104          TST     KTFLG           ;SYSTEM HAVE ABOVE 28K?
020302  001433                  BEQ     10$             ;BR IF NO
020304  010102                  MOV     R1,R2           ;SAVE LOW ORDER BITS
        000006                  .REPT   6
                                ASR     R0              ;CONVERT WORD ADDRESS TO 32W BLOCKS
                                ROR     R1              ;MAKE IT DOUBLE PRECISION
                                .ENDR
020336  042701  000177          BIC     #177,R1         ;ALINE FOR LOWER 4K BOUNDARY
020342  020137  003104          CMP     R1,KTFLG        ;HIGHER THAN EXISTING MEMORY?
020346  103011                  BHIS    10$             ;BR IF YES
020350  010137  172354          MOV     R1,@#KIPAR6     ;SETUP MAPPING REGISTER PAR6
020354  042702  160000          BIC     #160000,R2      ;SETUP DISPLACEMENT IN PAGE
020360  062702  140000          ADD     #140000,R2      ;ADD IN PAR6 BIAS
020364  010200                  MOV     R2,R0           ;RETURN IN R0
020366  000261                  SEC                     ;SET SUCCESS
020370  000401                  BR      15$             ;
020372  000241          10$:    CLC                     ;SET FAILURE
020374  000207          15$:    RTS     PC              ;RETURN
```

```
                                        .SBTTL  FILLMEM - FILL MEMORY WITH BACKGROUND PATTERN
                                ;+
                                ;  FILL MEMORY WITH A BACKGROUND PATTERN
                                ;
                                ; INPUTS:
                                ;
                                ;       R0 = BACKGROUND PATTERN
                                ;       FREE    = FIRST LOCATION AVAILABLE TO DIAGNOSTIC
                                ;       KTFLG   = SET TO HIGHEST MEMORY LOCATION IF > 28K.
                                ;
                                ; OUTPUTS:
                                ;
                                ;       NONE
                                ;-
                                ;
020376                          FILLMEM:
020376                                  SAVREG                  ;SAVE R1-R5 UNTIL NEXT RETURN
020402  004737  020250                  JSR     PC,KTOFF        ;DISABLE KT.
020406  010003                          MOV     R0,R3           ;COPY TEST PATTERN
020410  013701  003076                  MOV     FREE,R1         ;GET FIRST FREE LOCATION
020414  013702  003100                  MOV     FRESIZ,R2       ;SIZE OF FREE SPACE BELOW 28K.
020420  010321                  10$:    MOV     R3,(R1)+        ;STORE A BACKGROUND WORD
020422  005302                          DEC     R2              ;DONE ALL MEMORY IN FREE SPACE?
020424  003375                          BGT     10$             ;BR IF NO
020426  005737  003104                  TST     KTFLG           ; GOT KT?
020432  001452                          BEQ     55$             ; NO. GET OUT.
020434  004737  020232                  JSR     PC,KTON         ; YES. ENABLE KT.
020440  005000                          CLR     R0              ;HIGH ORDER ADDRESS START
020442  013701  003110                  MOV     PST32W,R1       ;GET >28K START ADDRESS (IN 32W BLOCKS)
        000006                          .REPT   6
                                        CLC                     ;CLEAR C BIT
                                        ROL     R1              ;CONVERT BLOCKS TO WORDS
                                        ROL     R0              ;MAKE IT DOUBLE PRECISION
                                        .ENDR
020512  004737  020272                  JSR     PC,SETMAP       ;SETUP PAR6 MAPPING REGISTER
020516  010320                  30$:    MOV     R3,(R0)+        ;STORE TEST PATTERN IN >28K ADDRESS
020520  020027  160000                  CMP     R0,#160000      ;END OF PAR6 MAPPING AREA?
020524  103774                          BLO     30$             ;BR IF NO
020526  162700  020000                  SUB     #20000,R0       ;BACKUP INTO PAR6 MAPPING BEGIN
020532  062737  000200  172354          ADD     #200,@#KIPAR6   ;POINT TO NEXT 4K BLOCK >28K.
020540  023737  172354  003104          CMP     @#KIPAR6,KTFLG  ;END OF MEMORY?
020546  001402                          BEQ     50$             ;BR IF YES
020550  000137  020516                  JMP     30$             ;KEEP GOING ON ETC.
020554  004737  020250          50$:    JSR     PC,KTOFF        ; DISABLE KT.
020560  000207                  55$:    RTS     PC
```

```
                                .SBTTL  CMPMEM  - COMPARE MEMORY TO BACKGROUND PATTERN
                              ;+
                              ; COMPARE MEMORY WITH A BACKGROUND PATTERN
                              ;
                              ; INPUTS:
                              ;
                              ;       R0 = BACKGROUND PATTERN
                              ;       FREE    = FIRST LOCATION AVAILABLE TO DIAGNOSTIC
                              ;       KTFLG   = SET TO HIGHEST MEMORY LOCATION IF > 28K.
                              ;
                              ; OUTPUTS:
                              ;
                              ;       CARRY   - SET IF NO ERROR
                              ;       CARRY   - CLR IF ERROR
                              ;
                              ; IMPLICIT OUTPUTS:
                              ;
                              ;       ERRHI   - ERROR HIGH ADDRESS
                              ;       ERRLO   - ERROR LOW ADDRESS
                              ;       EXPD    - EXPECTED DATA
                              ;       RECV    - RECEIVED DATA
                              ;-
020562                        CMPMEM:
020562                                SAVREG                  ;SAVE R1-R5 UNTIL NEXT RETURN
020566  010003                        MOV     R0,R3           ;COPY TEST PATTERN
020570  004737  020250                JSR     PC,KTOFF        ;DISABLE KT.
020574  013701  003076                MOV     FREE,R1         ;GET FIRST FREE LOCATION
020600  013702  003100                MOV     FRESIZ,R2       ;SIZE OF FREE SPACE BELOW 28K.
020604  020311                10$:    CMP     R3,(R1)         ;FREE SPACE LOCATION EQUAL TO EXPD?
020606  001411                        BEQ     15$             ;BR IF YES
020610  010137  002206                MOV     R1,ERRLO        ;SAVE ADDRESS IN ERROR
020614  005037  002204                CLR     ERRHI           ;NO HIGH ADDRESS
020620  010337  002200                MOV     R3,EXPD         ;SAVE EXPD FOR ERROR REPORT
020624  011137  002202                MOV     (R1),RECV       ;SAVE RECV FOR ERROR REPORT
020630  000474                        BR      50$             ;
020632  005721                15$:    TST     (R1)+           ;POINT TO NEXT ADDRESS
020634  005302                        DEC     R2              ;DONE ALL MEMORY IN FREE SPACE?
020636  003362                        BGT     10$             ;BR IF NO
020640  005737  003104                TST     KTFLG           ; GOT KT?
020644  001472                        BEQ     55$             ; NO. GET OUT.
020646  004737  020232                JSR     PC,KTON         ; YES. ENABLE KT.
020652  005000                        CLR     R0              ;HIGH ORDER ADDRESS START
020654  013701  003110                MOV     PST32W,R1       ;GET >28K START ADDRESS (IN 32W BLOCKS)
        000006                        .REPT   6
                                      ROL     R1              ;CONVERT BLOCKS TO WORDS
                                      ROL     R0              ;MAKE IT DOUBLE PRECISION
                                      .ENDR
020710  042701  000177                BIC     #177,R1         ;ALINE 4K BOUNDARY
020714  010046                        MOV     R0,-(SP)        ;SAVE HIGH ORDER
020716  010146                        MOV     R1,-(SP)        ;SAVE LOW ORDER
020720  004737  020272                JSR     PC,SETMAP       ;SETUP PAR6 MAPPING REGISTER
020724  010004                        MOV     R0,R4           ;COPY ADDRESS BIASED TO PAR6
020726  012601                        MOV     (SP)+,R1        ;RESTORE LOW ORDER IN NON PAR6 FORMAT
020730  012600                        MOV     (SP)+,R0        ;RESTORE HIGH ORDER IN NON PAR6 FORMAT
020732  020314                30$:    CMP     R3,(R4)         ;ABOVE 28K LOCATION EQUAL EXPD?
020734  001411                        BEQ     32$             ;BR IF YES
020736  010037  002204                MOV     R0,ERRHI        ;SAVE HIGH ORDER IN ERROR
```

```
3462 020742  010137  002206                MOV     R1,ERRLO        ;SAVE LOW ORDER IN ERROR
3463 020746  010337  002200                MOV     R3,EXPD         ;SAVE EXPD FOR ERROR REPORT
3464 020752  011437  002202                MOV     (R4),RECV       ;SAVE RECV FOR ERROR REPORT
3465 020756  000421                        BR      50$             ;
3466 020760  062701  000002          32$:  ADD     #2,R1           ;UPDATE NON PAR6 ADDRESS
3467 020764  005500                        ADC     R0              ;MAKE IT DOUBLE PRECISION ADD
3468 020766  062704  000002                ADD     #2,R4           ;UPDATE PAR FORMAT ADDRESS
3469 020772  020427  160000                CMP     R4,#160000      ;END OF PAR6 MAPPING AREA?
3470 020776  103755                        BLO     30$             ;BR IF NO
3471 021000  162704  020000                SUB     #20000,R4       ;BACKUP INTO PAR6 MAPPING BEGIN
3472 021004  062737  000200  172354        ADD     #200,@#KIPAR6   ;POINT TO NEXT 4K BLOCK >28K.
3473 021012  023737  172354  003104        CMP     @#KIPAR6,KTFLG  ;END OF MEMORY?
3474 021020  101744                        BLOS    30$             ;BR IF NO
3475 021022  004737  020250          50$:  JSR     PC,KTOFF        ;TURN OFF MEMORY MAPPING
3476 021026  000241                        CLC                     ;SET FAILURE
3477 021030  000403                        BR      60$             ;
3478 021032  004737  020250          55$:  JSR     PC,KTOFF        ;TURN OFF MEMORY MAPPING
3479 021036  000261                        SEC                     ;SET SUCCESS
3480 021040  000207                60$:    RTS     PC
3481
```

```
                                .SBTTL  REGSAV  - SAVE R1-R5 ON STACK
                        ;+
                        ;
                        ;ROUTINE TO
                        ;SAVE R1 THROUGH R5 ON THE STACK
                        ;
                        ;CALLING SEQUENCE:
                        ;
                        ;       JSR     R5,REGSAV
                        ;
                        ;THIS IS A COOROUTINE WHICH TRANSFER CONTROL BACK TO
                        ;THE CALLING ROUTINE. AT THE END OF THE CALLING ROUTINE,
                        ;THE RTS PC RETURNS CONTROL TO THIS ROUTINE TO RESTORE
                        ;REGISTERS.
                        ;
                        ;THIS ROUTINE SHOULD ONLY BE CALLED FROM ROUTINES WHICH ARE
                        ;CALLED VIA A JSR PC INSTRUCTION
                        ;
                        ;-

021042                  REGSAV:
021042                          BREAK                   ;LOOK FOR CNTL C
021042  104422                  TRAP    C$BRK
021044  010446                  MOV     R4,-(SP)
021046  010346                  MOV     R3,-(SP)
021050  010246                  MOV     R2,-(SP)
021052  010146                  MOV     R1,-(SP)
021054  010546                  MOV     R5,-(SP)
021056  016605  000012          MOV     10.(SP),R5
021062  004736                  JSR     PC,@(SP)+
021064  012601                  MOV     (SP)+,R1
021066  012602                  MOV     (SP)+,R2
021070  012603                  MOV     (SP)+,R3
021072  012604                  MOV     (SP)+,R4
021074  012605                  MOV     (SP)+,R5
021076                          BREAK                   ;LOOK FOR CNTL C
021076  104422                  TRAP    C$BRK
021100  000207                  RTS     PC
```

```
                                .SBTTL  GETPAT  - GET 8 BIT PATTERN FROM OPERATOR
                            ;+
                            ;
                            ;ROUTINE TO REQUEST AN 8 BIT DATA PATTERN FROM THE OPERATOR
                            ;
                            ;INPUTS:
                            ;
                            ;       NONE.
                            ;
                            ;OUTPUTS:
                            ;
                            ;       R0      OCTAL NUMBER FROM THE OPERATOR
                            ;
                            ;CALLING SEQUENCE:
                            ;
                            ;       JSR     PC,GETPAT
                            ;
                            ;-

021102                      GETPAT::
021102                              SAVREG                      ;SAVE THE GENERAL REGISTERS
021106                      1$:     GMANID  DATASC,PATDAT,0,377,0,377,NO
021106  104443                      TRAP    C$GMAN
021110  000406                      BR      10000$
021112  021136                      .WORD   PATDAT
021114  000022                      .WORD   T$CODE
021116  021140                      .WORD   DATASC
021120  000377                      .WORD   377
021122  000000                      .WORD   T$LOLIM
021124  000377                      .WORD   T$HILIM
021126                      10000$:
021126                              BNCOMPLETE      1$      ;RETRY IF ERROR
021126  103367                      BCC     1$
021130  013700  021136            MOV     PATDAT,R0       ;DATA PATTERN FROM OPERATOR
021134  000207                      RTS     PC              ;RETURN TO CALLER

                            ;+
                            ;LOCAL DATA AREA
                            ;-

021136  000000              PATDAT: .WORD   0               ;TEMPORARY STORAGE FOR DATA
021140    105   116   124   DATASC: .ASCIZ  'ENTER DATA PATTERN'
                                    .EVEN
```

```
                                .SBTTL  GETSEL  - ISSUE MENU AND GET OPERATOR RESPONSE
                        ;+
                        ;ROUTINE TO ISSUE A MENU AND GET
                        ;THE OPERATOR'S RESPONSE.
                        ;
                        ;INPUTS:
                        ;
                        ;       R0      ADDRESS OF ASCIZ STRING OF MENU
                        ;       R1      MAXIMUM ALLOWABLE OPERATOR RESPONSE
                        ;
                        ;OUTPUTS:
                        ;
                        ;       R0      NUMBER OF THE OPERATOR'S SELECTION
                        ;-
021164                  GETSEL::
021164                          SAVREG                          ;SAVE GENERAL REGISTERS
021170  010002                  MOV     R0,R2                   ;SAVE THE MENU ADDRESS
021172  010203          1$:     MOV     R2,R3                   ;START OF MENU STRING
021174  005713          2$:     TST     (R3)                    ;END OF ASCII ?
021176  001412                  BEQ     3$                      ;BRANCH IF ALL LINES DISPLAYED
021200                          PRINTF  #SELASC,(R3)+           ;DISPLAY THE MENU
021200  012346                  MOV     (R3)+,-(SP)
021202  012746  021350          MOV     #SELASC,-(SP)
021206  012746  000002          MOV     #2,-(SP)
021212  010600                  MOV     SP,R0
021214  104417                  TRAP    C$PNTF
021216  062706  000006          ADD     #6,SP
021222  000764                  BR      2$
021224                  3$:     GMANID  MENASC,MENRES,D,-1,0,-1,NO
021224  104443                  TRAP    C$GMAN
021226  000406                  BR      10001$
021230  021404                  .WORD   MENRES
021232  000042                  .WORD   T$CODE
021234  021355                  .WORD   MENASC
021236  177777                  .WORD   -1
021240  000000                  .WORD   T$LOLIM
021242  177777                  .WORD   T$HILIM
021244                  10001$:
021244                          BNCOMPLETE      1$      ;RETRY IF ERROR
021244  103352                  BCC     1$
021246  013700  021404          MOV     MENRES,R0       ;GET THE OPERATOR'S REPLY
021252  020001                  CMP     R0,R1           ;COMPARE TO MAXIMUM ALLOWED
021254  101411                  BLOS    5$              ;BRANCH IF OK
021256                          PRINTF  #MENERR         ;DISPLAY ERROR MESSAGE
021256  012746  021302          MOV     #MENERR,-(SP)
021262  012746  000001          MOV     #1,-(SP)
021266  010600                  MOV     SP,R0
021270  104417                  TRAP    C$PNTF
021272  062706  000004          ADD     #4,SP
021276  000735                  BR      1$              ;RETRY
021300  000207          5$:     RTS     PC              ;RETURN TO CALLER
021302    045   116   045 MENERR: .ASCIZ  '<N><A *** Menu Selection Too Large ***'
021350    045   116   045 SELASC: .ASCIZ  '<N><T'
021355    105   156   164 MENASC: .ASCIZ  'Enter Menu Selection: '
                        .EVEN
021404  000000          MENRES:         .WORD   0
```

SEQ 0103

```
                                        .SBTTL  CHKMAN  - CHECK MANUAL INTERVENTION LEGALITY
                                ;+
                                ;
                                ;ROUTINE TO TEST FOR MANUAL INTERVENTION LEGALITY.
                                ;
                                ;INPUT:
                                ;
                                ;       NONE.
                                ;
                                ;OUTPUT:
                                ;
                                ;       CARRY   0       MANUAL INTERVENTION NOT ALLOWED
                                ;               1       MANUAL INTERVENTION IS OK
                                ;
                                ;SIDE EFFECTS:
                                ;
                                ;       A MESSAGE IS DISPLAYED WARNING THAT TEST IS
                                ;       NOT EXECUTED IF MANUAL INTERVENTION IS NOT
                                ;       ALLOWED.
                                ;
                                ;-

021406                          CHKMAN::
                                        SAVREG                  ;SAVE THE REGISTERS
021406                                  MANUAL                  ;SEE IF MANUAL INTERVENTION OK
021412  104450                          TRAP    C$MANI
021414                                  BCOMPLETE 1$            ;BRANCH IF ALLOWED
021414  103411                          BCS     1$
021416                                  PRINTF  #NOMAN          ;PRINT THE WARNING MESSAGE
021416  012746  021442                  MOV     #NOMAN,-(SP)
021422  012746  000001                  MOV     #1,-(SP)
021426  010600                          MOV     SP,R0
021430  104417                          TRAP    C$PNTF
021432  062706  000004                  ADD     #4,SP
021436  000241                          CLC                     ;CLEAR CARRY FOR ERROR
021440  000207                  1$:     RTS     PC              ;RETURN

021442      045     116     045 NOMAN:  .ASCIZ  '%N%A *** Manual Intervention not Allowed - Test Aborted ***'
                                        .even
```

```
                                        .SBTTL  ENVIRN  - SETUP FREE DIAGNOSTIC SPACE
                                ;
                                ; SUBROUTINE TO SET-UP VARIOUS ENVIRONMENTAL PARAMETERS.
                                ;
021536                          ENVIRN: MEMORY  R0
021536  104431                          TRAP    C$MEM
021540  010037  003076                  MOV     R0,FREE         ; GET 1ST FREE ADDRESS...
021544  062737  000002  003076          ADD     #2,FREE
021552  011037  003100                  MOV     (R0),FRESIZ     ;...AND WORD COUNT.
021556  162737  000004  003100          SUB     #4,FRESIZ
021564  013702  002012                  MOV     L$UNIT,R2       ; GET NUMBER OF UNITS
021570  162737  000007  003100  10$:     SUB     #7,FRESIZ       ; TAKE AWAY 7 WORDS PER UNIT
021576  005302                          DEC     R2
021600  001373                          BNE     10$
021602  013700  003076                  MOV     FREE,R0         ;GET FIRST FREE ADDRESS
021606  063700  003100                  ADD     FRESIZ,R0       ;POINT TO LAST FREE ADDRESS
021612  162700  000002                  SUB     #2,R0           ;BACKUP 1 WORD
021616  010037  003102                  MOV     R0,FREEHI       ;STORE LAST FREE ADDRESS
021622  000207                          RTS     PC              ;RETURN
```

```
                                .SBTTL  KTINIT  - SETUP KT11 MEMORY MANAGEMENT REGISTERS
                        ;*
                        ;
                        ;ROUTINE TO INIT KT-11
                        ;
                        ;-

021624                  KTINIT:
021624  005037  003104          CLR     KTFLG           ; INIT >28K MEMORY FLAG
021630  005037  003106          CLR     KTENABLE        ; INIT TEST >28K FLAG
021634  023727  002120  001577  CMP     L$HIME,#1577    ; GOT ENOUGH MEMORY (>28K)?
021642  101444                  BLOS    9$              ; NO.
021644  013700  000004          MOV     @#ERRVEC,R0     ; SAVE OLD ERR VEC PTR.
021650  012737  021742  000004  MOV     #2$,@#ERRVEC    ; SET ERR VEC PTR.
021656  005737  177572          TST     @#SR0           ; GOT KT11?
021662  000240                  NOP                     ;  (TRAP IF NO).
021664  013737  002120  003104  MOV     L$HIME,KTFLG    ; YES. SET KT FLAG.
021672  042737  000177  003104  BIC     #177,KTFLG      ;
021700  010037  000004          MOV     R0,@#ERRVEC     ; RESTORE OLD ERR VEC PTR.
021704  005000                  CLR     R0              ; R0 = AR DATA.
021706  012701  172340          MOV     #KIPAR0,R1      ; R1 = KI REGS PTR.
021712  012761  077406  177740  1$:     MOV     #77406,-40(R1)  ; SET DESCRIPTOR REG.
021720  010021                  MOV     R0,(R1)+        ; SET KIPAR REG.
021722  062700  000200          ADD     #200,R0         ; BUMP AR DATA BY "4K".
021726  020027  002000          CMP     R0,#2000        ; AT "I/O"?
021732  001367                  BNE     1$              ; NO.
021734  012741  177600          MOV     #177600,-(R1)   ; YES. SET KTPAR7 FOR I/O.
021740  000405                  BR      9$
                        
021742  012716  021750  2$:     MOV     #6$,(SP)        ; SET UP RETURN
021746  000002                  RTI                     ; RTI TO NEXT LOCATION
                        
021750  010037  000004  6$:     MOV     R0,@#ERRVEC     ; RESTORE OLD ERR VEC PTR.
                        
021754  000207          9$:     RTS     PC
                        
                        
                        
```

```
                                        .SBTTL  PROTECTION TABLE
021756                                  BGNPROT
021756                          L$PROT::
021756  177777  177777  177777          .WORD   -1, -1, -1, -1  ;NO DEVICE PROTECTION REQUIRED.
021766                                  ENDPROT

```

```
                                .SBTTL  INITIALIZE SECTION

                        ;++
                        ;THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                        ;AT THE BEGINNING OF EACH PASS.
                        ;
                        ;IF "START" OR "RESTART", SET QUICK-PASS FLAG AND BUS-INIT.
                        ;IF "CONTINUE", NOTHING IS REQUIRED.
                        ;
                        ;--
021766                          BGNINIT
021766                  L$INIT::
021766                  40$:
021766  012737  005676  002150          MOV     #EPRT1,EPRTSW   ;SET UP PRIMARY MESSAGE FOR REPLACEMENT
021774  005037  003112                  CLR     SIFLAG          ;CLEAR "SOFT INIT" FLAG
022000  005037  003106                  CLR     KTENABLE        ;CLEAR TEST ABOVE 28K FLAG
022004  005037  002250                  CLR     RAMSIZ          ;CLEAR RAM SIZE FOR RAMERR ROUTINE
022010                                  READEF  #EF.CONTINUE
022010  012700  000076                  MOV     #EF.CONTINUE,R0
022014  104447                          TRAP    C$REFG
022016                                  BNCOMPLETE 1$
022016  103023                          BCC     1$
022020  023737  002152  002012          CMP     UNITN,L$UNIT    ;UNIT IN RANGE?
022026  103066                          BHIS    4$              ;BR IF NO.
022030  005737  003064                  TST     DUFLG           ;DROPPED UNIT?
022034  100470                          BMI     NXTU            ;BR IF YES
022036  013701  002152                  MOV     UNITN,R1
022042  006301                          ASL     R1
022044  005761  003134                  TST     ERTABL(R1)
022050  001514                          BEQ     SETU
022052  032761  040000  003134          BIT     #BIT14,ERTABL(R1) ;DROPPED?
022060  001056                          BNE     NXTU
022062                                  EXIT    INIT            ;DO NOTHING IF "CONTINUE".
022062  104432                          TRAP    C$EXIT
022064  000416                          .WORD   L10030-.
022066                          1$:     READEF  #EF.NEW
022066  012700  000035                  MOV     #EF.NEW,R0
022072  104447                          TRAP    C$REFG
022074                                  BNCOMPLETE NXTU         ;TAKE NEXT UNIT IF NOT NEW PASS.
022074  103050                          BCC     NXTU
022076                                  READEF  #EF.START
022076  012700  000040                  MOV     #EF.START,R0
022102  104447                          TRAP    C$REFG
022104                                  BCOMPLETE 2$
022104  103404                          BCS     2$
022106                                  READEF  #EF.RESTART
022106  012700  000037                  MOV     #EF.RESTART,R0
022112  104447                          TRAP    C$REFG
022114                                  BNCOMPLETE 31$
022114  103027                          BCC     31$
022116                          2$:                             ;1ST PASS, BUS-INIT...
022116                                  BRESET                  ;BUS RESET.
022116  104433                          TRAP    C$RESET
022120  005037  002164                  CLR     TSTCNT          ;NUMBER OF TESTS RUN IN PASS
022124  005037  002722                  CLR     FLLTSW          ;SHOW 1ST PASS ON FAULT LIGHT MESSAGE SW
022130  005037  002172                  CLR     FATFLG          ;RESET FLAG TO ZERO "FATAL ERRORS"
022134  005037  003336          19$:    CLR     SKIPT           ;CLEAR THE SUBTEST "SKIPPER"
```

```
                                20$:
                                        MOV     #-1,QVP         ;...QUICK VERIFY...
                                        JSR     PC,ENVIRN       ;SET ENVIRONMENT.
                                        JSR     PC,KTINIT       ;INITIALIZE KT MEMORY MANAGEMENT
                                        MOV     #ERTABL,R0
                                30$:    CLR     (R0)+           ;CLEAR THE ERROR TABLE
                                        CMP     R0,#ERTABE
                                        BLO     30$
                                        BR      4$
                                31$:    CLR     QVP
                                        JMP     PASRPT          ;GO REPORT THE STATUS

                                4$:
                                NEWPAS: MOV     #-1,UNITN       ;INIT UNIT NUMBER...
                                        CLR     DEVCNT          ;CLEAR COUNT OF DEVICES RUNNING
                                NXTU:   BREAK
                                        TRAP    C$BRK
                                        INC     UNITN           ;...AND SET NEXT UNIT NUMBER.
                                        CMP     UNITN,L$UNIT
                                        BLO     SETU
                                        MOV     #-1,DUFLG
                                        BR      11$
                                        DOCLN                   ;ABORT, NO MORE UNITS.
                                        TRAP    C$DCLN
                                11$:    NOP
                                PASRPT:
                                        CMP     L$UNIT,#1       ;HOW MANY UNITS SELECTED?
                                        BLOS    NEWPAS          ;BR IF ONLY 1
                                        TST     DEVCNT          ;ARE ANY STILL RUNNING?
                                        BEQ     NEWPAS          ;BR IF NO
                                        RFLAGS  R0
                                        TRAP    C$RFLA
                                        BIT     #ISR,R0         ;SHOULD WE PRINT STATISTICS
                                        BNE     NEWPAS          ;BR IF NO

                                        DORPT
                                        TRAP    C$DRPT
                                        BR      NEWPAS
                                10$:

                                SETU:   GPHARD  UNITN,R0        ;GET UNIT N P-TABLE POINTER.
                                        MOV     UNITN,R0
                                        TRAP    C$GPHRD
                                        BNCOMPLETE NXTU         ;BR IF UNIT NOT AVAILABLE.
                                        BCC     NXTU
                                        CLR     DUFLG           ;CLEAR "DROPPED" FLAG.
                                        INC     DEVCNT
                                        MOV     (R0)+,R1        ;GET 1ST REGISTER ADDRESS.
                                        MOV     R1,CSRADDR      ;ADDRESS OF REGISTERS OF UNIT UNDER TEST

                                        MOV     (R0)+,R1        ;GET VECTOR ADDRESS.
                                        MOV     (R0),R2         ;GET INTERRUPT PRIORITY
                                        MOV     R2,IPRI         ;SET INTERRUPT PRIORITY.
                                        MOV     R1,IVEC         ;SET INTERRUPT VECTOR POINTER...
                                        MOV     #INTR,(R1)+     ;...VECTOR...
                                        MOV     R2,(R1)+        ;...AND PRIORITY.
```

```
3793 022352                          1$:
3794                                 ;       TST     QVP             ;1ST PASS ??
3795                                 ;       BEQ     5$              ;NO, SKIP THE PASS 1 STUFF.
3796
3797                                 ;
3798                                 ;1ST PASS, CHECK THAT DEVICE ADDRESSES ARE VALID, AND
3799                                 ;THAT THE DISPLAY STATUS IS PROPERLY INITIALIZED.
3800                                 ;
3801 022352  013701  002152                  MOV     UNITN,R1
3802 022356  006301                          ASL     R1
3803 022360  052761  100000  003134          BIS     #BIT15,ERTABL(R1) ;SAY DEVICE RUNNING
3804 022366  005037  005236                  CLR     EXTA            ;CLEAR ERROR EXTENSION FLAG.
3805 022372  023727  002012  000001          CMP     L$UNIT,#1       ;ARE WE TESTING MULTIPLE UNITS?
3806 022400  101416                          BLOS    10$             ;BR IF NO.
3807 022402                                  RFLAGS  R0              ;YES -- GET OPERATOR FLAGS.
     022402  104421                          TRAP    C$RFLA
3808 022404  032700  001000                  BIT     #PNT,R0         ;SHOULD WE PRINT UNIT #?
3809 022410  001412                          BEQ     10$             ;BR IF NOT.
3810 022412                                  PRINTF  #PUNIT,UNITN    ;PRINT THE UNIT #
     022412  013746  002152                  MOV     UNITN,-(SP)
     022416  012746  022504                  MOV     #PUNIT,-(SP)
     022422  012746  000002                  MOV     #2,-(SP)
     022426  010600                          MOV     SP,R0
     022430  104417                          TRAP    C$PNTF
     022432  062706  000006                  ADD     #6,SP
3811 022436                          10$:
3812 022436  005037  003066                  CLR     NODEV
3813 022442  013701  002156                  MOV     CSRADDR,R1      ;ADDRESS OF FIRST REGISTER
3814 022446  010102                          MOV     R1,R2           ;START OF REGISTERS
3815 022450  062702  000000                  ADD     #TSSR,R2        ;ADDRESS OF TSSR REGISTER
3816 022454  004737  017300                  JSR     PC,XNXM         ;TEST BOTH CONTROLLER REGISTERS...
3817 022460  103005                          BCC     2$              ;...AND BR IF ALL OK.
3818 022462  010137  003066                  MOV     R1,NODEV        ;FLAG DEVICE AS NON-EXISTENT
3819 022466  012737  177777  003064          MOV     #-1,DUFLG       ;DROP THIS UNIT.
3820 022474                          2$:
3821                                 ;
3822                                 ;FINALLY, SET CPU PRIORITY AND WE'RE DONE.
3823                                 ;
3824 022474                          5$:     SETPRI  #PRI00          ;ENABLE INTERRUPTS.
     022474  012700  000000                  MOV     #PRI00,R0
     022500  104441                          TRAP    C$SPRI
3825 022502                                  ENDINIT
     022502                          L10030:
     022502  104411                          TRAP    C$INIT
3826
3827 022504     045     116     045  PUNIT:  .ASCIZ  /%N%N%A***** TESTING UNIT %D2%A *****/
3828                                         .EVEN
```

```
3830                                    .SBTTL  ADD AND DROP UNITS SECTIONS
3831
3832                                ;++
3833                                ; THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
3834                                ; TO BE (A) ADDED TO THE TEST LIST FOR THE FIRST TIME,
3835                                ;    OR (B) RE-INSERTED IF IT HAD BEEN PREVIOUSLY DROPPED.
3836                                ;--
3837 022552                            BGNAU
     022552                         L$AU::
3838 022552  010001                     MOV     R0,R1           ; GET UNIT TO BE ADDED (R0)
3839 022554  006301                     ASL     R1              ; MAKE IT A WORD INDEX
3840 022556  052761  100000  003134      BIS     #100000,ERTABL(R1) ; SET THE "ACTIVE" BIT
3841 022564  042761  040000  003134      BIC     #40000,ERTABL(R1)  ; CLEAR THE "DROPPED" BIT
3842 022572                             PRINTF  #1$,R0
     022572  010046                     MOV     R0,-(SP)
     022574  012746  022620             MOV     #1$,-(SP)
     022600  012746  000002             MOV     #2,-(SP)
     022604  010600                     MOV     SP,R0
     022606  104417                     TRAP    C$PNTF
     022610  062706  000006             ADD     #6,SP
3843 022614                             EXIT    AU
     022614  000167                     .WORD   J$JMP
     022616  000026                     .WORD   L10031-2-.
3844 022620     045     116     045  1$:     .ASCIZ  /%N%A  UNIT %D%A ADDED/
3845                                    .EVEN
3846
3847 022646                             ENDAU                   ; UNUSED.
     022646                         L10031:
     022646  104452                     TRAP    C$AU
3848                                ;++
3849                                ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
3850                                ; TO BE REMOVED FROM THE TEST LIST.
3851                                ;
3852                                ; SUPVSR DOES THE "DROPPING".  THIS IS JUST TO TELL THE MAN.
3853                                ; "DROPPED" UNITS ARE RE-SELECTED ON OPERATOR "STA" OR "ADD"
3854                                ; COMMAND, OTHERWISE REMAIN INACTIVE.  THE "DISPLAY" COMMAND
3855                                ; WILL PRINT ALL DROPPED UNITS, AND THE P-TABLES OF THOSE
3856                                ; WHICH ARE STILL ACTIVE.
3857                                ; UPON ENTRY, R0 CONTAINS THE UNIT TO BE DROPPED.
3858
3859 022650                             BGNDU
     022650                         L$DU::
3860 022650  012737  177777  003064      MOV     #-1,DUFLG
3861 022656  010001                     MOV     R0,R1
3862 022660  006301                     ASL     R1
3863 022662  052761  140000  003134      BIS     #140000,ERTABL(R1)      ; SAY DROPPED
3864 022670  000240  000240  000240      240,240,240                     ; ?????????
3865 022676                             PRINTF  #1$,R0
     022676  010046                     MOV     R0,-(SP)
     022700  012746  022724             MOV     #1$,-(SP)
     022704  012746  000002             MOV     #2,-(SP)
     022710  010600                     MOV     SP,R0
     022712  104417                     TRAP    C$PNTF
     022714  062706  000006             ADD     #6,SP
3866 022720                             EXIT    DU
     022720  000167                     .WORD   J$JMP
     022722  000030                     .WORD   L10032-2-.
```

```
3867 022724     045     116     045  1$:     .ASCIZ  /%N%A  UNIT %D%A DROPPED/
3868                                         .EVEN
3869 022754                                  ENDDU
     022754                             L10032:
     022754  104453                           TRAP    C$DU
3870                                     ;++
3871                                     ; AUTO-DROP CODE SECTION.
3872                                     ;--
3873 022756                                  BGNAUTO
     022756                             L$AUTO::
3874 022756  012703  000550                 MOV     #360.,R3        ;ENOUGH TIME FOR 2400' REEL TO REWIND
3875 022762  004737  017124            10$: JSR     PC,WAITF        ;WAIT FOR SSR TO SET
3876 022766  103420                         BCS     20$             ;LEAVE WHEN SSR IS SET
3877 022770                                  DELAY   250.            ;WAIT FOR .25 SECONDS
     022770  012727  000372                 MOV     #250.,(PC)+
     022774  000000                         .WORD   0
     022776  013727  002116                 MOV     L$DLY,(PC)+
     023002  000000                         .WORD   0
     023004  005367  177772                 DEC     -6(PC)
     023010  001375                         BNE     .-4
     023012  005367  177756                 DEC     -22(PC)
     023016  001367                         BNE     .-20
3878 023020  005303                         DEC     R3              ;BUMP COUNTER DOWN
3879 023022  001357                         BNE     10$             ;KEEP GOING
3880 023024  004737  020156                 JSR     PC,CKDROP       ;TRY AND DROP UNIT
3881 023030                             20$:
3882 023030                                  ENDAUTO                 ; UNUSED.
     023030                             L10033:
     023030  104461                           TRAP    C$AUTO
```

```
3884                                            .SBTTL  CLEAN-UP AND REPORT CODING SECTIONS
3885
3886                                    ;++
3887                                    ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS
3888                                    ; EXECUTED AT THE END OF EACH PASS (OR SUB-PASS).
3889                                    ; USE TO RETURN DEVICE UNDER TEST TO A NEUTRAL STATE.
3890                                    ;--
3891 023032                                    BGNCLN
     023032                             L$CLEAN::
3892 023032  005737  003064                    TST     DUFLG           ;"DROPPED" FLAG IS SET ON...
3893 023036  100407                            BMI     1$              ;...AND GROSS CONTROLLER FAULT...
3894                                                                   ;...DON'T TRY TO XCT CLEANUP CODE.
3895
3896 023040  013705  002156                    MOV     CSRADDR,R5      ; ADDRESS OF TSV REGISTERS ON UNIBUS
3897 023044  012765  000000  000000            MOV     #0,TSSR(R5)     ;DO SOFT INIT
3898 023052  004737  017124                    JSR     PC,WAITF
3899 023056                             1$:
3900 023056                             2$:    ENDCLN
     023056                             L10034:
     023056  104412                            TRAP    C$CLEAN
3901                                    ;++
3902                                    ; THE REPORT CODING SECTION CONTAINS THE
3903                                    ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
3904                                    ;--
3905 023060                                    BGNRPT
     023060                             L$RPT::
3906 023060                                    PRINTS  #DEVSUM
     023060  012746  023322                    MOV     #DEVSUM,-(SP)
     023064  012746  000001                    MOV     #1,-(SP)
     023070  010600                            MOV     SP,R0
     023072  104416                            TRAP    C$PNTS
     023074  062706  000004                    ADD     #4,SP
3907 023100  010246                            MOV     R2,-(SP)
3908 023102  010346                            MOV     R3,-(SP)
3909 023104  010446                            MOV     R4,-(SP)
3910 023106  012704  003134                    MOV     #ERTABL,R4      ; GET START OF ERROR TABLE.
3911 023112  005003                            CLR     R3              ; CLEAR UNIT NUMBER
3912 023114  011402                     1$:    MOV     (R4),R2         ; GET ERROR TABLE ENTRY & TEST IT.
3913 023116  001467                            BEQ     4$              ; ZERO IF UNIT NOT RUN
3914 023120  100066                            BPL     4$
3915 023122  032702  040000                    BIT     #BIT14,R2       ; WAS UNIT DROPPED?
3916 023126  001015                            BNE     2$              ; BR IF YES
3917 023130  042702  170000                    BIC     #^C7777,R2      ; GET ERROR COUNT FIELD
3918 023134                                    PRINTS  #DEVONL,R3,R2   ; PRINT
     023134  010246                            MOV     R2,-(SP)
     023136  010346                            MOV     R3,-(SP)
     023140  012746  023357                    MOV     #DEVONL,-(SP)
     023144  012746  000003                    MOV     #3,-(SP)
     023150  010600                            MOV     SP,R0
     023152  104416                            TRAP    C$PNTS
     023154  062706  000010                    ADD     #10,SP
3919 023160  000446                            BR      4$
3920 023162  020227  160000             2$:    CMP     R2,#160000      ; WAS UNIT NON-EXISTENT?
3921 023166  001012                            BNE     3$              ; BR IF NO
3922 023170                                    PRINTS  #DEVNXR,R3
     023170  010346                            MOV     R3,-(SP)
     023172  012746  023427                    MOV     #DEVNXR,-(SP)
```

```
        023176  012746  000002                  MOV     #2,-(SP)
        023202  010600                          MOV     SP,R0
        023204  104416                          TRAP    C$PNTS
        023206  062706  000006                  ADD     #6,SP
  3923  023212  000431                          BR      4$
  3924  023214  020227  160001          3$:     CMP     R2,#160001      ; WAS UNIT NOT READY AT STARTUP?
  3925  023220  001012                          BNE     30$             ; BR IF NO.
  3926  023222                                  PRINTS  #DEVNRD,R3
        023222  010346                          MOV     R3,-(SP)
        023224  012746  023511                  MOV     #DEVNRD,-(SP)
        023230  012746  000002                  MOV     #2,-(SP)
        023234  010600                          MOV     SP,R0
        023236  104416                          TRAP    C$PNTS
        023240  062706  000006                  ADD     #6,SP
  3927  023244  000414                          BR      4$
  3928  023246  042702  170000          30$:    BIC     #^C7777,R2
  3929  023252                                  PRINTS  #DEVDRO,R3,R2
        023252  010246                          MOV     R2,-(SP)
        023254  010346                          MOV     R3,-(SP)
        023256  012746  023572                  MOV     #DEVDRO,-(SP)
        023262  012746  000003                  MOV     #3,-(SP)
        023266  010600                          MOV     SP,R0
        023270  104416                          TRAP    C$PNTS
        023272  062706  000010                  ADD     #10,SP
  3930  023276  062704  000002          4$:     ADD     #2,R4
  3931  023302  005203                          INC     R3
  3932  023304  020427  003334                  CMP     R4,#ERTABE
  3933  023310  103701                          BLO     1$
  3934  023312  012604                          MOV     (SP)+,R4
  3935  023314  012603                          MOV     (SP)+,R3
  3936  023316  012602                          MOV     (SP)+,R2
  3937  023320                                  ENDRPT                  ; UNUSED.
        023320                          L10035:
        023320  104425                          TRAP    C$RPT
  3938
  3939
  3940  023322      045     116     045  DEVSUM: .ASCIZ  /%N%ADEVICE STATUS SUMMARY:%N/
  3941  023357      045     101     040  DEVONL: .ASCIZ  /%A  UNIT %D3%A   ONLINE,  ERRORS = %D%N/
  3942  023427      045     101     040  DEVNXR: .ASCIZ  /%A  UNIT %D3%A   DROPPED, NON-EXISTENT REGISTER%N/
  3943  023511      045     101     040  DEVNRD: .ASCIZ  /%A  UNIT %D3%A   DROPPED, NOT READY AT STARTUP%N/
  3944  023572      045     101     040  DEVDRO: .ASCIZ  /%A  UNIT %D3%A   DROPPED, ERRORS = %D%N/
  3945                                          .EVEN
  3948
  3955
  3961
```

```
3971                                    .SBTTL  TEST  1: WRITE TAPE MARK RETRY
3972                                ;*
3973                                ;
3974                                ;THIS TEST VERIFIES PROPER OPERATION OF THE WRITE TAPE MARK RETRY COMMAND (SPACE
3975                                ;REVERSE, ERASE, WRITE TAPE MARK). SUBTESTS ARE AS FOLLOWS:
3976                                ;
3977                                ;
3978                                ;THE TEST CONSISTS OF THE FOLLOWING 4 SUBTESTS
3979                                ;
3980                                ;
3981                                ;
3982                                ;-
3983 023642                             BGNTST
     023642                                                                  T1::
3984 023642  005037  002172             CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
3985 023646  005037  003104             CLR     KTFLG           ;HOLD OFF KT11
3986 023652  012737  005676  002150     MOV     #EPRT1,EPRTSW           ;PRIMARY ERROR MESSAGE
3991 023660  012700  032111             MOV     #TST29ID,R0             ;ASCII MESSAGE TO IDENTIFY TEST
3992 023664  004737  017412             JSR     PC,TSTSETUP             ;DO INITIAL TEST SETUP
3993 023670  012737  000001  002166     MOV     #1,LOOPCNT              ;PERFORM 1 ITERATIONS
3994 023676  005037  026544             CLR     T29CNT                  ;CLEAR TAPE RECORD COUNTER
3995 023702                         T29LOOP:
```

```
                                        ;+
                                        ;
                                        ;TEST 1, SUBTEST 1
                                        ;
                                        ;VERIFIES THAT A WRITE TAPE MARK RETRY COMMAND ISSUED WHILE THE
                                        ;TAPE IS POSITIONED AT BOT CAUSES FUNCTION REJECT TERMINATION, WITH THE
                                        ;NON-EXECUTABLE FUNCTION (NEF) ERROR BIT SET.
                                        ;
                                        ;
                                        ;-
023702                                          BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
023702                                                                                  T1.1:
023702  104402                                                                          TRAP    C$BSUB
023704  004737  032140                        JSR     PC,T29REST              ;SET COMMAND PACKET
023710  004737  032232                        JSR     PC,T29RT2               ;SET UP OTHER COMMAND PACKET
023714  004737  032274                        JSR     PC,T29RT3               ;SET UP OTHER COMMAND PACKET
023720  012737  023420  026550                MOV     #10000..T29DLY          ;SET UP DELAY ROUTINE
023726  004737  016650                10$:    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
023732  103426                              BCS     20$                     ;BR IF INIT WAS OK
023734                                      DELAY   250                     ;DELAY ABOUT .25 SECONDS
023734  012727  000250                                                            MOV     #250,(PC)+
023740  000000                                                                  .WORD   0
023742  013727  002116                                                          MOV     L$DLY,(PC)+
023746  000000                                                                  .WORD   0
023750  005367  177772                                                          DEC     -6(PC)
023754  001375                                                                  BNE     .-4
023756  005367  177756                                                          DEC     -22(PC)
023762  001367                                                                  BNE     .-20
023764  005337  026550                      DEC     T29DLY                  ;BUMP DELAY ROUTINE DOWN
023770  001356                              BNE     10$                     ;BR, IF MORE DELAY TIME LEFT
023772  004737  020104                      JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
023776  010001                              MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
024000                                      ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
024000  104455                                                                  TRAP    C$ERDF
024002  000145                                                                  .WORD   101
024004  003554                                                                  .WORD   SFIERR
024006  011666                                                                  .WORD   SFIMSG
024010                                  20$:
                                        ;
024010  012704  026360                      MOV     #T29PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
024014  004737  010332                      JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
024020  103407                              BCS     25$                     ;BR, IF COMMAND ISSUED OK
024022  004737  020104                      JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
024026  010001                              MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
024030                                      ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
024030  104456                                                                  TRAP    C$ERHRD
024032  000146                                                                  .WORD   102
024034  004760                                                                  .WORD   WRTMSG
024036  011666                                                                  .WORD   SFIMSG
024040                                  25$:    CKLOOP                          ;LOOP IF SELECTED
024040  104406                                                                  TRAP    C$CLP1
024042  016501  000000                      MOV     TSSR(R5),R1             ;GET THE TSSR
024046  010102                              MOV     R1,R2                   ;SET UP EXPECTED
024050  042702  000100                      BIC     #OFL,R2                 ;OFF LINE SHOULD NOT BE SET
024054  020102                              CMP     R1,R2                   ;THEY SHOULD BE EQUAL
024056  001406                              BEQ     26$                     ;BR, IF OFL IS NOT SET
```

```
4044 024060                              ERRDF   ERRNO,T290FL,EXPREC     ;DRIVE IS OFF LINE
     024060  104455                                                              TRAP    C$ERDF
     024062  000147                                                              .WORD   103
     024064  026552                                                              .WORD   T290FL
     024066  016350                                                              .WORD   EXPREC
4045 024070  004737  020156          26$:    JSR     PC,CKDROP               ;TRY AND DROP DRIVE
4046 024074  004737  010434                  JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
4047 024100  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR
4048 024104  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED TSSR
4049 024110  103407                          BCS     30$                     ;BR, IF NO PROBLEM
4050 024112  010004                          MOV     R0,R4                   ;PACKET ADDRESS SET UP
4051 024114  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4055 024120                              ERRHRD  ERRNO,T29RWN,PKTSSR     ;REWIND NOT ACCEPTED
     024120  104456                                                              TRAP    C$ERHRD
     024122  000150                                                              .WORD   104
     024124  030270                                                              .WORD   T29RWN
     024126  011700                                                              .WORD   PKTSSR
4056 024130                  30$:    CKLOOP                                  ;LOOP IF SELECTED
     024130  104406                                                              TRAP    C$CLP1
4057 024132  013701  026406                  MOV     T29BFR+6,R1             ;PICK UP XSTO
4058 024136  010102                          MOV     R1,R2                   ;SET UP EXPECTED
4059 024140  052702  000002                  BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
4060 024144  020102                          CMP     R1,R2                   ;DOES EXP = REC'D
4061 024146  001406                          BEQ     40$                     ;BR, IF EQUAL (OK)
4062 024150  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4066 024154                              ERRHRD  ERRNO,T29BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     024154  104456                                                              TRAP    C$ERHRD
     024156  000151                                                              .WORD   105
     024160  027761                                                              .WORD   T29BOT
     024162  016350                                                              .WORD   EXPREC
4067 024164                  40$:    CKLOOP                                  ;LOOP IF SELECTED
     024164  104406                                                              TRAP    C$CLP1
4068 024166  013737  003076  026512          MOV     FREE,T29RB              ;ADDRESS OF READ BUFFER
4069 024174  012737  141011  026510          MOV     #141011,T29PK3          ;WRITE TAPE MARK RETRY,CVC=1,ACK COMMAND
4070 024202  012704  026510                  MOV     #T29PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
4071 024206  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
4072 024212  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
4073 024216  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
4074 024222  012702  100206                  MOV     #SSR!SC!BIT1!BIT2,R2    ;SET UP EXPECTED
4075 024226  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
4076 024230  001406                          BEQ     75$                     ;BR, IF OK
4077 024232  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4081 024236                              ERRHRD  ERRNO,T29WDE,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
     024236  104456                                                              TRAP    C$ERHRD
     024240  000152                                                              .WORD   106
     024242  027632                                                              .WORD   T29WDE
     024244  011700                                                              .WORD   PKTSSR
4082 024246                  75$:    CKLOOP                                  ;LOOP IF SELECTED
     024246  104406                                                              TRAP    C$CLP1
4083 024250  013701  026406                  MOV     T29BFR+6,R1             ;GET XSTO STATUS WORD
4084 024254  010102                          MOV     R1,R2                   ;SET UP EXPECTED
4085 024256  052702  002000                  BIS     #BIT10,R2               ;SET THE NEF BIT
4086 024262  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
4087 024264  001406                          BEQ     170$                    ;BR, IF EQUAL (GOOD)
4088 024266  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4092 024272                              ERRHRD  ERRNO,T29NEF,EXPREC     ;NEF SHOULD BE SET
     024272  104456                                                              TRAP    C$ERHRD
```

```
        024274  000153                                                  .WORD   107
        024276  026700                                                  .WORD   T29NEF
        024300  016350                                                  .WORD   EXPREC
  4093  024302                  170$:
  4094  024302  005103                   COM     R3          ;RESET THE SWITCH
  4095  024304  001273                   BNE     26$         ;BR, IF FIRST TIME THROUGH HERE
  4096  024306                           ENDSUB
        024306                                                  L10037:
        024306  104403                                                 TRAP    C$ESUB
```

```
                                ;*
                                ;
                                ;TEST 1, SUBTEST 2
                                ;
                                ;VERIFIES THAT A WRITE TAPE MARK RETRY COMMAND ISSUED WHILE THE TAPE
                                ;IS POSITIONED BEFORE THE FIRST RECORD ON TAPE (BUT NOT AT BOT) RESULTS IN
                                ;TAPE STATUS ALERT TERMINATION, WITH THE REVERSE INTO BOT (RIB) STATUS BIT
                                ;SET.
                                ;
                                ;
                                ;-
024310                                  BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
024310                                                                                  T1.2:
024310  104402                                                                          TRAP    C$BSUB
024312  004737  032140                  JSR     PC,T29REST              ;SET COMMAND PACKET
024316  004737  032232                  JSR     PC,T29RT2               ;SET UP OTHER COMMAND PACKET
024322  004737  032274                  JSR     PC,T29RT3               ;SET UP OTHER COMMAND PACKET
024326  004737  016650                  JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
024332  103407                          BCS     20$                     ;BR IF INIT WAS OK
024334  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
024340  010001                          MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
024342                                  ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
024342  104455                                                                          TRAP    C$ERDF
024344  000154                                                                          .WORD   108
024346  003554                                                                          .WORD   SFIERR
024350  011666                                                                          .WORD   SFIMSG
024352                          20$:
                                ;
024352  012704  026360                  MOV     #T29PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
024356  004737  010332                  JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
024362  103407                          BCS     25$                     ;BR, IF COMMAND ISSUED OK
024364  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
024370  010001                          MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
024372                                  ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
024372  104456                                                                          TRAP    C$ERHRD
024374  000155                                                                          .WORD   109
024376  004760                                                                          .WORD   WRTMSG
024400  011666                                                                          .WORD   SFIMSG
024402                          25$:    CKLOOP                          ;LOOP IF SELECTED
024402  104406                                                                          TRAP    C$CLP1
024404  004737  010434          26$:    JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
024410  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR
024414  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED TSSR
024420  103407                          BCS     30$                     ;BR, IF NO PROBLEM
024422  010004                          MOV     R0,R4                   ;PACKET ADDRESS SET UP
024424  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
024430                                  ERRHRD  ERRNO,T29RWN,PKTSSR     ;REWIND NOT ACCEPTED
024430  104456                                                                          TRAP    C$ERHRD
024432  000156                                                                          .WORD   110
024434  030270                                                                          .WORD   T29RWN
024436  011700                                                                          .WORD   PKTSSR
024440                          30$:    CKLOOP                          ;LOOP IF SELECTED
024440  104406                                                                          TRAP    C$CLP1
024442  013701  026406                  MOV     T29BFR+6,R1             ;PICK UP XSTO
024446  010102                          MOV     R1,R2                   ;SET UP EXPECTED
024450  052702  000002                  BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
```

```
4148 024454  020102                          CMP     R1,R2                   ;DOES EXP = REC'D
4149 024456  001406                          BEQ     40$                     ;BR, IF EQUAL (OK)
4150 024460  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4154 024464                                  ERRHRD  ERRNO,T29BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     024464  104456                                                          TRAP    C$ERHRD
     024466  000157                                                          .WORD   111
     024470  027761                                                          .WORD   T29BOT
     024472  016350                                                          .WORD   EXPREC
4155 024474  012737  000001  026512  40$:    MOV     #1,T29RB                ;NUMBER OF RECORDS TO SPACE OVER
4156 024502  012737  000400  026516          MOV     #256.,T29SZ             ;SET UP RECORD SIZE
4157 024510  012737  140005  026510          MOV     #140005,T29PK3          ;WRITE FORWARD,CVC=1,ACK COMMAND
4158 024516  012704  026510                  MOV     #T29PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
4159 024522  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
4160 024526  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
4161 024532  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
4162 024536  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
4163 024542  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
4164 024544  001406                          BEQ     75$                     ;BR, IF OK
4165 024546  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4169                                                                         ;SOFT ERROR, DON'T CARE ABOUT WRITE
4170                                                                         ;COMMAND'S RESULTS - CHECKING WRITE
4171                                                                         ;TAPE MARK COMMAND
4172 024552                                  ERRSOFT ERRNO,T29WRT,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     024552  104457                                                          TRAP    C$ERSOFT
     024554  000160                                                          .WORD   112
     024556  027714                                                          .WORD   T29WRT
     024560  011700                                                          .WORD   PKTSSR
4173 024562                          75$:    CKLOOP                          ;LOOP IF SELECTED
     024562  104406                                                          TRAP    C$CLP1
4174 024564  012737  000001  026512          MOV     #1,T29RB                ;NUMBER OF RECORDS TO SPACE OVER
4175 024572  012737  140410  026510          MOV     #140410,T29PK3          ;SET UP COMMAND IN APCKET                ;SET
4176 024600  012704  026510                  MOV     #T29PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
4177 024604  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
4178 024610  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
4179 024614  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
4180 024620  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
4181 024624  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
4182 024626  001406                          BEQ     175$                    ;BR, IF OK
4183 024630  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4187 024634                                  ERRHRD  ERRNO,T29WDE,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
     024634  104456                                                          TRAP    C$ERHRD
     024636  000161                                                          .WORD   113
     024640  027632                                                          .WORD   T29WDE
     024642  011700                                                          .WORD   PKTSSR
4188 024644                          175$:   CKLOOP                          ;LOOP IF SELECTED
     024644  104406                                                          TRAP    C$CLP1
4189 024646  013737  003076  026512          MOV     FREE,T29RB              ;ADDRESS OF BUFFER
4190 024654  012737  141011  026510          MOV     #141011,T29PK3          ;WRITE TAPE MARK RETRY,ACK,CVC=1 COMD.
4191 024662  012704  026510                  MOV     #T29PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
4192 024666  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
4193 024672  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
4194 024676  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
4195 024702  012702  100204                  MOV     #SSR!SC!BIT2,R2         ;SET UP EXPECTED
4196 024706  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
4197 024710  001406                          BEQ     180$                    ;BR, IF OK
4198 024712  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4202 024716                                  ERRHRD  ERRNO,T29WDE,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
```

```
                024716  104456                                                          TRAP    C$ERHRD
                024720  000162                                                          .WORD   114
                024722  027632                                                          .WORD   T29WDE
                024724  011700                                                          .WORD   PKTSSR
          4203  024726                          180$:   CKLOOP                          ;LOOP IF SELECTED
                024726  104406                                                          TRAP    C$CLP1
          4204  024730  013701  026414                  MOV     T29BFR+14,R1            ;GET XST3 STATUS WORD
          4205  024734  010102                          MOV     R1,R2                   ;SET UP EXPECTED
          4206  024736  052702  000001                  BIS     #BIT0,R2                ;SET THE RIB BIT
          4207  024742  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
          4208  024744  001406                          BEQ     190$                    ;BR, IF EQUAL (GOOD)
          4209  024746  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
          4213  024752                                  ERRHRD  ERRNO,T29RIB,EXPREC     ;NEF SHOULD BE SET
                024752  104456                                                          TRAP    C$ERHRD
                024754  000163                                                          .WORD   115
                024756  031716                                                          .WORD   T29RIB
                024760  016350                                                          .WORD   EXPREC
          4214  024762                          190$:
          4215  024762                          ENDSUB                  ;>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>>
                024762                                                          L10040:
                024762  104403                                                          TRAP    C$ESUB
```

```
                                        ;*
                                        ;
                                        ;TEST 1, SUBTEST 3
                                        ;
                                        ;VERIFIES THAT A WRITE TAPE MARK RETRY COMMAND TERMINATES
                                        ;PROPERLY AND WRITES THE TAPE MARK ONTO TAPE (BY ISSUING A READ REVERSE
                                        ;COMMAND AND CHECKING FOR TAPE STATUS ALERT TERMINATION AND TMK=1).
                                        ;
                                        ;-
4226 024764                             BGNSUB                  ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
     024764                                                                       T1.3:
     024764  104402                                                                 TRAP    C$BSUB
4227 024766  004737  032140             JSR     PC,T29REST      ;SET COMMAND PACKET
4228 024772  004737  032232             JSR     PC,T29RT2       ;SET UP OTHER COMMAND PACKET
4229 024776  004737  032274             JSR     PC,T29RT3       ;SET UP OTHER COMMAND PACKET
4230 025002  012737  023420  026550     MOV     #10000.,T29DLY  ;SET UP DELAY ROUTINE
4231 025010  004737  016650     10$:    JSR     PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
4232 025014  103426                     BCS     20$             ;BR IF INIT WAS OK
4233 025016                             DELAY   250             ;DELAY ABOUT .25 SECONDS
     025016  012727  000250                                                        MOV     #250,(PC)+
     025022  000000                                                                .WORD   0
     025024  013727  002116                                                        MOV     L$DLY,(PC)+
     025030  000000                                                                .WORD   0
     025032  005367  177772                                                        DEC     -6(PC)
     025036  001375                                                                BNE     .-4
     025040  005367  177756                                                        DEC     -22(PC)
     025044  001367                                                                BNE     .-20
4234 025046  005337  026550             DEC     T29DLY          ;BUMP DELAY ROUTINE DOWN
4235 025052  001356                     BNE     10$             ;BR, IF MORE DELAY TIME LEFT
4236 025054  004737  020104             JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
4240 025060  010001                     MOV     R0,R1           ;CONTENTS OF TSSR REGISTER
4241 025062                             ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
     025062  104455                                                                 TRAP    C$ERDF
     025064  000164                                                                .WORD   116
     025066  003554                                                                .WORD   SFIERR
     025070  011666                                                                .WORD   SFIMSG
4242 025072                     20$:
4243 025072  012704  026360             MOV     #T29PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
4244 025076  004737  010332             JSR     PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
4245 025102  103407                     BCS     23$             ;BR, IF COMMAND ISSUED OK
4246 025104  004737  020104             JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
4250 025110  010001                     MOV     R0,R1           ;SAVE CONTENTS OF TSSR
4251 025112                             ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     025112  104456                                                                 TRAP    C$ERHRD
     025114  000165                                                                .WORD   117
     025116  004760                                                                .WORD   WRTMSG
     025120  011666                                                                .WORD   SFIMSG
4252 025122                     23$:    CKLOOP                  ;LOOP IF SELECTED
     025122  104406                                                                 TRAP    C$CLP1
4253 025124  004737  010434             JSR     PC,REWIND       ;CALL TAPE REWIND COMMAND
4254 025130  103411                     BCS     30$             ;BR, IF NO PROBLEM
4255 025132  016501  000000             MOV     TSSR(R5),R1     ;GET TSSR
4256 025136  010004                     MOV     R0,R4           ;SAVE PACKET ADDRESS
4257 025140  004737  020104             JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
4261 025144                             ERRHRD  ERRNO,T29RWN,PKTSSR     ;REWIND NOT ACCEPTED
     025144  104456                                                                 TRAP    C$ERHRD
     025146  000166                                                                .WORD   118
```

```
                025150  030270                                                          .WORD   T29RWN
                025152  011700                                                          .WORD   PKTSSR
4262            025154                          30$:    CKLOOP                  ;LOOP IF SELECTED
                025154  104406                                                          TRAP    C$CLP1
4263            025156  013701  026406                  MOV     T29BFR+6,R1     ;PICK UP XSTO
4264            025162  010102                          MOV     R1,R2           ;SET UP EXPECTED
4265            025164  052702  000002                  BIS     #BIT1,R2        ;SET BOT BIT IN EXPECTED
4266            025170  020102                          CMP     R1,R2           ;DOES EXP = REC'D
4267            025172  001406                          BEQ     40$             ;BR, IF EQUAL (OK)
4268            025174  004737  020104                  JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
4272            025200                                  ERRHRD  ERRNO,T29BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
                025200  104456                                                          TRAP    C$ERHRD
                025202  000167                                                          .WORD   119
                025204  027761                                                          .WORD   T29BOT
                025206  016350                                                          .WORD   EXPREC
4273            025210                          40$:    CKLOOP                  ;LOOP IF SELECTED
                025210  104406                                                          TRAP    C$CLP1
4274            025212  012737  140011  026510          MOV     #140011,T29PK3  ;WRITE TAPE MARK,ACK,CVC=1 COMMAND
4275            025220  012704  026510                  MOV     #T29PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
4276            025224  010465  177776                  MOV     R4,TSDB(R5)     ;ISSUE COMMAND
4277            025230  004737  017124                  JSR     PC,WAITF        ;WAIT FOR SSR TO SET
4278            025234  016501  000000                  MOV     TSSR(R5),R1     ;GET TSSR CONTENTS
4279            025240  012702  000200                  MOV     #SSR,R2         ;SET UP EXPECTED
4280            025244  020102                          CMP     R1,R2           ;ARE THEY EQUAL
4281            025246  001406                          BEQ     70$             ;BR, IF OK
4282            025250  004737  020104                  JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
4286            025254                                  ERRHRD  ERRNO,T29WDC,PKTSSR     ;TSSR INCORRECT AFTER WRITE TAPE MARK
                025254  104456                                                          TRAP    C$ERHRD
                025256  000170                                                          .WORD   120
                025260  030607                                                          .WORD   T29WDC
                025262  011700                                                          .WORD   PKTSSR
4287            025264                          70$:    CKLOOP                  ;LOOP IF SELECTED
                025264  104406                                                          TRAP    C$CLP1
4288            025266  012703  000001          150$:   MOV     #1.,R3          ;NUMBER OF RECORDS TO WRITE TM
4289            025272  012737  141011  026510          MOV     #141011,T29PK3  ;WRITE TAPE MARK RETRY,ACK,CVC=1 COMMAND
4290            025300  012704  026510                  MOV     #T29PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
4291            025304  010465  177776          155$:   MOV     R4,TSDB(R5)     ;ISSUE COMMAND
4292            025310  004737  017124                  JSR     PC,WAITF        ;WAIT FOR SSR TO SET
4293            025314  016501  000000                  MOV     TSSR(R5),R1     ;PICK UP TSSR
4294            025320  012702  000200                  MOV     #SSR,R2         ;SET UP EXPECTED (SSR ONLY)
4295            025324  020102                          CMP     R1,R2           ;WAS STATUS GOOD
4296            025326  001406                          BEQ     165$            ;BR, IF TERMINATION WAS GOOD
4297            025330  004737  020104                  JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
4301            025334                                  ERRHRD  ERRNO,T29WDC,PKTSSR     ;TSSR NOT CORRECT AFTER WRT TAPE M.
                025334  104456                                                          TRAP    C$ERHRD
                025336  000171                                                          .WORD   121
                025340  030607                                                          .WORD   T29WDC
                025342  011700                                                          .WORD   PKTSSR
4302            025344                          165$:   CKLOOP                  ;LOOP IF SELECTED
                025344  104406                                                          TRAP    C$CLP1
4303            025346  012737  140401  026510          MOV     #140401,T29PK3  ;READ REVERSE,ACK, COMMAND
4304            025354  013737  003076  026512          MOV     FREE,T29RB      ;NUMBER OF RECORDS TO SPACE BACK
4305            025362  012704  026510                  MOV     #T29PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
4306            025366  010465  177776                  MOV     R4,TSDB(R5)     ;ISSUE COMMAND
4307            025372  004737  017124                  JSR     PC,WAITF        ;WAIT FOR SSR TO SET
4308            025376  016501  000000                  MOV     TSSR(R5),R1     ;GET TSSR CONTENTS
4309            025402  012702  100204                  MOV     #SSR!SC!BIT2,R2 ;SET UP EXPECTED
```

```
4310 025406  020102                    CMP     R1,R2                   ;ARE THEY EQUAL
4311 025410  001406                    BEQ     222$                    ;BR, IF OK
4312 025412  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4316 025416                            ERRHRD  ERRNO,T29RDG,PKTSSR     ;TSSR INCORRECT AFTER SPACE CMD.
     025416  104456                                                            TRAP    C$ERHRD
     025420  000172                                                            .WORD   122
     025422  031553                                                            .WORD   T29RDG
     025424  011700                                                            .WORD   PKTSSR
4317 025426                    222$:   CKLOOP                          ;LOOP IF SELECTED
     025426  104406                                                            TRAP    C$CLP1
4318 025430  013701  026406            MOV     T29BFR+6,R1             ;PICK UP XSTO
4319 025434  010102                    MOV     R1,R2                   ;SET UP EXPECTED
4320 025436  052702  100000            BIS     #BIT15,R2               ;TMK SHOULD BE SET
4321 025442  020102                    CMP     R1,R2                   ;IS TMK SET
4322 025444  001406                    BEQ     226$                    ;BR, IF TMK WAS SET (GOOD)
4323 025446  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4327 025452                            ERRHRD  ERRNO,T29RRN,EXPREC     ;TMK NOT SET AFTER READ REV
     025452  104456                                                            TRAP    C$ERHRD
     025454  000173                                                            .WORD   123
     025456  032016                                                            .WORD   T29RRN
     025460  016350                                                            .WORD   EXPREC
4328 025462                    226$:   CKLOOP                          ;LOOP IF SELECTED
     025462  104406                                                            TRAP    C$CLP1
4329 025464                            ENDSUB                          ;<<<<<<<<<<< END SUBTEST >>>>>>>>>>>
     025464                                                         L10041:
     025464  104403                                                            TRAP    C$ESUB
```

```
                                        ;*
                                        ;
                                        ;TEST 1, SUBTEST 4
                                        ;
                                        ;VERIFIES THAT THE SPACE-REVERSE PORTION OF THE WRITE TAPE MARK
                                        ;RETRY OPERATION IS PERFORMED BY REWINDING THE TAPE, ISSUING SEVERAL
                                        ;WRITE TAPE MARK RETRY COMMANDS IN SUCCESSION, THEN ISSUING TWO SPACE
                                        ;RECORDS REVERSE COMMANDS IN SUCCESSION.  THE SECOND SPACE RECORDS REVERSE
                                        ;COMMAND SHOULD TERMINATE WITH REVERSE INTO BOT (RIB) STATUS SET.
                                        ;
                                        ;-
4342 025466                                     BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
     025466                                                                                     T1.4:
     025466  104402                                                                             TRAP    C$BSUB
4343 025470  004737  032140                     JSR     PC,T29REST              ;SET COMMAND PACKET
4344 025474  004737  032232                     JSR     PC,T29RT2               ;SET UP OTHER COMMAND PACKET
4345 025500  004737  032274                     JSR     PC,T29RT3               ;SET UP OTHER COMMAND PACKET
4346 025504  012737  023420  026550             MOV     #10000.,T29DLY          ;SET UP DELAY ROUTINE
4347 025512  004737  016650           10$:      JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
4348 025516  103426                            BCS     20$                     ;BR IF INIT WAS OK
4349 025520                                     DELAY   250                     ;DELAY ABOUT .25 SECONDS
     025520  012727  000250                                                       MOV     #250,(PC)+
     025524  000000                                                             .WORD   0
     025526  013727  002116                                                       MOV     L$DLY,(PC)+
     025532  000000                                                             .WORD   0
     025534  005367  177772                                                     DEC     -6(PC)
     025540  001375                                                             BNE     .-4
     025542  005367  177756                                                     DEC     -22(PC)
     025546  001367                                                             BNE     .-20
4350 025550  005337  026550                     DEC     T29DLY                  ;BUMP DELAY ROUTINE DOWN
4351 025554  001356                            BNE     10$                     ;BR, IF MORE DELAY TIME LEFT
4352 025556  004737  020104                     JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4356 025562  010001                            MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
4357 025564                                     ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
     025564  104455                                                             TRAP    C$ERDF
     025566  000174                                                             .WORD   124
     025570  003554                                                             .WORD   SFIERR
     025572  011666                                                             .WORD   SFIMSG
4358 025574                             20$:
4359 025574  012704  026360                     MOV     #T29PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
4360 025600  004737  010332                     JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
4361 025604  103407                            BCS     23$                     ;BR, IF COMMAND ISSUED OK
4362 025606  004737  020104                     JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4366 025612  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
4367 025614                                     ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     025614  104456                                                             TRAP    C$ERHRD
     025616  000175                                                             .WORD   125
     025620  004760                                                             .WORD   WRTMSG
     025622  011666                                                             .WORD   SFIMSG
4368 025624                             23$:    CKLOOP                          ;LOOP IF SELECTED
     025624  104406                                                             TRAP    C$CLP1
4369 025626  004737  010434                     JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
4370 025632  103411                            BCS     30$                     ;BR, IF NO PROBLEM
4371 025634  016501  000000                     MOV     TSSR(R5),R1             ;GET TSSR
4372 025640  010004                            MOV     R0,R4                   ;SAVE PACKET ADDRESS
4373 025642  004737  020104                     JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4377 025646                                     ERRHRD  ERRNO,T29RWN,PKTSSR     ;REWIND NOT ACCEPTED
```

```
        025646  104456                                                          TRAP    C$ERHRD
        025650  000176                                                          .WORD   126
        025652  030270                                                          .WORD   T29RWN
        025654  011700                                                          .WORD   PKTSSR
4378    025656                          30$:    CKLOOP                          ;LOOP IF SELECTED
        025656  104406                                                          TRAP    C$CLP1
4379    025660  013701  026406                  MOV     T29BFR+6,R1             ;PICK UP XSTO
4380    025664  010102                          MOV     R1,R2                   ;SET UP EXPECTED
4381    025666  052702  000002                  BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
4382    025672  020102                          CMP     R1,R2                   ;DOES EXP = REC'D
4383    025674  001406                          BEQ     40$                     ;BR, IF EQUAL (OK)
4384    025676  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4388    025702                                  ERRHRD  ERRNO,T29BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
        025702  104456                                                          TRAP    C$ERHRD
        025704  000177                                                          .WORD   127
        025706  027761                                                          .WORD   T29BOT
        025710  016350                                                          .WORD   EXPREC
4389    025712                          40$:    CKLOOP                          ;LOOP IF SELECTED
        025712  104406                                                          TRAP    C$CLP1
4390    025714  012737  140011  026510          MOV     #140011,T29PK3          ;WRITE TAPE MARK,ACK,CVC=1 COMMAND
4391    025722  012704  026510                  MOV     #T29PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
4392    025726  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
4393    025732  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
4394    025736  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
4395    025742  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
4396    025746  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
4397    025750  001406                          BEQ     70$                     ;BR, IF OK
4398    025752  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4402    025756                                  ERRHRD  ERRNO,T29WDC,PKTSSR     ;TSSR INCORRECT AFTER WRITE TAPE MARK
        025756  104456                                                          TRAP    C$ERHRD
        025760  000200                                                          .WORD   128
        025762  030607                                                          .WORD   T29WDC
        025764  011700                                                          .WORD   PKTSSR
4403    025766                          70$:    CKLOOP                          ;LOOP IF SELECTED
        025766  104406                                                          TRAP    C$CLP1
4404    025770  012703  000012          150$:   MOV     #10.,R3                 ;NUMBER OF RECORDS TO WRITE TM
4405    025774  012737  000001  026512          MOV     #1,T29RB                ;SET UP PACKET
4406    026002  012737  141011  026510          MOV     #141011,T29PK3          ;WRITE TAPE MARK RETRY,ACK,CVC=1 COMMAND
4407    026010  012704  026510                  MOV     #T29PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
4408    026014  010465  177776          155$:   MOV     R4,TSDB(R5)             ;ISSUE COMMAND
4409    026020  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
4410    026024  016501  000000                  MOV     TSSR(R5),R1             ;PICK UP TSSR
4411    026030  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED (SSR ONLY)
4412    026034  020102                          CMP     R1,R2                   ;WAS STATUS GOOD
4413    026036  001406                          BEQ     165$                    ;BR, IF TERMINATION WAS GOOD
4414    026040  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4418    026044                                  ERRHRD  ERRNO,T29WDC,PKTSSR     ;TSSR NOT CORRECT AFTER WRT TAPE M.
        026044  104456                                                          TRAP    C$ERHRD
        026046  000201                                                          .WORD   129
        026050  030607                                                          .WORD   T29WDC
        026052  011700                                                          .WORD   PKTSSR
4419    026054                          165$:   CKLOOP                          ;LOOP IF SELECTED
        026054  104406                                                          TRAP    C$CLP1
4420    026056  005303                          DEC     R3                      ;BUMP COUNTER DOWN
4421    026060  001355                          BNE     155$                    ;BR, IF LESS THAN 10 TAPE MARKS
4422    026062  012737  140410  026510          MOV     #140410,T29PK3          ;SPACE REVERSE,ACK,CVC=1, COMMAND
4423    026070  012737  000001  026512          MOV     #1,T29RB                ;NUMBER OF RECORDS TO SPACE BACK
```

```
4424 026076  012704  026510                MOV     #T29PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
4425 026102  010465  177776                MOV     R4,TSDB(R5)         ;ISSUE COMMAND
4426 026106  004737  017124                JSR     PC,WAITF            ;WAIT FOR SSR TO SET
4427 026112  016501  000000                MOV     TSSR(R5),R1         ;GET TSSR CONTENTS
4428 026116  012702  100204                MOV     #SSR!SC!BIT2,R2     ;SET UP EXPECTED
4429 026122  020102                        CMP     R1,R2               ;ARE THEY EQUAL
4430 026124  001406                        BEQ     222$                ;BR, IF OK
4431 026126  004737  020104                JSR     PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
4435 026132                                ERRHRD  ERRNO,T29WDE,PKTSSR ;TSSR INCORRECT AFTER SPACE CMD.
     026132  104456                                                        TRAP    C$ERHRD
     026134  000202                                                        .WORD   130
     026136  027632                                                        .WORD   T29WDE
     026140  011700                                                        .WORD   PKTSSR
4436 026142                        222$:   CKLOOP                         ;LOOP IF SELECTED
     026142  104406                                                        TRAP    C$CLP1
4437 026144  012737  100410  026510        MOV     #100410,T29PK3      ;SPACE REVERSE,ACK, COMMAND
4438 026152  012737  000005  026512        MOV     #5,T29RB            ;NUMBER OF RECORDS TO SPACE BACK
4439 026160  012704  026510                MOV     #T29PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
4440 026164  010465  177776                MOV     R4,TSDB(R5)         ;ISSUE COMMAND
4441 026170  012737  000310  026550        MOV     #200.,T29DLY        ;NEED DELAY
4442 026176  004737  017124        230$:   JSR     PC,WAITF            ;WAIT FOR SSR TO SET
4443 026202  016501  000000                MOV     TSSR(R5),R1         ;GET TSSR CONTENTS
4444 026206  012702  100204                MOV     #SSR!SC!BIT2,R2     ;SET UP EXPECTED
4445 026212  020102                        CMP     R1,R2               ;ARE THEY EQUAL
4446 026214  001425                        BEQ     260$                ;BR, IF OK
4447 026216                                DELAY   250                 ;DELAY ABOUT .25 SECONDS
     026216  012727  000250                                                MOV     #250,(PC)+
     026222  000000                                                        .WORD   0
     026224  013727  002116                                                MOV     L$DLY,(PC)+
     026230  000000                                                        .WORD   0
     026232  005367  177772                                                DEC     -6(PC)
     026236  001375                                                        BNE     .-4
     026240  005367  177756                                                DEC     -22(PC)
     026244  001367                                                        BNE     .-20
4448 026246  005337  026550                DEC     T29DLY              ;LOOP ROUTINE
4449 026252  001351                        BNE     230$                ;LOOP BACK IF NOT ENOUGH DELAY
4450 026254  004737  020104                JSR     PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
4454 026260                                ERRHRD  ERRNO,T29SDG,PKTSSR ;TSSR INCORRECT AFTER SPACE REV CMD.
     026260  104456                                                        TRAP    C$ERHRD
     026262  000203                                                        .WORD   131
     026264  031634                                                        .WORD   T29SDG
     026266  011700                                                        .WORD   PKTSSR
4455 026270                        260$:   CKLOOP                         ;LOOP IF SELECTED
     026270  104406                                                        TRAP    C$CLP1
4456 026272  013701  026414                MOV     T29BFR+14,R1        ;PICK UP XST3
4457 026276  010102                        MOV     R1,R2               ;SET UP EXPECTED
4458 026300  052702  000001                BIS     #BIT0,R2            ;RIB SHOULD BE SET
4459 026304  020102                        CMP     R1,R2               ;IS RIB SET
4460 026306  001406                        BEQ     270$                ;BR, IF RIB WAS SET (GOOD)
4461 026310  004737  020104                JSR     PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
4465 026314                                ERRHRD  ERRNO,T29RIB,EXPREC ;TMK NOT SET AFTER READ REV
     026314  104456                                                        TRAP    C$ERHRD
     026316  000204                                                        .WORD   132
     026320  031716                                                        .WORD   T29RIB
     026322  016350                                                        .WORD   EXPREC
4466 026324                        270$:   CKLOOP                         ;LOOP IF SELECTED
     026324  104406                                                        TRAP    C$CLP1
```

```
4467 026326                          330$:  CKLOOP                          ;LOOP IF SELECTED
     026326  104406                                                                 TRAP    C$CLP1
4468 026330                                 ENDSUB                          ;<<<<<<<<<<< END SUBTEST >>>>>>>>>>>
     026330                                                                    L10042:
     026330  104403                                                                 TRAP    C$ESUB
4469
4470                                 ;
4471                                 ;
4472                                 ;
4473 026332  004737  017360                    JSR     PC,TSTLOOP              ;DO WE NEED TO ITERATE TEST
4474 026336  103002                            BCC     163$                    ;BR, IF NO LOOP REQUIRED
4475 026340  000137  023702                    JMP     T29LOOP                 ;EXECUTE AGAIN
4476 026344                          163$:  EXIT    TST                     ;ALL DONE THIS TEST
     026344  104432                                                                 TRAP    C$EXIT
     026346  003756                                                                 .WORD   L10036-.
```

```
 4478                                ;+
 4479                                ;LOCAL STORAGE FOR THIS TEST
 4480                                ;-
 4482 026350                                .BLKB   10-<.-TUV2A&7>
 4484 026360                        T29PACKET:                      ;COMMAND PACKET FOR TEST
 4485 026360  014004                        .WORD   14004           ;WRITE CHARACTERISTICS COMMAND, WITH CVC=1, ACK
 4486 026362  026370                        .WORD   T29DATA         ;ADDRESS OF CHARACTERISTICS BLOCK
 4487 026364  000000                        .WORD   0
 4488 026366  000012                        .WORD   10.             ;STARTING VALUE OF BLOCK SIZE
 4489 026370                        T29DATA:                        ;CHARACTERISTICS DATA BLOCK
 4490 026370  026400                        .WORD   T29BFR          ;ADDRESS OF MESSAGE BUFFER
 4491 026372  000000                        .WORD   0
 4492 026374  000024                        .WORD   20.             ;LENGTH OF MESSAGE BUFFER
 4493 026376  000000                        .WORD   0
 4494 026400                        T29BFR: .BLKW   25.             ;MESSAGE BUFFER
 4495                                ;
 4496                                ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
 4497                                ;
 4499 026462                                .BLKB   10-<.-TUV2A&7>
 4501 026470                        T29PK2:
 4502 026470  100006                        .WORD   100006          ;WRITE SUB SYS MEM COMMAND, AND ACK
 4503 026472  026520                        .WORD   T29BF2          ;ADDRESS OF SELECT BLOCK DATA
 4504 026474  000000                        .WORD   0
 4505 026476  000006                        .WORD   6.              ;SIZE OF DATA PACKET
 4507 026500                                .BLKB   10-<.-TUV2A&7>
 4509 026510                        T29PK3:
 4510 026510  140005                        .WORD   140005          ;WRITE TAPE MARK RETRY COMMAND, CVC=1 AND ACK
 4511 026512                        T29RB:
 4512 026512  003076                T29WB:  .WORD   FREE            ;ADDRESS OF WRITE BUFFER
 4513 026514  000000                        .WORD   0
 4514 026516  000000                T29SZ:  .WORD   0               ;SIZE OF BUFFER (EXTENT)
 4515                                       .EVEN
 4516                                ;
 4517 026520                        T29BF2:
 4518 026520     010                T29BS0: .BYTE   10              ;BSEL0 AREA
 4519 026521     200                T29BS1: .BYTE   200             ;BSEL1 AREA
 4520 026522  000000                T29S2:  .WORD   0               ;SEL 2 AREA
 4521 026524  000000                T29S3:  .WORD   0               ;DATA AREA
 4522                                       .EVEN
 4523                                ;TAPE MOTION PACKET COMMAND VALUES
 4524
 4525 026526  140001                T29RN:  .WORD   140001          ;READ DATA
 4526 026530  140401                T29WDR: .WORD   140401          ;READ DATA REVERSE
 4527 026532  141001                T29CON: .WORD   141001          ;READ PREVIOUS OPP=0
 4528 026534  161001                        .WORD   161001          ;READ PREVIOUS OPP=1
 4529 026536  141401                        .WORD   141401          ;WRITE TAPE MARK RETRY NEXT OPP=0
 4530 026540  161401                        .WORD   161401          ;WRITE TAPE MARK RETRY NEXT OPP=1
 4531 026542  177777                        .WORD   177777          ;END OF DATA
 4532                                ;
 4533 026544  000000                T29CNT: .WORD   0               ;TAPE RECORD COUNTER STORAGE AREA
 4534
 4535 026546  000000                T29RSZ: .WORD   0               ;RECORD STORAGE SIZE  AREA
 4536 026550  000000                T29DLY: .WORD                   ;DELAY COUNTER STORAGE AREA
```

```
4538
4539                                    ;+
4540                                    ;LOCAL TEXT MESSAGES FOR TEST
4541                                    ;-
4542
4543
4544 026552  104  162  151  T29OFL: .ASCIZ  'Drive is OFFLINE'
4545 026573  124  141  160  T29WNG: .ASCIZ  'Tape Position Incorrect After WRITE TAPE MARK RETRY Previous (OPP=1)'
4546 026700  127  122  111  T29NEF: .ASCIZ  'WRITE TAPE MARK RETRY, At BOT, Failed To Set NEF (XST0)'
4547 026770  124  123  123  T29RDF: .ASCIZ  'TSSR Incorrect After READ DATA Command'
4548 027037  127  122  111  T29RRF: .ASCIZ  'WRITE TAPE MARK RETRY Previous (Space Reverse, Read Forward) Command Failed
4549 027153  127  122  111  T29RRG: .ASCIZ  'WRITE TAPE MARK RETRY Previous (Read Forward, Space Reverse) Command Failed
4550 027267  120  117  123  T29SC:  .ASCIZ  'POSITION (Space Command) Failed, TSSR Not Correct'
4551 027351  122  111  102  T29LOR: .ASCIZ  'RIB NOT SET AFTER READ REVERSE INTO BOT'
4552 027421  124  123  123  T29WDF: .ASCIZ  'TSSR Not Correct After Illegal Mode Bits Set'
4553 027476  111  154  154  T29LOQ: .ASCIZ  'Illegal Mode Bits, Failed To Set ILC Bit In XST0'
4554 027557  127  122  111  T29SSR: .ASCIZ  'WRITE TAPE MARK RETRY COMMAND Not Accepted'
4555 027632  124  123  123  T29WDE: .ASCIZ  'TSSR Not Correct After SPACE REVERSE DATA Command'
4556
4557 027714  124  123  123  T29WRT: .ASCIZ  'TSSR Not Correct After WRITE Command'
4558 027761  124  141  160  T29BOT: .ASCIZ  'Tape Not At BOT After REWIND Command'
4559 030026  104  141  164  T29DTA: .ASCIZ  'Data Written To Tape Not Equal To Data Read From Tape'
4560 030114  127  122  111  T29EOT: .ASCIZ  'WRITE TAPE MARK RETRY DATA OVER EOT GAVE NO TAPE STATUS ALERT'
4561 030212  124  123  123  T29TM:  .ASCIZ  'TSSR Not Correct After SPACE REVERSE Into BOT'
4562 030270  122  145  167  T29RWN: .ASCIZ  'Rewind (POSITION) Command Not Accepted'
4563 030337  122  101  115  T29RNC: .ASCIZ  'RAM Error, Correct Data Pattern Not In Ram'
4564 030412  124  123  123  T29AM3: .ASCIZ  'TSSR Init. Failed After WRITE TAPE MARK RETRY COMMAND'
4565 030500  124  123  123  T29WDD: .ASCIZ  'TSSR Not Correct After WRITE TAPE MARK RETRY DATA Command, SWB Bit Set'
4566 030607  124  123  123  T29WDC: .ASCIZ  'TSSR Not Correct After WRITE TAPE MARK RETRY DATA Command'
4567 030701  103  126  103  T29VCK: .ASCIZ  'CVC Set, Didn''t Reset VCK In Message Buffer'
4568 030754  124  123  102  T29BA:  .ASCIZ  'TSBA Not Correct After WRITE TAPE MARK RETRY DATA Command'
4569 031046  127  122  111  T29WSS: .ASCIZ  'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
4570 031135  122  145  141  T29LON: .ASCIZ  'Reading Long Record Failed To Set RLL Bit In XST0'
4571 031217  122  145  141  T29LOP: .ASCIZ  'Reading Long Record Failed To Set RLS Bit In XST0'
4572 031301  122  145  163  T29PBP: .ASCIZ  'Residual Byte Count Incorrect After Short Record Read'
4573 031367  122  145  141  T29TRL: .ASCIZ  'Reading Long Record  Failed To Give Tape Status Alert'
4574 031455  104  141  164  T29NEQ: .ASCIZ  'Data WRITE TAPE MARK RETRY From Tape Not Correct, After SWB=1'
4575 031553  124  123  123  T29RDG: .ASCIZ  'TSSR Incorrect After READ REVERSE Into Tape Mark'
4576 031634  124  123  123  T29SDG: .ASCIZ  'TSSR Incorrect After SPACE REVERSE Into Tape Mark'
4577 031716  127  122  111  T29RIB: .ASCIZ  'WRITE TAPE MARK RETRY At First Record, Failed To Set RIB (XST3)'
4578 032016  124  115  113  T29RRN: .ASCIZ  'TMK (XST0) Failed To Set After READ REVERSE Into Tape Mark'
4579 032111  127  162  151  TST29ID:        .ASCIZ  'Write Tape Mark Retry'
4580                              .EVEN
4581                                    ;+
4582                                    ;
4583                                    ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
4584                                    ;WRITE SUBSYSTEM MEMORY COMMAND
4585                                    ;
4586                                    ;-
4587
4588 032140                         T29REST:
4589 032140                                 SAVREG                          ;SAVE THE REGISTERS
4590 032144  012701  026360                MOV     #T29PACKET,R1           ;START OF THE PACKET
4591 032150  012721  140004                MOV     #140004,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK, CVC=1
4592 032154  012721  026370                MOV     #T29DATA,(R1)+          ;ADDRESS OF CHARAISTICS DATA BLOCK
4593 032160  005021                        CLR     (R1)+                   ;EXTENDED ADDRESS
4594 032162  012721  000012                MOV     #10.,(R1)+              ;SIZE OF DATA BLOCK IN BYTES
```

```
4595 032166  012721  026400                    MOV     #T29BFR,(R1)+           ;ADDRESS OF MESSAGE BUFFER
4596 032172  005021                            CLR     (R1)+
4597 032174  012721  000024                    MOV     #20.,(R1)+              ;LENGTH OF MESSAGE BUFFER
4598 032200  005021                            CLR     (R1)+
4599 032202  012711  000000                    MOV     #0,(R1)                 ;SELECT DRIVE ZERO (0)
4600 032206  012702  000030                    MOV     #24.,R2                 ;NUMBER OF LOCATIONS TO BE CLEARED
4601 032212  012762  177777  026400  64$:      MOV     #177777,T29BFR(R2)      ;ALL ONES TO MESSAGE BUFFER
4602 032220  005742                            TST     -(R2)                   ;NEXT LOCATION
4603 032222  020227  000000                    CMP     R2,#0                   ;CHECK FOR END OF LOOP
4604 032226  001371                            BNE     64$                     ;KEEP GOING UNTIL DONE
4605 032230  000207                            RTS     PC                      ;RETURN
4606
4607
4608 032232                            T29RT2:
4609 032232                                    SAVREG                          ;SAVE THE REGISTERS
4610 032236  012701  026470                    MOV     #T29PK2,R1              ;START OF THE PACKET
4611 032242  012721  140006                    MOV     #140006,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK,CVC=1,
4612 032246  012721  026520                    MOV     #T29BF2,(R1)+           ;ADDRESS OF DATA BLOCK
4613 032252  005021                            CLR     (R1)+                   ;EXTENDED ADDRESS
4614 032254  012721  000006                    MOV     #6.,(R1)+               ;SIZE OF DATA BLOCK IN BYTES
4615 032260  005021                            CLR     (R1)+
4616 032262  012701  026520                    MOV     #T29BF2,R1              ;POINT TO DATA SEL AREA
4617 032266  005021                            CLR     (R1)+
4618 032270  005011                            CLR     (R1)
4619 032272  000207                            RTS     PC                      ;RETURN
4620 032274                            T29RT3:
4621 032274                                    SAVREG                          ;SAVE THE REGISTERS
4622 032300  012701  026510                    MOV     #T29PK3,R1              ;START OF THE PACKET
4623 032304  012721  000000                    MOV     #0,(R1)+                ;WRITE SUBSYSTEM MEM. WITH ACK,
4624 032310  012721  000000                    MOV     #0,(R1)+                ;ADDRESS OF DATA BLOCK
4625 032314  005021                            CLR     (R1)+                   ;EXTENDED ADDRESS
4626 032316  012711  000000                    MOV     #0,(R1)         ;SIZE OF DATA BLOCK IN BYTES
4627 032322  000207                            RTS     PC                      ;RETURN
4628 032324                                    ENDTST
     032324
     032324                                                    L10036:
     032324  104401                                                TRAP    C$ETST
```

```
                                .SBTTL  TEST  2: SKIP TAPE MARKS
                              ;*
                              ;
                              ;THIS TEST VERIFIES PROPER OPERATION OF THE SKIP TAPE MARKS
                              ;FORWARD AND SKIP TAPE MARKS REVERSE COMMANDS. PROPER OPERATION
                              ;UNDER CONTROL OF ALL COMBINATIONS OF THE ENABLE SKIP TAPE MARKS
                              ;STOP (ESS) AND ENABLE TAPE MARKS STOP OFF BOT (ENB) BITS SPECIFIED
                              ;BY THE WRITE CHARACTERISTICS COMMAND. THE TEST CONSISTS OF THE
                              ;FOLLOWING SUBTESTS (FOR EACH SUBTEST, THE TAPE IS FIRST WRITTEN
                              ;WITH AN APPROPRIATE SERIES OF DATA RECORDS AND/OR TAPE MARKS
                              ;AND/OR DOUBLE TAPE MARKS.
                              ;
                              ;
                              ;THE TEST CONSISTS OF THE FOLLOWING 11 SUBTESTS
                              ;
                              ;
                              ;
                              ;-
032326                                BGNTST
032326                                                                  T2::
032326  005037  002172                  CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
032332  005037  003104                  CLR     KTFLG           ;HOLD OFF KT11
032336  012737  005676  002150          MOV     @EPRT1,EPRTSW           ;PRIMARY ERROR MESSAGE
032344  012700  041121                  MOV     @TST30ID,R0             ;ASCII MESSAGE TO IDENTIFY TEST
032350  004737  017412                  JSR     PC,TSTSETUP             ;DO INITIAL TEST SETUP
032354  012737  000001  002166          MOV     #1,LOOPCNT              ;PERFORM 1 ITERATIONS
                              ;*
                              ;
                              ;TEST 2, SUBTEST 1
                              ;
                              ;
                              ;VERIFIES THAT A SKIP TAPE MARKS FORWARD COMMAND WITH
                              ;A TAPE MARK COUNT OF 1 OPERATES OPERATES PROPERLY. THE TAPE
                              ;IS FIRST REWOUND, THEN WRITTEN WITH SEVERAL "FILES";
                              ;EACH FILE CONSISTS OF A NUMBER OF DATA RECORDS
                              ;FOLLOWED BY A TAPE MARK. THE FINAL FILE IS
                              ;TERMINATED BY A DOUBLE TAPE MARK. EACH DATA RECORD
                              ;CONTAINS A FILE NUMBER AND THE RECORD NUMBER WITHIN
                              ;THE FILE SO THAT TAPE POSITION CAN BE SUBSEQUENTLY
                              ;VERIFIED BY READING THE DATA. THE TAPE IS AGAIN
                              ;REWOUND AND A SERIES OF SKIP TAPE MARKS FORWARD
                              ;COMMANDS ARE ISSUED AND THE RESULTS (TAPE STATUS ALERT
                              ;TERMINATION, TMK=1 STATUS, TAPE POSITION VIA READ
                              ;COMMAND) IS CHECKED. PRIOR TO ISSUANCE OF EACH SKIP
                              ;COMMAND, A WRITE CHARACTERISTICS COMMAND IS ISSUED TO
                              ;SET UP THE ESS AND ENB CONTROL BITS. ALL
                              ;COMBINATIONS OF ESS AND ENB ARE USED (00,01,10,
                              ;11); OPERATION SHOULD BE THE SAME IN EACH CASE FOR
                              ;THIS SUBTEST.
                              ;
                              ;
                              ;-

032362                        T30LOOP:
032362                                BGNSUB                    ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
032362                                                                  T2.1:
032362  104402                                                          TRAP    C$BSUB
```

```
 4689 032364  004737  041142                JSR     PC,T3OREST              ;SET COMMAND PACKET
 4690 032370  005037  036544                CLR     T3OFCN                  ;CLEAR FILE COUNTER
 4691 032374  004737  041234                JSR     PC,T3ORT2               ;SET UP OTHER COMMAND PACKET
 4692 032400  004737  041276                JSR     PC,T3ORT3               ;SET UP OTHER COMMAND PACKET
 4693 032404  012737  176750  036546        MOV     #65000.,T3ODLY          ;SET UP DELAY COUNTER
 4694 032412  004737  016650          10$:  JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
 4695 032416  103426                BCS     20$                     ;BR IF INIT WAS OK
 4696 032420                        DELAY   250                     ;DELAY ROUTINE CALL
      032420  012727  000250                                                MOV     #250,(PC)+
      032424  000000                                                        .WORD   0
      032426  013727  002116                                                MOV     L$DLY,(PC)+
      032432  000000                                                        .WORD   0
      032434  005367  177772                                                DEC     -6(PC)
      032440  001375                                                        BNE     .-4
      032442  005367  177756                                                DEC     -22(PC)
      032446  001367                                                        BNE     .-20
 4697 032450  005337  036546                DEC     T3ODLY                  ;BUMP COUNTER
 4698 032454  001356                BNE     10$                     ;BR, IF MORE COUNTING TO DO
 4699 032456  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4703 032462  010001                MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
 4704 032464                        ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      032464  104455                                                        TRAP    C$ERDF
      032466  000311                                                        .WORD   201
      032470  003554                                                        .WORD   SFIERR
      032472  011666                                                        .WORD   SFIMSG
 4705 032474                  20$:
 4706
 4707 032474  012704  036360                MOV     #T3OPACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
 4708
 4709                        ;*************************************************************
 4710                        ;
 4711                        ;ISSUE WRITE CHARACTERISTICS COMMAND
 4712                        ;
 4713                        ;*************************************************************
 4714
 4715 032500  004737  010332                JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
 4716 032504  103407                BCS     23$                     ;BR, IF COMMAND ISSUED OK
 4717 032506  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4721 032512  010001                MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
 4722 032514                        ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      032514  104456                                                        TRAP    C$ERHRD
      032516  000312                                                        .WORD   202
      032520  004760                                                        .WORD   WRTMSG
      032522  011666                                                        .WORD   SFIMSG
 4723 032524                  23$:  CKLOOP                          ;LOOP IF SELECTED
      032524  104406                                                        TRAP    C$CLP1
 4724
 4725                        ;*************************************************************
 4726                        ;
 4727                        ;ISSUE A REWIND TO TAPE DRIVE AND WAIT FOR SSR TO SET
 4728                        ;
 4729                        ;*************************************************************
 4730
 4731 032526  004737  010434                JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
 4732 032532  103411                BCS     30$                     ;BR, IF NO PROBLEM
 4733 032534  010004                MOV     R0,R4                   ;GET PACKET ADDRESS
 4734 032536  016501  000000                MOV     TSSR(R5),R1             ;GET STATUS REGISTER
```

```
4735 032542  004737  020104            JSR     PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
4739 032546                            ERRHRD  ERRNO,T30RWN,PKTSSR ;REWIND NOT ACCEPTED
     032546  104456                                                TRAP    C$ERHRD
     032550  000313                                                .WORD   203
     032552  040130                                                .WORD   T30RWN
     032554  011700                                                .WORD   PKTSSR
4740 032556                    30$:    CKLOOP                      ;LOOP IF SELECTED
     032556  104406                                                TRAP    C$CLP1
4741
4742                           ;*************************************************************
4743                           ;
4744                           ;GET EXTENDED STATUS REGISTER ZERO (XSTO) FROM MESSAGE BUFFER
4745                           ;
4746                           ;*************************************************************
4747
4748 032560  013701  036406            MOV     T30BFR+6,R1         ;PICK UP XSTO
4749 032564  010102                    MOV     R1,R2               ;SET UP EXPECTED
4750 032566  052702  000002            BIS     #BIT1,R2            ;SET BOT BIT IN EXPECTED
4751 032572  020102                    CMP     R1,R2               ;DOES EXP = REC'D
4752 032574  001406                    BEQ     40$                 ;BR, IF EQUAL (OK)
4753 032576  004737  020104            JSR     PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
4757 032602                            ERRHRD  ERRNO,T30BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
     032602  104456                                                TRAP    C$ERHRD
     032604  000314                                                .WORD   204
     032606  037731                                                .WORD   T30BOT
     032610  016350                                                .WORD   EXPREC
4758 032612                    40$:    CKLOOP                      ;LOOP IF SELECTED
     032612  104406                                                TRAP    C$CLP1
4759 032614  012737  000001  036544        MOV     #1.,T30FCN          ;SET "FILE" COUNTER AT 1 DECIMAL
4760 032622  012703  000001        64$:    MOV     #1,R3               ;ONE RECORD  PER "FILE"
4761 032626  013737  003076  036512  65$:  MOV     FREE,T30WB          ;SET UP PACKETS'S WRITE BUFFER
4762 032634  012737  003720  036516        MOV     #2000.,T30SZ        ;SET RECORD SIZE AT 2000 BYTES
4763
4764                           ;*************************************************************
4765                           ;
4766                           ;WRITE DATA,ACK,CVC=1 COMMAND
4767                           ;
4768                           ;*************************************************************
4769
4770 032642  012737  140005  036510        MOV     #140005,T30PK3      ;WRITE DATA,ACK,CVC=1 COMMAND
4771 032650  012704  036510                MOV     #T30PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
4772 032654  013702  036544                MOV     T30FCN,R2           ;GET FILE COUNTER
4773 032660  000302                        SWAB    R2                  ;MOVE TO UPPER BYTE
4774 032662  010301                        MOV     R3,R1               ;GET RECORD COUNTER
4775 032664  060201                        ADD     R2,R1               ;FILE COUNTER IN UPPER, RECORD # LOW
4776 032666  010177  150204                MOV     R1,@FREE            ;MOV TO OUT PUT BUFFER
4777 032672  010465  177776                MOV     R4,TSDB(R5)         ;ISSUE COMMAND
4778 032676  004737  017124                JSR     PC,WAITF            ;WAIT FOR SSR TO SET
4779 032702  016501  000000                MOV     TSSR(R5),R1         ;GET TSSR CONTENTS
4780 032706  012702  000200                MOV     #SSR,R2             ;SET UP EXPECTED
4781 032712  020102                        CMP     R1,R2               ;ARE THEY EQUAL
4782 032714  001406                        BEQ     70$                 ;BR, IF OK
4783 032716  004737  020104                JSR     PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
4787                                                                   ;SOFT ERROR, DON'T CARE ABOUT WRITE
4788                                                                   ;COMMAND'S RESULTS - CHECKING SKIP
4789                                                                   ;TAPE MARK COMMAND
4790 032722                            ERRSOFT ERRNO,T30WDD,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
```

```
                032722  104457                                                          TRAP    C$ERSOFT
                032724  000315                                                          .WORD   205
                032726  037060                                                          .WORD   T30WDD
                032730  011700                                                          .WORD   PKTSSR
 4791           032732                          70$:    CKLOOP                  ;LOOP IF SELECTED
                032732  104406                                                          TRAP    C$CLP1
 4792           032734  005203                  INC     R3                      ;COUNT THE RECORD COUNTER DOWN
 4793           032736  020327  000021          CMP     R3,#21                  ;AT 20 YET
 4794           032742  001331                  BNE     65$                     ;BR, IF NOT AT 20 RECORDS WRITTEN
 4795
 4796                                  ;*************************************************************
 4797                                  ;
 4798                                  ;WRITE TAPE MARK,ACK,CVC=1 COMMAND
 4799                                  ;
 4800                                  ;*************************************************************
 4801
 4802           032744  012737  141011  036510          MOV     #141011,T30PK3          ;WRITE TAPE MARK,ACK,CVC=1 COMMAND
 4803           032752  012704  036510                  MOV     #T30PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
 4804           032756  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
 4805           032762  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
 4806           032766  016501  000000                  MOV     TSSR(R5),R1             ;PICK UP TSSR
 4807           032772  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED (SSR ONLY)
 4808           032776  020102                          CMP     R1,R2                   ;WAS STATUS GOOD
 4809           033000  001406                          BEQ     160$                    ;BR, IF TERMINATION WAS GOOD
 4810           033002  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4814           033006                                  ERRHRD  ERRNO,T30WDC,PKTSSR     ;TSSR NOT CORRECT AFTER WRT TAPE M.
                033006  104456                                                          TRAP    C$ERHRD
                033010  000316                                                          .WORD   206
                033012  040252                                                          .WORD   T30WDC
                033014  011700                                                          .WORD   PKTSSR
 4815           033016                          160$:   CKLOOP                  ;LOOP IF SELECTED
                033016  104406                                                          TRAP    C$CLP1
 4816           033020  005237  036544                  INC     T30FCN                  ;COUNT THE "FILE" COUNTER DOWN
 4817           033024  023727  036544  000006          CMP     T30FCN,#6               ;WRITE 5 FILE TO TAPE
 4818           033032  001273                          BNE     64$                     ;BR, IF NOT AT 5 FILES WRITTEN
 4819
 4820                                  ;*************************************************************
 4821                                  ;
 4822                                  ;WRITE TAPE MARK,ACK,CVC=1 COMMAND
 4823                                  ;
 4824                                  ;*************************************************************
 4825
 4826           033034  012737  141011  036510          MOV     #141011,T30PK3          ;WRITE TAPE MARK,ACK,CVC=1 COMMAND
 4827           033042  012704  036510                  MOV     #T30PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
 4828           033046  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
 4829           033052  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
 4830           033056  016501  000000                  MOV     TSSR(R5),R1             ;PICK UP TSSR
 4831           033062  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED (SSR ONLY)
 4832           033066  020102                          CMP     R1,R2                   ;WAS STATUS GOOD
 4833           033070  001406                          BEQ     165$                    ;BR, IF TERMINATION WAS GOOD
 4834           033072  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4838           033076                                  ERRHRD  ERRNO,T30WDC,PKTSSR     ;TSSR NOT CORRECT AFTER WRT TAPE M.
                033076  104456                                                          TRAP    C$ERHRD
                033100  000317                                                          .WORD   207
                033102  040252                                                          .WORD   T30WDC
                033104  011700                                                          .WORD   PKTSSR
 4839           033106                          165$:   CKLOOP                  ;LOOP IF SELECTED
```

```
      033106  104406                                                          TRAP    C$CLP1
 4840
 4841                                ;*************************************************************
 4842                                ;
 4843                                ;ISSUE A REWIND TO TAPE DRIVE AND WAIT FOR SSR TO SET
 4844                                ;
 4845                                ;*************************************************************
 4846
 4847 033110  004737  010434              JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
 4848 033114  103411                      BCS     170$                    ;BR, IF NO PROBLEM
 4849 033116  010004                      MOV     R0,R4                   ;GET PACKET ADDRESS
 4850 033120  016501  000000              MOV     TSSR(R5),R1             ;GET STATUS REGISTER
 4851 033124  004737  020104              JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4855 033130                              ERRHRD  ERRNO,T30RWN,PKTSSR     ;REWIND NOT ACCEPTED
      033130  104456                                                          TRAP    C$ERHRD
      033132  000320                                                          .WORD   208
      033134  040130                                                          .WORD   T30RWN
      033136  011700                                                          .WORD   PKTSSR
 4856 033140                      170$:   CKLOOP                          ;LOOP IF SELECTED
      033140  104406                                                          TRAP    C$CLP1
 4857
 4858                                ;*************************************************************
 4859                                ;
 4860                                ;GET EXTENDED STATUS REGISTER ZERO (XST0) FROM MESSAGE BUFFER
 4861                                ;
 4862                                ;*************************************************************
 4863
 4864 033142  013701  036406              MOV     T30BFR+6,R1             ;PICK UP XST0
 4865 033146  010102                      MOV     R1,R2                   ;SET UP EXPECTED
 4866 033150  052702  000002              BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
 4867 033154  020102                      CMP     R1,R2                   ;DOES EXP = REC'D
 4868 033156  001406                      BEQ     180$                    ;BR, IF EQUAL (OK)
 4869 033160  004737  020104              JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4873 033164                              ERRHRD  ERRNO,T30BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
      033164  104456                                                          TRAP    C$ERHRD
      033166  000321                                                          .WORD   209
      033170  037731                                                          .WORD   T30BOT
      033172  016350                                                          .WORD   EXPREC
 4874 033174                      180$:   CKLOOP                          ;LOOP IF SELECTED
      033174  104406                                                          TRAP    C$CLP1
 4875 033176  012703  036526              MOV     #T30IMV,R3              ;SET UP POINTER TO COMMAND TABLE
 4876
 4877 033202  011337  036376      182$:   MOV     (R3),T30ETM             ;GET NEXT COMMAND
 4878 033206  012704  036360              MOV     #T30PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
 4879
 4880                                ;*************************************************************
 4881                                ;
 4882                                ;ISSUE WRITE CHARACTERISTICS COMMAND
 4883                                ;
 4884                                ;*************************************************************
 4885
 4886 033212  004737  010332              JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
 4887 033216  103407                      BCS     188$                    ;BR, IF COMMAND ISSUED OK
 4888 033220  004737  020104              JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4892 033224  010001                      MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
 4893 033226                              ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      033226  104456                                                          TRAP    C$ERHRD
```

```
      033230  000322                                                          .WORD   210
      033232  004760                                                          .WORD   WRTMSG
      033234  011666                                                          .WORD   SFIMSG
 4894 033236                  188$:   CKLOOP                  ;LOOP IF SELECTED
      033236  104406                                                          TRAP    C$CLP1
 4895
 4896                         ;*************************************************************
 4897                         ;
 4898                         ;SKIP TAPE MARK,ACK,CVC=1 COMMAND
 4899                         ;
 4900                         ;*************************************************************
 4901
 4902 033240  012737  141010  036510          MOV     #141010,T30PK3      ;SKIP TAPE MARK,ACK,CVC=1 COMMAND
 4903 033246  012737  000001  036512          MOV     #1,T30RB            ;SET UP NUMBER TO SKIP
 4904 033254  012704  036510                  MOV     #T30PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
 4905 033260  010465  177776          189$:   MOV     R4,TSDB(R5)         ;ISSUE COMMAND
 4906 033264  012737  176750  036546          MOV     #65000.,T30DLY      ;SET UP DELAY COUNTER
 4907 033272  004737  017124          190$:   JSR     PC,WAITF            ;WAIT FOR SSR TO SET
 4908 033276  016501  000000                  MOV     TSSR(R5),R1         ;PICK UP TSSR
 4909 033302  032701  000200                  BIT     #SSR,R1             ;IS SSR SET YET
 4910 033306  001017                          BNE     191$                ;BR, IF SSR IS SET
 4911 033310                                  DELAY   250                 ;CALL DELAY ROUTINE
      033310  012727  000250                                                  MOV     #250.,(PC)+
      033314  000000                                                          .WORD   0
      033316  013727  002116                                                  MOV     L$DLY,(PC)+
      033322  000000                                                          .WORD   0
      033324  005367  177772                                                  DEC     -6(PC)
      033330  001375                                                          BNE     .-4
      033332  005367  177756                                                  DEC     -22(PC)
      033336  001367                                                          BNE     .-20
 4912 033340  005337  036546                  DEC     T30DLY              ;BUMP DELAY ROUTINE
 4913 033344  001352                          BNE     190$                ;BR, IF MORE DELAY TO GO
 4914 033346  012702  000200          191$:   MOV     #SSR,R2             ;SET UP EXPECTED (SSR ONLY)
 4915 033352  020102                          CMP     R1,R2               ;WAS STATUS GOOD
 4916 033354  001406                          BEQ     192$                ;BR, IF TERMINATION WAS GOOD
 4917 033356  004737  020104                  JSR     PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4921 033362                                  ERRHRD  ERRNO,T30SKM,PKTSSR ;TSSR NOT CORRECT AFTER SKIP TAPE M.
      033362  104456                                                          TRAP    C$ERHRD
      033364  000323                                                          .WORD   211
      033366  037004                                                          .WORD   T30SKM
      033370  011700                                                          .WORD   PKTSSR
 4922 033372                  192$:   CKLOOP                  ;LOOP IF SELECTED
      033372  104406                                                          TRAP    C$CLP1
 4923
 4924                         ;*************************************************************
 4925                         ;
 4926                         ;GET EXTENDED STATUS REGISTER ZERO (XST0) FROM MESSAGE BUFFER
 4927                         ;
 4928                         ;*************************************************************
 4929
 4930 033374  013701  036406          MOV     T30BFR+6,R1         ;PICK UP XST0
 4931 033400  010102                  MOV     R1,R2               ;SET UP EXPECTED
 4932 033402  052702  100000          BIS     #BIT15,R2           ;SET TMK BIT IN EXPECTED
 4933 033406  020102                  CMP     R1,R2               ;DOES EXP = REC'D
 4934 033410  001406                  BEQ     195$                ;BR, IF EQUAL (OK)
 4935 033412  004737  020104          JSR     PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4939 033416                          ERRHRD  ERRNO,T30TMK,EXPREC ;TMK NOT SET AFTER WRT TAPE MARK
```

```
        033416  104456                                                          TRAP    C$ERHRD
        033420  000324                                                          .WORD   212
        033422  040404                                                          .WORD   T30TMK
        033424  016350                                                          .WORD   EXPREC
 4940   033426                          195$:   CKLOOP                          ;LOOP IF SELECTED
        033426  104406                                                          TRAP    C$CLP1
 4941   033430  012700  177777                  MOV     #177777,R0              ;VALUE TO WRITTEN TO MEMORY
 4942   033434  004737  020376                  JSR     PC,FILLMEM              ;FILL MEM WITH ALL ONES
 4943   033440  013737  003076  036512          MOV     FREE,T30RB              ;STARTING READ BUFFER ADDRESS
 4944
 4945                          ;*************************************************************
 4946                          ;
 4947                          ;READ FORWARD,ACK,CVC=1 COMMAND
 4948                          ;
 4949                          ;*************************************************************
 4950
 4951   033446  012737  140001  036510          MOV     #140001,T30PK3          ;READ FORWARD,ACK,CVC=1 COMMAND
 4952   033454  012704  036510                  MOV     #T30PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
 4953   033460  012737  003720  036516          MOV     #2000.,T30SZ            ;SET UP RECORD SIZE IN PACKET
 4954   033466  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
 4955   033472  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
 4956   033476  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
 4957   033502  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
 4958   033506  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
 4959   033510  001406                          BEQ     200$                    ;BR, IF OK
 4960   033512  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4964   033516                                  ERRHRD  ERRNO,T30RDF,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
        033516  104456                                                          TRAP    C$ERHRD
        033520  000325                                                          .WORD   213
        033522  037303                                                          .WORD   T30RDF
        033524  011700                                                          .WORD   PKTSSR
 4965   033526                          200$:   CKLOOP                          ;LOOP IF SELECTED
        033526  104406                                                          TRAP    C$CLP1
 4966   033530  017701  147342                  MOV     @FREE,R1                ;FIRST LOC IN READ BUFFER
 4967   033534  012702  177777                  MOV     #177777,R2              ;EXPECTED IF NO DATA TRANS.
 4968   033540  020102                          CMP     R1,R2                   ;DID ANY DATA GET TRANSFERRED
 4969   033542  001006                          BNE     220$                    ;BR, IF NO DATA TRANS (GOOD)
 4970   033544  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4974   033550                                  ERRHRD  ERRNO,T30DTR,EXPREC     ;DATA TRANSFERRED ON READ TAPE MARK
        033550  104456                                                          TRAP    C$ERHRD
        033552  000326                                                          .WORD   214
        033554  040760                                                          .WORD   T30DTR
        033556  016350                                                          .WORD   EXPREC
 4975   033560                          220$:   CKLOOP                          ;LOOP IF SELECTED
        033560  104406                                                          TRAP    C$CLP1
 4976   033562  012702  001001                  MOV     #1001,R2                ;SET UP RECORD NUMBER EXPECTED (FILE 2)
 4977   033566  017701  147304                  MOV     @FREE,R1                ;GET INFO FROM BUFFER
 4978   033572  020201                          CMP     R2,R1                   ;ARE THEY EQUAL
 4979   033574  001406                          BEQ     228$                    ;BR, IF EQUAL (OK)
 4980   033576  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4984   033602                                  ERRHRD  ERRNO,T30PTB,EXPREC     ;RECORD POSITION WAS NOT CORRECT
        033602  104456                                                          TRAP    C$ERHRD
        033604  000327                                                          .WORD   215
        033606  037132                                                          .WORD   T30PTB
        033610  016350                                                          .WORD   EXPREC
 4985   033612                          228$:   CKLOOP                          ;LOOP IF SELECTED
        033612  104406                                                          TRAP    C$CLP1
```

```
                        ;*************************************************************
                        ;
                        ;ISSUE A REWIND TO TAPE DRIVE AND WAIT FOR SSR TO SET
                        ;
                        ;*************************************************************

033614  004737  010434            JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
033620  103411                    BCS     230$                    ;BR, IF NO PROBLEM
033622  010004                    MOV     R0,R4                   ;SAVE PACKET ADDRESS
033624  016501  000000            MOV     TSSR(R5),R1             ;GET TSSR STATUS
033630  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
033634                            ERRHRD  ERRNO,T30RWN,PKTSSR     ;REWIND NOT ACCEPTED
033634  104456                                                            TRAP    C$ERHRD
033636  000330                                                            .WORD   216
033640  040130                                                            .WORD   T30RWN
033642  011700                                                            .WORD   PKTSSR
033644                  230$:   CKLOOP                          ;LOOP IF SELECTED
033644  104406                                                            TRAP    C$CLP1

                        ;*************************************************************
                        ;
                        ;GET EXTENDED STATUS REGISTER ZERO (XST0) FROM MESSAGE BUFFER
                        ;
                        ;*************************************************************

033646  013701  036406            MOV     T30BFR+6,R1             ;PICK UP XST0
033652  010102                    MOV     R1,R2                   ;SET UP EXPECTED
033654  052702  000002            BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
033660  020102                    CMP     R1,R2                   ;DOES EXP = REC'D
033662  001406                    BEQ     240$                    ;BR, IF EQUAL (OK)
033664  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
033670                            ERRHRD  ERRNO,T30BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
033670  104456                                                            TRAP    C$ERHRD
033672  000331                                                            .WORD   217
033674  037731                                                            .WORD   T30BOT
033676  016350                                                            .WORD   EXPREC
033700                  240$:   CKLOOP                          ;LOOP IF SELECTED
033700  104406                                                            TRAP    C$CLP1
033702  005723                    TST     (R3)+                   ;POINT TO NEXT POSITION
033704  011301                    MOV     (R3),R1                 ;GET NEXT COMMAND ETC.
033706  020127  177777            CMP     R1,#177777              ;END OF TABLE MARKER
033712  001402                    BEQ     330$                    ;BR, IF AT END OF TABLE
033714  000137  033202            JMP     182$                    ;JUMP TO MORE COMMANDS TO DO
033720                  330$:   CKLOOP                          ;LOOP IF SELECTED
033720  104406                                                            TRAP    C$CLP1
033722                          ENDSUB                          ;<<<<<<<<<<< END SUBTEST >>>>>>>>>>>
033722                                                                  L10044:
033722  104403                                                            TRAP    C$ESUB
```

```
5029                                    ;*
5030                                    ;
5031                                    ;TEST 2, SUBTEST 2
5032                                    ;
5033                                    ;VERIFIES THAT SKIP TAPE MARKS COMMANDS WITH A TAPE
5034                                    ;MARK COUNT GREATER THAN 1 OPERATE PROPERLY. COUNTS
5035                                    ;OF 2,3,8,64,256, AND 512 ARE TESTED. THE
5036                                    ;TESTING SEQUENCE IS SIMILAR TO THAT USED IN SUBTEST 1.
5037                                    ;
5038                                    ;
5039                                    ;
5040                                    ;-
5041
5042 033724                                     BGNSUB                  ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
      033724                                                                    T2.2:
      033724  104402                                                              TRAP    C$BSUB
5043 033726  004737  041142                     JSR     PC,T3OREST      ;SET COMMAND PACKET
5044 033732  005037  036544                     CLR     T3OFCN          ;CLEAR FILE COUNTER
5045 033736  004737  041234                     JSR     PC,T3ORT2       ;SET UP OTHER COMMAND PACKET
5046 033742  004737  041276                     JSR     PC,T3ORT3       ;SET UP OTHER COMMAND PACKET
5047 033746  012737  176750  036546             MOV     #65000.,T3ODLY  ;SET UP DELAY COUNTER
5048 033754  004737  016650           10$:      JSR     PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
5049 033760  103426                             BCS     20$             ;BR IF INIT WAS OK
5050 033762                                     DELAY   250             ;DELAY ROUTINE CALL
      033762  012727  000250                                                   MOV     #250,(PC)+
      033766  000000                                                           .WORD   0
      033770  013727  002116                                                   MOV     L$DLY,(PC)+
      033774  000000                                                           .WORD   0
      033776  005367  177772                                                   DEC     -6(PC)
      034002  001375                                                           BNE     .-4
      034004  005367  177756                                                   DEC     -22(PC)
      034010  001367                                                           BNE     .-20
5051 034012  005337  036546                     DEC     T3ODLY          ;BUMP COUNTER
5052 034016  001356                             BNE     10$             ;BR, IF MORE COUNTING TO DO
5053 034020  004737  020104                     JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
5057 034024  010001                             MOV     R0,R1           ;CONTENTS OF TSSR REGISTER
5058 034026                                     ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      034026  104455                                                           TRAP    C$ERDF
      034030  000332                                                           .WORD   218
      034032  003554                                                           .WORD   SFIERR
      034034  011666                                                           .WORD   SFIMSG
5059 034036                             20$:
5060
5061 034036  012704  036360                     MOV     #T3OPACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
5062
5063                                    ;*************************************************************
5064                                    ;
5065                                    ;ISSUE WRITE CHARACTERISTICS COMMAND
5066                                    ;
5067                                    ;*************************************************************
5068
5069 034042  004737  010332                     JSR     PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
5070 034046  103407                             BCS     23$             ;BR, IF COMMAND ISSUED OK
5071 034050  004737  020104                     JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
5075 034054  010001                             MOV     R0,R1           ;SAVE CONTENTS OF TSSR
5076 034056                                     ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      034056  104456                                                           TRAP    C$ERHRD
```

```
                034060  000333                                                  .WORD   219
                034062  004760                                                  .WORD   WRTMSG
                034064  011666                                                  .WORD   SFIMSG
 5077           034066                  23$:    CKLOOP                          ;LOOP IF SELECTED
                034066  104406                                                  TRAP    C$CLP1
 5078
 5079                                   ;*************************************************************
 5080                                   ;
 5081                                   ;ISSUE A REWIND TO TAPE DRIVE AND WAIT FOR SSR TO SET
 5082                                   ;
 5083                                   ;*************************************************************
 5084
 5085           034070  004737  010434          JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
 5086           034074  103411                  BCS     30$                     ;BR, IF NO PROBLEM
 5087           034076  010004                  MOV     R0,R4                   ;GET PACKET ADDRESS
 5088           034100  016501  000000          MOV     TSSR(R5),R1             ;GET STATUS REGISTER
 5089           034104  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 5093           034110                  ERRHRD  ERRNO,T30RWN,PKTSSR             ;REWIND NOT ACCEPTED
                034110  104456                                                  TRAP    C$ERHRD
                034112  000334                                                  .WORD   220
                034114  040130                                                  .WORD   T30RWN
                034116  011700                                                  .WORD   PKTSSR
 5094           034120                  30$:    CKLOOP                          ;LOOP IF SELECTED
                034120  104406                                                  TRAP    C$CLP1
 5095
 5096                                   ;*************************************************************
 5097                                   ;
 5098                                   ;GET EXTENDED STATUS REGISTER ZERO (XST0) FROM MESSAGE BUFFER
 5099                                   ;
 5100                                   ;*************************************************************
 5101
 5102           034122  013701  036406          MOV     T30BFR+6,R1             ;PICK UP XST0
 5103           034126  010102                  MOV     R1,R2                   ;SET UP EXPECTED
 5104           034130  052702  000002          BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
 5105           034134  020102                  CMP     R1,R2                   ;DOES EXP = REC'D
 5106           034136  001406                  BEQ     40$                     ;BR, IF EQUAL (OK)
 5107           034140  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 5111           034144                  ERRHRD  ERRNO,T30BOT,EXPREC             ;TAPE NOT AT BOT AFTER REWIND
                034144  104456                                                  TRAP    C$ERHRD
                034146  000335                                                  .WORD   221
                034150  037731                                                  .WORD   T30BOT
                034152  016350                                                  .WORD   EXPREC
 5112           034154                  40$:    CKLOOP                          ;LOOP IF SELECTED
                034154  104406                                                  TRAP    C$CLP1
 5113           034156  012737  000001  036544          MOV     #1.,T30FCN      ;SET "FILE" COUNTER AT 1 DECIMAL
 5114           034164  012703  000001          64$:    MOV     #1,R3           ;ONE RECORD  PER "FILE"
 5115           034170  013737  003076  036512  65$:    MOV     FREE,T30WB      ;SET UP PACKETS'S WRITE BUFFER
 5116           034176  012737  000024  036516          MOV     #20.,T30SZ      ;SET RECORD SIZE AT 2000 BYTES
 5117
 5118                                   ;*************************************************************
 5119                                   ;
 5120                                   ;WRITE DATA,ACK,CVC=1 COMMAND
 5121                                   ;
 5122                                   ;*************************************************************
 5123
 5124           034204  012737  140005  036510          MOV     #140005,T30PK3  ;WRITE DATA,ACK,CVC=1 COMMAND
 5125           034212  012704  036510                  MOV     #T30PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
```

```
5126 034216  013702  036544            MOV     T30FCN,R2               ;GET FILE COUNTER
5127 034222  000302                    SWAB    R2                      ;MOVE TO UPPER BYTE
5128 034224  010301                    MOV     R3,R1                   ;GET RECORD COUNTER
5129 034226  060201                    ADD     R2,R1                   ;FILE COUNTER IN UPPER, RECORD # LOW
5130 034230  010177  146642            MOV     R1,@FREE                ;MOV TO OUT PUT BUFFER
5131 034234  010465  177776            MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5132 034240  004737  017124            JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5133 034244  016501  000000            MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5134 034250  012702  000200            MOV     #SSR,R2                 ;SET UP EXPECTED
5135 034254  020102                    CMP     R1,R2                   ;ARE THEY EQUAL
5136 034256  001406                    BEQ     70$                     ;BR, IF OK
5137 034260  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5141                                                                    ;SOFT ERROR, DON'T CARE ABOUT WRITE
5142                                                                    ;COMMAND'S RESULTS - CHECKING SKIP
5143                                                                    ;TAPE MARK COMMAND
5144 034264                            ERRSOFT ERRNO,T30WDD,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     034264  104457                                                            TRAP    C$ERSOFT
     034266  000336                                                            .WORD   222
     034270  037060                                                            .WORD   T30WDD
     034272  011700                                                            .WORD   PKTSSR
5145 034274                    70$:    CKLOOP                          ;LOOP IF SELECTED
     034274  104406                                                            TRAP    C$CLP1
5146 034276  005203                    INC     R3                      ;COUNT THE RECORD COUNTER DOWN
5147 034300  020327  000021            CMP     R3,#21                  ;AT 20 YET
5148 034304  001331                    BNE     65$                     ;BR, IF NOT AT 20 RECORDS WRITTEN
5149
5150                           ;*************************************************************
5151                           ;
5152                           ;WRITE TAPE MARK,ACK,CVC=1 COMMAND
5153                           ;
5154                           ;*************************************************************
5155
5156 034306  012737  141011  036510        MOV     #141011,T30PK3          ;WRITE TAPE MARK,ACK,CVC=1 COMMAND
5157 034314  012704  036510            MOV     #T30PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5158 034320  010465  177776            MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5159 034324  004737  017124            JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5160 034330  016501  000000            MOV     TSSR(R5),R1             ;PICK UP TSSR
5161 034334  012702  000200            MOV     #SSR,R2                 ;SET UP EXPECTED (SSR ONLY)
5162 034340  020102                    CMP     R1,R2                   ;WAS STATUS GOOD
5163 034342  001406                    BEQ     160$                    ;BR, IF TERMINATION WAS GOOD
5164 034344  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5168 034350                            ERRHRD  ERRNO,T30WDC,PKTSSR     ;TSSR NOT CORRECT AFTER WRT TAPE M.
     034350  104456                                                            TRAP    C$ERHRD
     034352  000337                                                            .WORD   223
     034354  040252                                                            .WORD   T30WDC
     034356  011700                                                            .WORD   PKTSSR
5169 034360                    160$:   CKLOOP                          ;LOOP IF SELECTED
     034360  104406                                                            TRAP    C$CLP1
5170 034362  005237  036544            INC     T30FCN                  ;COUNT THE "FILE" COUNTER DOWN
5171 034366  023727  036544  000031    CMP     T30FCN,#25.             ;WRITE 25 FILES TO TAPE
5172 034374  001273                    BNE     64$                     ;BR, IF NOT AT 25 FILES WRITTEN
5173
5174                           ;*************************************************************
5175                           ;
5176                           ;WRITE TAPE MARK,ACK,CVC=1 COMMAND
5177                           ;
5178                           ;*************************************************************
```

```
5179
5180 034376  012737  141011  036510          MOV     #141011,T30PK3          ;WRITE TAPE MARK,ACK,CVC=1 COMMAND
5181 034404  012704  036510                  MOV     #T30PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5182 034410  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5183 034414  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5184 034420  016501  000000                  MOV     TSSR(R5),R1             ;PICK UP TSSR
5185 034424  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED (SSR ONLY)
5186 034430  020102                          CMP     R1,R2                   ;WAS STATUS GOOD
5187 034432  001406                          BEQ     165$                    ;BR, IF TERMINATION WAS GOOD
5188 034434  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5192 034440                                  ERRHRD  ERRNO,T30WDC,PKTSSR     ;TSSR NOT CORRECT AFTER WRT TAPE M.
     034440  104456                                                                  TRAP    C$ERHRD
     034442  000340                                                                  .WORD   224
     034444  040252                                                                  .WORD   T30WDC
     034446  011700                                                                  .WORD   PKTSSR
5193 034450                          165$:   CKLOOP                          ;LOOP IF SELECTED
     034450  104406                                                                  TRAP    C$CLP1
5194
5195                                 ;*************************************************************
5196                                 ;
5197                                 ;ISSUE A REWIND TO TAPE DRIVE AND WAIT FOR SSR TO SET
5198                                 ;
5199                                 ;*************************************************************
5200
5201 034452  004737  010434                  JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
5202 034456  103411                          BCS     170$                    ;BR, IF NO PROBLEM
5203 034460  010004                          MOV     R0,R4                   ;GET PACKET ADDRESS
5204 034462  016501  000000                  MOV     TSSR(R5),R1             ;GET STATUS REGISTER
5205 034466  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5209 034472                                  ERRHRD  ERRNO,T30RWN,PKTSSR     ;REWIND NOT ACCEPTED
     034472  104456                                                                  TRAP    C$ERHRD
     034474  000341                                                                  .WORD   225
     034476  040130                                                                  .WORD   T30RWN
     034500  011700                                                                  .WORD   PKTSSR
5210 034502                          170$:   CKLOOP                          ;LOOP IF SELECTED
     034502  104406                                                                  TRAP    C$CLP1
5211
5212                                 ;*************************************************************
5213                                 ;
5214                                 ;GET EXTENDED STATUS REGISTER ZERO (XST0) FROM MESSAGE BUFFER
5215                                 ;
5216                                 ;*************************************************************
5217
5218 034504  013701  036406                  MOV     T30BFR+6,R1             ;PICK UP XST0
5219 034510  010102                          MOV     R1,R2                   ;SET UP EXPECTED
5220 034512  052702  000002                  BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
5221 034516  020102                          CMP     R1,R2                   ;DOES EXP = REC'D
5222 034520  001406                          BEQ     180$                    ;BR, IF EQUAL (OK)
5223 034522  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5227 034526                                  ERRHRD  ERRNO,T30BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     034526  104456                                                                  TRAP    C$ERHRD
     034530  000342                                                                  .WORD   226
     034532  037731                                                                  .WORD   T30BOT
     034534  016350                                                                  .WORD   EXPREC
5228 034536                          180$:   CKLOOP                          ;LOOP IF SELECTED
     034536  104406                                                                  TRAP    C$CLP1
5229 034540  012737  000002  036544          MOV     #2,T30FCN               ;SET TO NUMBER OF SKIP "FILES"
```

```
5230 034546  012703  036526              MOV     #T30IMV,R3              ;SET UP POINTER TO COMMAND TABLE
5231
5232 034552  011337  036376      182$:   MOV     (R3),T30ETM             ;GET NEXT COMMAND
5233 034556  012704  036360              MOV     #T30PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
5234
5235                             ;*************************************************************
5236                             ;
5237                             ;ISSUE WRITE CHARACTERISTICS COMMAND
5238                             ;
5239                             ;*************************************************************
5240
5241 034562  004737  010332              JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
5242 034566  103407                      BCS     188$                    ;BR, IF COMMAND ISSUED OK
5243 034570  004737  020104              JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5247 034574  010001                      MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
5248 034576                              ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     034576  104456                                                              TRAP    C$ERHRD
     034600  000343                                                              .WORD   227
     034602  004760                                                              .WORD   WRTMSG
     034604  011666                                                              .WORD   SFIMSG
5249 034606                      188$:   CKLOOP                          ;LOOP IF SELECTED
     034606  104406                                                              TRAP    C$CLP1
5250
5251                             ;*************************************************************
5252                             ;
5253                             ;SKIP TAPE MARK,ACK,CVC=1 COMMAND
5254                             ;
5255                             ;*************************************************************
5256
5257 034610  012737  141010  036510          MOV     #141010,T30PK3          ;SKIP TAPE MARK,ACK,CVC=1 COMMAND
5258 034616  013737  036544  036512          MOV     T30FCN,T30RB            ;SET UP NUMBER TO SKIP
5259 034624  012704  036510                  MOV     #T30PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5260 034630  010465  177776          189$:   MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5261 034634  012737  176750  036546          MOV     #65000.,T30DLY          ;SET UP DELAY COUNTER
5262 034642  004737  017124          190$:   JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5263 034646  016501  000000                  MOV     TSSR(R5),R1             ;PICK UP TSSR
5264 034652  032701  000200                  BIT     #SSR,R1                 ;IS SSR SET YET
5265 034656  001017                          BNE     191$                    ;BR, IF SSR IS SET
5266 034660                                  DELAY   250                     ;CALL DELAY ROUTINE
     034660  012727  000250                                                          MOV     #250,(PC)+
     034664  000000                                                                  .WORD   0
     034666  013727  002116                                                          MOV     L$DLY,(PC)+
     034672  000000                                                                  .WORD   0
     034674  005367  177772                                                          DEC     -6(PC)
     034700  001375                                                                  BNE     .-4
     034702  005367  177756                                                          DEC     -22(PC)
     034706  001367                                                                  BNE     .-20
5267 034710  005337  036546                  DEC     T30DLY                  ;BUMP DELAY ROUTINE
5268 034714  001352                          BNE     190$                    ;BR, IF MORE DELAY TO GO
5269 034716  012702  000200          191$:   MOV     #SSR,R2                 ;SET UP EXPECTED (SSR ONLY)
5270 034722  020102                          CMP     R1,R2                   ;WAS STATUS GOOD
5271 034724  001406                          BEQ     192$                    ;BR, IF TERMINATION WAS GOOD
5272 034726  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5276 034732                                  ERRHRD  ERRNO,T30SKM,PKTSSR     ;TSSR NOT CORRECT AFTER SKIP TAPE M.
     034732  104456                                                              TRAP    C$ERHRD
     034734  000344                                                              .WORD   228
     034736  037004                                                              .WORD   T30SKM
```

```
                034740  011700                                                          .WORD   PKTSSR
5277            034742                  192$:   CKLOOP                  ;LOOP IF SELECTED
                034742  104406                                                          TRAP    C$CLP1
5278
5279                                    ;*************************************************************
5280                                    ;
5281                                    ;GET EXTENDED STATUS REGISTER ZERO (XST0) FROM MESSAGE BUFFER
5282                                    ;
5283                                    ;*************************************************************
5284
5285 034744    013701  036406                  MOV     T30BFR+6,R1             ;PICK UP XST0
5286 034750    010102                          MOV     R1,R2                   ;SET UP EXPECTED
5287 034752    052702  100000                  BIS     #BIT15,R2               ;SET TMK BIT IN EXPECTED
5288 034756    0201C2                          CMP     R1,R2                   ;DOES EXP = REC'D
5289 034760    001406                          BEQ     195$                    ;BR, IF EQUAL (OK)
5290 034762    004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5294 034766                                    ERRHRD  ERRNO,T30TMK,EXPREC     ;TMK NOT SET AFTER WRT TAPE MARK
                034766  104456                                                          TRAP    C$ERHRD
                034770  000345                                                          .WORD   229
                034772  040404                                                          .WORD   T30TMK
                034774  016350                                                          .WORD   EXPREC
5295 034776                     195$:   CKLOOP                  ;LOOP IF SELECTED
                034776  104406                                                          TRAP    C$CLP1
5296 035000    012700  177777                  MOV     #177777,R0              ;VALUE TO WRITTEN TO MEMORY
5297 035004    004737  020376                  JSR     PC,FILLMEM              ;FILL MEM WITH ALL ONES
5298 035010    013737  003076  036512          MOV     FREE,T30RB              ;STARTING READ BUFFER ADDRESS
5299
5300                                    ;*************************************************************
5301                                    ;
5302                                    ;READ FORWARD,ACK,CVC=1 COMMAND
5303                                    ;
5304                                    ;*************************************************************
5305
5306 035016    012737  140001  036510          MOV     #140001,T30PK3          ;READ FORWARD,ACK,CVC=1 COMMAND
5307 035024    012704  036510                  MOV     #T30PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5308 035030    012737  000024  036516          MOV     #20.,T30SZ              ;SET UP RECORD SIZE IN PACKET
5309 035036    010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5310 035042    004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5311 035046    016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5312 035052    012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
5313 035056    020102                          CMP     R1,R2                   ;ARE THEY EQUAL
5314 035060    001406                          BEQ     200$                    ;BR, IF OK
5315 035062    004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5319 035066                                    ERRHRD  ERRNO,T30RDF,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
                035066  104456                                                          TRAP    C$ERHRD
                035070  000346                                                          .WORD   230
                035072  037303                                                          .WORD   T30RDF
                035074  011700                                                          .WORD   PKTSSR
5320 035076                     200$:   CKLOOP                  ;LOOP IF SELECTED
                035076  104406                                                          TRAP    C$CLP1
5321 035100    017701  145772                  MOV     @FREE,R1                ;FIRST LOC IN READ BUFFER
5322 035104    012702  177777                  MOV     #177777,R2              ;EXPECTED IF NO DATA TRANS.
5323 035110    020102                          CMP     R1,R2                   ;DID ANY DATA GET TRANSFERRED
5324 035112    001006                          BNE     220$                    ;BR, IF NO DATA TRANS (GOOD)
5325 035114    004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5329 035120                                    ERRHRD  ERRNO,T30DTR,EXPREC     ;DATA TRANSFERRED ON READ TAPE MARK
                035120  104456                                                          TRAP    C$ERHRD
```

```
                035122  000347                                                          .WORD   231
                035124  040760                                                          .WORD   T30DTR
                035126  016350                                                          .WORD   EXPREC
5330            035130                  220$:   CKLOOP                  ;LOOP IF SELECTED
                035130  104406                                                          TRAP    C$CLP1
5331            035132  013702  036544          MOV     T30FCN,R2       ;GET NUMBER OF SKIPS
5332            035136  005202                  INC     R2              ;SET TO CORRECT FILE VALUE
5333            035140  000302                  SWAB    R2              ;SWAP BYTE HALVES
5334            035142  052702  000001          BIS     #BIT0,R2        ;SET FOR RECORD #1
5335            035146  017701  145724          MOV     @FREE,R1        ;GET INFO FROM BUFFER
5336            035152  020201                  CMP     R2,R1           ;ARE THEY EQUAL
5337            035154  001406                  BEQ     228$            ;BR, IF EQUAL (OK)
5338            035156  004737  020104          JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
5342            035162                  ERRHRD  ERRNO,T30PTB,EXPREC     ;RECORD POSITION WAS NOT CORRECT
                035162  104456                                                          TRAP    C$ERHRD
                035164  000350                                                          .WORD   232
                035166  037132                                                          .WORD   T30PTB
                035170  016350                                                          .WORD   EXPREC
5343            035172                  228$:   CKLOOP                  ;LOOP IF SELECTED
                035172  104406                                                          TRAP    C$CLP1
5344
5345                                    ;***************************************************************
5346                                    ;
5347                                    ;ISSUE A REWIND TO TAPE DRIVE AND WAIT FOR SSR TO SET
5348                                    ;
5349                                    ;***************************************************************
5350
5351            035174  004737  010434          JSR     PC,REWIND       ;CALL TAPE REWIND COMMAND
5352            035200  103411                  BCS     230$            ;BR, IF NO PROBLEM
5353            035202  010004                  MOV     R0,R4           ;SAVE PACKET ADDRESS
5354            035204  016501  000000          MOV     TSSR(R5),R1     ;GET TSSR STATUS
5355            035210  004737  020104          JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
5359            035214                  ERRHRD  ERRNO,T30RWN,PKTSSR     ;REWIND NOT ACCEPTED
                035214  104456                                                          TRAP    C$ERHRD
                035216  000351                                                          .WORD   233
                035220  040130                                                          .WORD   T30RWN
                035222  011700                                                          .WORD   PKTSSR
5360            035224                  230$:   CKLOOP                  ;LOOP IF SELECTED
                035224  104406                                                          TRAP    C$CLP1
5361
5362                                    ;***************************************************************
5363                                    ;
5364                                    ;GET EXTENDED STATUS REGISTER ZERO (XST0) FROM MESSAGE BUFFER
5365                                    ;
5366                                    ;***************************************************************
5367
5368            035226  013701  036406          MOV     T30BFR+6,R1     ;PICK UP XST0
5369            035232  010102                  MOV     R1,R2           ;SET UP EXPECTED
5370            035234  052702  000002          BIS     #BIT1,R2        ;SET BOT BIT IN EXPECTED
5371            035240  020102                  CMP     R1,R2           ;DOES EXP = REC'D
5372            035242  001406                  BEQ     240$            ;BR, IF EQUAL (OK)
5373            035244  004737  020104          JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
5377            035250                  ERRHRD  ERRNO,T30BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
                035250  104456                                                          TRAP    C$ERHRD
                035252  000352                                                          .WORD   234
                035254  037731                                                          .WORD   T30BOT
                035256  016350                                                          .WORD   EXPREC
```

```
5378 035260                          240$:   CKLOOP                          ;LOOP IF SELECTED
     035260  104406                                                                  TRAP    C$CLP1
5379 035262  005723                          TST     (R3)+                   ;POINT TO NEXT POSITION
5380 035264  011301                          MOV     (R3),R1                 ;GET NEXT COMMAND ETC.
5381 035266  020127  177777                  CMP     R1,#177777              ;END OF TABLE MARKER
5382 035272  001410                          BEQ     330$                    ;BR, IF AT END OF TABLE
5383 035274  013701  036544                  MOV     T30FCN,R1               ;GET NUMBER OF SKIPS
5384 035300  000241                          CLC                             ;CLEAR THE CARRY BIT
5385 035302  006101                          ROL     R1                      ;PUSH OVER ONE POSITION
5386 035304  010137  036544                  MOV     R1,T30FCN               ;PUT BACK IN COUNTER
5387 035310  000137  034552                  JMP     182$                    ;JUMP TO MORE COMMANDS TO DO
5388 035314                          330$:   CKLOOP                          ;LOOP IF SELECTED
     035314  104406                                                                  TRAP    C$CLP1
5389 035316                          L10045: ENDSUB                          ;<<<<<<<<<<< END SUBTEST >>>>>>>>>>>
     035316
     035316  104403                                                                  TRAP    C$ESUB
```

```
                                        ;*
                                        ;
                                        ;TEST 2, SUBTEST 3
                                        ;
                                        ;
                                        ;VERIFIES THAT A SKIP TAPE MARKS REVERSE COMMAND
                                        ;ISSUED WHILE THE TAPE IS POSITIONED AT BOT CAUSES
                                        ;FUNCTION REJECT TERMINATION, WITH THE NON-EXECUTABLE
                                        ;FUNCTION (NEF) ERROR BIT SET.
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;-
035320                                          BGNSUB                  ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
035320                                                                                      T2.3:
035320  104402                                                                                      TRAP    C$BSUB
035322  004737  041142                         JSR     PC,T30REST      ;SET COMMAND PACKET
035326  005037  036544                         CLR     T30FCN          ;CLEAR FILE COUNTER
035332  004737  041234                         JSR     PC,T30RT2       ;SET UP OTHER COMMAND PACKET
035336  004737  041276                         JSR     PC,T30RT3       ;SET UP OTHER COMMAND PACKET
035342  012737  176750  036546                 MOV     #65000.,T30DLY  ;SET UP DELAY COUNTER
035350  004737  016650                 10$:    JSR     PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
035354  103426                               BCS     20$             ;BR IF INIT WAS OK
035356                                       DELAY   250             ;DELAY ROUTINE CALL
035356  012727  000250                                                        MOV     #250,(PC)+
035362  000000                                                                .WORD   0
035364  013727  002116                                                        MOV     L$DLY,(PC)+
035370  000000                                                                .WORD   0
035372  005367  177772                                                        DEC     -6(PC)
035376  001375                                                                BNE     .-4
035400  005367  177756                                                        DEC     -22(PC)
035404  001367                                                                BNE     .-20
035406  005337  036546                         DEC     T30DLY          ;BUMP COUNTER
035412  001356                               BNE     10$             ;BR, IF MORE COUNTING TO DO
035414  004737  020104                         JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
035420  010001                               MOV     R0,R1           ;CONTENTS OF TSSR REGISTER
035422                                       ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
035422  104455                                                                  TRAP    C$ERDF
035424  000353                                                                  .WORD   235
035426  003554                                                                  .WORD   SFIERR
035430  011666                                                                  .WORD   SFIMSG
035432                                  20$:
                                        ;
035432  012704  036360                         MOV     #T30PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
                                        ;
                                        ;*************************************************************
                                        ;
                                        ;ISSUE WRITE CHARACTERISTICS COMMAND
                                        ;
                                        ;*************************************************************
                                        ;
035436  004737  010332                         JSR     PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
035442  103407                               BCS     23$             ;BR, IF COMMAND ISSUED OK
035444  004737  020104                         JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
035450  010001                               MOV     R0,R1           ;SAVE CONTENTS OF TSSR
```

```
5440 035452                          ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     035452  104456                                                          TRAP    C$ERHRD
     035454  000354                                                          .WORD   236
     035456  004760                                                          .WORD   WRTMSG
     035460  011666                                                          .WORD   SFIMSG
5441 035462                    23$:  CKLOOP                          ;LOOP IF SELECTED
     035462  104406                                                          TRAP    C$CLP1
5442
5443                           ;*************************************************************
5444                           ;
5445                           ;ISSUE A REWIND TO TAPE DRIVE AND WAIT FOR SSR TO SET
5446                           ;
5447                           ;*************************************************************
5448
5449 035464  004737  010434          JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
5450 035470  103411                  BCS     30$                     ;BR, IF NO PROBLEM
5451 035472  010004                  MOV     R0,R4                   ;GET PACKET ADDRESS
5452 035474  016501  000000          MOV     TSSR(R5),R1             ;GET STATUS REGISTER
5453 035500  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5457 035504                          ERRHRD  ERRNO,T30RWN,PKTSSR     ;REWIND NOT ACCEPTED
     035504  104456                                                          TRAP    C$ERHRD
     035506  000355                                                          .WORD   237
     035510  040130                                                          .WORD   T30RWN
     035512  011700                                                          .WORD   PKTSSR
5458 035514                    30$:  CKLOOP                          ;LOOP IF SELECTED
     035514  104406                                                          TRAP    C$CLP1
5459
5460                           ;*************************************************************
5461                           ;
5462                           ;GET EXTENDED STATUS REGISTER ZERO (XST0) FROM MESSAGE BUFFER
5463                           ;
5464                           ;*************************************************************
5465
5466 035516  013701  036406          MOV     T30BFR+6,R1             ;PICK UP XST0
5467 035522  010102                  MOV     R1,R2                   ;SET UP EXPECTED
5468 035524  052702  000002          BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
5469 035530  020102                  CMP     R1,R2                   ;DOES EXP = REC'D
5470 035532  001406                  BEQ     40$                     ;BR, IF EQUAL (OK)
5471 035534  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5475 035540                          ERRHRD  ERRNO,T30BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     035540  104456                                                          TRAP    C$ERHRD
     035542  000356                                                          .WORD   238
     035544  037731                                                          .WORD   T30BOT
     035546  016350                                                          .WORD   EXPREC
5476 035550                    40$:  CKLOOP                          ;LOOP IF SELECTED
     035550  104406                                                          TRAP    C$CLP1
5477 035552  012737  000001  036512          MOV     #1,T30WB                ;SET # OF TM TO SKIP
5478
5479                           ;*************************************************************
5480                           ;
5481                           ;SKIP TAPE MARK REVERSE,ACK,CVC=1 COMMAND
5482                           ;
5483                           ;*************************************************************
5484
5485 035560  012737  141410  036510          MOV     #141410,T30PK3          ;SKIP TAPE MARK REVERSE,ACK,CVC=1 CMD
5486 035566  012704  036510                  MOV     #T30PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5487 035572  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
```

```
5488 035576  004737  017124          JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5489 035602  016501  000000          MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5490 035606  012702  100206          MOV     #SSR!SC!BIT1!BIT2,R2    ;SET UP EXPECTED
5491 035612  020102                  CMP     R1,R2                   ;ARE THEY EQUAL
5492 035614  001406                  BEQ     70$                     ;BR, IF OK
5493 035616  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5497 035622                          ERRHRD  ERRNO,T30IBT,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     035622  104456                                                          TRAP    C$ERHRD
     035624  000357                                                          .WORD   239
     035626  036721                                                          .WORD   T30IBT
     035630  011700                                                          .WORD   PKTSSR
5498 035632                  70$:    CKLOOP                          ;LOOP IF SELECTED
     035632  104406                                                          TRAP    C$CLP1
5499
5500                         ;*************************************************************
5501                         ;
5502                         ;GET EXTENDED STATUS REGISTER ZERO (XST0) FROM MESSAGE BUFFER
5503                         ;
5504                         ;*************************************************************
5505
5506 035634  013701  036406          MOV     T30BFR+6,R1             ;PICK UP XST0
5507 035640  010102                  MOV     R1,R2                   ;SET UP EXPECTED
5508 035642  052702  002000          BIS     #BIT10,R2               ;SET NEF BIT IN EXPECTED
5509 035646  020102                  CMP     R1,R2                   ;DOES EXP = REC'D
5510 035650  001406                  BEQ     180$                    ;BR, IF EQUAL (OK)
5511 035652  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5515 035656                          ERRHRD  ERRNO,T30NEF,EXPREC     ;TAPE NOT AT NEF
     035656  104456                                                          TRAP    C$ERHRD
     035660  000360                                                          .WORD   240
     035662  040466                                                          .WORD   T30NEF
     035664  016350                                                          .WORD   EXPREC
5516 035666                  180$:   CKLOOP                          ;LOOP IF SELECTED
     035666  104406                                                          TRAP    C$CLP1
5517 035670                          ENDSUB                          ;<<<<<<<<<<< END SUBTEST >>>>>>>>>>>
     035670                                                                L10046:
     035670  104403                                                          TRAP    C$ESUB
```

```
                                        ;+
                                        ;
                                        ;TEST 2, SUBTEST 4
                                        ;
                                        ;
                                        ;VERIFIES THAT A SKIP TAPE MARKS REVERSE COMMAND
                                        ;ISSUED WHILE THE TAPE IS POSITIONED JUST BEFORE THE
                                        ;FIRST RECORD ON ON TAPE (BUT NOT AT BOT) CAUSES TAPE
                                        ;STATUS ALERT TERMINATION, WITH THE REVERSE INTO BOT
                                        ;(RIB) STATUS BIT SET.
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;-
035672                                          BGNSUB                  ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
035672                                                                                  T2.4:
035672  104402                                                                                  TRAP    C$BSUB
035674  004737  041142                        JSR     PC,T30REST      ;SET COMMAND PACKET
035700  005037  036544                        CLR     T30FCN          ;CLEAR FILE COUNTER
035704  004737  041234                        JSR     PC,T30RT2       ;SET UP OTHER COMMAND PACKET
035710  004737  041276                        JSR     PC,T30RT3       ;SET UP OTHER COMMAND PACKET
035714  012737  176750  036546                MOV     #65000.,T30DLY  ;SET UP DELAY COUNTER
035722  004737  016650          10$:          JSR     PC,$OFINIT      ;DO INITIALIZE ON CONTROLLER
035726  103426                                BCS     20$             ;BR IF INIT WAS OK
035730                                        DELAY   250             ;DELAY ROUTINE CALL
035730  012727  000250                                                          MOV     #250.,(PC)+
035734  000000                                                                  .WORD   0
035736  013727  002116                                                          MOV     L$DLY,(PC)+
035742  000000                                                                  .WORD   0
035744  005367  177772                                                          DEC     -6(PC)
035750  001375                                                                  BNE     .-4
035752  005367  177756                                                          DEC     -22(PC)
035756  001367                                                                  BNE     .-20
035760  005337  036546                        DEC     T30DLY          ;BUMP COUNTER
035764  001356                                BNE     10$             ;BR, IF MORE COUNTING TO DO
035766  004737  020104                        JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
035772  010001                                MOV     R0,R1           ;CONTENTS OF TSSR REGISTER
035774                                        ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
035774  104455                                                                  TRAP    C$ERDF
035776  000361                                                                  .WORD   241
036000  003554                                                                  .WORD   SFIERR
036002  011666                                                                  .WORD   SFIMSG
036004                          20$:
                                        ;
036004  012704  036360                        MOV     #T30PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
                                        ;
                                        ;*************************************************************
                                        ;
                                        ;ISSUE WRITE CHARACTERISTICS COMMAND
                                        ;
                                        ;*************************************************************
                                        ;
036010  004737  010332                        JSR     PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
036014  103407                                BCS     23$             ;BR, IF COMMAND ISSUED OK
036016  004737  020104                        JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
```

```
5568 036022  010001                    MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
5569 036024                            ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     036024  104456                                                            TRAP    C$ERHRD
     036026  000362                                                            .WORD   242
     036030  004760                                                            .WORD   WRTMSG
     036032  011666                                                            .WORD   SFIMSG
5570 036034                    23$:    CKLOOP                          ;LOOP IF SELECTED
     036034  104406                                                            TRAP    C$CLP1
5571
5572                           ;*************************************************************
5573                           ;
5574                           ;ISSUE A REWIND TO TAPE DRIVE AND WAIT FOR SSR TO SET
5575                           ;
5576                           ;*************************************************************
5577
5578 036036  004737  010434            JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
5579 036042  103411                    BCS     30$                     ;BR, IF NO PROBLEM
5580 036044  010004                    MOV     R0,R4                   ;GET PACKET ADDRESS
5581 036046  016501  000000            MOV     TSSR(R5),R1             ;GET STATUS REGISTER
5582 036052  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5586 036056                            ERRHRD  ERRNO,T3ORWN,PKTSSR     ;REWIND NOT ACCEPTED
     036056  104456                                                            TRAP    C$ERHRD
     036060  000363                                                            .WORD   243
     036062  040130                                                            .WORD   T3ORWN
     036064  011700                                                            .WORD   PKTSSR
5587 036066                    30$:    CKLOOP                          ;LOOP IF SELECTED
     036066  104406                                                            TRAP    C$CLP1
5588
5589                           ;*************************************************************
5590                           ;
5591                           ;GET EXTENDED STATUS REGISTER ZERO (XST0) FROM MESSAGE BUFFER
5592                           ;
5593                           ;*************************************************************
5594
5595 036070  013701  036406            MOV     T30BFR+6,R1             ;PICK UP XST0
5596 036074  010102                    MOV     R1,R2                   ;SET UP EXPECTED
5597 036076  052702  000002            BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
5598 036102  020102                    CMP     R1,R2                   ;DOES EXP = REC'D
5599 036104  001406                    BEQ     40$                     ;BR, IF EQUAL (OK)
5600 036106  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5604 036112                            ERRHRD  ERRNO,T30BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     036112  104456                                                            TRAP    C$ERHRD
     036114  000364                                                            .WORD   244
     036116  037731                                                            .WORD   T30BOT
     036120  016350                                                            .WORD   EXPREC
5605 036122                    40$:    CKLOOP                          ;LOOP IF SELECTED
     036122  104406                                                            TRAP    C$CLP1
5606 036124  013737  003076  036512    MOV     FREE,T30WB              ;SET UP GOOD WRITE BUFFER
5607 036132  012737  000400  036516    MOV     #256.,T30SZ             ;SET UP SIZE
5608
5609                           ;*************************************************************
5610                           ;
5611                           ;WRITE DATA,ACK,CVC=1 COMMAND
5612                           ;
5613                           ;*************************************************************
5614
5615 036140  012737  140005  036510    MOV     #140005,T30PK3          ;WRITE DATA,ACK,CVC=1 COMMAND
```

```
 5616 036146  012704  036510            MOV     #T30PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
 5617 036152  010465  177776            MOV     R4,TSDB(R5)             ;ISSUE COMMAND
 5618 036156  004737  017124            JSR     PC,WAITF                ;WAIT FOR SSR TO SET
 5619 036162  016501  000000            MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
 5620 036166  012702  000200            MOV     #SSR,R2                 ;SET UP EXPECTED
 5621 036172  020102                    CMP     R1,R2                   ;ARE THEY EQUAL
 5622 036174  001406                    BEQ     70$                     ;BR, IF OK
 5623 036176  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 5627                                                                   ;SOFT ERROR, DON'T CARE ABOUT WRITE
 5628                                                                   ;COMMAND'S RESULTS - CHECKING SKIP
 5629                                                                   ;TAPE MARK COMMAND
 5630 036202                            ERRSOFT ERRNO,T30WDD,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
      036202  104457                                                                  TRAP    C$ERSOFT
      036204  000365                                                                  .WORD   245
      036206  037060                                                                  .WORD   T30WDD
      036210  011700                                                                  .WORD   PKTSSR
 5631 036212                    70$:    CKLOOP                          ;LOOP IF SELECTED
      036212  104406                                                                  TRAP    C$CLP1
 5632
 5633                           ;*************************************************************
 5634                           ;
 5635                           ;SKIP TAPE MARK REVERSE,ACK,CVC=1 COMMAND
 5636                           ;
 5637                           ;*************************************************************
 5638
 5639 036214  012737  000001  036512        MOV     #1,T30WB                ;# OF TM TO SKIP
 5640 036222  012737  141410  036510        MOV     #141410,T30PK3          ;SKIP TAPE MARK REVERSE,ACK,CVC=1 CMD
 5641 036230  012704  036510            MOV     #T30PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
 5642 036234  010465  177776            MOV     R4,TSDB(R5)             ;ISSUE COMMAND
 5643 036240  004737  017124            JSR     PC,WAITF                ;WAIT FOR SSR TO SET
 5644 036244  016501  000000            MOV     TSSR(R5),R1             ;PICK UP TSSR
 5645 036250  012702  100204            MOV     #SSR!BIT2!SC,R2         ;SET UP EXPECTED (SSR AND SC ONLY)
 5646 036254  020102                    CMP     R1,R2                   ;WAS STATUS GOOD
 5647 036256  001406                    BEQ     160$                    ;BR, IF TERMINATION WAS GOOD
 5648 036260  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 5652 036264                            ERRHRD  ERRNO,T30IBU,PKTSSR     ;TSSR NOT CORRECT AFTER WRT TAPE M.
      036264  104456                                                                  TRAP    C$ERHRD
      036266  000366                                                                  .WORD   246
      036270  036550                                                                  .WORD   T30IBU
      036272  011700                                                                  .WORD   PKTSSR
 5653 036274                    160$:   CKLOOP                          ;LOOP IF SELECTED
      036274  104406                                                                  TRAP    C$CLP1
 5654
 5655                           ;*************************************************************
 5656                           ;
 5657                           ;GET EXTENDED STATUS REGISTER ZERO (XST3) FROM MESSAGE BUFFER
 5658                           ;
 5659                           ;*************************************************************
 5660
 5661 036276  013701  036414            MOV     T30BFR+14,R1            ;PICK UP XST3
 5662 036302  010102                    MOV     R1,R2                   ;SET UP EXPECTED
 5663 036304  052702  000001            BIS     #BIT0,R2                ;SET RIB BIT IN EXPECTED
 5664 036310  020102                    CMP     R1,R2                   ;DOES EXP = REC'D
 5665 036312  001406                    BEQ     170$                    ;BR, IF EQUAL (OK)
 5666 036314  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 5670 036320                            ERRHRD  ERRNO,T30RIB,EXPREC     ;TAPE NOT AT RIB
      036320  104456                                                                  TRAP    C$ERHRD
```

```
      036322  000367                                                  .WORD   247
      036324  036635                                                  .WORD   T3ORIB
      036326  016350                                                  .WORD   EXPREC
 5671 036330                          170$:   CKLOOP                 ;LOOP IF SELECTED
      036330  104406                                                  TRAP    C$CLP1
 5672 036332                                  ENDSUB                 ;<<<<<<<<<<< END SUBTEST >>>>>>>>>>>
      036332                                                        L10047:
      036332  104403                                                  TRAP    C$ESUB
 5673                                 ;SUBTEST END
 5674                                 ;
 5675                                 ;
 5676 036334  004737  017360                 JSR     PC,TSTLOOP     ;DO WE NEED TO ITERATE TEST
 5677 036340  103002                         BCC     400$           ;BR, IF NO LOOP REQUIRED
 5678 036342  000137  032362                 JMP     T3OLOOP        ;EXECUTE AGAIN
 5679 036346                          400$:   EXIT    TST            ;ALL DONE THIS TEST
      036346  104432                                                  TRAP    C$EXIT
      036350  002750                                                  .WORD   L10043-.
```

```
                                ;+
                                ;LOCAL STORAGE FOR THIS TEST
                                ;-
                                        .BLKB   10-<.-TUV2A&7>
                                T30PACKET:                              ;COMMAND PACKET FOR TEST
                                        .WORD   100004                  ;WRITE CHARACTERISTICS COMMAND, WITH , ACK
                                        .WORD   T30DATA                 ;ADDRESS OF CHARACTERISTICS BLOCK
                                        .WORD   0
                                        .WORD   10.                     ;STARTING VALUE OF BLOCK SIZE
                                T30DATA:                                ;CHARACTERISTICS DATA BLOCK
                                        .WORD   T30BFR                  ;ADDRESS OF MESSAGE BUFFER
                                        .WORD   0
                                        .WORD   20.                     ;LENGTH OF MESSAGE BUFFER
                                T30ETM: .WORD   0                       ;SKIP TAPE MARK CONTROL
                                T30BFR: .BLKW   25.                     ;MESSAGE BUFFER
                                ;
                                ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
                                ;
                                        .BLKB   10-<.-TUV2A&7>
                                T30PK2:
                                        .WORD   100006                  ;WRITE SUB SYS MEM COMMAND,  AND ACK
                                        .WORD   T30BF2                  ;ADDRESS OF SELECT BLOCK DATA
                                        .WORD   0
                                        .WORD   6.                      ;SIZE OF DATA PACKET
                                        .BLKB   10-<.-TUV2A&7>
                                T30PK3:
                                        .WORD   100205                  ;REREAD COMMAND, IE AND ACK
                                T30RB:
                                T30WB:  .WORD   FREE                    ;ADDRESS OF WRITE BUFFER
                                        .WORD   0
                                T30SZ:  .WORD   0                       ;SIZE OF BUFFER (EXTENT)
                                        .EVEN
                                T30BF2:
                                T30BS0: .BYTE   10                      ;BSEL0 AREA
                                T30BS1: .BYTE   200                     ;BSEL1 AREA
                                T30S2:  .WORD   0                       ;SEL 2 AREA
                                T30S3:  .WORD   0                       ;DATA AREA
                                ;
                                ;
                                        .EVEN
                                ;TAPE MOTION PACKET COMMAND VALUES

                                T30IMV:
                                T30RN:
                                        .WORD   000000                  ;NEITHER EWB NOR ESS
                                        .WORD   000100                  ;EWB SET
                                        .WORD   000200                  ;ESS SET
                                        .WORD   000300                  ;BOTH EWB AND ESS SET
                                        .WORD   177777                  ;END OF DATA
                                T30CNT: .WORD   0                       ;TAPE TIMER COUNTER STORAGE AREA
                                T30CNU: .WORD   0                       ;TAPE TIMER COUNTER STORAGE AREA
                                T30FCN: .WORD   0                       ;FILE NUMBER COUNTER
                                T30DLY: .WORD   0                       ;DELAY COUNTER STORAGE
```

```
                                        ;+
                                        ;LOCAL TEXT MESSAGES FOR TEST
                                        ;-

036550  124  123  123  T30IBU: .ASCIZ  'TSSR Incorrect After SKIP TAPE MARK REVERSE Into BOT'
036635  122  111  102  T30RIB: .ASCIZ  'RIB Bit (XST3) Failed To Set After Reverse Into BOT'
036721  124  123  123  T30IBT: .ASCIZ  'TSSR Incorrect After SKIP TAPE MARK REVERSE At BOT'
037004  124  123  123  T30SKM: .ASCIZ  'TSSR Incorrect After SKIP TAPE MARK Command'
037060  124  123  123  T30WDD: .ASCIZ  'TSSR Not Correct After WRITE DATA Command'
037132  124  141  160  T30PTB: .ASCIZ  'Tape Not Positioned On Correct Record After READ REVERSE'
037223  124  141  160  T30TPB: .ASCIZ  'Tape Not Positioned On Second File First Record'
037303  124  123  123  T30RDF: .ASCIZ  'TSSR Incorrect After READ FORWARD Into "File"'
037361  124  123  123  T30RDG: .ASCIZ  'TSSR Incorrect After SPACE Command Into TAPE MARK'
037443  124  123  123  T30WDF: .ASCIZ  'TSSR Not Correct After Illegal Mode Bits Set'
037520  111  154  154  T30LOQ: .ASCIZ  'Illegal Mode Bits, Failed To Set ILC Bit In XST0'
037601  127  122  111  T30SSR: .ASCIZ  'WRITE MISCELLANEOUS Command Not Accepted'
037652  124  123  123  T30WDE: .ASCIZ  'TSSR Not Correct After SKIP TAPE MARKS, At BOT'
037731  124  141  160  T30BOT: .ASCIZ  'Tape Not At BOT After REWIND Command'
037776  124  123  123  T30TM:  .ASCIZ  'TSSR Not Correct After SPACE FORWARD Command'
040053  124  123  123  T30TM2: .ASCIZ  'TSSR Not Correct After SPACE REVERSE Command'
040130  122  145  167  T30RWN: .ASCIZ  'Rewind (POSITION) Command Not Accepted'
040177  104  162  151  T30OFL: .ASCIZ  'Drive 7 Select Failed To Set "OFL" In TSSR'
040252  124  123  123  T30WDC: .ASCIZ  'TSSR Not Correct After WRITE TAPE MARK Command'
040331  103  126  103  T30VCK: .ASCIZ  'CVC Set, Didn''t Reset VCK In Message Buffer'
040404  124  115  113  T30TMK: .ASCIZ  'TMK Not Set After WRITE TAPE MARK (RETRY) Command'
040466  123  113  111  T30NEF: .ASCIZ  'SKIP TAPE MARKS, At BOT, Failed To Set NEF Bit'
040545  124  115  113  T30RRM: .ASCIZ  'TMK Not Set After READ REVERSE Into TAPE MARK'
040623  124  115  113  T30RRN: .ASCIZ  'TMK Not Set After SPACE REVERSE Into TAPE MARK'
040702  124  115  113  T30RRP: .ASCIZ  'TMK Not Set After READ FORWARD Into TAPE MARK'
040760  116  117  040  T30DTR: .ASCIZ  'NO Data Transferred On READ FORWARD'
041024  104  141  164  T30DTA: .ASCIZ  'Data Compare Error, Data Read From Tape Not Equal To Written'
041121  123  153  151  TST30ID:        .ASCIZ  'Skip Tape Marks'
                                        .EVEN
                                        ;+
                                        ;
                                        ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
                                        ;WRITE SUBSYSTEM MEMORY COMMAND
                                        ;
                                        ;-

041142                                  T30REST:
041142                                          SAVREG                          ;SAVE THE REGISTERS
041146  012701  036360                          MOV     #T30PACKET,R1           ;START OF THE PACKET
041152  012721  100004                          MOV     #100004,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK,
041156  012721  036370                          MOV     #T30DATA,(R1)+          ;ADDRESS OF CHARAISTICS DATA BLOCK
041162  005021                                  CLR     (R1)+                   ;EXTENDED ADDRESS
041164  012721  000012                          MOV     #10.,(R1)+              ;SIZE OF DATA BLOCK IN BYTES
041170  012721  036400                          MOV     #T30BFR,(R1)+           ;ADDRESS OF MESSAGE BUFFER
041174  005021                                  CLR     (R1)+
041176  012721  000024                          MOV     #20.,(R1)+              ;LENGTH OF MESSAGE BUFFER
041202  005021                                  CLR     (R1)+
041204  012711  000000                          MOV     #0,(R1)                 ;SELECT DRIVE ZERO
041210  012702  000030                          MOV     #24.,R2                 ;NUMBER OF LOCATIONS TO BE CLEARED
041214  012762  177777  036400  64$:           MOV     #177777,T30BFR(R2)      ;ALL ONES TO MESSAGE BUFFER
041222  005742                                  TST     -(R2)                   ;NEXT LOCATION
```

```
 5798 041224  022702  000000                 CMP     #0.,R2                  ;CHECK R2 FOR DONE
 5799 041230  001371                         BNE     64$                     ;KEEP GOING UNTIL DONE
 5800 041232  000207                         RTS     PC                      ;RETURN
 5801
 5802
 5803 041234                         T30RT2:
 5804 041234                                 SAVREG                          ;SAVE THE REGISTERS
 5805 041240  012701  036470                MOV     #T30PK2,R1              ;START OF THE PACKET
 5806 041244  012721  100006                MOV     #100006,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK,
 5807 041250  012721  036520                MOV     #T30BF2,(R1)+           ;ADDRESS OF DATA BLOCK
 5808 041254  005021                        CLR     (R1)+                   ;EXTENDED ADDRESS
 5809 041256  012721  000006                MOV     #6.,(R1)+               ;SIZE OF DATA BLOCK IN BYTES
 5810 041262  005021                        CLR     (R1)+
 5811 041264  012701  036520                MOV     #T30BF2,R1              ;POINT TO DATA SEL AREA
 5812 041270  005021                        CLR     (R1)+
 5813 041272  005011                        CLR     (R1)
 5814 041274  000207                        RTS     PC                      ;RETURN
 5815 041276                         T30RT3:
 5816 041276                                 SAVREG                          ;SAVE REGISTERS
 5817 041302  012701  036510                MOV     #T30PK3,R1              ;SET UP POINTER ADDRESS
 5818 041306  005021                        CLR     (R1)+                   ;COMMAND SPACE
 5819 041310  005021                        CLR     (R1)+                   ;ADDRESS OF DATA BLOCK
 5820 041312  005021                        CLR     (R1)+                   ;EXTENDED ADDRESS
 5821 041314  005011                        CLR     (R1)                    ;SIZE OF DATA TRANSFER BLOCK
 5822 041316  000207                        RTS     PC                      ;RETURN
 5823 041320                                 ENDTST
      041320
      041320  104401                                                L10043:
                                                                             TRAP    C$ETST
```

```
                                        .SBTTL  TEST  3: NO-OP ("CLEAN TAPE") AND INITIALIZE
                                ;*
                                ;
                                ;THIS TEST VERIFIES PROPER OPERATION OF THE NO-OP ("CLEAN TAPE") AND INITIALIZE
                                ;COMMAND (SPACE REVERSE, ERASE, WRITE DATA)
                                ;
                                ;
                                ;THE TEST CONSISTS OF THE FOLLOWING 2 SUBTESTS
                                ;
                                ;
                                ;
                                ;-
041322                                  BGNTST
041322                                                                  T3::
041322  005037  002172                  CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
041326  005037  003104                  CLR     KTFLG                   ;HOLD OFF KT11
041332  012737  005676  002150          MOV     #EPRT1,EPRTSW           ;PRIMARY ERROR MESSAGE
041340  012700  046413                  MOV     #TST31ID,R0             ;ASCII MESSAGE TO IDENTIFY TEST
041344  004737  017412                  JSR     PC,TSTSETUP             ;DO INITIAL TEST SETUP
041350  012737  000002  002166          MOV     #2,LOOPCNT              ;PERFORM 2 ITERATIONS
041356  005037  043206                  CLR     T31CNT                  ;CLEAR TAPE RECORD COUNTER
                                ;
                                ;-
041362                          T31LOOP:
```

```
                                ;*
                                ;
                                ;TEST 3, SUBTEST 1
                                ;
                                ;       VERIFIES THAT THE NO-OP COMMAND (CORRESPONDS TO THE CLEAN TAPE
                                ;       COMMAND) TERMINATES PROPERLY (NORMAL TERMINATION), STORES PROPER
                                ;       STATUS IN THE MESSAGE BUFFER (LIKE THE GET STATUS COMMAND), AND
                                ;       INDEED DOES NOT MOVE TAPE.  THE TAPE IS FIRST REWOUND AND
                                ;       WRITTEN WITH SEQUENCED TEST RECORDS.  IT IS THEN REWOUND AGAIN
                                ;       AND THE NO-OP COMMAND IS ISSUED.  IT IS VERIFIED THAT THE TAPE
                                ;       IS STILL AT BOT AND THAT PROPER STATUS IS STORED.  THE FIRST
                                ;       RECORD ON TAPE IS READ AND VERIFIED (TO CHECK TAPE POSITION AND
                                ;       VERIFY THAT DATA WAS NOT CHANGED), THEN THE NO-OP COMMAND IS
                                ;       ISSUED AGAIN AND STATUS AND POSITION VERIFIED.
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;-
041362                              BGNSUB                  ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
041362                                                                      T3.1:
041362  104402                                                                      TRAP    C$BSUB
041364  004737  046460                  JSR     PC,T31REST  ;SET COMMAND PACKET
041370  004737  046552                  JSR     PC,T31RT2   ;SET UP OTHER COMMAND PACKET
041374  004737  046614                  JSR     PC,T31RT3   ;SET UP OTHER COMMAND PACKET
041400  012737  176750  043212          MOV     #65000.,T31DLY  ;SET UP DELAY COUNTER
041406  004737  016650          10$:    JSR     PC,SOFINIT  ;DO INITIALIZE ON CONTROLLER
041412  103426                  BCS     20$         ;BR IF INIT WAS OK
041414                          DELAY   250         ;DELAY ABOUT .25 SEC
041414  012727  000250                                          MOV     #250,(PC)+
041420  000000                                                  .WORD   0
041422  013727  002116                                          MOV     L$DLY,(PC)+
041426  000000                                                  .WORD   0
041430  005367  177772                                          DEC     -6(PC)
041434  001375                                                  BNE     .-4
041436  005367  177756                                          DEC     -22(PC)
041442  001367                                                  BNE     .-20
041444  005337  043212                  DEC     T31DLY      ;BUMP COUNTER
041450  001356                  BNE     10$         ;BR, IF COUNTER NOT DONE
041452  004737  020104                  JSR     PC,FATCHK   ;INC AND CHECK FOR MORE THAN 25 ERRORS
041456  010001                  MOV     R0,R1       ;CONTENTS OF TSSR REGISTER
041460                          ERRDF   ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
041460  104455                                                  TRAP    C$ERDF
041462  000455                                                  .WORD   301
041464  003554                                                  .WORD   SFIERR
041466  011666                                                  .WORD   SFIMSG
041470                  20$:
041470  012704  043030                  MOV     #T31PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
041474  004737  010332                  JSR     PC,WRTCHR   ;ISSUE WRITE CHARACTERISTICS
041500  103407                  BCS     23$         ;BR, IF COMMAND ISSUED OK
041502  004737  020104                  JSR     PC,FATCHK   ;INC AND CHECK FOR MORE THAN 25 ERRORS
041506  010001                  MOV     R0,R1       ;SAVE CONTENTS OF TSSR
041510                          ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTISC FAILED
041510  104456                                                  TRAP    C$ERHRD
041512  000456                                                  .WORD   302
```

```
                041514  004760                                                          .WORD   WRTMSG
                041516  011666                                                          .WORD   SFIMSG
5901            041520                          23$:    CKLOOP                  ;LOOP IF SELECTED
                041520  104406                                                          TRAP    C$CLP1
5902            041522  004737  010434                  JSR     PC,REWIND       ;CALL TAPE REWIND COMMAND
5903            041526  103407                  BCS     30$                     ;BR, IF NO PROBLEM
5904            041530  010004                  MOV     R0,R4                   ;SET UP REWIND PACKET ADDRESS
5905            041532  004737  020104                  JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
5909            041536                          ERRHRD  ERRNO,T31RWN,PKTSSR     ;REWIND NOT ACCEPTED
                041536  104456                                                          TRAP    C$ERHRD
                041540  000457                                                          .WORD   303
                041542  044544                                                          .WORD   T31RWN
                041544  011700                                                          .WORD   PKTSSR
5910            041546                          30$:    CKLOOP                  ;LOOP IF SELECTED
                041546  104406                                                          TRAP    C$CLP1
5911            041550  013701  043056                  MOV     T31BFR+6,R1     ;PICK UP XSTO
5912            041554  010102                  MOV     R1,R2                   ;SET UP EXPECTED
5913            041556  052702  000002                  BIS     #BIT1,R2        ;SET BOT BIT IN EXPECTED
5914            041562  020102                  CMP     R1,R2                   ;DOES EXP = REC'D
5915            041564  001406                  BEQ     40$                     ;BR, IF EQUAL (OK)
5916            041566  004737  020104                  JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
5920            041572                          ERRHRD  ERRNO,T31BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
                041572  104456                                                          TRAP    C$ERHRD
                041574  000460                                                          .WORD   304
                041576  044215                                                          .WORD   T31BOT
                041600  016350                                                          .WORD   EXPREC
5921            041602                          40$:    CKLOOP                  ;LOOP IF SELECTED
                041602  104406                                                          TRAP    C$CLP1
5922            041604  013737  003076  043162          MOV     FREE,T31WB      ;STARTING WRITE BUFFER ADDRESS
5923            041612  012737  140005  043160  65$:    MOV     #140005,T31PK3  ;WRITE DATA,CVC=1,ACK COMMAND
5924            041620  012704  043160                  MOV     #T31PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
5925            041624  012700  000144                  MOV     #100.,R0        ;SET PATTERN IN CORRECT REGISTER
5926            041630  004737  020376                  JSR     PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
5927            041634  012737  000144  043166          MOV     #100.,T31SZ     ;SET UP RECORD SIZE IN PACKET
5928            041642  010465  177776                  MOV     R4,TSDB(R5)     ;ISSUE COMMAND
5929            041646  004737  017124                  JSR     PC,WAITF        ;WAIT FOR SSR TO SET
5930            041652  016501  000000                  MOV     TSSR(R5),R1     ;GET TSSR CONTENTS
5931            041656  012702  000200                  MOV     #SSR,R2         ;SET UP EXPECTED
5932            041662  020102                  CMP     R1,R2                   ;ARE THEY EQUAL
5933            041664  001406                  BEQ     80$                     ;BR, IF OK
5934            041666  004737  020104                  JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
5938                                                                            ;SOFT ERROR, DON'T CARE ABOUT WRITE
5939                                                                            ;COMMAND'S RESULTS - CHECKING
5940                                                                            ;NO-OP COMMAND
5941            041672                          ERRSOFT ERRNO,T31WDC,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
                041672  104457                                                          TRAP    C$ERSOFT
                041674  000461                                                          .WORD   305
                041676  045100                                                          .WORD   T31WDC
                041700  011700                                                          .WORD   PKTSSR
5942            041702                          80$:    CKLOOP                  ;LOOP IF SELECTED
                041702  104406                                                          TRAP    C$CLP1
5943            041704  004737  010434                  JSR     PC,REWIND       ;CALL TAPE REWIND COMMAND
5944            041710  103407                  BCS     230$                    ;BR, IF NO PROBLEM
5945            041712  010001                  MOV     R0,R1                   ;SAVE TSSR
5946            041714  004737  020104                  JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
5950            041720                          ERRHRD  ERRNO,T31RWN,EXPREC     ;REWIND NOT ACCEPTED
                041720  104456                                                          TRAP    C$ERHRD
```

```
                041722  000462                                                          .WORD   306
                041724  044544                                                          .WORD   T31RWN
                041726  016350                                                          .WORD   EXPREC
5951            041730                          230$:   CKLOOP                          ;LOOP IF SELECTED
                041730  104406                                                          TRAP    C$CLP1
5952            041732  013701  043056                  MOV     T31BFR+6,R1             ;PICK UP XSTO
5953            041736  010102                          MOV     R1,R2                   ;SET UP EXPECTED
5954            041740  052702  000002                  BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
5955            041744  020102                          CMP     R1,R2                   ;DOES EXP = REC'D
5956            041746  001406                          BEQ     240$                    ;BR, IF EQUAL (OK)
5957            041750  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5961            041754                          ERRHRD  ERRNO,T31BOT,EXPREC             ;TAPE NOT AT BOT AFTER REWIND
                041754  104456                                                          TRAP    C$ERHRD
                041756  000463                                                          .WORD   307
                041760  044215                                                          .WORD   T31BOT
                041762  016350                                                          .WORD   EXPREC
5962            041764                          240$:   CKLOOP                          ;LOOP IF SELECTED
                041764  104406                                                          TRAP    C$CLP1
5963            041766  012737  041012  043160  265$:   MOV     #041012,T31PK3          ;NO-OP,CVC=1 COMMAND
5964            041774  012704  043160                  MOV     #T31PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5965            042000  010337  043166                  MOV     R3,T31SZ                ;SET UP RECORD SIZE IN PACKET
5966            042004  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5967            042010  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5968            042014  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5969            042020  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
5970            042024  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
5971            042026  001406                          BEQ     280$                    ;BR, IF OK
5972            042030  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5976            042034                          ERRHRD  ERRNO,T31RDF,PKTSSR             ;TSSR INCORRECT AFTER READ DATA
                042034  104456                                                          TRAP    C$ERHRD
                042036  000464                                                          .WORD   308
                042040  043413                                                          .WORD   T31RDF
                042042  011700                                                          .WORD   PKTSSR
5977            042044                          280$:   CKLOOP                          ;LOOP IF SELECTED
                042044  104406                                                          TRAP    C$CLP1
5978            042046  013701  043056                  MOV     T31BFR+6,R1             ;PICK UP XSTO
5979            042052  010102                          MOV     R1,R2                   ;SET UP EXPECTED
5980            042054  052702  000002                  BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
5981            042060  020102                          CMP     R1,R2                   ;DOES EXP = REC'D
5982            042062  001406                          BEQ     285$                    ;BR, IF EQUAL (OK)
5983            042064  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5987            042070                          ERRHRD  ERRNO,T31BOT,EXPREC             ;TAPE NOT AT BOT AFTER REWIND
                042070  104456                                                          TRAP    C$ERHRD
                042072  000465                                                          .WORD   309
                042074  044215                                                          .WORD   T31BOT
                042076  016350                                                          .WORD   EXPREC
5988            042100                          285$:   CKLOOP                          ;LOOP IF SELECTED
                042100  104406                                                          TRAP    C$CLP1
5989            042102  012737  140001  043160          MOV     #140001,T31PK3          ;READ,ACK,CVC=1 COMMAND
5990            042110  012704  043160                  MOV     #T31PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5991            042114  012737  000144  043166          MOV     #100.,T31SZ             ;SET UP RECORD SIZE IN PACKET
5992            042122  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5993            042126  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5994            042132  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5995            042136  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
5996            042142  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
5997            042144  001406                          BEQ     290$                    ;BR, IF OK
```

```
 5998 042146  004737  020104              JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
 6002 042152                              ERRHRD  ERRNO,T31RDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      042152  104456                                                  TRAP    C$ERHRD
      042154  000466                                                  .WORD   310
      042156  043214                                                  .WORD   T31RDE
      042160  011700                                                  .WORD   PKTSSR
 6003 042162                        290$:  CKLOOP                     ;LOOP IF SELECTED
      042162  104406                                                  TRAP    C$CLP1
 6004 042164  017701  140706              MOV     @FREE,R1           ;GET DATA READ
 6005 042170  012702  000144              MOV     #100.,R2           ;READ EXPECTED
 6006 042174  020102                      CMP     R1,R2              ;DID TAPE STAY POSITIONED
 6007 042176  001406                      BEQ     330$               ;BR, IF EXPD = RECD
 6008 042200  004737  020104              JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
 6012 042204                              ERRHRD  ERRNO,T31WNG,EXPREC ;TAPE DATA NOT CORRECT
      042204  104456                                                  TRAP    C$ERHRD
      042206  000467                                                  .WORD   311
      042210  043341                                                  .WORD   T31WNG
      042212  016350                                                  .WORD   EXPREC
 6013 042214                        330$:
 6014 042214                              ENDSUB                     ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
      042214                                                        L10051:
      042214  104403                                                  TRAP    C$ESUB
```

```
                                ;+
                                ;
                                ;TEST 3, SUBTEST 2
                                ;
                                ;
                                ;       VERIFIES THAT THE INITIALIZE COMMAND OPERATES AS A NO-OP,
                                ;       ASSUMING NO MICRODIAGNOSTIC ERRORS ARE PRESENT (THEY WOULD
                                ;       ALREADY HAVE BEEN DETECTED IN PREVIOUS TESTS).  THE TEST
                                ;       SEQUENCE IS SIMILAR TO THAT USED IN SUBTEST 1.
                                ;
                                ;
                                ;
                                ;-
042216                          BGNSUB                  ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
042216                                                                  T3.2:
042216  104402                                                          TRAP    C$BSUB
042220  004737  046460          JSR     PC,T31REST      ;SET COMMAND PACKET
042224  004737  046552          JSR     PC,T31RT2       ;SET UP OTHER COMMAND PACKET
042230  004737  046614          JSR     PC,T31RT3       ;SET UP OTHER COMMAND PACKET
042234  004737  016650          JSR     PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
042240  103407                  BCS     20$             ;BR IF INIT WAS OK
042242  004737  020104          JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
042246  010001                  MOV     R0,R1           ;CONTENTS OF TSSR REGISTER
042250                          ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
042250  104455                                                          TRAP    C$ERDF
042252  000470                                                          .WORD   312
042254  003554                                                          .WORD   SFIERR
042256  011666                                                          .WORD   SFIMSG
042260                  20$:
042260  012704  043030          MOV     #T31PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
042264  004737  010332          JSR     PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
042270  103407                  BCS     23$             ;BR, IF COMMAND ISSUED OK
042272  004737  020104          JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
042276  010001                  MOV     R0,R1           ;SAVE CONTENTS OF TSSR
042300                          ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
042300  104456                                                          TRAP    C$ERHRD
042302  000471                                                          .WORD   313
042304  004760                                                          .WORD   WRTMSG
042306  011666                                                          .WORD   SFIMSG
042310                  23$:    CKLOOP                  ;LOOP IF SELECTED
042310  104406                                                          TRAP    C$CLP1
042312  004737  010434          JSR     PC,REWIND       ;CALL TAPE REWIND COMMAND
042316  103407                  BCS     30$             ;BR, IF NO PROBLEM
042320  010004                  MOV     R0,R4           ;SET UP REWIND PACKET ADDRESS
042322  004737  020104          JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
042326                          ERRHRD  ERRNO,T31RWN,PKTSSR     ;REWIND NOT ACCEPTED
042326  104456                                                          TRAP    C$ERHRD
042330  000472                                                          .WORD   314
042332  044544                                                          .WORD   T31RWN
042334  011700                                                          .WORD   PKTSSR
042336                  30$:    CKLOOP                  ;LOOP IF SELECTED
042336  104406                                                          TRAP    C$CLP1
042340  013701  043056          MOV     T31BFR+6,R1     ;PICK UP XSTO
042344  010102                  MOV     R1,R2           ;SET UP EXPECTED
042346  052702  000002          BIS     #BIT1,R2        ;SET BOT BIT IN EXPECTED
042352  020102                  CMP     R1,R2           ;DOES EXP = REC'D
042354  001406                  BEQ     40$             ;BR, IF EQUAL (OK)
```

```
6066 042356  004737  020104                    JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6070 042362                                    ERRHRD  ERRNO,T31BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
     042362  104456                                                        TRAP    C$ERHRD
     042364  000473                                                        .WORD   315
     042366  044215                                                        .WORD   T31BOT
     042370  016350                                                        .WORD   EXPREC
6071 042372                            40$:    CKLOOP                     ;LOOP IF SELECTED
     042372  104406                                                        TRAP    C$CLP1
6072 042374  013737  003076  043162             MOV     FREE,T31WB         ;STARTING WRITE BUFFER ADDRESS
6073 042402  012737  140005  043160  65$:       MOV     #140005,T31PK3     ;WRITE DATA,CVC=1,ACK COMMAND
6074 042410  012704  043160                    MOV     #T31PK3,R4         ;SET UP R4 WITH PACKET ADDRESS
6075 042414  012700  000144                    MOV     #100.,R0                ;SET PATTERN IN CORRECT REGISTER
6076 042420  004737  020376                    JSR     PC,FILLMEM         ;FILL MEMORY WITH RECORD SIZE
6077 042424  012737  000144  043166             MOV     #100.,T31SZ        ;SET UP RECORD SIZE IN PACKET
6078 042432  010465  177776                    MOV     R4,TSDB(R5)        ;ISSUE COMMAND
6079 042436  004737  017124                    JSR     PC,WAITF           ;WAIT FOR SSR TO SET
6080 042442  016501  000000                    MOV     TSSR(R5),R1        ;GET TSSR CONTENTS
6081 042446  012702  000200                    MOV     #SSR,R2            ;SET UP EXPECTED
6082 042452  020102                           CMP     R1,R2              ;ARE THEY EQUAL
6083 042454  001406                           BEQ     80$                ;BR, IF OK
6084 042456  004737  020104                    JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6088                                                                      ;SOFT ERROR, DON'T CARE ABOUT WRITE
6089                                                                      ;COMMAND'S RESULTS - CHECKING
6090                                                                      ;THE INITIALIZE COMMAND
6091 042462                                    ERRSOFT ERRNO,T31WDC,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
     042462  104457                                                        TRAP    C$ERSOFT
     042464  000474                                                        .WORD   316
     042466  045100                                                        .WORD   T31WDC
     042470  011700                                                        .WORD   PKTSSR
6092 042472                            80$:    CKLOOP                     ;LOOP IF SELECTED
     042472  104406                                                        TRAP    C$CLP1
6093 042474  004737  010434                    JSR     PC,REWIND          ;CALL TAPE REWIND COMMAND
6094 042500  103407                           BCS     230$               ;BR, IF NO PROBLEM
6095 042502  010001                           MOV     R0,R1              ;SAVE TSSR
6096 042504  004737  020104                    JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6100 042510                                    ERRHRD  ERRNO,T31RWN,EXPREC ;REWIND NOT ACCEPTED
     042510  104456                                                        TRAP    C$ERHRD
     042512  000475                                                        .WORD   317
     042514  044544                                                        .WORD   T31RWN
     042516  016350                                                        .WORD   EXPREC
6101 042520                            230$:   CKLOOP                     ;LOOP IF SELECTED
     042520  104406                                                        TRAP    C$CLP1
6102 042522  013701  043056                    MOV     T31BFR+6,R1        ;PICK UP XSTO
6103 042526  010102                           MOV     R1,R2              ;SET UP EXPECTED
6104 042530  052702  000002                    BIS     #BIT1,R2           ;SET BOT BIT IN EXPECTED
6105 042534  020102                           CMP     R1,R2              ;DOES EXP = REC'D
6106 042536  001406                           BEQ     240$               ;BR, IF EQUAL (OK)
6107 042540  004737  020104                    JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6111 042544                                    ERRHRD  ERRNO,T31BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
     042544  104456                                                        TRAP    C$ERHRD
     042546  000476                                                        .WORD   318
     042550  044215                                                        .WORD   T31BOT
     042552  016350                                                        .WORD   EXPREC
6112 042554                            240$:   CKLOOP                     ;LOOP IF SELECTED
     042554  104406                                                        TRAP    C$CLP1
6113 042556  012737  041012  043160  265$:       MOV     #041012,T31PK3     ;INITIALIZE,CVC=1 COMMAND
6114 042564  012704  043160                    MOV     #T31PK3,R4         ;SET UP R4 WITH PACKET ADDRESS
```

```
6115 042570  010337  043166                MOV     R3,T31SZ                ;SET UP RECORD SIZE IN PACKET
6116 042574  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6117 042600  004737  017124                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6118 042604  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6119 042610  012702  000200                MOV     #SSR,R2                 ;SET UP EXPECTED
6120 042614  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
6121 042616  001406                        BEQ     280$                    ;BR, IF OK
6122 042620  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6126 042624                                ERRHRD  ERRNO,T31RDF,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
     042624  104456                                                            TRAP    C$ERHRD
     042626  000477                                                            .WORD   319
     042630  043413                                                            .WORD   T31RDF
     042632  011700                                                            .WORD   PKTSSR
6127 042634                        280$:   CKLOOP                          ;LOOP IF SELECTED
     042634  104406                                                            TRAP    C$CLP1
6128 042636  013701  043056                MOV     T31BFR+6,R1             ;PICK UP XSTO
6129 042642  010102                        MOV     R1,R2                   ;SET UP EXPECTED
6130 042644  052702  000002                BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
6131 042650  020102                        CMP     R1,R2                   ;DOES EXP = REC'D
6132 042652  001406                        BEQ     285$                    ;BR, IF EQUAL (OK)
6133 042654  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6137 042660                                ERRHRD  ERRNO,T31BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     042660  104456                                                            TRAP    C$ERHRD
     042662  000500                                                            .WORD   320
     042664  044215                                                            .WORD   T31BOT
     042666  016350                                                            .WORD   EXPREC
6138 042670                        285$:   CKLOOP                          ;LOOP IF SELECTED
     042670  104406                                                            TRAP    C$CLP1
6139 042672  012737  140001  043160          MOV     #140001,T31PK3          ;READ,ACK,CVC=1 COMMAND
6140 042700  012704  043160                MOV     #T31PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6141 042704  012737  000144  043166          MOV     #100.,T31SZ             ;SET UP RECORD SIZE IN PACKET
6142 042712  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6143 042716  004737  017124                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6144 042722  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6145 042726  012702  000200                MOV     #SSR,R2                 ;SET UP EXPECTED
6146 042732  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
6147 042734  001406                        BEQ     290$                    ;BR, IF OK
6148 042736  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6152 042742                                ERRHRD  ERRNO,T31RDE,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
     042742  104456                                                            TRAP    C$ERHRD
     042744  000501                                                            .WORD   321
     042746  043214                                                            .WORD   T31RDE
     042750  011700                                                            .WORD   PKTSSR
6153 042752                        290$:   CKLOOP                          ;LOOP IF SELECTED
     042752  104406                                                            TRAP    C$CLP1
6154 042754  017701  140116                MOV     @FREE,R1                ;GET DATA READ
6155 042760  012702  000144                MOV     #100.,R2                ;READ EXPECTED
6156 042764  020102                        CMP     R1,R2                   ;DID TAPE STAY POSITIONED
6157 042766  001406                        BEQ     330$                    ;BR, IF EXPD = RECD
6158 042770  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6162 042774                                ERRHRD  ERRNO,T31WNH,EXPREC     ;TAPE POSITION NOT CORRECT AFTER INIT
     042774  104456                                                            TRAP    C$ERHRD
     042776  000502                                                            .WORD   322
     043000  043260                                                            .WORD   T31WNH
     043002  016350                                                            .WORD   EXPREC
6163 043004                        330$:
6164 043004                                ENDSUB                          ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
```

```
        043004
        043004  104403                                                  L10052:
                                                                                TRAP    C$ESUB
 6165                                   ;
 6166                                   ;
 6167                                   ;
 6168 043006  004737  017360                    JSR     PC,TSTLOOP      ;DO WE NEED TO ITERATE TEST
 6169 043012  103002                            BCC     163$            ;BR, IF NO LOOP REQUIRED
 6170 043014  000137  041362                    JMP     T31LOOP         ;EXECUTE AGAIN
 6171 043020                            163$:   EXIT    TST             ;ALL DONE THIS TEST
        043020  104432                                                          TRAP    C$EXIT
        043022  003614                                                          .WORD   L10050-.
```

```
                                ;+
                                ;LOCAL STORAGE FOR THIS TEST
                                ;-
043024                                  .BLKB   10-<.-TUV2A&7>
043030                          T31PACKET:                              ;COMMAND PACKET FOR TEST
043030  100004                          .WORD   100004                  ;WRITE CHARACTERISTICS COMMAND, WITH , ACK
043032  043040                          .WORD   T31DATA                 ;ADDRESS OF CHARACTERISTICS BLOCK
043034  000000                          .WORD   0
043036  000012                          .WORD   10.                     ;STARTING VALUE OF BLOCK SIZE
043040                          T31DATA:                                ;CHARACTERISTICS DATA BLOCK
043040  043050                          .WORD   T31BFR                  ;ADDRESS OF MESSAGE BUFFER
043042  000000                          .WORD   0
043044  000024                          .WORD   20.                     ;LENGTH OF MESSAGE BUFFER
043046  000000                          .WORD   0
043050                          T31BFR: .BLKW   25.                     ;MESSAGE BUFFER
                                ;
                                ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
                                ;
043132                                  .BLKB   10-<.-TUV2A&7>
043140                          T31PK2:
043140  100006                          .WORD   100006                  ;WRITE SUB SYS MEM COMMAND,  AND ACK
043142  043170                          .WORD   T31BF2                  ;ADDRESS OF SELECT BLOCK DATA
043144  000000                          .WORD   0
043146  000006                          .WORD   6.                      ;SIZE OF DATA PACKET

043150                                  .BLKB   10-<.-TUV2A&7>
043160                          T31PK3:
043160  100005                          .WORD   100005                  ;REREAD COMMAND, AND ACK
043162                          T31RB:
043162  003076                  T31WB:  .WORD   FREE                    ;ADDRESS OF WRITE BUFFER
043164  000000                          .WORD   0
043166  000000                  T31SZ:  .WORD   0                       ;SIZE OF BUFFER (EXTENT)
                                        .EVEN
                                ;
                                ;
043170                          T31BF2:
043170     010                  T31BS0: .BYTE   10                      ;BSEL0 AREA
043171     200                  T31BS1: .BYTE   200                     ;BSEL1 AREA
043172  000000                  T31S2:  .WORD   0                       ;SEL 2 AREA
043174  000000                  T31S3:  .WORD   0                       ;DATA AREA
                                ;
                                ;
                                        .EVEN
                                ;TAPE MOTION PACKET COMMAND VALUES

043176  100205                  T31RN:  .WORD   100205                  ;REREAD DATA (NEXT)
043200  100605                  T31WDR: .WORD   100605                  ;REREAD DATA RETRY
043202  102205                  T31CON: .WORD   102205                  ;WRITE CONTINOUS
043204  177777                          .WORD   177777                  ;END OF DATA

                                ;
043206  000000                  T31CNT: .WORD   0                       ;TAPE TIMER COUNTER STORAGE AREA
043210  000000                  T31CNU: .WORD   0                       ;TAPE TIMER COUNTER STORAGE AREA
043212  000000                  T31DLY: .WORD   0                       ;DELAY COUNTER
```

```
                                ;+
                                ;LOCAL TEXT MESSAGES FOR TEST
                                ;-


043214  124  123  123  T31RDE: .ASCIZ  'TSSR Not Correct After READ Command'
043260  124  141  160  T31WNH: .ASCIZ  'Tape Position Incorrect After INITIALIZE Command'
043341  124  141  160  T31WNG: .ASCIZ  'Tape Position Incorrect After NOP Command'
043413  124  123  123  T31RDF: .ASCIZ  'TSSR Incorrect After READ DATA Command'
043462  122  105  122  T31RRF: .ASCIZ  'REREAD Previous (Space Reverse, Read Forward) Command Failed'
043557  120  117  123  T31SC:  .ASCIZ  'POSITION (Space Command) Failed, TSSR Not Correct'
043641  122  111  102  T31LOR: .ASCIZ  'RIB NOT SET AFTER READ REVERSE INTO BOT'
043711  124  123  123  T31WDF: .ASCIZ  'TSSR Not Correct After Illegal Mode Bits Set'
043766  111  154  154  T31LOQ: .ASCIZ  'Illegal Mode Bits, Failed To Set ILC Bit In XSTO'
044047  122  105  122  T31SSR: .ASCIZ  'REREAD COMMAND Not Accepted'
044103  124  123  123  T31WDE: .ASCIZ  'TSSR Not Correct After NO-OP ("CLEAN TAPE") AND INITIALIZE Command,At BOT'
044215  124  141  160  T31BOT: .ASCIZ  'Tape Not At BOT After REWIND Command (BOT Not Set In XSTO)'
044310  116  117  055  T31TIM: .ASCIZ  'NO-OP ("CLEAN TAPE") AND INITIALIZE''S Erase Tape Not Long Enough'
044410  122  105  122  T31EOT: .ASCIZ  'REREAD DATA OVER EOT GAVE NO TAPE STATUS ALERT'
044467  124  123  123  T31TM:  .ASCIZ  'TSSR Not Correct After REREAD COMMAND Reject'
044544  122  145  167  T31RWN: .ASCIZ  'Rewind (POSITION) Command Not Accepted'
044613  122  101  115  T31RNC: .ASCIZ  'RAM Error, Correct Data Pattern Not In Ram'
044666  124  123  123  T31AM3: .ASCIZ  'TSSR Init. Failed After REREAD COMMAND'
044735  104  162  151  T31OFL: .ASCIZ  'Drive 7 Select Failed To Set "OFL" In TSSR'
045010  124  123  123  T31WDD: .ASCIZ  'TSSR Not Correct After REREAD DATA Command, SWB Bit Set'
045100  124  123  123  T31WDC: .ASCIZ  'TSSR Not Correct After REREAD DATA Command'
045153  103  126  103  T31VCK: .ASCIZ  'CVC Set, Didn''t Reset VCK In Message Buffer'
045226  124  123  102  T31BA:  .ASCIZ  'TSBA Not Correct After REREAD DATA Command'
045301  127  122  111  T31WSS: .ASCIZ  'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
045370  122  145  141  T31LON: .ASCIZ  'Reading Long Record Failed To Set RLL Bit In XSTO'
045452  122  145  141  T31LOP: .ASCIZ  'Reading Long Record Failed To Set RLS Bit In XSTO'
045534  122  145  163  T31PBP: .ASCIZ  'Residual Byte Count Incorrect After Short Record Read'
045622  122  145  141  T31TRL: .ASCIZ  'Reading Long Record  Failed To Give Tape Status Alert'
045710  116  117  055  T31NEF: .ASCIZ  'NO-OP ("CLEAN TAPE") AND INITIALIZE, At First Record, Failed To Set RIB Bit
046031  124  123  123  T31SCF: .ASCIZ  'TSSR Not Correct After SPACE RECORDS Command'
046106  124  123  123  T31TSA: .ASCIZ  'TSSR Not Correct After NO-OP ("CLEAN TAPE") AND INITIALIZE, Into BOT'
046213  124  123  123  T31WRF: .ASCIZ  'TSSR Not Correct After NO-OP ("CLEAN TAPE") AND INITIALIZE Command'
046316  104  141  164  T31DTA: .ASCIZ  'Data Compare Error, Data Read From Tape Not Equal To Written'
046413  116  117  055  TST31ID:       .ASCIZ  'NO-OP ("Clean Tape") And INITIALIZE'
                              .EVEN
                                ;+
                                ;
                                ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
                                ;WRITE SUBSYSTEM MEMORY COMMAND
                                ;
                                ;-

046460                          T31REST:
046460                                SAVREG                          ;SAVE THE REGISTERS
046464  012701  043030                MOV     #T31PACKET,R1           ;START OF THE PACKET
046470  012721  100004                MOV     #100004,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK.
046474  012721  043040                MOV     #T31DATA,(R1)+          ;ADDRESS OF CHARAISTICS DATA BLOCK
046500  005021                       CLR     (R1)+                   ;EXTENDED ADDRESS
046502  012721  000012                MOV     #10.,(R1)+              ;SIZE OF DATA BLOCK IN BYTES
046506  012721  043050                MOV     #T31BFR,(R1)+           ;ADDRESS OF MESSAGE BUFFER
```

```
6293 046512  005021                          CLR     (R1)+
6294 046514  012721  000024                  MOV     #20.,(R1)+          ;LENGTH OF MESSAGE BUFFER
6295 046520  005021                          CLR     (R1)+
6296 046522  012711  000000                  MOV     #0,(R1)             ;SELECT DRIVE ZERO
6297 046526  012702  000030                  MOV     #24.,R2             ;NUMBER OF LOCATIONS TO BE CLEARED
6298 046532  012762  177777  043050  64$:    MOV     #177777,T31BFR(R2)  ;ALL ONES TO MESSAGE BUFFER
6299 046540  005742                          TST     -(R2)               ;NEXT LOCATION
6300 046542  022702  000000                  CMP     #0,R2               ;AT END OF LOOP YET
6301 046546  001371                          BNE     64$                 ;KEEP GOING UNTIL DONE
6302 046550  000207                          RTS     PC                  ;RETURN
6303
6304
6305 046552                          T31RT2:
6306 046552                                  SAVREG                      ;SAVE THE REGISTERS
6307 046556  012701  043140                  MOV     #T31PK2,R1          ;START OF THE PACKET
6308 046562  012721  100006                  MOV     #100006,(R1)+       ;WRITE SUBSYSTEM MEM. WITH ACK,
6309 046566  012721  043170                  MOV     #T31BF2,(R1)+       ;ADDRESS OF DATA BLOCK
6310 046572  005021                          CLR     (R1)+               ;EXTENDED ADDRESS
6311 046574  012721  000006                  MOV     #6.,(R1)+           ;SIZE OF DATA BLOCK IN BYTES
6312 046600  005021                          CLR     (R1)+
6313 046602  012701  043170                  MOV     #T31BF2,R1          ;POINT TO DATA SEL AREA
6314 046606  005021                          CLR     (R1)+
6315 046610  005011                          CLR     (R1)
6316 046612  000207                          RTS     PC                  ;RETURN
6317 046614                          T31RT3:
6318 046614                                  SAVREG                      ;SAVE REGISTERS
6319 046620  012701  043160                  MOV     #T31PK3,R1          ;SET UP POINTER ADDRESS
6320 046624  005021                          CLR     (R1)+               ;COMMAND SPACE
6321 046626  005021                          CLR     (R1)+               ;ADDRESS OF DATA BLOCK
6322 046630  005021                          CLR     (R1)+               ;EXTENDED ADDRESS
6323 046632  005011                          CLR     (R1)                ;SIZE OF DATA TRANSFER BLOCK
6324 046634  000207                          RTS     PC                  ;RETURN
6325 046636                                  ENDTST
     046636
     046636  104401                                              L10050:
                                                                         TRAP    C$ETST
```

```
                                .SBTTL  TEST  4: Erase And Operation Incomplete
                        ;+
                        ;
                        ;       VERIFIES THAT AN ERASE COMMAND ISSUED WHEN THE TAPE IS
                        ;       POSITIONED AT BOT OPERATES PROPERLY AND ACTUALLY ERASES TAPE.
                        ;       THE FOLLOWING TEST SEQUENCE IS PERFORMED:
                        ;
                        ;
                        ;          1.  THE TAPE IS FIRST REWOUND, SEVERAL TEST RECORDS ARE
                        ;              WRITTEN, AND THE TAPE IS REWOUND AGAIN.
                        ;
                        ;          2.  AN ERASE COMMAND IS ISSUED, WHICH SHOULD ERASE A NUMBER
                        ;              OF THE TEST RECORDS.
                        ;
                        ;          3.  NORMAL TERMINATION IS VERIFIED AND STATUS IS CHECKED
                        ;              (BOT SHOULD BE 0).
                        ;
                        ;          4.  A READ REVERSE COMMAND IS ISSUED.  IT IS VERIFIED THAT
                        ;              THE COMMAND TERMINATES WITH TAPE STATUS ALERT, THAT THE
                        ;              REVERSE INTO BOT (RIB) STATUS BIT IS SET, AND THAT NO
                        ;              DATA IS TRANSFERRED.  THIS DEMONSTRATES THAT NO DATA
                        ;              WAS ENCOUNTERED IN THE AREA ERASED BY THE ERASE
                        ;              COMMAND.
                        ;
                        ;
                        ;
                        ;
                        ;
                        ;
                        ;THE TEST CONSISTS OF THE FOLLOWING 3 SUBTESTS
                        ;
                        ;
                        ;
                        ;-
                                BGNTST
046640                                                          T4::
046640  005037  002172          CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
046644  005037  003104          CLR     KTFLG           ;HOLD OFF KT11
046650  012737  005676  002150  MOV     #EPRT1,EPRTSW           ;PRIMARY ERROR MESSAGE
046656  012700  052470          MOV     #TST32ID,R0             ;ASCII MESSAGE TO IDENTIFY TEST
046662  004737  017412          JSR     PC,TSTSETUP             ;DO INITIAL TEST SETUP
046666  012737  000001  002166  MOV     #1,LOOPCNT              ;PERFORM 1 ITERATIONS
046674  005037  051340          CLR     T32CNT                  ;CLEAR TAPE RECORD COUNTER

                        ;+
                        ;       CHECK FOR 1ST PASS, IF 1ST PASS PRINT FAULT LIGHT MESSAGE
                        ;       ELSE SKIP MESSAGE
                        ;-
046700  005737  002722          TST     FLLTSW                  ;CHECK FAULT SWITCH
046704  001012                  BNE     5$                      ;BR, IF NOT 1ST PASS
046706  005237  002722          INC     FLLTSW                  ;IT IS 1ST PASS, SET SW FOR LATER
046712                          PRINTX  #FAULTM                 ;"THIS TEST MAY ILLUMINATE FAULT LIGHT"
046712  012746  052527                                                  MOV     #FAULTM,-(SP)
046716  012746  000001                                                  MOV     #1,-(SP)
046722  010600                                                          MOV     SP,R0
046724  104415                                                          TRAP    C$PNTX
```

```
      046726  062706  000004                                                  ADD     #4,SP
 6384 046732                          5$:
 6385
 6386
 6387                                 ;*
 6388                                 ;
 6389                                 ;TEST 4, SUBTEST 1
 6390                                 ;
 6391                                 ;
 6392                                 ;VERIFIES THAT A Erase And Operation Incomplete COMMAND ISSUED WHILE
 6393                                 ;THE TAPE IS POSITIONED AT BOT CAUSES FUNCTION REJECT
 6394                                 ;TERMINATION, WITH THE NON-EXECUTABLE FUNCTION (NEF)
 6395                                 ;ERROR BIT SET.
 6396                                 ;
 6397                                 ;
 6398                                 ;-
 6399
 6400 046732                          T32LOOP:
 6401                                 ;
 6402                                 ;
 6403                                 ;-
 6404 046732                                BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
      046732                                                                      T4.1:
      046732  104402                                                              TRAP    C$BSUB
 6405 046734  004737  052632                JSR     PC,T32REST              ;SET COMMAND PACKET
 6406 046740  004737  052724                JSR     PC,T32RT2               ;SET UP OTHER COMMAND PACKET
 6407 046744  004737  052754                JSR     PC,T32RT3               ;SET UP OTHER COMMAND PACKET
 6408 046750  012737  176750  051344        MOV     #65000.,T32DLY          ;SET UP DELAY COUNTER
 6409 046756  004737  016650          10$:  JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
 6410 046762  103426                       BCS     20$                     ;BR IF INIT WAS OK
 6411 046764                                DELAY   250                     ;DELAY ABOUT .25 SEC
      046764  012727  000250                                                  MOV     #250.,(PC)+
      046770  000000                                                          .WORD   0
      046772  013727  002116                                                  MOV     L$DLY,(PC)+
      046776  000000                                                          .WORD   0
      047000  005367  177772                                                  DEC     -6(PC)
      047004  001375                                                          BNE     .-4
      047006  005367  177756                                                  DEC     -22(PC)
      047012  001367                                                          BNE     .-20
 6412 047014  005337  051344                DEC     T32DLY                  ;BUMP COUNTER
 6413 047020  001356                       BNE     10$                     ;BR, IF COUNTER NOT DONE
 6414 047022  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 6418 047026  010001                       MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
 6419 047030                                ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      047030  104455                                                          TRAP    C$ERDF
      047032  000621                                                          .WORD   401
      047034  003554                                                          .WORD   SFIERR
      047036  011666                                                          .WORD   SFIMSG
 6420 047040                          20$:
 6421 047040  012704  051150                MOV     #T32PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
 6422 047044  004737  010332                JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
 6423 047050  103407                       BCS     25$                     ;BR, IF COMMAND ISSUED OK
 6424 047052  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 6428 047056  010001                       MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
 6429 047060                                ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      047060  104456                                                          TRAP    C$ERHRD
      047062  000622                                                          .WORD   402
```

```
                047064  004760                                                          .WORD   WRTMSG
                047066  011666                                                          .WORD   SFIMSG
6430            047070                          25$:    CKLOOP                          ;LOOP IF SELECTED
                047070  104406                                                          TRAP    C$CLP1
6431            047072  004737  010434                  JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
6432            047076  103411                          BCS     26$                     ;BR, IF NO PROBLEM
6433            047100  010004                          MOV     R0,R4                   ;SET UP REWIND PACKET ADDRESS
6434            047102  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6435            047106  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6439            047112                                  ERRHRD  ERRNO,T32RWN,PKTSSR     ;REWIND NOT ACCEPTED
                047112  104456                                                          TRAP    C$ERHRD
                047114  000623                                                          .WORD   403
                047116  051530                                                          .WORD   T32RWN
                047120  011700                                                          .WORD   PKTSSR
6440            047122                          26$:    CKLOOP                          ;LOOP IF SELECTED
                047122  104406                                                          TRAP    C$CLP1
6441            047124  012703  000400                  MOV     #256.,R3                ;STARTING RECORD SIZE
6442            047130  013737  003076  051302          MOV     FREE,T32WB              ;STARTING WRITE BUFFER ADDRESS
6443            047136  012737  140005  051300          MOV     #140005,T32PK3          ;WRITE DATA,CVC=1,ACK COMMAND
6444            047144  012704  051300                  MOV     #T32PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6445            047150  010337  051306          27$:    MOV     R3,T32SZ                ;SET UP RECORD SIZE IN PACKET
6446            047154  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6447            047160  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6448            047164  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6449            047170  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
6450            047174  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
6451            047176  001406                          BEQ     28$                     ;BR, IF OK
6452            047200  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6456                                                                                    ;SOFT ERROR, DON'T CARE ABOUT WRITE
6457                                                                                    ;COMMAND'S RESULTS - CHECKING THE
6458                                                                                    ;ERASE COMMAND
6459            047204                                  ERRSOFT ERRNO,T32WDC,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
                047204  104457                                                          TRAP    C$ERSOFT
                047206  000624                                                          .WORD   404
                047210  052366                                                          .WORD   T32WDC
                047212  011700                                                          .WORD   PKTSSR
6460            047214                          28$:    CKLOOP                          ;LOOP IF SELECTED
                047214  104406                                                          TRAP    C$CLP1
6461            047216  005723                          TST     (R3)+                   ;BUMP RECORD COUNTER
6462            047220  020327  001002                  CMP     R3,#514.                ;AT MAX SIZE YET
6463            047224  001351                          BNE     27$                     ;BR, IF NOT AT END OF LOOP
6464            047226  004737  010434                  JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
6465            047232  103411                          BCS     30$                     ;BR, IF NO PROBLEM
6466            047234  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6467            047240  010004                          MOV     R0,R4                   ;SET UP REWIND PACKET ADDRESS
6468            047242  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6472            047246                                  ERRHRD  ERRNO,T32RWN,PKTSSR     ;REWIND NOT ACCEPTED
                047246  104456                                                          TRAP    C$ERHRD
                047250  000625                                                          .WORD   405
                047252  051530                                                          .WORD   T32RWN
                047254  011700                                                          .WORD   PKTSSR
6473            047256                          30$:    CKLOOP                          ;LOOP IF SELECTED
                047256  104406                                                          TRAP    C$CLP1
6474            047260  013701  051176                  MOV     T32BFR+6,R1             ;PICK UP XSTO
6475            047264  010102                          MOV     R1,R2                   ;SET UP EXPECTED
6476            047266  052702  000002                  BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
6477            047272  020102                          CMP     R1,R2                   ;DOES EXP = REC'D
```

```
6478 047274  001406                          BEQ     40$                     ;BR, IF EQUAL (OK)
6479 047276  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6483 047302                                  ERRHRD  ERRNO,T32BOE,EXPREC     ;TAPE AT BOT AFTER ERASE
     047302  104456                                                          TRAP    C$ERHRD
     047304  000626                                                          .WORD   406
     047306  052216                                                          .WORD   T32BOE
     047310  016350                                                          .WORD   EXPREC
6484 047312                          40$:    CKLOOP                          ;LOOP IF SELECTED
     047312  104406                                                          TRAP    C$CLP1
6485 047314  012737  140411  051300          MOV     #140411,T32PK3          ;ERASE TAPE,CVC=1,ACK COMMAND
6486 047322  012704  051300                  MOV     #T32PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6487 047326  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6488 047332  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6489 047336  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6490 047342  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
6491 047346  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
6492 047350  001406                          BEQ     50$                     ;BR, IF OK
6493 047352  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6497 047356                                  ERRHRD  ERRNO,T32ERA,PKTSSR     ;TSSR INCORRECT AFTER ERASE DATA
     047356  104456                                                          TRAP    C$ERHRD
     047360  000627                                                          .WORD   407
     047362  051646                                                          .WORD   T32ERA
     047364  011700                                                          .WORD   PKTSSR
6498 047366                          50$:    CKLOOP                          ;LOOP IF SELECTED
     047366  104406                                                          TRAP    C$CLP1
6499 047370  013701  051176                  MOV     T32BFR+6,R1             ;PICK UP XSTO
6500 047374  010102                          MOV     R1,R2                   ;SET UP EXPECTED
6501 047376  042702  000002                  BIC     #BIT1,R2                ;SET BOT BIT IN EXPECTED
6502 047402  020102                          CMP     R1,R2                   ;DOES EXP = REC'D
6503 047404  001406                          BEQ     55$                     ;BR, IF EQUAL (OK)
6504 047406  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6508 047412                                  ERRHRD  ERRNO,T32BOE,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     047412  104456                                                          TRAP    C$ERHRD
     047414  000630                                                          .WORD   408
     047416  052216                                                          .WORD   T32BOE
     047420  016350                                                          .WORD   EXPREC
6509 047422                          55$:    CKLOOP                          ;LOOP IF SELECTED
     047422  104406                                                          TRAP    C$CLP1
6510 047424  013737  003076  051302          MOV     FREE,T32RB              ;ADDRESS OF BUFFER
6511 047432  012737  140401  051300          MOV     #140401,T32PK3          ;READ REVERSE,ACK,CVC=1 COMMAND
6512 047440  012737  000400  051306          MOV     #256.,T32SZ             ;SET UP THE SIZE OF RECORD
6513 047446  012704  051300                  MOV     #T32PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6514 047452  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6515 047456  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6516 047462  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6517 047466  012702  100204                  MOV     #SSR!SC!BIT2,R2         ;SET UP EXPECTED TAPE STATUS ALERT
6518 047472  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
6519 047474  001406                          BEQ     180$                    ;BR, IF OK
6520 047476  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6524 047502                                  ERRHRD  ERRNO,T32TSA,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
     047502  104456                                                          TRAP    C$ERHRD
     047504  000631                                                          .WORD   409
     047506  052141                                                          .WORD   T32TSA
     047510  011700                                                          .WORD   PKTSSR
6525 047512                          180$:   CKLOOP                          ;LOOP IF SELECTED
     047512  104406                                                          TRAP    C$CLP1
6526 047514  013701  051204                  MOV     T32BFR+14,R1            ;GET XST3 STATUS WORD
```

TEST 4: Erase And Operation Incomplete SEQ 0173

```
 6527 047520  010102                        MOV     R1,R2                   ;SET UP EXPECTED
 6528 047522  052702  000001                BIS     #BIT0,R2                ;SET THE RIB BIT
 6529 047526  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
 6530 047530  001406                        BEQ     190$                    ;BR, IF EQUAL (GOOD)
 6531 047532  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 6535 047536                                ERRHRD  ERRNO,T32RIB,EXPREC     ;RIB SHOULD BE SET
      047536  104456                                                            TRAP    C$ERHRD
      047540  000632                                                            .WORD   410
      047542  051766                                                            .WORD   T32RIB
      047544  016350                                                            .WORD   EXPREC
 6536 047546                        190$:
 6537 047546                                ENDSUB                  ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
      047546                                                                L10054:
      047546  104403                                                            TRAP    C$ESUB
```

```
6539                                    ;*
6540                                    ;
6541                                    ;TEST 4, SUBTEST 2
6542                                    ;
6543                                    ;       VERIFIES THAT AN ERASE COMMAND EXECUTED WHEN THE TAPE IS NOT
6544                                    ;       POSITIONED AT BOT OPERATES PROPERLY AND DOES NOT CORRUPT
6545                                    ;       PREVIOUS TAPE RECORDS.  THE TEST SEQUENCE IS:
6546                                    ;
6547                                    ;            1.  THE TAPE IS FIRST REWOUND, SEVERAL TEST RECORDS ARE
6548                                    ;                WRITTEN, AND THE TAPE IS REWOUND AGAIN.
6549                                    ;
6550
6551                                    ;            2.  A SPACE RECORDS FORWARD COMMAND IS ISSUED TO MOVE THE
6552                                    ;                TAPE OFF OF BOT AND SKIP OVER THE FIRST SEVERAL
6553                                    ;                RECORDS.
6554                                    ;
6555                                    ;            3.  AN ERASE COMMAND IS ISSUED, WHICH SHOULD ERASE A NUMBER
6556                                    ;                OF THE TEST RECORDS.
6557                                    ;
6558                                    ;            4.  NORMAL TERMINATION IS VERIFIED AND STATUS IS CHECKED.
6559                                    ;
6560                                    ;            5.  A READ REVERSE COMMAND IS ISSUED.  IT IS VERIFIED THAT
6561                                    ;                NORMAL TERMINATION IS ACCOMPLISHED AND THAT THE DATA
6562                                    ;                TRANSFERRED CORRESPONDS TO THAT FOR THE EXPECTED
6563                                    ;                RECORD.  THIS DEMONSTRATES THAT NO DATA WAS ENCOUNTERED
6564                                    ;                IN THE AREA ERASED BY THE ERASE COMMAND, AND THAT THE
6565                                    ;                PREVIOUS RECORD WAS NOT CORRUPTED.
6566                                    ;
6567                                    ;
6568                                    ;
6569                                    ;
6570                                    ;
6571                                    ;
6572                                    ;-
6573 047550                                BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
      047550                                                                        T4.2:
      047550  104402                                                                TRAP    C$BSUB
6574 047552  004737  052632                JSR     PC,T32REST              ;SET COMMAND PACKET
6575 047556  004737  052724                JSR     PC,T32RT2               ;SET UP OTHER COMMAND PACKET
6576 047562  004737  052754                JSR     PC,T32RT3               ;SET UP OTHER COMMAND PACKET
6577 047566  004737  016650                JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
6578 047572  103407                       BCS     20$                     ;BR IF INIT WAS OK
6579 047574  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6583 047600  010001                       MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
6584 047602                                ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      047602  104455                                                                TRAP    C$ERDF
      047604  000633                                                                .WORD   411
      047606  003554                                                                .WORD   SFIERR
      047610  011666                                                                .WORD   SFIMSG
6585 047612                             20$:
6586 047612  012704  051150                MOV     #T32PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
6587 047616  004737  010332                JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
6588 047622  103407                       BCS     23$                     ;BR, IF COMMAND ISSUED OK
6589 047624  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6593 047630  010001                       MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
6594 047632                                ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      047632  104456                                                                TRAP    C$ERHRD
```

```
                047634  000634                                                  .WORD   412
                047636  004760                                                  .WORD   WRTMSG
                047640  011666                                                  .WORD   SFIMSG
6595            047642                  23$:    CKLOOP                          ;LOOP IF SELECTED
                047642  104406                                                  TRAP    C$CLP1
6596            047644  004737  010434          JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
6597            047650  103407          BCS     30$                             ;BR, IF NO PROBLEM
6598            047652  010004          MOV     R0,R4                           ;SET UP REWIND PACKET ADDRESS
6599            047654  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6603            047660                  ERRHRD  ERRNO,T32RWN,PKTSSR             ;REWIND NOT ACCEPTED
                047660  104456                                                  TRAP    C$ERHRD
                047662  000635                                                  .WORD   413
                047664  051530                                                  .WORD   T32RWN
                047666  011700                                                  .WORD   PKTSSR
6604            047670                  30$:    CKLOOP                          ;LOOP IF SELECTED
                047670  104406                                                  TRAP    C$CLP1
6605            047672  013701  051176          MOV     T32BFR+6,R1             ;PICK UP XSTO
6606            047676  010102          MOV     R1,R2                           ;SET UP EXPECTED
6607            047700  052702  000002          BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
6608            047704  020102          CMP     R1,R2                           ;DOES EXP = REC'D
6609            047706  001406          BEQ     40$                             ;BR, IF EQUAL (OK)
6610            047710  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6614            047714                  ERRHRD  ERRNO,T32BOT,EXPREC             ;TAPE NOT AT BOT AFTER REWIND
                047714  104456                                                  TRAP    C$ERHRD
                047716  000636                                                  .WORD   414
                047720  051346                                                  .WORD   T32BOT
                047722  016350                                                  .WORD   EXPREC
6615            047724                  40$:    CKLOOP                          ;LOOP IF SELECTED
                047724  104406                                                  TRAP    C$CLP1
6616            047726  012703  000144          MOV     #100.,R3                ;STARTING RECORD SIZE
6617            047732  010300          MOV     R3,R0                           ;SET UP MEMORY FILL
6618            047734  004737  020376          JSR     PC,FILLMEM              ;CALL MEMORY FILLER
6619            047740  013737  003076  051302  MOV     FREE,T32WB              ;STARTING WRITE BUFFER ADDRESS
6620            047746  012737  140005  051300  65$:    MOV     #140005,T32PK3  ;WRITE DATA,CVC=1,ACK COMMAND
6621            047754  012704  051300          MOV     #T32PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6622            047760  010300          MOV     R3,R0                           ;SET PATTERN IN CORRECT REGISTER
6623            047762  004737  020376          JSR     PC,FILLMEM              ;FILL MEMORY WITH RECORD SIZE
6624            047766  010337  051306          MOV     R3,T32SZ                ;SET UP RECORD SIZE IN PACKET
6625            047772  010465  177776          MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6626            047776  004737  017124          JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6627            050002  016501  000000          MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6628            050006  012702  000200          MOV     #SSR,R2                 ;SET UP EXPECTED
6629            050012  020102          CMP     R1,R2                           ;ARE THEY EQUAL
6630            050014  001406          BEQ     80$                             ;BR, IF OK
6631            050016  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6635                                                                            ;SOFT ERROR, DON'T CARE ABOUT WRITE
6636                                                                            ;COMMAND'S RESULTS - CHECKING THE
6637                                                                            ;ERASE COMMAND
6638            050022                  ERRSOFT ERRNO,T32WDC,PKTSSR             ;TSSR INCORRECT AFTER WRITE DATA
                050022  104457                                                  TRAP    C$ERSOFT
                050024  000637                                                  .WORD   415
                050026  052366                                                  .WORD   T32WDC
                050030  011700                                                  .WORD   PKTSSR
6639            050032                  80$:    CKLOOP                          ;LOOP IF SELECTED
                050032  104406                                                  TRAP    C$CLP1
6640            050034  005723          TST     (R3)+                           ;BUMP RECORD SIZE COUNTER
6641            050036  020327  000156          CMP     R3,#110.                ;AT 160 SIZE YET
```

```
6642 050042  001341                        BNE     65$                ;BR, IF MORE RECORDS TO WRITE
6643 050044  004737  010434                JSR     PC,REWIND          ;CALL TAPE REWIND COMMAND
6644 050050  103407                        BCS     230$               ;BR, IF NO PROBLEM
6645 050052  010001                        MOV     R0,R1              ;SAVE TSSR
6646 050054  004737  020104                JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6650 050060                                ERRHRD  ERRNO,T32RWN,EXPREC ;REWIND NOT ACCEPTED
     050060  104456                                                    TRAP    C$ERHRD
     050062  000640                                                    .WORD   416
     050064  051530                                                    .WORD   T32RWN
     050066  016350                                                    .WORD   EXPREC
6651 050070                        230$:   CKLOOP                     ;LOOP IF SELECTED
     050070  104406                                                    TRAP    C$CLP1
6652 050072  013701  051176                MOV     T32BFR+6,R1        ;PICK UP XSTO
6653 050076  010102                        MOV     R1,R2              ;SET UP EXPECTED
6654 050100  052702  000002                BIS     #BIT1,R2           ;SET BOT BIT IN EXPECTED
6655 050104  020102                        CMP     R1,R2              ;DOES EXP = REC'D
6656 050106  001406                        BEQ     240$               ;BR, IF EQUAL (OK)
6657 050110  004737  020104                JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6661 050114                                ERRHRD  ERRNO,T32BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
     050114  104456                                                    TRAP    C$ERHRD
     050116  000641                                                    .WORD   417
     050120  051346                                                    .WORD   T32BOT
     050122  016350                                                    .WORD   EXPREC
6662 050124                        240$:   CKLOOP                     ;LOOP IF SELECTED
     050124  104406                                                    TRAP    C$CLP1
6663 050126  012703  000001                MOV     #1,R3              ;SET UP FOR SPACE COMMAND
6664 050132  004737  010134                JSR     PC,SPACE           ;ISSUE SPACE COMMAND 1 FORWARD
6665 050136  012737  140411  051300  265$: MOV     #140411,T32PK3     ;ERASE DATA,ACK COMMAND
6666 050144  012704  051300                MOV     #T32PK3,R4         ;SET UP R4 WITH PACKET ADDRESS
6667 050150  010465  177776                MOV     R4,TSDB(R5)        ;ISSUE COMMAND
6668 050154  004737  017124                JSR     PC,WAITF           ;WAIT FOR SSR TO SET
6669 050160  016501  000000                MOV     TSSR(R5),R1        ;GET TSSR CONTENTS
6670 050164  012702  000200                MOV     #SSR,R2            ;SET UP EXPECTED
6671 050170  020102                        CMP     R1,R2              ;ARE THEY EQUAL
6672 050172  001406                        BEQ     280$               ;BR, IF OK
6673 050174  004737  020104                JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6677 050200                                ERRHRD  ERRNO,T32ERA,PKTSSR ;TSSR INCORRECT AFTER READ DATA
     050200  104456                                                    TRAP    C$ERHRD
     050202  000642                                                    .WORD   418
     050204  051646                                                    .WORD   T32ERA
     050206  011700                                                    .WORD   PKTSSR
6678 050210                        280$:   CKLOOP                     ;LOOP IF SELECTED
     050210  104406                                                    TRAP    C$CLP1
6679 050212  013737  003076  051302        MOV     FREE,T32RB         ;ADDRESS OF BUFFER
6680 050220  012737  140401  051300        MOV     #140401,T32PK3     ;READ REVERSE,ACK,CVC=1 COMMAND
6681 050226  012737  000144  051306        MOV     #100.,T32SZ        ;SET UP THE SIZE OF RECORD
6682 050234  012704  051300                MOV     #T32PK3,R4         ;SET UP R4 WITH PACKET ADDRESS
6683 050240  010465  177776                MOV     R4,TSDB(R5)        ;ISSUE COMMAND
6684 050244  004737  017124                JSR     PC,WAITF           ;WAIT FOR SSR TO SET
6685 050250  016501  000000                MOV     TSSR(R5),R1        ;GET TSSR CONTENTS
6686 050254  012702  000200                MOV     #SSR,R2            ;SET UP EXPECTED TAPE STATUS ALERT
6687 050260  020102                        CMP     R1,R2              ;ARE THEY EQUAL
6688 050262  001406                        BEQ     290$               ;BR, IF OK
6689 050264  004737  020104                JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6693 050270                                ERRHRD  ERRNO,T32TSA,PKTSSR ;TSSR INCORRECT AFTER READ DATA
     050270  104456                                                    TRAP    C$ERHRD
     050272  000643                                                    .WORD   419
```

```
      050274  052141                                                    .WORD   T32TSA
      050276  011700                                                    .WORD   PKTSSR
6694  050300                        290$:   CKLOOP                      ;LOOP IF SELECTED
      050300  104406                                                    TRAP    C$CLP1
6695  050302  017701  132570              MOV     @FREE,R1              ;GET DATA READ
6696  050306  012702  000144              MOV     #100.,R2              ;SHOULD BE 100
6697  050312  020102                      CMP     R1,R2                 ;CHECK'EM OUT
6698  050314  001406                      BEQ     300$                  ;BR, IF OK
6699  050316  004737  020104              JSR     PC,FATCHK             ;INC AND CHECK FOR MORE THAN 25 ERRORS
6703  050322                              ERRHRD  ERRNO,T32ECF,EXPREC   ;ERASE COMMAND DIDN'T WORK
      050322  104456                                                    TRAP    C$ERHRD
      050324  000644                                                    .WORD   420
      050326  052305                                                    .WORD   T32ECF
      050330  016350                                                    .WORD   EXPREC
6704  050332                        300$:   CKLOOP                      ;LOOP IF SELECTED
      050332  104406                                                    TRAP    C$CLP1
6705  050334                        330$:
6706  050334                              ENDSUB                        ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
      050334                                                          L10055:
      050334  104403                                                    TRAP    C$ESUB
```

```
                        ;+
                        ;
                        ;TEST 4, SUBTEST 3
                        ;
                        ;       VERIFIES THAT THE OTHER TAPE
                        ;       MOTION COMMANDS EXECUTED WHEN THE TAPE IS BLANK
                        ;       RESULT IN UNRECOVERABLE ERROR TERMINATION AND OPERATION
                        ;       INCOMPLETE STATUS.  THE FOLLOWING TEST SEQUENCE IS EXECUTED:
                        ;
                        ;            1.  THE TAPE IS REWOUND.
                        ;
                        ;            2.  ERASE COMMANDS ARE ISSUED SUCH THAT THE TAPE IS POSITIONED
                        ;                ABOUT IN THE MIDDLE OF THE FIRST TRACK.
                        ;
                        ;
                        ;
                        ;            3.  IT IS VERIFIED THAT EACH OF THE FOLLOWING COMMANDS
                        ;                (ISSUED IN THE ORDER GIVEN) RESULTS IN UNRECOVERABLE
                        ;                ERROR TERMINATION WITH OPI=1:  SPACE RECORDS REVERSE
                        ;                SKIP TAPE MARKS REVERSE READ REVERSE REREAD PREVIOUS
                        ;                (OPP=0) REREAD PREVIOUS (OPP=1) REREAD NEXT (OPP=1)
                        ;                REREAD NEXT (OPP=0) READ NEXT SKIP TAPE MARKS REVERSE
                        ;                SKIP TAPE MARKS FORWARD SPACE RECORDS FORWARD WRITE
                        ;                DATA RETRY
                        ;
                        ;
                        ;
                        ;
                        ;-
050336                          BGNSUB                  ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
                                                                        T4.3:
050336  104402                                                          TRAP    C$BSUB
050340  004737  052724        9$:     JSR     PC,T32RT2       ;SET UP OTHER COMMAND PACKET
050344  004737  052632                JSR     PC,T32REST      ;SET COMMAND PACKET
050350  004737  052754                JSR     PC,T32RT3       ;SET UP OTHER COMMAND PACKET
050354  012737  176750  051344        MOV     #65000.,T32DLY  ;SET UP DELAY COUNTER
050362  004737  016650        10$:    JSR     PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
050366  103426                BCS     20$             ;BR IF INIT WAS OK
050370                        DELAY   250             ;DELAY ABOUT .25 SEC
050370  012727  000250                                          MOV     #250,(PC)+
050374  000000                                                  .WORD   0
050376  013727  002116                                          MOV     L$DLY,(PC)+
050402  000000                                                  .WORD   0
050404  005367  177772                                          DEC     -6(PC)
050410  001375                                                  BNE     .-4
050412  005367  177756                                          DEC     -22(PC)
050416  001367                                                  BNE     .-20
050420  005337  051344                DEC     T32DLY          ;BUMP COUNTER
050424  001356                BNE     10$             ;BR, IF COUNTER NOT DONE
050426  004737  020104                JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
050432  010001                MOV     R0,R1           ;CONTENTS OF TSSR REGISTER
050434                        ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
050434  104455                                                  TRAP    C$ERDF
050436  000645                                                  .WORD   421
050440  003554                                                  .WORD   SFIERR
050442  011666                                                  .WORD   SFIMSG
050444                  20$:
```

```
6754
6755 050444  012704  051150              MOV     #T32PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
6756 050450  004737  010332              JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
6757 050454  103407                      BCS     23$                     ;BR, IF COMMAND ISSUED OK
6758 050456  004737  020104              JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6762 050462  010001                      MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
6763 050464                              ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     050464  104456                                                      TRAP    C$ERHRD
     050466  000646                                                      .WORD   422
     050470  004760                                                      .WORD   WRTMSG
     050472  011666                                                      .WORD   SFIMSG
6764 050474                      23$:    CKLOOP                          ;LOOP IF SELECTED
     050474  104406                                                      TRAP    C$CLP1
6765 050476  004737  010434              JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
6766 050502  103411                      BCS     30$                     ;BR, IF NO PROBLEM
6767 050504  016501  000000              MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6768 050510  010004                      MOV     R0,R4                   ;GET PACKET ADDRESS
6769 050512  004737  020104              JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6773 050516                              ERRHRD  ERRNO,T32RWN,PKTSSR     ;REWIND NOT ACCEPTED
     050516  104456                                                      TRAP    C$ERHRD
     050520  000647                                                      .WORD   423
     050522  051530                                                      .WORD   T32RWN
     050524  011700                                                      .WORD   PKTSSR
6774 050526                      30$:    CKLOOP                          ;LOOP IF SELECTED
     050526  104406                                                      TRAP    C$CLP1
6775 050530  013701  051176              MOV     T32BFR+6,R1             ;PICK UP XSTO
6776 050534  010102                      MOV     R1,R2                   ;SET UP EXPECTED
6777 050536  052702  000002              BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
6778 050542  020102                      CMP     R1,R2                   ;DOES EXP = REC'D
6779 050544  001406                      BEQ     40$                     ;BR, IF EQUAL (OK)
6780 050546  004737  020104              JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6784 050552                              ERRHRD  ERRNO,T32BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     050552  104456                                                      TRAP    C$ERHRD
     050554  000650                                                      .WORD   424
     050556  051346                                                      .WORD   T32BOT
     050560  016350                                                      .WORD   EXPREC
6785 050562                      40$:    CKLOOP                          ;LOOP IF SELECTED
     050562  104406                                                      TRAP    C$CLP1
6786 050564  012703  000454              MOV     #300.,R3                ;# OF ERASES SO TAPE IS HALF 1ST TRACK
6787
6788 050570  012737  140411  051300  65$:    MOV     #140411,T32PK3          ;ERASE DATA,CVC=1,ACK COMMAND
6789 050576  012704  051300              MOV     #T32PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6790 050602  010465  177776              MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6791 050606  004737  017124              JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6792 050612  016501  000000              MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6793 050616  012702  000200              MOV     #SSR,R2                 ;SET UP EXPECTED
6794 050622  020102                      CMP     R1,R2                   ;ARE THEY EQUAL
6795 050624  001407                      BEQ     70$                     ;BR, IF OK
6796 050626  010102                      MOV     R1,R2                   ;SAVE ORIG TSSR
6797 050630  004737  020104              JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6801 050634                              ERRHRD  ERRNO,T32WDC,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     050634  104456                                                      TRAP    C$ERHRD
     050636  000651                                                      .WORD   425
     050640  052366                                                      .WORD   T32WDC
     050642  011700                                                      .WORD   PKTSSR
6802 050644  162703  000001      70$:    SUB     #1,R3                   ;BUMP DOWN TO NEXT VALUE
6803 050650  001401                      BEQ     80$                     ;BR, IF 300 ERASES WRITTEN
```

```
6804 050652  000746                    BR      65$                ;KEEP GOING
6805 050654                    80$:    CKLOOP                     ;LOOP IF SELECTED
     050654  104406                                                       TRAP    C$CLP1
6806 050656  012703  051310            MOV     #T32CMD,R3         ;STARTING RECORD SIZE
6807 050662  013737  003076  051302    MOV     FREE,T32RB         ;STARTING READ BUFFER ADDRESS
6808 050670  011337  051300    265$:   MOV     (R3),T32PK3        ;READ DATA,ACK COMMAND
6809 050674  012704  051300            MOV     #T32PK3,R4         ;SET UP R4 WITH PACKET ADDRESS
6810 050700  012700  177777            MOV     #177777,R0         ;SET PATTERN IN CORRECT REGISTER
6811 050704  004737  020376            JSR     PC,FILLMEM         ;FILL MEMORY WITH ALL ONES
6812 050710  012737  000144  051306    MOV     #100.,T32SZ        ;SET UP RECORD SIZE IN PACKET
6813 050716  010465  177776            MOV     R4,TSDB(R5)        ;ISSUE COMMAND
6814 050722  012737  000012  051344    MOV     #10.,T32DLY        ;SET UP DELAY COUNTER
6815 050730  004737  017124    270$:   JSR     PC,WAITF           ;WAIT FOR SSR TO SET
6816 050734  016501  000000            MOV     TSSR(R5),R1        ;GET TSSR CONTENTS
6817 050740  012702  100214            MOV     #SSR!SC!BIT2!BIT3,R2 ;SET UP EXPECTED
6818 050744  020102                    CMP     R1,R2              ;ARE THEY EQUAL
6819 050746  001425                    BEQ     280$               ;BR, IF OK
6820 050750                            DELAY   250                ;DELAY FOR SSR TO BE SET
     050750  012727  000250                                                MOV     #250,(PC)+
     050754  000000                                                .WORD   0
     050756  013727  002116                                                MOV     L$DLY,(PC)+
     050762  000000                                                .WORD   0
     050764  005367  177772                                                DEC     -6(PC)
     050770  001375                                                BNE     .-4
     050772  005367  177756                                                DEC     -22(PC)
     050776  001367                                                BNE     .-20
6821 051000  005337  051344            DEC     T32DLY             ;COUNT DELAY ROUTINE DOWN
6822 051004  001351                    BNE     270$               ;BR, IF DELAY HAS NOT ENDED
6823 051006  004737  020104            JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6827 051012                            ERRHRD  ERRNO,T32ECF,PKTSSR ;TSSR INCORRECT AFTER READ DATA
     051012  104456                                                       TRAP    C$ERHRD
     051014  000652                                                .WORD   426
     051016  052305                                                .WORD   T32ECF
     051020  011700                                                .WORD   PKTSSR
6828 051022                    280$:   CKLOOP                     ;LOOP IF SELECTED
     051022  104406                                                       TRAP    C$CLP1
6829 051024  013701  051204            MOV     T32BFR+14,R1       ;PICK UP XST3
6830 051030  010102                    MOV     R1,R2              ;SET UP EXPECTED
6831 051032  052702  000100            BIS     #BIT6,R2           ;SET OPI BIT IN EXPECTED
6832 051036  020102                    CMP     R1,R2              ;IS OPI BIT SET
6833 051040  001406                    BEQ     290$               ;BR, IF BIT IS SET
6834 051042  004737  020104            JSR     PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
6838 051046                            ERRHRD  ERRNO,T320PI,EXPREC ;OPI BIT NOT SET
     051046  104456                                                       TRAP    C$ERHRD
     051050  000653                                                .WORD   427
     051052  052433                                                .WORD   T320PI
     051054  016350                                                .WORD   EXPREC
6839 051056                    290$:   CKLOOP                     ;LOOP IF SELECTED
     051056  104406                                                       TRAP    C$CLP1
6840 051060  005723                    TST     (R3)+              ;BUMP COMMAND POINTER
6841 051062  021327  177777            CMP     (R3),#177777       ;AT END OF TABLE YET
6842 051066  001300                    BNE     265$               ;BR, KEEP TRYING COMMANDS
6843                           ;
6844                           ;       REWIND THE DRIVE
6845                           ;
6846 051070  004737  010434            JSR     PC,REWIND          ;CALL TAPE REWIND COMMAND
6847 051074  103411                    BCS     226$               ;BR, IF NO PROBLEM
```

```
 6848 051076  010004                        MOV     R0,R4                   ;SET UP REWIND PACKET ADDRESS
 6849 051100  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
 6850 051104  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 6854 051110                                ERRHRD  ERRNO,T32RWN,PKTSSR     ;REWIND NOT ACCEPTED
      051110  104456                                                            TRAP    C$ERHRD
      051112  000654                                                            .WORD   428
      051114  051530                                                            .WORD   T32RWN
      051116  011700                                                            .WORD   PKTSSR
 6855 051120                        226$:   CKLOOP                          ;LOOP IF SELECTED
      051120  104406                                                            TRAP    C$CLP1
 6856
 6857
 6858
 6859 051122                        291$:   ENDSUB                          ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
      051122                                                                  L10056:
      051122  104403                                                            TRAP    C$ESUB
 6860                               ;
 6861                               ;
 6862                               ;
 6863 051124  004737  017360                JSR     PC,TSTLOOP              ;DO WE NEED TO ITERATE TEST
 6864 051130  103002                        BCC     163$                    ;BR, IF NO LOOP REQUIRED
 6865 051132  000137  046732                JMP     T32LOOP                 ;EXECUTE AGAIN
 6866 051136                        163$:   EXIT    TST                     ;ALL DONE THIS TEST
      051136  104432                                                            TRAP    C$EXIT
      051140  001636                                                            .WORD   L10053-.
```

```
 6868                                   ;+
 6869                                   ;LOCAL STORAGE FOR THIS TEST
 6870                                   ;-
 6872 051142                                    .BLKB   10-<.-TUV2A&7>
 6874 051150                            T32PACKET:                      ;COMMAND PACKET FOR TEST
 6875 051150  100004                            .WORD   100004          ;WRITE CHARACTERISTICS COMMAND, WITH , ACK
 6876 051152  051160                            .WORD   T32DATA         ;ADDRESS OF CHARACTERISTICS BLOCK
 6877 051154  000000                            .WORD   0
 6878 051156  000012                            .WORD   10.             ;STARTING VALUE OF BLOCK SIZE
 6879 051160                            T32DATA:                        ;CHARACTERISTICS DATA BLOCK
 6880 051160  051170                            .WORD   T32BFR          ;ADDRESS OF MESSAGE BUFFER
 6881 051162  000000                            .WORD   0
 6882 051164  000024                            .WORD   20.             ;LENGTH OF MESSAGE BUFFER
 6883 051166  000000                            .WORD   0
 6884 051170                            T32BFR: .BLKW   25.             ;MESSAGE BUFFER
 6885                                   ;
 6886                                   ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
 6887                                   ;
 6889 051252                                    .BLKB   10-<.-TUV2A&7>
 6891 051260                            T32PK2:
 6892 051260  100006                            .WORD   100006          ;WRITE SUB SYS MEM COMMAND,  AND ACK
 6893 051262  000000                            .WORD   0               ;ADDRESS OF SELECT BLOCK DATA
 6894 051264  000000                            .WORD   0
 6895 051266  000006                            .WORD   6.              ;SIZE OF DATA PACKET
 6897 051270                                    .BLKB   10-<.-TUV2A&7>
 6899 051300                            T32PK3:
 6900 051300  100005                            .WORD   100005          ;REREAD COMMAND, AND ACK
 6901 051302                            T32RB:
 6902 051302  003076                    T32WB:  .WORD   FREE            ;ADDRESS OF WRITE BUFFER
 6903 051304  000000                            .WORD   0
 6904 051306  000000                    T32SZ:  .WORD   0               ;SIZE OF BUFFER (EXTENT)
 6905                                           .EVEN
 6906                                   ;TAPE MOTION PACKET COMMAND VALUES
 6907
 6908 051310                            T32CMD:
 6909 051310  140410                            .WORD   140410          ;SPACE RECORDS REVERSE
 6910 051312  141410                            .WORD   141410          ;SKIP TAPE MARKS REVERSE
 6911 051314  140401                            .WORD   140401          ;READ REVERSE
 6912 051316  141001                            .WORD   141001          ;REREAD PREVIOUS (OPP=0)
 6913 051320  161401                            .WORD   161401          ;REREAD NEXT (OPP=1)
 6914 051322  161001                            .WORD   161001          ;REREAD PREVIOUS (OPP=1)
 6915 051324  141401                            .WORD   141401          ;REREAD NEXT (OPP=0)
 6916 051326  140001                            .WORD   140001          ;READ NEXT
 6917 051330  141410                            .WORD   141410          ;SKIP TAPE MARKS REVERSE
 6918 051332  141010                            .WORD   141010          ;SKIP RECORDS FORWARD
 6919 051334  141005                            .WORD   141005          ;WRITE DATA RETRY
 6920 051336  177777                            .WORD   177777          ;END OF DATA
 6921                                   ;
 6922 051340  000000                    T32CNT: .WORD   0               ;TAPE TIMER COUNTER STORAGE AREA
 6923 051342  000000                    T32CNU: .WORD   0               ;TAPE TIMER COUNTER STORAGE AREA
 6924 051344  000000                    T32DLY: .WORD   0               ;DELAY COUNTER
```

```
                                        ;*
                                        ;LOCAL TEXT MESSAGES FOR TEST
                                        ;-

051346  124  141  160  T32BOT: .ASCIZ  'Tape Not At BOT After REWIND Command (BOT Not Set In XST0)'
051441  124  141  160  T32EOT: .ASCIZ  'Tape Status Alert During Erase To EOT, But EOT Not Set'
051530  122  145  167  T32RWN: .ASCIZ  'Rewind (POSITION) Command Not Accepted'
051577  124  123  123  T32AM3: .ASCIZ  'TSSR Init. Failed After REREAD COMMAND'
051646  124  123  123  T32ERA: .ASCIZ  'TSSR Not Correct After ERASE Command'
051713  124  123  102  T32BA:  .ASCIZ  'TSBA Not Correct After REREAD DATA Command'
051766  122  105  101  T32RIB: .ASCIZ  'READ REVERSE, After ERASE From BOT, Failed To Set RIB In XST3'
052064  124  123  123  T32SCF: .ASCIZ  'TSSR Not Correct After SPACE RECORDS Command'
052141  124  123  123  T32TSA: .ASCIZ  'TSSR Not Correct After READ REVERSE Into BOT'
052216  102  117  124  T32BOE: .ASCIZ 'BOT (XST0) Still Set After Erase From Tape''s BOT Marker'
052305  105  122  101  T32ECF: .ASCIZ  'ERASE Failed To Clear Tape (Erase) Tape Properly'

052366  124  123  123  T32WDC: .ASCIZ  'TSSR Not Correct After ERASE Command'
052433  117  120  111  T32OPI: .ASCIZ  'OPI Bit (XST3) Failed To Set'
052470  105  162  141  TST32ID:        .ASCIZ  'Erase And Operation Incomplete'
052527  045  116  045  FAULTM: .ASCIZ  '%N%A This Test May Illuminate The Drive Fault Light, Not An Error'
                             .EVEN
                                        ;*
                                        ;
                                        ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
                                        ;WRITE SUBSYSTEM MEMORY COMMAND
                                        ;
                                        ;-

052632                                  T32REST:
052632                                          SAVREG                  ;SAVE THE REGISTERS
052636  012701  051150                          MOV     #T32PACKET,R1   ;START OF THE PACKET
052642  012721  100004                          MOV     #100004,(R1)+   ;WRITE SUBSYSTEM MEM. WITH ACK,
052646  012721  051160                          MOV     #T32DATA,(R1)+  ;ADDRESS OF CHARAISTICS DATA BLOCK
052652  005021                                  CLR     (R1)+           ;EXTENDED ADDRESS
052654  012721  000012                          MOV     #10.,(R1)+      ;SIZE OF DATA BLOCK IN BYTES
052660  012721  051170                          MOV     #T32BFR,(R1)+   ;ADDRESS OF MESSAGE BUFFER
052664  005021                                  CLR     (R1)+
052666  012721  000024                          MOV     #20.,(R1)+      ;LENGTH OF MESSAGE BUFFER
052672  005021                                  CLR     (R1)+
052674  012711  000000                          MOV     #0,(R1)         ;SELECT DRIVE ZERO
052700  012702  000030                          MOV     #24.,R2         ;NUMBER OF LOCATIONS TO BE CLEARED
052704  012762  177777  051170  64$:   MOV     #177777,T32BFR(R2)      ;ALL ONES TO MESSAGE BUFFER
052712  005742                                  TST     -(R2)           ;NEXT LOCATION
052714  022702  000000                          CMP     #0,R2           ;AT END OF LOOP YET
052720  001371                                  BNE     64$             ;KEEP GOING UNTIL DONE
052722  000207                                  RTS     PC              ;RETURN

052724                                  T32RT2:
052724                                          SAVREG                  ;SAVE THE REGISTERS
052730  012701  051260                          MOV     #T32PK2,R1      ;START OF THE PACKET
052734  012721  100006                          MOV     #100006,(R1)+   ;WRITE SUBSYSTEM MEM. WITH ACK,
052740  005021                                  CLR     (R1)+           ;ADDRESS OF DATA BLOCK
052742  005021                                  CLR     (R1)+           ;EXTENDED ADDRESS
052744  012721  000006                          MOV     #6.,(R1)+       ;SIZE OF DATA BLOCK IN BYTES
```

```
6983 052750  005021                    CLR     (R1)+
6984 052752  000207                    RTS     PC                      ;RETURN
6985 052754                    T32RT3:
6986 052754                            SAVREG                          ;SAVE REGISTERS
6987 052760  012701  051300            MOV     @T32PK3,R1              ;SET UP POINTER ADDRESS
6988 052764  005021                    CLR     (R1)+                   ;COMMAND SPACE
6989 052766  005021                    CLR     (R1)+                   ;ADDRESS OF DATA BLOCK
6990 052770  005021                    CLR     (R1)+                   ;EXTENDED ADDRESS
6991 052772  005011                    CLR     (R1)                    ;SIZE OF DATA TRANSFER BLOCK
6992 052774  000207                    RTS     PC                      ;RETURN
6993 052776                            ENDTST
     052776
     052776  104401                                            L10053:
                                                                        TRAP    C$ETST
```

```
                                        .SBTTL  TEST  5: OPERATIONS AT EOT
                                ;*
                                ;
                                ;THIS TEST VERIFIES PROPER OPERATION OF THE WRITE DATA RETRY
                                ;COMMAND (SPACE REVERSE, ERASE, WRITE DATA)
                                ;
                                ;
                                ;THE TEST CONSISTS OF THE FOLLOWING 1 SUBTEST
                                ;
                                ;
                                ;
                                ;-
053000                                  BGNTST
053000                                                                  T5::
053000  005037  002172                  CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
053004  005037  003104                  CLR     KTFLG           ;HOLD OFF KT11
053010  012737  005676  002150          MOV     #EPRT1,EPRTSW           ;PRIMARY ERROR MESSAGE
053016  012700  057224                  MOV     #TST34ID,R0             ;ASCII MESSAGE TO IDENTIFY TEST
053022  004737  017412                  JSR     PC,TSTSETUP             ;DO INITIAL TEST SETUP
053026  012737  000001  002166          MOV     #1,LOOPCNT              ;PERFORM 1 ITERATIONS
053034  005037  055562                  CLR     T34CNT                  ;CLEAR TAPE RECORD COUNTER
                                ;*
                                ;
                                ;TEST 5, SUBTEST 1
                                ;
                                ;
                                ;       THIS TEST VERIFIES THAT THE EOT STATUS IS HANDLED PROPERLY BY
                                ;       THE VARIOUS TAPE MOTION COMMANDS.  THE FOLLOWING TEST SEQUENCE
                                ;       IS PERFORMED:
                                ;
                                ;            1.  THE TAPE IS REWOUND.
                                ;
                                ;            2.  WRITE DATA COMMANDS ARE REPEATEDLY ISSUED UNTIL TAPE
                                ;                STATUS ALERT TERMINATION IS SEEN WITH EOT=1.  ERRORS
                                ;                OTHER THAN OCCASIONAL CORRECTABLE OR UNCORRECTABLE DATA
                                ;                ERRORS CAUSE A FATAL ERROR REPORT.  RECORDS WITH DATA
                                ;                ERRORS ARE RETRIED, SO THE TAPE ENDS UP WITH GOOD DATA.
                                ;
                                ;            3.  ANOTHER WRITE DATA COMMAND IS ISSUED, AND IT IS CHECKED
                                ;                THAT TAPE STATUS ALERT TERMINATION OCCURS, WITH EOT=1.
                                ;
                                ;            4.  A WRITE TAPE MARK COMMAND IS ISSUED, AND IT IS CHECKED
                                ;                THAT TAPE STATUS ALERT TERMINATION OCCURS, WITH EOT=1.
                                ;
                                ;            5.  A SKIP TAPE MARKS REVERSE COMMAND IS ISSUED, AND IT IS
                                ;                CHECKED THAT TAPE STATUS ALERT TERMINATION OCCURS, WITH
                                ;                EOT=1 AND TMK=1.
                                ;
                                ;            6.  A SPACE RECORDS REVERSE COMMAND, WITH A RECORD COUNT OF
                                ;                1, IS ISSUED, AND IT IS CHECKED THAT TAPE STATUS ALERT
                                ;                TERMINATION OCCURS, WITH EOT=1 AND TMK=1.
                                ;
                                ;            7.  A SPACE RECORDS REVERSE COMMAND, WITH A RECORD COUNT OF
                                ;                1, IS ISSUED, AND IT IS CHECKED THAT NORMAL TERMINATION
                                ;                OCCURS, WITH EOT=1.
                                ;
                                ;            8.  A SPACE RECORDS FORWARD COMMAND, WITH A RECORD COUNT OF
```

```
7057                                ;               1, IS ISSUED, AND IT IS CHECKED THAT NORMAL TERMINATION
7058                                ;               OCCURS, WITH EOT=1.
7059                                ;
7060                                ;           9.  A READ REVERSE COMMAND IS ISSUED, AND IT IS CHECKED
7061                                ;               THAT NORMAL TERMINATION OCCURS, WITH EOT=1.
7062                                ;
7063                                ;           10. A READ FORWARD COMMAND IS ISSUED, AND IT IS CHECKED
7064                                ;               THAT NORMAL TERMINATION OCCURS, WITH EOT=1.
7065                                ;
7066                                ;           11. A SPACE RECORDS REVERSE COMMAND, WITH A RECORD COUNT OF
7067                                ;               3, IS ISSUED, AND IT IS CHECKED THAT NORMAL TERMINATION
7068                                ;               OCCURS, WITH EOT=0.
7069                                ;
7070                                ;           12. A SPACE RECORDS FORWARD COMMAND, WITH A RECORD COUNT OF
7071                                ;               3, IS ISSUED, AND IT IS CHECKED THAT NORMAL TERMINATION
7072                                ;               OCCURS, WITH EOT=1.
7073                                ;
7074                                ;           13. A SKIP FILE MARKS REVERSE COMMAND IS ISSUED, WHICH
7075                                ;               SHOULD SKIP ALL THE WAY TO BOT, AND IT IS CHECKED THAT
7076                                ;               TAPE STATUS ALERT TERMINATION OCCURS, WITH EOT=0,
7077                                ;               BOT=1, AND RIB=1.
7078                                ;
7079                                ;
7080                                ;
7081                                ;
7082                                ;-
7083
7084 053040                        T34LOOP:
```

```
                                ;+
                                ;
                                ;TEST 5, SUBTEST 1
                                ;
                                ;VERIFIES THAT WRITING OVER THE END-OF-TAPE (EOT)
                                ;MARKER CAUSES TAPE STATUS ALERT TERMINATION. IF THE
                                ;TAPE TRANSPORT DOES NOT RECOGNIZE THE EOT MARKER, THE
                                ;TAPE WILL RUN OFF THE END OF THE REEL, CAUSING THE
                                ;PROGRAM SEQUENCE TO BE ABORTED WITH A FATAL ERROR
                                ;INDICATION. IN THIS CASE, CORRECTIVE MAINTENANCE
                                ;MUST BE PERFORMED USING THE TRANSPORT'S BUILT-IN
                                ;MAINTENANCE ROUTINE AVAILIABLE VIA THE FRONT PANEL.
                                ;IF THE CONTROLLER DOES NOT RECOGNIZE THE EOT, THE
                                ;TRANSPORT WILL FAULT BUT THE TAPE WILL NOT RUN OFF
                                ;THE END OF THE REEL BUT THE TRANSPORT MUST BE
                                ;MANUALLY PLACED BACK ON-LINE TO BE USABLE. THIS IS A
                                ;FATAL DIAGNOSTIC ERROR.
                                ;
                                ;
                                ;
                                ;
                                ;-

053040                                  BGNSUB                  ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
053040                                                                  T5.1:
053040  104402                                                                  TRAP    C$BSUB

                                ;
                                ;
                                ;
                                ;
                                ;*************************************************************************
                                ;
                                ;       SPECIAL SOFTWARE PARAMETER CHECK TO REDUCE THE MTTR OF THE
                                ;       DRIVE I.E. SKIPS THIS SUBTEST IF SELECTED
                                ;
                                ;*************************************************************************
                                ;
                                ;
                                ;
053042  005737  002140                  TST     EOTSEL                  ;CHECK SWITCH FOR PARAMETER
053046  001402                          BEQ     5$                      ;BR, IF OPTION NOT SELECTED
053050  000137  055372                  JMP     600$                    ;BR, NO EOT TEST SELECTED
053054                          5$:
                                ;
                                ;
                                ;
                                ;*************************************************************************
                                ;
                                ;       SET UP THE PACKET BUFFERS AND WAIT LONG TIME FOR SSR
                                ;       THEN ISSUE A SOFT INITIALIZE
                                ;
                                ;*************************************************************************
                                ;
053054  004737  057726                  JSR     PC,T34REST              ;SET COMMAND PACKET
053060  004737  060164                  JSR     PC,T34RT3               ;RESTORE PACKET
053064  004737  060122                  JSR     PC,T34RT2               ;SET UP OTHER COMMAND PACKET
053070  012737  176750  055564          MOV     #65000.,T34DLY          ;SET UP COUNTER
```

```
7141 053076  004737  016650          10$:    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
7142 053102  103433                          BCS     20$                     ;BR IF INIT WAS OK
7143 053104                                  DELAY   250                     ;DELAY A WHILE
     053104  012727  000250                                                          MOV     #250,(PC)+
     053110  000000                                                                  .WORD   0
     053112  013727  002116                                                          MOV     L$DLY,(PC)+
     053116  000000                                                                  .WORD   0
     053120  005367  177772                                                          DEC     -6(PC)
     053124  001375                                                                  BNE     .-4
     053126  005367  177756                                                          DEC     -22(PC)
     053132  001367                                                                  BNE     .-20
7144 053134  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR STATUS
7145 053140  032701  000200                  BIT     #SSR,R1                 ;CHECK FOR SSR SET
7146 053144  001012                          BNE     20$                     ;BR, WHEN SSR IS SET
7147 053146  005337  055564                  DEC     T34DLY                  ;BUMP COUNTER DOWN
7148 053152  001351                          BNE     10$                     ;BR, IF MORE DELAY REQUIRED
7149 053154  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7153 053160  010001                          MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
7154 053162                                  ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
     053162  104455                                                                  TRAP    C$ERDF
     053164  000765                                                                  .WORD   501
     053166  003554                                                                  .WORD   SFIERR
     053170  011666                                                                  .WORD   SFIMSG
7155 053172                          20$:    CKLOOP                          ;LOOP IF SELECTED
     053172  104406                                                                  TRAP    C$CLP1
7156
7157
7158                                 ;
7159                                 ;*************************************************************************
7160                                 ;
7161                                 ;       ISSUE A WRITE CHARACTERISTICS COMMAND TO CONTROLLER
7162                                 ;
7163                                 ;*************************************************************************
7164                                 ;
7165 053174  012704  055420                  MOV     #T34PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
7166 053200  004737  010332                  JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
7167 053204  103407                          BCS     30$                     ;BR, IF COMMAND ISSUED OK
7168 053206  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7172 053212  010001                          MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
7173 053214                                  ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     053214  104456                                                                  TRAP    C$ERHRD
     053216  000766                                                                  .WORD   502
     053220  004760                                                                  .WORD   WRTMSG
     053222  011666                                                                  .WORD   SFIMSG
7174 053224                          30$:    CKLOOP                          ;LOOP IF SELECTED
     053224  104406                                                                  TRAP    C$CLP1
7175                                 ;
7176                                 ;*************************************************************************
7177                                 ;
7178                                 ;       ISSUE A REWIND COMMAND
7179                                 ;
7180                                 ;*************************************************************************
7181                                 ;
7182 053226  004737  010434                  JSR     PC,REWIND               ;REWIND CALL
7183 053232  103411                          BCS     35$                     ;BR, IF TSSR IS OK (GOOD)
7184 053234  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR
7185 053240  010004                          MOV     R0,R4                   ;SET UP PACKET
```

```
7186 053242  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7190 053246                            ERRHRD  ERRNO,T34RWN,PKTSSR     ;TSSR IS INCORRECT AFTER REWIND
     053246  104456                                                    TRAP    C$ERHRD
     053250  000767                                                    .WORD   503
     053252  057246                                                    .WORD   T34RWN
     053254  011700                                                    .WORD   PKTSSR
7191 053256                     35$:   CKLOOP                          ;LOOP IF SELECTED
     053256  104406                                                    TRAP    C$CLP1
7192                            ;
7193                            ;*************************************************************************
7194                            ;
7195                            ;       ISSUE A WRITE COMMAND, CHECK FOR ERRORS, THIS IS SO THAT THE
7196                            ;       DRIVE WILL NOT JUST HANG IF AN ERROR OCCURS.
7197                            ;
7198                            ;*************************************************************************
7199                            ;
7200 053260  012737  140005  055550        MOV     #140005,T34PK3          ;WRITE DATA, ACK, CVC=1
7201 053266  013737  003076  055552        MOV     FREE,T34WB              ;SET UP WRITE BUFFER ADDRESS
7202 053274  012737  066540  055556        MOV     #28000.,T34SZ           ;SET UP BUFFER SIZE (INC # OF BYTES)
7203 053302  012704  055550               MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7204 053306  010465  177776        36$:   MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7205 053312  004737  017124               JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7206 053316  016501  000000               MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7207 053322  012702  000200               MOV     #SSR,R2                 ;SET UP EXPECTED
7208 053326  020102                       CMP     R1,R2                   ;ARE THEY EQUAL
7209 053330  001407                       BEQ     39$                     ;BR, IF ALL IS WELL NO PROBLEMS
7210 053332  004737  020104               JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7214 053336                               ERRSOFT ERRNO,WRTERR,PKTSSR     ;TSSR INCORRECT AFTER WRITE TAPE
     053336  104457                                                    TRAP    C$ERSOFT
     053340  000770                                                    .WORD   504
     053342  005015                                                    .WORD   WRTERR
     053344  011700                                                    .WORD   PKTSSR
7215 053346  000757                       BR      36$                     ;BR, TO DO MORE CONTROLLED WRITES
7216 053350                     39$:   CKLOOP                          ;LOOP ON ERROR IF SELECTED
     053350  104406                                                    TRAP    C$CLP1
7217
7218
7219                            ;
7220                            ;*************************************************************************
7221                            ;
7222                            ;       ISSUE A WRITE COMMAND, KEEP GOING UNTIL TAPE STATUS ALERT
7223                            ;
7224                            ;*************************************************************************
7225                            ;
7226 053352  012737  140005  055550        MOV     #140005,T34PK3          ;WRITE DATA, ACK, CVC=1
7227 053360  012703  176750               MOV     #65000.,R3              ;SET MAX NUMBER OF WRITES
7228 053364  013737  003076  055552        MOV     FREE,T34WB              ;SET UP WRITE BUFFER ADDRESS
7229 053372  012737  066540  055556        MOV     #28000.,T34SZ           ;SET UP BUFFER SIZE (INC # OF BYTES)
7230 053400  012704  055550               MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7231 053404  010465  177776        40$:   MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7232 053410  004737  017124               JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7233 053414  016501  000000               MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7234 053420  012702  000200               MOV     #SSR,R2                 ;SET UP EXPECTED
7235 053424  020102                       CMP     R1,R2                   ;ARE THEY EQUAL
7236 053426  001010                       BNE     50$                     ;BR, IT MIGHT BE END OF TAPE
7237 053430  005303                       DEC     R3                      ;DEC RECORD COUNTER
7238 053432  001364                       BNE     40$                     ;BR, IF MORE TO GO
```

```
7239 053434  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7243 053440                          ERRDF   ERRNO,T34ET,PKTSSR      ;EOT NOT FOUND (USE SHORTER TAPE?)
     053440  104455                                                          TRAP    C$ERDF
     053442  000771                                                          .WORD   505
     053444  057057                                                          .WORD   T34ET
     053446  011700                                                          .WORD   PKTSSR
7244                                 ;
7245                                 ;*************************************************************************
7246                                 ;
7247                                 ;       HAVE TAPE STATUS ALERT, NOW CHECK FOR EOT. IF NEITHER KEEP GOING
7248                                 ;
7249                                 ;*************************************************************************
7250                                 ;
7251 053450                          50$:
7252 053450  022701  100210                 CMP     #100210,R1              ;CHECK FOR UNCORRECTABLE ERROR
7253 053454  001003                         BNE     55$                     ;BR, IF IT WASN'T UNCORR.
7254 053456  004737  060020                 JSR     PC,EWCHK                ;CHECK FOR EARLY WARNING
7255 053462  103750                         BCS     40$                     ;BR, IF EARLY WARNING FOUND
7256 053464  032701  000004          55$:   BIT     #BIT2,R1                ;CHECK FOR TAPE STATUS ALERT
7257 053470  001001                         BNE     60$                     ;BR, IF SET
7258 053472  000744                         BR      40$                     ;KEEP GOING
7259 053474  013701  055446          60$:   MOV     T34BFR+6,R1             ;PICK UP XSTO
7260 053500  010102                         MOV     R1,R2                   ;SET UP EXPECTED
7261 053502  052702  000001                 BIS     #BIT0,R2                ;SET THE EOT BIT ON IN EXPECTED
7262 053506  020102                         CMP     R1,R2                   ;WAS THE BIT ON
7263 053510  001402                         BEQ     80$                     ;BR, IF EOT WAS FOUND
7264 053512  000137  053404                 JMP     40$                     ;KEEP LOOKING
7265 053516                          80$:   CKLOOP                          ;LOOP IF SELECTED
     053516  104406                                                                 TRAP    C$CLP1
7266                                 ;
7267                                 ;*************************************************************************
7268                                 ;
7269                                 ;       ISSUE ONE MORE WRITE AFTER EOT DETECTED
7270                                 ;
7271                                 ;*************************************************************************
7272                                 ;
7273 053520  012737  140005  055550         MOV     #140005,T34PK3          ;WRITE DATA, ACK, CVC=1
7274 053526  013737  003076  055552         MOV     FREE,T34WB              ;SET UP WRITE BUFFER ADDRESS
7275 053534  012737  066540  055556         MOV     #28000.,T34SZ           ;SET UP BUFFER SIZE (INC # OF BYTES)
7276 053542  012704  055550                 MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7277 053546  010465  177776                 MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7278 053552  004737  017124                 JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7279 053556  016501  000000                 MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7280 053562  012702  100204                 MOV     #SC!SSR!BIT2,R2         ;SET UP EXPECTED
7281 053566  020102                         CMP     R1,R2                   ;ARE THEY EQUAL
7282 053570  001406                         BEQ     90$                     ;BR, IF THEY ARE OK
7283 053572  004737  020104                 JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7287 053576                                 ERRHRD  ERRNO,T34ET2,PKTSSR     ;WRITE TAPE AT EOT FAILED TO SET TSA
     053576  104456                                                                 TRAP    C$ERHRD
     053600  000772                                                                 .WORD   506
     053602  056430                                                                 .WORD   T34ET2
     053604  011700                                                                 .WORD   PKTSSR
7288 053606                          90$:   CKLOOP                          ;LOOP IF SELECTED
     053606  104406                                                                 TRAP    C$CLP1
7289                                 ;
7290                                 ;*************************************************************************
7291                                 ;
```

```
7292                                    ;       CHECK TO BE SURE EOT IS STILL SET
7293                                    ;
7294                                    ;*******************************************************************************
7295                                    ;
7296 053610  013701  055446                    MOV     T34BFR+6,R1             ;PICK UP XSTO
7297 053614  010102                            MOV     R1,R2                   ;SET UP EXPECTED
7298 053616  052702  000001                    BIS     #BIT0,R2                ;SET THE EOT BIT ON IN EXPECTED
7299 053622  020102                            CMP     R1,R2                   ;WAS THE BIT ON
7300 053624  001406                            BEQ     100$                    ;BR, IF EOT WAS FOUND
7301 053626  004737  020104                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7305 053632                                    ERRHRD  ERRNO,T34ETN,EXPREC     ;EOT BIT (XSTO) NOT SET
     053632  104456                                                            TRAP    C$ERHRD
     053634  000773                                                            .WORD   507
     053636  056516                                                            .WORD   T34ETN
     053640  016350                                                            .WORD   EXPREC
7306 053642                            100$:   CKLOOP                          ;LOOP IF SELECTED
     053642  104406                                                            TRAP    C$CLP1
7307                                    ;
7308                                    ;*******************************************************************************
7309                                    ;
7310                                    ;       NOW ISSUE A WRITE TAPE MARK, STILL BEYOND EOT
7311                                    ;
7312                                    ;*******************************************************************************
7313                                    ;
7314 053644  012737  140011  055550            MOV     #140011,T34PK3          ;WRITE TAPE MARK, ACK, CVC=1 COMMAND
7315 053652  012704  055550                    MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7316 053656  010465  177776                    MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7317 053662  004737  017124                    JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7318 053666  016501  000000                    MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7319 053672  012702  100204                    MOV     #SC!SSR!BIT2,R2         ;SET UP EXPECTED
7320 053676  020102                            CMP     R1,R2                   ;ARE THEY EQUAL
7321 053700  001406                            BEQ     110$                    ;BR, IF STATUS IS GOOD (OK)
7322 053702  004737  020104                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7326 053706                                    ERRHRD  ERRNO,T34WTM,PKTSSR     ;WRITE TAPE MARK FAILED
     053706  104456                                                            TRAP    C$ERHRD
     053710  000774                                                            .WORD   508
     053712  056341                                                            .WORD   T34WTM
     053714  011700                                                            .WORD   PKTSSR
7327 053716                            110$:   CKLOOP                          ;LOOP IF SELECTED
     053716  104406                                                            TRAP    C$CLP1
7328                                    ;
7329                                    ;*******************************************************************************
7330                                    ;
7331                                    ;       NOW CHECK TO BE SURE EOT IS STILL SET
7332                                    ;
7333                                    ;*******************************************************************************
7334                                    ;
7335 053720  013701  055446                    MOV     T34BFR+6,R1             ;PICK UP XSTO
7336 053724  010102                            MOV     R1,R2                   ;SET UP EXPECTED
7337 053726  052702  000001                    BIS     #BIT0,R2                ;SET THE EOT BIT ON IN EXPECTED
7338 053732  020102                            CMP     R1,R2                   ;WAS THE BIT ON
7339 053734  001406                            BEQ     120$                    ;BR, IF EOT WAS FOUND
7340 053736  004737  020104                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7344 053742                                    ERRHRD  ERRNO,T34ET0,EXPREC     ;EOT BIT (XSTO) NOT SET
     053742  104456                                                            TRAP    C$ERHRD
     053744  000775                                                            .WORD   509
     053746  055764                                                            .WORD   T34ET0
```

```
     053750  016350                                                          .WORD   EXPREC
7345 053752                  120$:  CKLOOP                  ;LOOP IF SELECTED
     053752  104406                                                          TRAP    C$CLP1
7346                         ;
7347                         ;*************************************************************************
7348                         ;
7349                         ;       NOW ISSUE A SKIP TAPE MARK REVERSE RIGHT BACK INTO THE JUST WRITTEN TM
7350                         ;
7351                         ;*************************************************************************
7352                         ;
7353 053754  012737  141410  055550        MOV     #141410,T34PK3          ;SKIP TAPE MARK REVERSE ACK,CVC=1 COMMAND
7354 053762  012737  000001  055552        MOV     #1,T34WB                ;SET NUMBER (1) OF TMS TO SKIP
7355 053770  012704  055550                MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7356 053774  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7357 054000  004737  017124                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7358 054004  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7359 054010  012702  000200                MOV     #SSR,R2                 ;SET UP EXPECTED
7360 054014  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
7361 054016  001406                        BEQ     130$                    ;BR, IF STATUS IS GOOD (OK)
7362 054020  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7366 054024                                ERRHRD  ERRNO,T34STM,PKTSSR     ;SKIP TAPE MARK REVERSE FAILED
     054024  104456                                                          TRAP    C$ERHRD
     054026  000776                                                          .WORD   510
     054030  057325                                                          .WORD   T34STM
     054032  011700                                                          .WORD   PKTSSR
7367 054034                  130$:  CKLOOP                  ;LOOP IF SELECTED
     054034  104406                                                          TRAP    C$CLP1
7368                         ;
7369                         ;*************************************************************************
7370                         ;
7371                         ;       EOT SHOULD STILL BE SET
7372                         ;
7373                         ;*************************************************************************
7374                         ;
7375 054036  013701  055446                MOV     T34BFR+6,R1             ;PICK UP XSTO
7376 054042  010102                        MOV     R1,R2                   ;SET UP EXPECTED
7377 054044  052702  000001                BIS     #BIT0,R2                ;SET THE EOT BIT ON IN EXPECTED
7378 054050  020102                        CMP     R1,R2                   ;WAS THE BIT ON
7379 054052  001406                        BEQ     140$                    ;BR, IF EOT WAS FOUND
7380 054054  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7384 054060                                ERRHRD  ERRNO,T34STE,EXPREC     ;EOT BIT (XSTO) NOT SET
     054060  104456                                                          TRAP    C$ERHRD
     054062  000777                                                          .WORD   511
     054064  057421                                                          .WORD   T34STE
     054066  016350                                                          .WORD   EXPREC
7385 054070                  140$:  CKLOOP                  ;LOOP IF SELECTED
     054070  104406                                                          TRAP    C$CLP1
7386                         ;
7387                         ;*************************************************************************
7388                         ;
7389                         ;       THE TMK BIT SHOULD BE SET ALSO
7390                         ;
7391                         ;*************************************************************************
7392                         ;
7393 054072  013701  055446                MOV     T34BFR+6,R1             ;PICK UP XSTO
7394 054076  010102                        MOV     R1,R2                   ;SET UP EXPECTED
7395 054100  052702  100000                BIS     #BIT15,R2               ;SET THE TMK BIT ON IN EXPECTED
```

```
7396 054104  020102                    CMP     R1,R2                   ;WAS THE BIT ON
7397 054106  001406                    BEQ     150$                    ;BR, IF TMK WAS FOUND
7398 054110  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7402 054114                            ERRHRD  ERRNO,T34TMK,EXPREC     ;TMK (XSTO) NOT SET
     054114  104456                                                            TRAP    C$ERHRD
     054116  001000                                                            .WORD   512
     054120  056751                                                            .WORD   T34TMK
     054122  016350                                                            .WORD   EXPREC
7403 054124                    150$:   CKLOOP                          ;LOOP IF SELECTED
     054124  104406                                                            TRAP    C$CLP1
7404                                   ;
7405                                   ;*************************************************************************
7406                                   ;
7407                                   ;       ISSUE SPACE RECORDS REVERSE FOR 1 RECORD, STILL BEYOND EOT
7408                                   ;
7409                                   ;*************************************************************************
7410                                   ;
7411 054126  012737  140410  055550            MOV     #140410,T34PK3          ;SPACE RECORDS REVERSE, ACK, CVC=1 CMD
7412 054134  012737  000001  055552            MOV     #1,T34WB                ;SPACE ONE RECORD REVERSE
7413 054142  012704  055550                    MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7414 054146  010465  177776                    MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7415 054152  004737  017124                    JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7416 054156  016501  000000                    MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7417 054162  012702  100204                    MOV     #SC!SSR!BIT2,R2         ;SET UP EXPECTED
7418 054166  020102                            CMP     R1,R2                   ;ARE THEY EQUAL
7419 054170  001006                            BNE     160$                    ;BR, IT MIGHT BE END OF TAPE
7420 054172  004737  020104                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7424 054176                                    ERRHRD  ERRNO,T34POS,PKTSSR     ;SPACE RECORDS REVERSE FAILED
     054176  104456                                                                    TRAP    C$ERHRD
     054200  001001                                                                    .WORD   513
     054202  055676                                                                    .WORD   T34POS
     054204  011700                                                                    .WORD   PKTSSR
7425 054206                            160$:   CKLOOP                          ;LOOP IF SELECTED
     054206  104406                                                                    TRAP    C$CLP1
7426                                   ;
7427                                   ;*************************************************************************
7428                                   ;
7429                                   ;       EOT SHOULD STILL BE SET
7430                                   ;
7431                                   ;*************************************************************************
7432                                   ;
7433 054210  013701  055446                    MOV     T34BFR+6,R1             ;PICK UP XSTO
7434 054214  010102                            MOV     R1,R2                   ;SET UP EXPECTED
7435 054216  052702  000001                    BIS     #BIT0,R2                ;SET THE EOT BIT ON IN EXPECTED
7436 054222  020102                            CMP     R1,R2                   ;WAS THE BIT ON
7437 054224  001406                            BEQ     163$                    ;BR, IF EOT WAS FOUND
7438 054226  004737  020104                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7442 054232                                    ERRHRD  ERRNO,T34ETS,EXPREC     ;EOT BIT (XSTO) NOT SET
     054232  104456                                                                    TRAP    C$ERHRD
     054234  001002                                                                    .WORD   514
     054236  056601                                                                    .WORD   T34ETS
     054240  016350                                                                    .WORD   EXPREC
7443 054242                            163$:   CKLOOP                          ;LOOP IF SELECTED
     054242  104406                                                                    TRAP    C$CLP1
7444                                   ;
7445                                   ;*************************************************************************
7446                                   ;
```

```
7447                                    ;       HOWEVER, THE TMK BIT SHOULD NOW BE CLEAR
7448                                    ;
7449                                    ;*******************************************************************************
7450                                    ;
7451 054244  013701  055446                    MOV     T34BFR+6,R1             ;PICK UP XSTO
7452 054250  010102                            MOV     R1,R2                   ;SET UP EXPECTED
7453 054252  042702  100000                    BIC     #BIT15,R2               ;CLEAR THE TMK BIT ON IN EXPECTED
7454 054256  020102                            CMP     R1,R2                   ;WAS THE BIT ON
7455 054260  001406                            BEQ     165$                    ;BR, IF TMK WAS FOUND
7456 054262  004737  020104                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7460 054266                                    ERRHRD  ERRNO,T34TMN,EXPREC     ;COULD NOT CLEAR TMK (ZSTO)
     054266  104456                                                                    TRAP    C$ERHRD
     054270  001003                                                                    .WORD   515
     054272  057515                                                                    .WORD   T34TMN
     054274  016350                                                                    .WORD   EXPREC
7461 054276                            165$:   CKLOOP                          ;LOOP IF SELECTED
     054276  104406                                                                    TRAP    C$CLP1
7462                                    ;
7463                                    ;*******************************************************************************
7464                                    ;
7465                                    ;       NOW SPACE 3 RECORDS IN REVERSE
7466                                    ;
7467                                    ;*******************************************************************************
7468                                    ;
7469 054300  012737  140410  055550            MOV     #140410,T34PK3          ;SPACE RECORDS REVERSE, ACK, CVC=1 CMD
7470 054306  012737  000003  055552            MOV     #3,T34WB                ;SPACE THREE RECORD REVERSE
7471 054314  012704  055550                    MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7472 054320  010465  177776                    MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7473 054324  004737  017124                    JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7474 054330  016501  000000                    MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7475 054334  012702  000200                    MOV     #SSR,R2                 ;SET UP EXPECTED
7476 054340  020102                            CMP     R1,R2                   ;ARE THEY EQUAL
7477 054342  001406                            BEQ     167$                    ;BR, IT MIGHT BE END OF TAPE
7478 054344  004737  020104                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7482 054350                                    ERRHRD  ERRNO,T34POS,PKTSSR     ;SPACE RECORDS COMMAND FAILED
     054350  104456                                                                    TRAP    C$ERHRD
     054352  001004                                                                    .WORD   516
     054354  055676                                                                    .WORD   T34POS
     054356  011700                                                                    .WORD   PKTSSR
7483 054360                            167$:   CKLOOP                          ;LOOP IF SELECTED
     054360  104406                                                                    TRAP    C$CLP1
7484                                    ;
7485                                    ;*******************************************************************************
7486                                    ;
7487                                    ;       NOW THE EOT BIT SHOULD BE CLEAR
7488                                    ;
7489                                    ;*******************************************************************************
7490                                    ;
7491 054362  013701  055446                    MOV     T34BFR+6,R1             ;PICK UP XSTO
7492 054366  010102                            MOV     R1,R2                   ;SET UP EXPECTED
7493 054370  042702  000001                    BIC     #BIT0,R2                ;CLEAR THE EOT BIT ON IN EXPECTED
7494 054374  020102                            CMP     R1,R2                   ;WAS THE BIT OFF
7495 054376  001404                            BEQ     170$                    ;BR, IF EOT WAS FOUND
7499 054400                                    ERRHRD  ERRNO,T34ETC,PKTSSR     ;UNABLE TO CLEAR EOT INDICATION
     054400  104456                                                                    TRAP    C$ERHRD
     054402  001005                                                                    .WORD   517
     054404  056155                                                                    .WORD   T34ETC
```

```
     054406  011700                                                          .WORD   PKTSSR
7500
7501 054410                          170$:   CKLOOP                          ;LOOP IF SELECTED
     054410  104406                                                          TRAP    C$CLP1
7502                                 ;
7503                                 ;*************************************************************************
7504                                 ;
7505                                 ;     NOW SPACE 4 RECORDS FORWARD, ONCE AGAIN OVER EOT MARKER
7506                                 ;
7507                                 ;*************************************************************************
7508                                 ;
7509 054412  012737  140010  055550          MOV     #140010,T34PK3          ;SPACE RECORDS FORWARD, ACK, CVC=1
7510 054420  012737  000004  055552          MOV     #4,T34WB                ;SPACE FOUR RECORDS
7511 054426  012704  055550                  MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7512 054432  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7513 054436  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7514 054442  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7515 054446  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
7516 054452  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
7517 054454  001406                          BEQ     190$                    ;BR, IT MIGHT BE END OF TAPE
7518 054456  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7522 054462                                  ERRHRD  ERRNO,T34POS,PKTSSR     ;SPACE RECORDS COMMAND FAILED
     054462  104456                                                          TRAP    C$ERHRD
     054464  001006                                                          .WORD   518
     054466  055676                                                          .WORD   T34POS
     054470  011700                                                          .WORD   PKTSSR
7523 054472                          190$:   CKLOOP                          ;LOOP IF SELECTED
     054472  104406                                                          TRAP    C$CLP1
7524                                 ;
7525                                 ;*************************************************************************
7526                                 ;
7527                                 ;       ONCE AGAIN THE EOT INDICATION SHOULD BE SET IN XSTATO
7528                                 ;
7529                                 ;*************************************************************************
7530                                 ;
7531 054474  013701  055446                  MOV     T34BFR+6,R1             ;PICK UP XSTO
7532 054500  010102                          MOV     R1,R2                   ;SET UP EXPECTED
7533 054502  052702  000001                  BIS     #BIT0,R2                ;SET THE EOT BIT ON IN EXPECTED
7534 054506  020102                          CMP     R1,R2                   ;WAS THE BIT ON
7535 054510  001406                          BEQ     200$                    ;BR, IF EOT WAS FOUND
7536 054512  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7540 054516                                  ERRHRD  ERRNO,T34ETS,EXPREC     ;EOT BIT (XSTO) NOT SET
     054516  104456                                                          TRAP    C$ERHRD
     054520  001007                                                          .WORD   519
     054522  056601                                                          .WORD   T34ETS
     054524  016350                                                          .WORD   EXPREC
7541 054526                          200$:   CKLOOP                          ;LOOP IF SELECTED
     054526  104406                                                          TRAP    C$CLP1
7542                                 ;
7543                                 ;*************************************************************************
7544                                 ;
7545                                 ;       NOW ISSUE A READ REVERSE COMMAND
7546                                 ;
7547                                 ;*************************************************************************
7548                                 ;
7549 054530  012737  140401  055550          MOV     #140401,T34PK3          ;READ REVERSE, ACK, CVC=1
7550 054536  013737  003076  055552          MOV     FREE,T34RB              ;SET UP WRITE BUFFER ADDRESS
```

```
7551 054544  012704  055550                MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7552 054550  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7553 054554  004737  017124                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7554 054560  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7555 054564  012702  000200                MOV     #SSR,R2                 ;SET UP EXPECTED
7556 054570  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
7557 054572  001406                        BEQ     205$                    ;BR, ONLY SSR IS SET
7558 054574  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7562 054600                                ERRHRD  ERRNO,T34RRE,PKTSSR     ;READ REVERSE COMMAND FAILED
     054600  104456                                                            TRAP    C$ERHRD
     054602  001010                                                            .WORD   520
     054604  056054                                                            .WORD   T34RRE
     054606  011700                                                            .WORD   PKTSSR
7563 054610                        205$:   CKLOOP                          ;LOOP IF SELECTED
     054610  104406                                                            TRAP    C$CLP1
7564                                ;
7565                                ;*************************************************************************
7566                                ;
7567                                ;       NOW ISSUE A READ REVERSE COMMAND
7568                                ;
7569                                ;*************************************************************************
7570 054612  012737  140401  055550        MOV     #140401,T34PK3          ;READ REVERSE, ACK, CVC=1
7571 054620  013737  003076  055552        MOV     FREE,T34RB              ;SET UP WRITE BUFFER ADDRESS
7572 054626  012704  055550                MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7573 054632  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7574 054636  004737  017124                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7575 054642  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7576 054646  012702  000200                MOV     #SSR,R2                 ;SET UP EXPECTED
7577 054652  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
7578 054654  001406                        BEQ     210$                    ;BR, IT MIGHT BE END OF TAPE
7579 054656  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7583 054662                                ERRHRD  ERRNO,T34RRE,PKTSSR     ;SECOND READ REVERSE COMMAND FAILED
     054662  104456                                                            TRAP    C$ERHRD
     054664  001011                                                            .WORD   521
     054666  056054                                                            .WORD   T34RRE
     054670  011700                                                            .WORD   PKTSSR
7584 054672                        210$:   CKLOOP                          ;LOOP IF SELECTED
     054672  104406                                                            TRAP    C$CLP1
7585                                ;
7586                                ;*************************************************************************
7587                                ;
7588                                ;       NOW ISSUE A READ COMMAND
7589                                ;
7590                                ;*************************************************************************
7591 054674  012737  140001  055550        MOV     #140001,T34PK3          ;READ DATA, ACK, CVC=1
7592 054702  013737  003076  055552        MOV     FREE,T34RB              ;SET UP WRITE BUFFER ADDRESS
7593 054710  012737  066540  055556        MOV     #28000.,T34SZ           ;SET UP BUFFER SIZE (INC # OF BYTES)
7594 054716  012704  055550                MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7595 054722  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7596 054726  004737  017124                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7597 054732  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7598 054736  012702  000200                MOV     #SSR,R2                 ;SET UP EXPECTED
7599 054742  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
7600 054744  001406                        BEQ     230$                    ;BR, IT MIGHT BE END OF TAPE
7601 054746  004737  020104                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7605 054752                                ERRHRD  ERRNO,T34RRF,PKTSSR     ;READ FORWARD COMMAND FAILED
     054752  104456                                                            TRAP    C$ERHRD
```

```
                     054754  001012                                                          .WORD   522
                     054756  057600                                                          .WORD   T34RRF
                     054760  011700                                                          .WORD   PKTSSR
7606                 054762                          230$:   CKLOOP                          ;LOOP IF SELECTED
                     054762  104406                                                          TRAP    C$CLP1
7607                 054764  012737  140001  055550          MOV     #140001,T34PK3          ;READ DATA, ACK, CVC=1
7608                 054772  013737  003076  055552          MOV     FREE,T34RB              ;SET UP WRITE BUFFER ADDRESS
7609                 055000  012737  066540  055556          MOV     #28000.,T34SZ           ;SET UP BUFFER SIZE (INC # OF BYTES)
7610                 055006  012704  055550                  MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7611                 055012  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7612                 055016  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7613                 055022  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7614                 055026  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
7615                 055032  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
7616                 055034  001406                          BEQ     235$                    ;BR, IT MIGHT BE END OF TAPE
7617                 055036  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7621                 055042                                  ERRHRD  ERRNO,T34RRF,PKTSSR     ;SECOND READ FORWARD FAILED
                     055042  104456                                                          TRAP    C$ERHRD
                     055044  001013                                                          .WORD   523
                     055046  057600                                                          .WORD   T34RRF
                     055050  011700                                                          .WORD   PKTSSR
7622                 055052                          235$:   CKLOOP                          ;LOOP IF SELECTED
                     055052  104406                                                          TRAP    C$CLP1
7623                                                 ;
7624                                                 ;*************************************************************************
7625                                                 ;
7626                                                 ;       THE EOT BIT SHOULD HAVE REMAINED SET
7627                                                 ;
7628                                                 ;*************************************************************************
7629                                                 ;
7630                 055054  013701  055446                  MOV     T34BFR+6,R1             ;PICK UP XSTO
7631                 055060  010102                          MOV     R1,R2                   ;SET UP EXPECTED
7632                 055062  052702  000001                  BIS     #BIT0,R2                ;SET THE EOT BIT ON IN EXPECTED
7633                 055066  020102                          CMP     R1,R2                   ;WAS THE BIT ON
7634                 055070  001406                          BEQ     240$                    ;BR, IF EOT WAS FOUND
7635                 055072  004737  020104                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
7639                 055076                                  ERRHRD  ERRNO,T34ETZ,EXPREC     ;EOT BIT (XSTO) NOT SET
                     055076  104456                                                          TRAP    C$ERHRD
                     055100  001014                                                          .WORD   524
                     055102  056667                                                          .WORD   T34ETZ
                     055104  016350                                                          .WORD   EXPREC
7640                 055106                          240$:   CKLOOP                          ;LOOP IF SELECTED
                     055106  104406                                                          TRAP    C$CLP1
7641                                                 ;
7642                                                 ;*************************************************************************
7643                                                 ;
7644                                                 ;       NOW ISSUE A SPACE RECORDS REVERSE FOR 5 RECORDS
7645                                                 ;
7646                                                 ;*************************************************************************
7647                                                 ;
7648                 055110  012737  140410  055550          MOV     #140410,T34PK3          ;SPACE RECORDS REVERSE, ACK, CVC=1 CMD.
7649                 055116  012737  000005  055552          MOV     #5,T34RB                ;NUMBER OF RECORDS TO SPACE
7650                 055124  012704  055550                  MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
7651                 055130  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
7652                 055134  004737  017124                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7653                 055140  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
7654                 055144  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
```

```
 7655 055150  020102                    CMP     R1,R2                   ;ARE THEY EQUAL
 7656 055152  001406                    BEQ     250$                    ;BR, IT MIGHT BE END OF TAPE
 7657 055154  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 7661 055160                            ERRHRD  ERRNO,T34POS,PKTSSR     ;SPACE 5 RECORDS REVERSE COMMAND FAILED
      055160  104456                                                            TRAP    C$ERHRD
      055162  001015                                                            .WORD   525
      055164  055676                                                            .WORD   T34POS
      055166  011700                                                            .WORD   PKTSSR
 7662 055170                    250$:   CKLOOP                          ;LOOP IF SELECTED
      055170  104406                                                            TRAP    C$CLP1
 7663                                   ;
 7664                                   ;*************************************************************************
 7665                                   ;
 7666                                   ;       EOT SHOULD BE CLEAR AS WE ARE NOW IN FRONT OF EOT
 7667                                   ;
 7668                                   ;*************************************************************************
 7669                                   ;
 7670 055172  013701  055446            MOV     T34BFR+6,R1             ;PICK UP XSTO
 7671 055176  010102                    MOV     R1,R2                   ;SET UP EXPECTED
 7672 055200  042702  000001            BIC     #BIT0,R2                ;CLEAR THE EOT BIT ON IN EXPECTED
 7673 055204  020102                    CMP     R1,R2                   ;WAS THE BIT ON
 7674 055206  001406                    BEQ     260$                    ;BR, IF EOT WAS FOUND
 7675 055210  004737  020104            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 7679 055214                            ERRHRD  ERRNO,T34ETC,EXPREC     ;EOT BIT (XSTO) NOT CLEAR
      055214  104456                                                            TRAP    C$ERHRD
      055216  001016                                                            .WORD   526
      055220  056155                                                            .WORD   T34ETC
      055222  016350                                                            .WORD   EXPREC
 7680 055224                    260$:   CKLOOP                          ;LOOP IF SELECTED
      055224  104406                                                            TRAP    C$CLP1
 7681                                   ;
 7682                                   ;*************************************************************************
 7683                                   ;
 7684                                   ;       NOW SPACE FORWARD 5 RECORDS AGAIN
 7685                                   ;
 7686                                   ;*************************************************************************
 7687                                   ;
 7688 055226  012737  140010  055550            MOV     #140010,T34PK3          ;SPACE RECORDS FORWARD, ACK, CVC=1 CMD.
 7689 055234  012737  000005  055552            MOV     #5,T34RB                ;NUMBER OF RECORDS TO SPACE
 7690 055242  012704  055550                    MOV     #T34PK3,R4              ;R4 = POINTER TO PACKET
 7691 055246  010465  177776                    MOV     R4,TSDB(R5)             ;ISSUE COMMAND
 7692 055252  004737  017124                    JSR     PC,WAITF                ;WAIT FOR SSR TO SET
 7693 055256  016501  000000                    MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
 7694 055262  012702  000200                    MOV     #SSR,R2                 ;SET UP EXPECTED
 7695 055266  020102                            CMP     R1,R2                   ;ARE THEY EQUAL
 7696 055270  001406                            BEQ     270$                    ;BR, IT MIGHT BE END OF TAPE
 7697 055272  004737  020104                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 7701 055276                                    ERRHRD  ERRNO,T34POS,PKTSSR     ;SPACE RECORDS FORWARD COMMAND FAILED
      055276  104456                                                                    TRAP    C$ERHRD
      055300  001017                                                                    .WORD   527
      055302  055676                                                                    .WORD   T34POS
      055304  011700                                                                    .WORD   PKTSSR
 7702 055306                            270$:   CKLOOP                          ;LOOP IF SELECTED
      055306  104406                                                                    TRAP    C$CLP1
 7703                                           ;
 7704                                           ;*************************************************************************
 7705                                           ;
```

```
                        ;       EOT SHOULD BE SET AGAIN
                        ;
                        ;*************************************************************************
                        ;
055310  013701  055446          MOV     T34BFR+6,R1             ;PICK UP XSTO
055314  010102                  MOV     R1,R2                   ;SET UP EXPECTED
055316  052702  000001          BIS     #BIT0,R2                ;SET THE EOT BIT ON IN EXPECTED
055322  020102                  CMP     R1,R2                   ;WAS THE BIT ON
055324  001404                  BEQ     280$                    ;BR, IF EOT WAS FOUND
055326                          ERRHRD  ERRNO,T34ETS,PKTSSR     ;EOT FAILED TO SET AFTER SPACE CMD.
055326  104456                                                          TRAP    C$ERHRD
055330  001020                                                          .WORD   528
055332  056601                                                          .WORD   T34ETS
055334  011700                                                          .WORD   PKTSSR
                        280$:   CKLOOP                          ;LOOP IF SELECTED
055336  104406                                                          TRAP    C$CLP1
                        ;
                        ;*************************************************************************
                        ;
                        ;       ISSUE A REWIND COMMAND
                        ;
                        ;*************************************************************************
                        ;
055340  004737  010434          JSR     PC,REWIND               ;REWIND CALL
055344  103411                  BCS     320$                    ;BR, IF TSSR IS OK (GOOD)
055346  016501  000000          MOV     TSSR(R5),R1             ;GET TSSR
055352  010004                  MOV     R0,R4                   ;SET UP PACKET
055354  004737  020104          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
055360                          ERRHRD  ERRNO,T34RWN,PKTSSR     ;TSSR IS INCORRECT AFTER REWIND
055360  104456                                                          TRAP    C$ERHRD
055362  001021                                                          .WORD   529
055364  057246                                                          .WORD   T34RWN
055366  011700                                                          .WORD   PKTSSR
                        320$:   CKLOOP                          ;LOOP IF SELECTED
055370  104406                                                          TRAP    C$CLP1
                        600$:
                        ENDSUB                                  ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
055372                                                                  L10060:
055372  104403                                                          TRAP    C$ESUB
055374  004737  017360          JSR     PC,TSTLOOP              ;DO WE NEED TO ITERATE TEST
055400  103002                  BCC     163$                    ;BR, IF NO LOOP REQUIRED
055402  000137  053040          JMP     T34LOOP                 ;EXECUTE AGAIN
055406                  163$:   EXIT    TST                     ;ALL DONE THIS TEST
055406  104432                                                          TRAP    C$EXIT
055410  002600                                                          .WORD   L10057-.
```

```
                        ;*
                        ;LOCAL STORAGE FOR THIS TEST
                        ;-
                                .BLKB   10-<.-TUV2A&7>
                        T34PACKET:                              ;COMMAND PACKET FOR TEST
                                .WORD   100004                  ;WRITE CHARACTERISTICS COMMAND, WITH ACK
                                .WORD   T34DATA                 ;ADDRESS OF CHARACTERISTICS BLOCK
                                .WORD   0
                                .WORD   8.                      ;STARTING VALUE OF BLOCK SIZE
                        T34DATA:                                ;CHARACTERISTICS DATA BLOCK
                                .WORD   T34BFR                  ;ADDRESS OF MESSAGE BUFFER
                                .WORD   0
                                .WORD   10.                     ;LENGTH OF MESSAGE BUFFER
                                .WORD   0
                        T34BFR: .BLKW   25.                     ;MESSAGE BUFFER
                        ;
                        ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
                        ;
                                .BLKB   10-<.-TUV2A&7>
                        T34PK2:
                                .WORD   100006                  ;WRITE SUB SYS MEM COMMAND, AND ACK
                                .WORD   T34BF2                  ;ADDRESS OF SELECT BLOCK DATA
                                .WORD   0
                                .WORD   6.                      ;SIZE OF DATA PACKET

                                .BLKB   10-<.-TUV2A&7>
                        T34PK3:
                                .WORD   100005                  ;WRITE COMMAND,  AND ACK
                        T34RB:
                        T34WB:  .WORD   0                       ;ADDRESS OF WRITE/READ BUFFER
                                .WORD   0
                        T34SZ:  .WORD   0                       ;SIZE OF BUFFER (EXTENT)
                                .EVEN
                        ;
                        T34RSZ: .WORD   0                       ;LARGEST TAPE RECORD IN BYTES
                        T34CNT: .WORD   0                       ;TAPE RECORD COUNTER
                        T34DLY: .WORD   0                       ;DELAY COUNTER

                        T34TRK: .WORD   0                       ;HOLD TRACK NUMBER
                        ;
                        ;
                        T34BF2:
                        T34BS0: .BYTE   10                      ;BSEL0 AREA
                        T34BS1: .BYTE   200                     ;BSEL1 AREA
                        T34S2:  .WORD   0                       ;SEL 2 AREA
                        T34S3:  .WORD   0                       ;DATA AREA
                        ;
                        ;
                                .EVEN
                        ;TAPE MOTION PACKET COMMAND VALUES

                        T34WD:  .WORD   100005                  ;WRITE DATA (NEXT)
                        T34WDR: .WORD   100405                  ;WRITE DATA RETRY
                        T34CON: .WORD   102005                  ;WRITE CONTINOUS
                                .WORD   177777                  ;END OF DATA
```

```
                                        ;+
                                        ;LOCAL TEXT MESSAGES FOR TEST
                                        ;-


055606  045  116  045  EWMSG:  .ASCIZ  '%N%A Early Warning Indicator Just Received, Track = %D2'
055676  124  123  123  T34POS: .ASCIZ  'TSSR Incorrect After Position (SPACE RECORDS) Command'
055764  127  122  111  T34ETO: .ASCIZ  'WRITE TAPE MARK Beyond EOT Failed To Set EOT Bit (XSTO)'
056054  122  105  101  T34RRE: .ASCIZ  'READ REVERSE Command At EOT Didn''t Give Normal Termination (TSSR)'
056155  125  156  141  T34ETC: .ASCIZ  'Unable To Clear EOT Indication, (XSTO) Bit 0'
056232  123  153  151  T34BOT: .ASCIZ  'Skip File Mark Reverse (over entire tape) Failed To Set BOT (XSTO) Bit'
056341  127  122  111  T34WTM: .ASCIZ  'WRITE TAPE MARK At EOT Failed To Set Tape Status Alert'
056430  127  122  111  T34ET2: .ASCIZ  'WRITE DATA Beyond EOT Failed To Set Tape Status Alert'
056516  127  122  111  T34ETN: .ASCIZ  'WRITE DATA Beyond EOT Failed To Set EOT Bit (XSTO)'
056601  123  120  101  T34ETS: .ASCIZ  'SPACE RECORDS Beyond EOT Failed To Set EOT Bit (XSTO)'
056667  122  105  101  T34ETZ: .ASCIZ  'READ DATA Beyond EOT Failed To Set EOT Bit (XSTO)'
056751  120  117  123  T34TMK: .ASCIZ  'POSITION Command Beyond EOT Into A Tape Mark Failed To Set TMK (XSTO)'
057057  105  117  124  T34ET:  .ASCIZ  'EOT Not Found In 65000 3.5K Writes, (Use Shorter Tape)'
057146  127  122  111  T34EOT: .ASCIZ  'WRITE DATA OVER EOT GAVE NO TAPE STATUS ALERT'

057224  117  160  145  TST34ID:        .ASCIZ  'Operations At EOT'
057246  124  123  123  T34RWN: .ASCIZ  'TSSR Incorrect After Position (REWIND) Command'
057325  124  123  123  T34STM: .ASCIZ  'TSSR Incorrect After SKIP TAPE MARK REVERSE Beyond EOT Mark'
057421  105  117  124  T34STE: .ASCIZ  'EOT (XSTO) Not Set After SKIP TAPE MARK REVERSE, Beyond EOT'
057515  125  156  141  T34TMN: .ASCIZ  'Unable To Clear TMK (XSTO) Bit Using Space Command'
057600  124  123  123  T34RRF: .ASCIZ  'TSSR Incorrect After READ FORWARD Command'
057652  124  123  123  T34WOL: .ASCIZ  'TSSR Incorrect After SKIP FILE MARK REVERSE'
                                .EVEN
                                ;+
                                ;
                                ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
                                ;WRITE SUBSYSTEM MEMORY COMMAND
                                ;
                                ;-

057726                          T34REST:
057726                                  SAVREG                          ;SAVE THE REGISTERS
057732  012701  055420                  MOV     #T34PACKET,R1           ;START OF THE PACKET
057736  012721  100004                  MOV     #100004,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK
057742  012721  055430                  MOV     #T34DATA,(R1)+          ;ADDRESS OF CHARAISTICS DATA BLOCK
057746  005021                          CLR     (R1)+                   ;EXTENDED ADDRESS
057750  012721  000012                  MOV     #10.,(R1)+              ;SIZE OF DATA BLOCK IN BYTES
057754  012721  055440                  MOV     #T34BFR,(R1)+           ;ADDRESS OF MESSAGE BUFFER
057760  005021                          CLR     (R1)+
057762  012721  000024                  MOV     #20.,(R1)+              ;LENGTH OF MESSAGE BUFFER
057766  005021                          CLR     (R1)+
057770  012711  000000                  MOV     #0,(R1)                 ;SELECT DRIVE ZERO
057774  012702  000030                  MOV     #24.,R2                 ;NUMBER OF LOCATIONS TO BE CLEARED
060000  012762  177777  055440  64$:    MOV     #177777,T34BFR(R2)      ;ALL ONES TO MESSAGE BUFFER
060006  005742                          TST     -(R2)                   ;BUMP DOWN TO NEXT LOCATION
060010  020227  000000                  CMP     R2,#0                   ;R2 AT ZERO YET
060014  001371                          BNE     64$                     ;KEEP GOING UNTIL DONE
060016  000207                          RTS     PC                      ;RETURN

                                ;
```

```
                                        ;       THIS SUBROUTINE CHECKS FOR EARLY WARNING SET AND IF SET
                                        ;       IT ISSUES A BACKSPACE AND AN ERASE TO GET TO THE NEXT TRACK
                                        ;
060020                                  EWCHK:
060020                                          SAVREG                          ;SAVE ALL REGISTERS ETC.
060024  013737  055450  055566                  MOV     T34BFR+10,T34TRK        ;READ XSTAT1 FOR EW
060032  032737  000010  055566                  BIT     #BIT3,T34TRK            ;WAS EW SET IN XSTAT1
060040  001424                                  BEQ     100$                    ;BR, IF IT WAS NOT
060042  012703  100001                          MOV     #100001,R3              ;PARAMETERS FOR SPACE ROUTINE
060046  004737  010134                          JSR     PC,SPACE                ;SPACE 1 RECORD REVERSE
060052  012704  060120                          MOV     #110$,R4                ;ADDRESS OF AN ERASE COMMAND
060056  010465  177776                          MOV     R4,TSDB(R5)             ;ISSUE THE ERASE COMMAND
060062  004737  017124                          JSR     PC,WAITF                ;WAIT FOR THE SSR BIT TO SET
060066  013702  055566                          MOV     T34TRK,R2               ;GET TRACK NUMBER
060072  006002                                  ROR     R2                      ;SHIFT OVER 4 BITS TO BITO
060074  006002                                  ROR     R2                      ;SHIFT OVER 4 BITS TO BITO
060076  006002                                  ROR     R2                      ;SHIFT OVER 4 BITS TO BITO
060100  006002                                  ROR     R2                      ;SHIFT OVER 4 BITS TO BITO
060102  042702  177760                          BIC     #177760,R2              ;ONLY FOUR BITS PASS
                                        ;+
                                        ;       THIS MESSAGE USED TO PRINT EARLY WARNING MESSAGE. TRACK NINE
                                        ;       DID NOT ALWAYS GIVE INDICATION. THIS WAS BECAUSE IT WASN'T
                                        ;       ALWAYS DETECTED DURING A WRITE. SO MESSAGE REMOVED.
                                        ;-

                                        ;       PRINTX  #EWMSG,R2               ;"JUST RECEIVED EARLY WARNING IND."
060106  000261                                  SEC                             ;SET THE CARRY BIT
060110  000401                                  BR      105$                    ;EXIT
060112  000241                          100$:   CLC                             ;CLEAR CARRY (NO EW FOUND)
060114  000207                          105$:   RTS     PC                      ;RETURN
060116                                          .BLKB   10-<.-TUV2A&7>
060120  140411                          110$:   .WORD   140411                  ;ERASE DATA, CVC=1, AND ACK COMMAND
060122                                  T34RT2:
060122                                          SAVREG                          ;SAVE THE REGISTERS
060126  012701  055530                          MOV     #T34PK2,R1              ;START OF THE PACKET
060132  012721  100006                          MOV     #100006,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK
060136  012721  055570                          MOV     #T34BF2,(R1)+           ;ADDRESS OF DATA BLOCK
060142  005021                                  CLR     (R1)+                   ;EXTENDED ADDRESS
060144  012721  000006                          MOV     #6.,(R1)+               ;SIZE OF DATA BLOCK IN BYTES
060150  012701  055570                          MOV     #T34BF2,R1              ;POINT TO DATA SEL AREA
060154  005021                                  CLR     (R1)+
060156  005021                                  CLR     (R1)+
060160  005011                                  CLR     (R1)
060162  000207                                  RTS     PC                      ;RETURN
060164                                  T34RT3:
060164                                          SAVREG                          ;SAVE THE REGISTERS
060170  012701  055550                          MOV     #T34PK3,R1              ;START OF THE PACKET
060174  012721  100005                          MOV     #100005,(R1)+           ;WRITE  TAPE. WITH ACK
060200  005021                                  CLR     (R1)+                   ;ADDRESS OF DATA BLOCK
060202  005021                                  CLR     (R1)+                   ;EXTENDED ADDRESS
060204  005011                                  CLR     (R1)                    ;SIZE OF DATA BLOCK
060206  000207                                  RTS     PC                              ;RETURN
060210                                          ENDTST
060210
060210  104401                                                          L10057:
                                                                                TRAP    C$ETST
7921
```

SEQ 0203

```
                                .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.
                               ;--
060212                                 BGNHRD
                               .MCALL  M$PUSH,M$INCR,M$GNINS,M$GNGBL
        000040                 I$HRD=F$BGN
060212                         M$PUSH  T$NS,T$NESTLEV,F$HARD
                               .MCALL  M$INCR,M$SETS
060212                         M$INCR  T$NESTLEV
        000001                 T$NESTLEV=T$NESTLEV+1
060212                         M$SETS  T$NS,\T$NESTLEV,F$HARD
        000004                 T$NS1=F$HARD
        010061                 T$$HARD=T$TAGNUM
060212                         M$INCR   T$TAGNUM
        010062                 T$TAGNUM=T$TAGNUM+1
                               .IRP    TAG,<\T$$HARD>
                               M$GNINS <.WORD L'TAG'-L$HARD/2>
                               .ENDM
060212                         M$GNINS <.WORD L10061-L$HARD/2>
                               .IF LT SVCINSINSTR
                               .MEXIT
                               .ENDC
                               .IF EQ SVCINS
                               .LIST
                                       .WORD L10061-L$HARD/2
                                             .NLIST
                               .MEXIT
                               .ENDC
                               .IF GT SVCINS
060212  000015                                                        .WORD L10061-L$HARD/2
                               .ENDC
060214                         M$GNGBL L$HARD
                               .MCALL  M$GEN
                               .IF NB,L$HARD
                               .IF NB,
                               M$GEN   L$HARD,:,SVCGBL,
                               .ENDC
                               .IF B,
060214                         M$GEN   L$HARD,:,SVCGBL,< >
                               .IF LE  SVCGBL
                               .IIF EQ SVCGBL,.LIST
                               L$HARD::
                               .IIF EQ SVCGBL,.NLIST
                               .MEXIT
                               .ENDC
060214                                                          L$HARD::
                               .ENDC
                               .ENDC

060214                                 GPRMA   HPM1,0,0,160000,177776,YES      ;GET TSBA/TSDB REGISTER ADDRESS.
```

```
                        .MCALL  M$RADIX,M$DEFAULT,M$EXCP,M$WORD,M$CNTOP
                        .IF IDN A,0
                        .ERROR  ;INVALID RADIX
                        .ENDC
        000000          T$TEMP=0&1
                        .IF NE  T$TEMP & 1
                        .ERROR  ;ODD OFFSET
                        .ENDC
                        .IF LT  G$OFSIZE-0
                        .ERROR  ;OFFSET TOO BIG
                        .ENDC
        000001          T$CODE=G$PRMA + <0 * G$OFFSET>
060214                  M$RADIX 0,T$TEMP
                        .IF IDN B,0
                        T$TEMP=G$RADB
                        .MEXIT
                        .ENDC
                        .IF IDN O,0
        000020          T$TEMP=G$RADO
                        .MEXIT
                        .ENDC
                        .IF IDN D,0
                        T$TEMP=G$RADD
                        .MEXIT
                        .ENDC
                        .IF IDN L,0
                        T$TEMP=G$RADL
                        .MEXIT
                        .ENDC
                        .IF IDN A,0
                        T$TEMP=G$RADA
                        .MEXIT
                        .ENDC
                        T$TEMP=-1
                        .ERROR  ;ILL. RADIX "0"
        000021          T$CODE=T$CODE ! T$TEMP
060214                  M$DEFAULT       YES,T$TEMP
                        .IF IDN YES,YES
        000010          T$TEMP=G$YES
                        .MEXIT
                        .ENDC
                        .IF IDN NO,YES
                        T$TEMP=G$NO
                        .MEXIT
                        .ENDC
                        T$TEMP=-1
                        .ERROR  ;DEFAULT "YES" MUST BE "YES" OR "NO"
        000031          T$CODE=T$CODE ! T$TEMP
        000000          T$EXCP=0
060214                  M$EXCP  T$CODE,T$EXCP,G$LOLIM,T$LOLIM,160000
                        .IF IDN <@>,<160000>
                        .IF LT G$OFSIZE-
                        .ERROR  ;INDIRECT PAR. TOO BIG
                        .MEXIT
                        .ENDC
                        T$LOLIM=/2
                        T$CODE=T$CODE ! G$EXCP
```

```
                        T$EXCP=T$EXCP ! G$LOLIM
                        .IFF
                        .IF B,
        160000          T$LOLIM=160000
                        .IFF
                        .ERROR  ;ILL. DEFERRED MODE
                        .ENDC
                        .ENDC
060214                  M$EXCP  T$CODE,T$EXCP,G$HILIM,T$HILIM,177776
                        .IF IDN <@>,<177776>
                        .IF LT G$OFSIZE-
                        .ERROR  ;INDIRECT PAR. TOO BIG
                        .MEXIT
                        .ENDC
                        T$HILIM=/2
                        T$CODE=T$CODE ! G$EXCP
                        T$EXCP=T$EXCP ! G$HILIM
                        .IFF
                        .IF B,
        177776          T$HILIM=177776
                        .IFF
                        .ERROR  ;ILL. DEFERRED MODE
                        .ENDC
                        .ENDC
060214                  M$CNTOP <>,<T$CODE>
                        .IF NB
                        .IF NE  I$SFT - F$BGN
                        .ERROR  ; "COUNT" OPTION VALID ONLY IN S.W. QUES.
                        .MEXIT
                        .ENDC
                        T$CODE=T$CODE ! G$CNTOP
                        .ENDC
060214                  M$WORD  <T$CODE,HPM1,T$LOLIM,T$HILIM>
                        .MCALL  M$GNINS
                        .IRP    N,<T$CODE,HPM1,T$LOLIM,T$HILIM>
                        M$GNINS <.WORD  N>
                        .ENDM
060214                  M$GNINS <.WORD  T$CODE>
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD   T$CODE
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
060214  000031                                                  .WORD   T$CODE
                        .ENDC
060216                  M$GNINS <.WORD  HPM1>
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD   HPM1
```

```
                                        .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060216  060246                                                          .WORD   HPM1
                                .ENDC
060220                          M$GNINS <.WORD  T$LOLIM>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   T$LOLIM
                                        .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060220  160000                                                          .WORD   T$LOLIM
                                .ENDC
060222                          M$GNINS <.WORD  T$HILIM>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   T$HILIM
                                        .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060222  177776                                                          .WORD   T$HILIM
                                .ENDC
                                .IF NE  T$EXCP
                                M$WORD  T$EXCP
                                .ENDC
                                .IF NB
                                M$WORD </2>
                                .ENDC
7937 060224                             GPRMA   HPM2,2,0,0,776,YES              ;GET VECTOR ADDRESS.
                                .MCALL  M$RADIX,M$DEFAULT,M$EXCP,M$WORD,M$CNTOP
                                .IF IDN A,0
                                .ERROR  ;INVALID RADIX
                                .ENDC
        000000                  T$TEMP=2&1
                                .IF NE  T$TEMP & 1
                                .ERROR  ;ODD OFFSET
                                .ENDC
                                .IF LT  G$OFSIZE-2
                                .ERROR  ;OFFSET TOO BIG
                                .ENDC
        001001                  T$CODE=G$PRMA + <2 * G$OFFSET>
060224                          M$RADIX 0,T$TEMP
                                .IF IDN B,0
                                T$TEMP=G$RADB
                                .MEXIT
                                .ENDC
                                .IF IDN 0,0
```

```
        000020                  T$TEMP=G$RADO
                                .MEXIT
                                .ENDC
                                .IF IDN D,O
                                T$TEMP=G$RADD
                                .MEXIT
                                .ENDC
                                .IF IDN L,O
                                T$TEMP=G$RADL
                                .MEXIT
                                .ENDC
                                .IF IDN A,O
                                T$TEMP=G$RADA
                                .MEXIT
                                .ENDC
                                T$TEMP=-1
                                .ERROR  ;ILL. RADIX "O"
        001021                  T$CODE=T$CODE ! T$TEMP
060224                          M$DEFAULT       YES,T$TEMP
                                .IF IDN YES,YES
        000010                  T$TEMP=G$YES
                                .MEXIT
                                .ENDC
                                .IF IDN NO,YES
                                T$TEMP=G$NO
                                .MEXIT
                                .ENDC
                                T$TEMP=-1
                                .ERROR  ;DEFAULT "YES" MUST BE "YES" OR "NO"
        001031                  T$CODE=T$CODE ! T$TEMP
        000000                  T$EXCP=0
060224                          M$EXCP  T$CODE,T$EXCP,G$LOLIM,T$LOLIM,O
                                .IF IDN <@>,<O>
                                .IF LT G$OFSIZE-
                                .ERROR  ;INDIRECT PAR. TOO BIG
                                .MEXIT
                                .ENDC
                                T$LOLIM=/2
                                T$CODE=T$CODE ! G$EXCP
                                T$EXCP=T$EXCP ! G$LOLIM
                                .IFF
                                .IF B,
        000000                  T$LOLIM=O
                                .IFF
                                .ERROR  ;ILL. DEFERRED MODE
                                .ENDC
                                .ENDC
060224                          M$EXCP  T$CODE,T$EXCP,G$HILIM,T$HILIM,776
                                .IF IDN <@>,<776>
                                .IF LT G$OFSIZE-
                                .ERROR  ;INDIRECT PAR. TOO BIG
                                .MEXIT
                                .ENDC
                                T$HILIM=/2
                                T$CODE=T$CODE ! G$EXCP
                                T$EXCP=T$EXCP ! G$HILIM
                                .IFF
```

```
                        .IF B,
        000776          T$HILIM=776
                        .IFF
                        .ERROR  ;ILL. DEFERRED MODE
                        .ENDC
                        .ENDC
060224                  M$CNTOP <>,<T$CODE>
                        .IF NB
                        .IF NE  I$SFT - F$BGN
                        .ERROR  ; "COUNT" OPTION VALID ONLY IN S.W. QUES.
                        .MEXIT
                        .ENDC
                        T$CODE=T$CODE ! G$CNTOP
                        .ENDC
060224                  M$WORD  <T$CODE,HPM2,T$LOLIM,T$HILIM>
                        .MCALL  M$GNINS
                        .IRP    N,<T$CODE,HPM2,T$LOLIM,T$HILIM>
                        M$GNINS <.WORD  N>
                        .ENDM
060224                  M$GNINS <.WORD  T$CODE>
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD   T$CODE
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
060224  001031                                                  .WORD   T$CODE
                        .ENDC
060226                  M$GNINS <.WORD  HPM2>
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD   HPM2
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
060226  060275                                                  .WORD   HPM2
                        .ENDC
060230                  M$GNINS <.WORD  T$LOLIM>
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD   T$LOLIM
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
060230  000000                                                  .WORD   T$LOLIM
```

```
                        .ENDC
      060232            M$GNINS <.WORD  T$HILIM>
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD   T$HILIM
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
      060232  000776                                                      .WORD   T$HILIM
                        .ENDC
                        .IF NE  T$EXCP
                        M$WORD  T$EXCP
                        .ENDC
                        .IF NB
                        M$WORD </2>
                        .ENDC
   7938 060234                  GPRMD   HPM3,4,0,340,0,7,YES            ;GET INTERRUPT PRIORITY.
                        .MCALL  M$RADIX,M$DEFAULT,M$EXCP,M$WORD,M$CNTOP
                        .IF IDN A,0
                        .IF EQ T$GMANID
                        .ERROR  ;ASCII RADIX VALID ONLY ON "GMANID"
                        .ENDC
                        .ENDC
            000000      T$TEMP=4&1
                        .IF NE  T$TEMP & 1
                        .ERROR  ;ODD OFFSET
                        .ENDC
                        .IF LT  G$OFSIZE-4
                        .ERROR  ;OFFSET TOO BIG
                        .ENDC
            002002      T$CODE=G$PRMD + <4 * G$OFFSET>
      060234            M$RADIX 0,T$TEMP
                        .IF IDN B,0
                        T$TEMP=G$RADB
                        .MEXIT
                        .ENDC
                        .IF IDN 0,0
            000020      T$TEMP=G$RAD0
                        .MEXIT
                        .ENDC
                        .IF IDN D,0
                        T$TEMP=G$RADD
                        .MEXIT
                        .ENDC
                        .IF IDN L,0
                        T$TEMP=G$RADL
                        .MEXIT
                        .ENDC
                        .IF IDN A,0
                        T$TEMP=G$RADA
                        .MEXIT
                        .ENDC
                        T$TEMP=-1
```

```
                        .ERROR  ;ILL. RADIX "O"
        002022          T$CODE=T$CODE ! T$TEMP
060234                  M$DEFAULT       YES,T$TEMP
                        .IF IDN YES,YES
        000010          T$TEMP=G$YES
                        .MEXIT
                        .ENDC
                        .IF IDN NO,YES
                        T$TEMP=G$NO
                        .MEXIT
                        .ENDC
                        T$TEMP=-1
                        .ERROR  ;DEFAULT "YES" MUST BE "YES" OR "NO"
        002032          T$CODE=T$CODE ! T$TEMP
        000000          T$EXCP=0
060234                  M$EXCP  T$CODE,T$EXCP,G$LOLIM,T$LOLIM,0
                        .IF IDN <@>,<0>
                        .IF LT G$OFSIZE-
                        .ERROR  ;INDIRECT PAR. TOO BIG
                        .MEXIT
                        .ENDC
                        T$LOLIM=/2
                        T$CODE=T$CODE ! G$EXCP
                        T$EXCP=T$EXCP ! G$LOLIM
                        .IFF
                        .IF B,
        000000          T$LOLIM=0
                        .IFF
                        .ERROR  ;ILL. DEFERRED MODE
                        .ENDC
                        .ENDC
060234                  M$EXCP  T$CODE,T$EXCP,G$HILIM,T$HILIM,7
                        .IF IDN <@>,<7>
                        .IF LT G$OFSIZE-
                        .ERROR  ;INDIRECT PAR. TOO BIG
                        .MEXIT
                        .ENDC
                        T$HILIM=/2
                        T$CODE=T$CODE ! G$EXCP
                        T$EXCP=T$EXCP ! G$HILIM
                        .IFF
                        .IF B,
        000007          T$HILIM=7
                        .IFF
                        .ERROR  ;ILL. DEFERRED MODE
                        .ENDC
                        .ENDC
060234                  M$CNTOP <>,<T$CODE>
                        .IF NB
                        .IF NE  I$SFT - F$BGN
                        .ERROR  ; "COUNT" OPTION VALID ONLY IN S.W. QUES.
                        .MEXIT
                        .ENDC
                        T$CODE=T$CODE ! G$CNTOP
                        .ENDC
060234                  M$WORD  <T$CODE,HPM3,340,T$LOLIM,T$HILIM>
                        .MCALL  M$GNINS
```

```
                                .IRP    N,<T$CODE,HPM3,340,T$LOLIM,T$HILIM>
                                M$GNINS <.WORD  N>
                                .ENDM
060234                          M$GNINS <.WORD  T$CODE>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   T$CODE
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060234  002032                                                          .WORD   T$CODE
                                .ENDC
060236                          M$GNINS <.WORD  HPM3>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   HPM3
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060236  060321                                                          .WORD   HPM3
                                .ENDC
060240                          M$GNINS <.WORD  340>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   340
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060240  000340                                                          .WORD   340
                                .ENDC
060242                          M$GNINS <.WORD  T$LOLIM>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   T$LOLIM
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060242  000000                                                          .WORD   T$LOLIM
                                .ENDC
060244                          M$GNINS <.WORD  T$HILIM>
                                .IF LT SVCINSINSTR
```

```
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   T$HILIM
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
     060244  000007                                                                    .WORD   T$HILIM
                                .ENDC
                                .IF NE  T$EXCP
                                M$WORD  T$EXCP
                                .ENDC
                                .IF NB
                                M$WORD </2>
                                .ENDC
  7939 060246                           ENDHRD
                                .MCALL  M$POP,M$GNINS,M$GNTAG,M$ENDERR
     060246                     M$POP   T$NS,T$NESTLEV,T$TEMP
                                .MCALL  M$GETS,M$DECR
                                .IF LT  T$NESTLEV
                                .ERROR  T$NESTLEV       ; MACRO T$NS UNDERFLOW
                                .MEXIT
                                .ENDC
     060246                     M$GETS  T$NS,\T$NESTLEV,T$TEMP
             000004             T$TEMP=T$NS1
     060246                     M$DECR  T$NESTLEV
             000000             T$NESTLEV=T$NESTLEV-1
                                .IF EQ F$HARD-T$TEMP
     060246                     M$GNINS .EVEN
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .EVEN
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
                                                                                        .EVEN
                                .ENDC
     060246                     M$GNTAG L,T$$HARD
                                .MCALL  M$GEN
     060246                     M$GEN   L,\T$$HARD,SVCTAG
                                .IF LE  SVCTAG
                                .IIF EQ SVCTAG,.LIST
                                L10061:
                                .IIF EQ SVCTAG,.NLIST
                                .MEXIT
                                .ENDC
     060246                                                                     L10061:
             010000             S$LSYM=T$LSYM
             000041             I$HRD=F$END
                                .IFF
                                M$ENDERR        ENDHRD,T$TEMP
```

```
                                        .ENDC
 7940 060246     104     105     126    HPM1:   .ASCIZ  'DEVICE ADDRESS (TSSR) '
 7941 050275     111     116     124    HPM2:   .ASCIZ  'INTERRUPT VECTOR    '
 7942 060321     111     116     124    HPM3:   .ASCIZ  'INTERRUPT PRIORITY      '
 7943                                            .EVEN
 7944
```

```
                                        .SBTTL  SOFTWARE PARAMETER CODING SECTION

                                ;++
                                ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;--
060352                                  BGNSFT
                                .MCALL  M$PUSH,M$INCR,M$GNINS,M$GNGBL
        000040                  I$SFT=F$BGN
060352                          M$PUSH  T$NS,T$NESTLEV,F$SOFT
                                .MCALL  M$INCR,M$SETS
060352                          M$INCR  T$NESTLEV
        000001                  T$NESTLEV=T$NESTLEV+1
060352                          M$SETS  T$NS,\T$NESTLEV,F$SOFT
        000005                  T$NS1=F$SOFT
        010062                  T$$SOFT=T$TAGNUM
060352                          M$INCR T$TAGNUM
        010063                  T$TAGNUM=T$TAGNUM+1
                                .IRP    TAG,<\T$$SOFT>
                                M$GNINS <.WORD L'TAG'-L$SOFT/2>
                                .ENDM
060352                          M$GNINS <.WORD L10062-L$SOFT/2>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD L10062-L$SOFT/2
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060352  000011                                                          .WORD L10062-L$SOFT/2
                                .ENDC
060354                          M$GNGBL L$SOFT
                                .MCALL  M$GEN
                                .IF NB,L$SOFT
                                .IF NB,
                                M$GEN   L$SOFT,:,SVCGBL,
                                .ENDC
                                .IF B,
060354                          M$GEN   L$SOFT,:,SVCGBL,< >
                                .IF LE  SVCGBL
                                .IIF EQ SVCGBL,.LIST
                                L$SOFT::
                                .IIF EQ SVCGBL,.NLIST
                                .MEXIT
                                .ENDC
060354                                                          L$SOFT::
                                .ENDC
                                .ENDC
7957 060354                             GPRML   SPM1,0,-1,YES           ;GET RAM DUMP FLAG
                                .MCALL  M$RADIX,M$DEFAULT,M$WORD,M$CNTOP
```

```
                000000          T$TEMP=0&1
                                .IF NE  T$TEMP & 1
                                .ERROR  ;ODD OFFSET
                                .ENDC
                                .IF LT  G$OFSIZE-0
                                .ERROR  ;OFFSET TOO BIG
                                .ENDC
                000000          T$CODE=G$PRML + <0 * G$OFFSET>
060354                          M$RADIX L,T$TEMP
                                .IF IDN B,L
                                T$TEMP=G$RADB
                                .MEXIT
                                .ENDC
                                .IF IDN O,L
                                T$TEMP=G$RADO
                                .MEXIT
                                .ENDC
                                .IF IDN D,L
                                T$TEMP=G$RADD
                                .MEXIT
                                .ENDC
                                .IF IDN L,L
                000120          T$TEMP=G$RADL
                                .MEXIT
                                .ENDC
                                .IF IDN A,L
                                T$TEMP=G$RADA
                                .MEXIT
                                .ENDC
                                T$TEMP=-1
                                .ERROR  ;ILL. RADIX "L"
                000120          T$CODE=T$CODE ! T$TEMP
060354                          M$DEFAULT       YES,T$TEMP
                                .IF IDN YES,YES
                000010          T$TEMP=G$YES
                                .MEXIT
                                .ENDC
                                .IF IDN NO,YES
                                T$TEMP=G$NO
                                .MEXIT
                                .ENDC
                                T$TEMP=-1
                                .ERROR  ;DEFAULT "YES" MUST BE "YES" OR "NO"
                000130          T$CODE=T$CODE ! T$TEMP
060354                          M$CNTOP <>,<T$CODE>
                                .IF NB
                                .IF NE  I$SFT - F$BGN
                                .ERROR  ; "COUNT" OPTION VALID ONLY IN S.W. QUES.
                                .MEXIT
                                .ENDC
                                T$CODE=T$CODE ! G$CNTOP
                                .ENDC
060354                          M$WORD  <T$CODE,SPM1,-1>
                                .MCALL  M$GNINS
                                .IRP    N,<T$CODE,SPM1,-1>
                                M$GNINS <.WORD  N>
                                .ENDM
```

```
060354                          M$GNINS <.WORD  T$CODE>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   T$CODE
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060354  000130                                                          .WORD   T$CODE
                                .ENDC
060356                          M$GNINS <.WORD  SPM1>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   SPM1
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060356  060376                                                          .WORD   SPM1
                                .ENDC
060360                          M$GNINS <.WORD  -1>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   -1
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060360  177777                                                          .WORD   -1
                                .ENDC
                                .IF NB
                                M$WORD  </2>
                                .ENDC
7958 060362                             GPRML   SPM4,2,-1,YES                   ; GET ITERATION CONTROL.
                                .MCALL  M$RADIX,M$DEFAULT,M$WORD,M$CNTOP
        000000                  T$TEMP=2&1
                                .IF NE  T$TEMP & 1
                                .ERROR  ;ODD OFFSET
                                .ENDC
                                .IF LT  G$OFSIZE-2
                                .ERROR  ;OFFSET TOO BIG
                                .ENDC
        001000                  T$CODE=G$PRML + <2 * G$OFFSET>
060362                          M$RADIX L,T$TEMP
                                .IF IDN B,L
                                T$TEMP=G$RADB
                                .MEXIT
                                .ENDC
```

```
                                .IF IDN O,L
                                T$TEMP=G$RADO
                                .MEXIT
                                .ENDC
                                .IF IDN D,L
                                T$TEMP=G$RADD
                                .MEXIT
                                .ENDC
                                .IF IDN L,L
        000120                  T$TEMP=G$RADL
                                .MEXIT
                                .ENDC
                                .IF IDN A,L
                                T$TEMP=G$RADA
                                .MEXIT
                                .ENDC
                                T$TEMP=-1
                                .ERROR  ;ILL. RADIX "L"
        001120                  T$CODE=T$CODE ! T$TEMP
060362                          M$DEFAULT       YES,T$TEMP
                                .IF IDN YES,YES
        000010                  T$TEMP=G$YES
                                .MEXIT
                                .ENDC
                                .IF IDN NO,YES
                                T$TEMP=G$NO
                                .MEXIT
                                .ENDC
                                T$TEMP=-1
                                .ERROR  ;DEFAULT "YES" MUST BE "YES" OR "NO"
        001130                  T$CODE=T$CODE ! T$TEMP
060362                          M$CNTOP <>,<T$CODE>
                                .IF NB
                                .IF NE  I$SFT - F$BGN
                                .ERROR  ; "COUNT" OPTION VALID ONLY IN S.W. QUES.
                                .MEXIT
                                .ENDC
                                T$CODE=T$CODE ! G$CNTOP
                                .ENDC
060362                          M$WORD  <T$CODE,SPM4,-1>
                                .MCALL  M$GNINS
                                .IRP    N,<T$CODE,SPM4,-1>
                                M$GNINS <.WORD  N>
                                .ENDM
060362                          M$GNINS <.WORD  T$CODE>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   T$CODE
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060362  001130                                                          .WORD   T$CODE
                                .ENDC
```

```
        060364                          M$GNINS <.WORD  SPM4>
                                        .IF LT SVCINSINSTR
                                        .MEXIT
                                        .ENDC
                                        .IF EQ SVCINS
                                        .LIST
                                                .WORD   SPM4
                                                .NLIST
                                        .MEXIT
                                        .ENDC
                                        .IF GT SVCINS
        060364  060442                                                          .WORD   SPM4
                                        .ENDC
        060366                          M$GNINS <.WORD  -1>
                                        .IF LT SVCINSINSTR
                                        .MEXIT
                                        .ENDC
                                        .IF EQ SVCINS
                                        .LIST
                                                .WORD   -1
                                                .NLIST
                                        .MEXIT
                                        .ENDC
                                        .IF GT SVCINS
        060366  177777                                                          .WORD   -1
                                        .ENDC
                                        .IF NB
                                        M$WORD  </2>
                                        .ENDC
     7959 060370                                GPRML   SPM6,4,-1,YES           ;GET EOT CHECK STATUS
                                        .MCALL  M$RADIX,M$DEFAULT,M$WORD,M$CNTOP
                000000                  T$TEMP=4&1
                                        .IF NE  T$TEMP & 1
                                        .ERROR  ;ODD OFFSET
                                        .ENDC
                                        .IF LT  G$OFSIZE-4
                                        .ERROR  ;OFFSET TOO BIG
                                        .ENDC
                002000                  T$CODE=G$PRML'+ <4 * G$OFFSET>
            060370                      M$RADIX L,T$TEMP
                                        .IF IDN B,L
                                        T$TEMP=G$RADB
                                        .MEXIT
                                        .ENDC
                                        .IF IDN O,L
                                        T$TEMP=G$RADO
                                        .MEXIT
                                        .ENDC
                                        .IF IDN D,L
                                        T$TEMP=G$RADD
                                        .MEXIT
                                        .ENDC
                                        .IF IDN L,L
                000120                  T$TEMP=G$RADL
                                        .MEXIT
                                        .ENDC
                                        .IF IDN A,L
```

```
                                T$TEMP=G$RADA
                                .MEXIT
                                .ENDC
                                T$TEMP=-1
                                .ERROR  ;ILL. RADIX "L"
        002120                  T$CODE=T$CODE ! T$TEMP
060370                          M$DEFAULT       YES,T$TEMP
                                .IF IDN YES,YES
        000010                  T$TEMP=G$YES
                                .MEXIT
                                .ENDC
                                .IF IDN NO,YES
                                T$TEMP=G$NO
                                .MEXIT
                                .ENDC
                                T$TEMP=-1
                                .ERROR  ;DEFAULT "YES" MUST BE "YES" OR "NO"
        002130                  T$CODE=T$CODE ! T$TEMP
060370                          M$CNTOP <>,<T$CODE>
                                .IF NB
                                .IF NE  I$SFT - F$BGN
                                .ERROR  ; "COUNT" OPTION VALID ONLY IN S.W. QUES.
                                .MEXIT
                                .ENDC
                                T$CODE=T$CODE ! G$CNTOP
                                .ENDC
060370                          M$WORD  <T$CODE,SPM6,-1>
                                .MCALL  M$GNINS
                                .IRP    N,<T$CODE,SPM6,-1>
                                M$GNINS <.WORD  N>
                                .ENDM
060370                          M$GNINS <.WORD  T$CODE>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   T$CODE
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060370  002130                                                          .WORD   T$CODE
                                .ENDC
060372                          M$GNINS <.WORD  SPM6>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   SPM6
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060372  060472                                                          .WORD   SPM6
                                .ENDC
```

```
      060374                        M$GNINS <.WORD  -1>
                                    .IF LT SVCINSINSTR
                                    .MEXIT
                                    .ENDC
                                    .IF EQ SVCINS
                                    .LIST
                                            .WORD   -1
                                                    .NLIST
                                    .MEXIT
                                    .ENDC
                                    .IF GT SVCINS
      060374  177777                .                                       .WORD   -1
                                    .ENDC
                                    .IF NB
                                    M$WORD  </2>
                                    .ENDC
 7960 060376                        ENDSFT
                                    .MCALL  M$POP,M$GNINS,M$GNTAG,M$ENDERR
      060376                        M$POP   T$NS,T$NESTLEV,T$TEMP
                                    .MCALL  M$GETS,M$DECR
                                    .IF LT  T$NESTLEV
                                    .ERROR  T$NESTLEV       ; MACRO T$NS UNDERFLOW
                                    .MEXIT
                                    .ENDC
      060376                        M$GETS  T$NS,\T$NESTLEV,T$TEMP
              000005                T$TEMP=T$NS1
      060376                        M$DECR  T$NESTLEV
              000000                T$NESTLEV=T$NESTLEV-1
                                    .IF EQ F$SOFT-T$TEMP
      060376                        M$GNINS .EVEN
                                    .IF LT SVCINSINSTR
                                    .MEXIT
                                    .ENDC
                                    .IF EQ SVCINS
                                    .LIST
                                            .EVEN
                                                    .NLIST
                                    .MEXIT
                                    .ENDC
                                    .IF GT SVCINS
                                    .                                               .EVEN
                                    .ENDC
      060376                        M$GNTAG L,T$$SOFT
                                    .MCALL  M$GEN
      060376                        M$GEN   L,\T$$SOFT,SVCTAG
                                    .IF LE  SVCTAG
                                    .IIF EQ SVCTAG,.LIST
                                    L10062:
                                    .IIF EQ SVCTAG,.NLIST
                                    .MEXIT
                                    .ENDC
      060376                                                                    L10062:
              010000                S$LSYM=T$LSYM
              000041                I$SFT=F$END
                                    .IFF
                                    M$ENDERR        ENDSFT,T$TEMP
                                    .ENDC
```

```
                                                SPM1:   .ASCIZ  'ENABLE CONTROLLER RAM DUMP ON ERROR'
                                                SPM4:   .ASCIZ  'INHIBIT ITERATIONS     '

                                                SPM6:   .ASCIZ  'INHIBIT EOT CHECKING (REDUCES RUN TIME BY 22 MINUTES)'
                                                        .EVEN
                                                        .SBTTL  PATCH AREA
                                                ;+
                                                ;DISPATCH TABLE
                                                ;
                                                ; *** MCVE TO FRONT OF PROGRAM FOR RELEASE ***
                                                ;-

                                                        DISPATCH        TESTNO
                                                .MCALL  M$WORD,M$GNGBL
                                                .RADIX  10
                                                M$WORD \TESTNO
                                                .MCALL  M$GNINS
                                                .IRP    N,<5>
                                                M$GNINS <.WORD  N>
                                                .ENDM
                                                M$GNINS <.WORD  5>
                                                .IF LT SVCINSINSTR
                                                .MEXIT
                                                .ENDC
                                                .IF EQ SVCINS
                                                .LIST
                                                        .WORD   5
                                                                .NLIST
                                                .MEXIT
                                                .ENDC
                                                .IF GT SVCINS
                                                                                        .WORD   5
                                                .ENDC
                                                M$GNGBL L$DISPATCH
                                                .MCALL  M$GEN
                                                .IF NB,L$DISPATCH
                                                .IF NB,
                                                M$GEN   L$DISPATCH,:,SVCGBL,
                                                .ENDC
                                                .IF B,
                                                M$GEN   L$DISPATCH,:,SVCGBL,< >
                                                .IF LE  SVCGBL
                                                .IIF EQ SVCGBL,.LIST
                                                L$DISPATCH::
                                                .IIF EQ SVCGBL,.NLIST
                                                .MEXIT
                                                .ENDC
                                                                                L$DISPATCH::
                                                .ENDC
                                                .ENDC
                                                T$TEMP=1
                                                .REPT   TESTNO
                                                .IRP    N,<\T$TEMP>
                                                M$WORD  T'N
                                                .ENDM
```

```
                                T$TEMP=T$TEMP + 1
                                .ENDR
                                .IRP    N,<\T$TEMP>
                                M$WORD  T'N
                                .ENDM
060562                          M$WORD  T1
                                .MCALL  M$GNINS
                                .IRP    N,<T1>
                                M$GNINS <.WORD  N>
                                .ENDM
060562                          M$GNINS <.WORD  T1>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   T1
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060562  023642                                                          .WORD   T1
                                .ENDC
        000002                  T$TEMP=T$TEMP + 1
                                .IRP    N,<\T$TEMP>
                                M$WORD  T'N
                                .ENDM
060564                          M$WORD  T2
                                .MCALL  M$GNINS
                                .IRP    N,<T2>
                                M$GNINS <.WORD  N>
                                .ENDM
060564                          M$GNINS <.WORD  T2>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   T2
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060564  032326                                                          .WORD   T2
                                .ENDC
        000003                  T$TEMP=T$TEMP + 1
                                .IRP    N,<\T$TEMP>
                                M$WORD  T'N
                                .ENDM
060566                          M$WORD  T3
                                .MCALL  M$GNINS
                                .IRP    N,<T3>
                                M$GNINS <.WORD  N>
                                .ENDM
060566                          M$GNINS <.WORD  T3>
                                .IF LT SVCINSINSTR
                                .MEXIT
```

```
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD   T3
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
060566  041322                                                          .WORD   T3
                        .ENDC
        000004          T$TEMP=T$TEMP + 1
                        .IRP    N,<\T$TEMP>
                        M$WORD  T'N
                        .ENDM
060570                  M$WORD  T4
                        .MCALL  M$GNINS
                        .IRP    N,<T4>
                        M$GNINS <.WORD  N>
                        .ENDM
060570                  M$GNINS <.WORD  T4>
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD   T4
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
060570  046640                                                          .WORD   T4
                        .ENDC
        000005          T$TEMP=T$TEMP + 1
                        .IRP    N,<\T$TEMP>
                        M$WORD  T'N
                        .ENDM
060572                  M$WORD  T5
                        .MCALL  M$GNINS
                        .IRP    N,<T5>
                        M$GNINS <.WORD  N>
                        .ENDM
060572                  M$GNINS <.WORD  T5>
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD   T5
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
060572  053000                                                          .WORD   T5
                        .ENDC
        000006          T$TEMP=T$TEMP + 1
        000010          .RADIX  8
7976
```

```
                        ;
                        ; FINALLY A GENEROUS PATCH AREA.
                        ;
                        ; AND AN ADJUSTMENT TO ACCOUNT FOR THE "LASTAD BIT7" HACK
                        ; DESCRIBED IN "SUPPRG.MEM" (FOR REV C).
                        ;
060574                  PATCH::
                        ;       .IF     NZ,.&377
                        ;       .=.!377+1
                        ;       .ENDC
060574                          LASTAD          ;SET LAST USED ADDRESS.
                        .MCALL  M$GNINS,M$GNGBL
060574                  M$GNINS .EVEN
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .EVEN
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
                                                                        .EVEN
                        .ENDC
000001                  T$LAST=1
                        .IF EQ 0$SETUP
                        M$WORD  <0>
                        M$WORD  <0>
                        .IFF
060574                  M$GNINS <.WORD T$FREE>
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD T$FREE
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
060574  060612                                                        .WORD T$FREE
                        .ENDC
060576                  M$GNINS <.WORD T$SIZE>
                        .IF LT SVCINSINSTR
                        .MEXIT
                        .ENDC
                        .IF EQ SVCINS
                        .LIST
                                .WORD T$SIZE
                                        .NLIST
                        .MEXIT
                        .ENDC
                        .IF GT SVCINS
060576  000005                                                        .WORD T$SIZE
                        .ENDC
```

```
                                .ENDC
                 000000         SVCGBL=0
           050600               M$GNGBL L$LAST
                                .MCALL  M$GEN
                                .IF NB,L$LAST
                                .IF NB,
                                M$GEN   L$LAST,:,SVCGBL,
                                .ENDC
                                .IF B,
           060600               M$GEN   L$LAST,:,SVCGBL,< >
                                .IF LE  SVCGBL
           060600               L$LAST::
                                .MEXIT
                                .ENDC
                                .LIST
                                                                        L$LAST::
                                        .NLIST
                                .ENDC
                                .ENDC
                 000005         T$LTNO=T$TESTNUM
      7989                              .SBTTL  HARD CODED P-TABLE
      7990                      ;++
      7991                      ;       DIAGNOSTIC IS PRE-PARAMETERIZED PER THIS TABLE
      7992                      ;--
      7993 060600                       BGNSETUP        1
                                .MCALL  M$INCR
                                .IF NE  F$END - I$SETUP
                                .ERROR  ; ONLY 1 SETUP!
                                .MEXIT
                                .ENDC
                                .IF NE  T$LAST-1
                                .ERROR  ; "LASTAD" MUST PRECEDE "BGNSETUP"
                                .MEXIT
                                .ENDC
                 000040         I$SETUP=F$BGN
                 010063         T$$PTAB=T$TAGNUM
           060600               M$INCR   T$TAGNUM
                 010064         T$TAGNUM=T$TAGNUM+1
                                .IF NB 1
                 000001         T$PCNT=1
                 000001         T$$PC=1
                                .IFF
                                .ERROR  ; MISSING # OF PTABLES
                                T$PCNT=1
                                .ENDC
      7994 060600                       BGNPTAB
                                .MCALL M$INCR,M$DECR,M$WORD,M$GNINS,M$GNTAG
                                .IF NE F$BGN-I$SETUP
                                .ERROR  ; MISSING "BGNSETUP"
                                .MEXIT
                                .ENDC
                                .IF NE F$END-I$PTAB
                                .ERROR  ; MISSING "ENDPTAB"
                                .MEXIT
                                .ENDC
                 000040         I$PTAB=F$BGN
                 010064         T$PTAB=T$TAGNUM
```

```
060600                          M$INCR   T$TAGNUM
        010065                  T$TAGNUM=T$TAGNUM+1
050600                          M$DECR   T$PCNT
        000000                  T$PCNT=T$PCNT-1
                                .IF LE T$PCNT
060600                          M$WORD <0>
                                .MCALL  M$GNINS
                                .IRP    N,<0>
                                M$GNINS <.WORD  N>
                                .ENDM
060600                          M$GNINS <.WORD  0>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   0
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060600  000000                                                          .WORD   0
                                .ENDC
                                .IFF
                                .IRP N,<\T$PTAB>
                                M$GNINS <.WORD  L'N>
                                .ENDM
                                .ENDC
        010065                  T$$DAT=T$TAGNUM
060602                          M$INCR   T$TAGNUM
        010066                  T$TAGNUM=T$TAGNUM+1
                                .IRP    N,<\T$$DAT>
                                M$GNINS <.WORD  L'N-./2-1>
                                .ENDM
060602                          M$GNINS <.WORD  L10065-./2-1>
                                .IF LT SVCINSINSTR
                                .MEXIT
                                .ENDC
                                .IF EQ SVCINS
                                .LIST
                                        .WORD   L10065-./2-1
                                                .NLIST
                                .MEXIT
                                .ENDC
                                .IF GT SVCINS
060602  000003                                                          .WORD   L10065-./2-1
                                .ENDC
060604                          M$GNTAG L,T$$PTAB
                                .MCALL  M$GEN
060604                          M$GEN   L,\T$$PTAB,SVCTAG
                                .IF LE  SVCTAG
                                .IIF EQ SVCTAG,.LIST
                                L10063:
                                .IIF EQ SVCTAG,.NLIST
                                .MEXIT
                                .ENDC
060604                                                          L10063:
```

```
                010064          T$$PTAB=T$PTAB
          060604                M$INCR  T$PTNUM
                000001          T$PTNUM=T$PTNUM+1
     7995 060604  172522                .WORD           172522
     7996 060606  000224                .WORD           224
     7997 060610  000240                .WORD           PRI05
     7998 060612                        ENDPTAB
                                .MCALL  M$GNTAG
                                .IF NE  F$BGN - I$PTAB
                                .ERROR  ; MISSING "BGNPTAB"
                                .MEXIT
                                .ENDC
                000041          I$PTAB=F$END
          060612                M$GNTAG L,T$$DAT
                                .MCALL  M$GEN
          060612                M$GEN   L,\T$$DAT,SVCTAG
                                .IF LE  SVCTAG
                                .IIF EQ SVCTAG,.LIST
                                L10065:
                                .IIF EQ SVCTAG,.NLIST
                                .MEXIT
                                .ENDC
          060612                                                  L10065:
     7999 060612                        ENDSETUP
                                .IF NE  F$BGN - I$SETUP
                                .ERROR  ; MISSING "BGNSETUP"
                                .MEXIT
                                .ENDC
                000041          I$SETUP=F$END
                                .IF NE  T$$PC - T$PTNUM
                                .ERROR  ; PTABLE COUNT INCORRECT
                                .ENDC
                060612          T$FREE=.
                000005          T$SIZE=.-L$LAST / 2
                000001          T$PTHV=T$PTNUM
     8000
     8001       000001                  .END
```

```
ADDSSR  011772 G
ADR   = 000020 G
AMBTSS  006332
ASSEMB= 000010
A1716 = 000003
BADDAT  003114 G
BADSSR  016554 G
BAR   = 174402
BENBSW  002176 G
BIE   = 040000
BIT0  = 000001 G
BIT00 = 000001 G
BIT01 = 000002 G
BIT02 = 000004 G
BIT03 = 000010 G
BIT04 = 000020 G
BIT05 = 000040 G
BIT06 = 000100 G
BIT07 = 000200 G
BIT08 = 000400 G
BIT09 = 001000 G
BIT1  = 000002 G
BIT10 = 002000 G
BIT11 = 004000 G
BIT12 = 010000 G
BIT13 = 020000 G
BIT14 = 040000 G
BIT15 = 100000 G
BIT2  = 000004 G
BIT3  = 000010 G
BIT4  = 000020 G
BIT5  = 000040 G
BIT6  = 000100 G
BIT7  = 000200 G
BIT8  = 000400 G
BIT9  = 001000 G
BOE   = 000400 G
BRINIT  004361
BSEL0 = 000000
BSEL1 = 000001
CHKAMB  016720
CHKMAN  021406 G
CHKTSS  017240
CKDROP  020156
CKEMAX  020004
CKMSG   011212 G
CKMSG2  011332 G
CKRAM   010534 G
CKRAM2  011110 G
CMPMEM  020562
CONFIG  020224
COUNT   002256 G
CSR   = 174400
CSRADD  002156 G
CTAB    003122 G
CTABE   003134 G
CTABM   003122 G
C$AU  = 000052
C$AUTO= 000061
C$BRK = 000022
C$BSEG= 000004
C$BSUB= 000002
C$CEFG= 000045
C$CLCK= 000062
C$CLEA= 000012
C$CLOS= 000035
C$CLP1= 000006
C$CVEC= 000036
C$DCLN= 000044
C$DODU= 000051
C$DRPT= 000024
C$DU  = 000053
C$EDIT= 000003
C$ERDF= 000055
C$ERHR= 000056
C$ERRO= 000060
C$ERSF= 000054
C$ERSO= 000057
C$ESCA= 000010
C$ESEG= 000005
C$ESUB= 000003
C$ETST= 000001
C$EXIT= 000032
C$GETB= 000026
C$GETW= 000027
C$GMAN= 000043
C$GPHR= 000042
C$GPLO= 000030
C$GPRI= 000040
C$INIT= 000011
C$INLP= 000020
C$MANI= 000050
C$MEM = 000031
C$MSG = 000023
C$OPEN= 000034
C$PNTB= 000014
C$PNTF= 000017
C$PNTS= 000016
C$PNTX= 000015
C$QIO = 000377
C$RDBU= 000007
C$REFG= 000047
C$RESE= 000033
C$REVI= 000003
C$RFLA= 000021
C$RPT = 000025
C$SEFG= 000046
C$SPRI= 000041
C$SVEC= 000037
C$TPRI= 000013
DAR   = 174404
DATA    002260 G
DATAFL  015070
DATASC  021140
DEBUGM  011464
DEVCNT  002170 G
DEVDRO  023572
DEVNRD  023511
DEVNXR  023427
DEVONL  023357
DEVSUM  023322
DFPTBL  002124 G
DIAGMC= 000000
DLCYL = 000177
DLDNER= 100200
DLERR = 177730
DLGETS= 000004
DLRDHD= 000010
DLRDNH= 000016
DLSR  = 000013
DLUN  = 000006
DSBINT  017060
DUAD12  004545
DUFLG   003064 G
DUMMY   003034
EF.CON= 000036 G
EF.NEW= 000035 G
EF.PWR= 000034 G
EF.RES= 000037 G
EF.STA= 000040 G
EMAXDU  017701
EN    = 000000
ENAINT  017026
ENVIRN  021536
EOTSEL  002140 G
EPRTSW  002150 G
EPRT1   005676
EPRT2   005737
EPRT3   006021
ERCM    011565
ERRHI   002204 G
ERRK    017660
ERRLO   002206 G
ERRNO = 001021
ERRVEC= 000004 G
ERTABE  003334
ERTABL  003134
ESUM    017662
EVL   = 000004 G
EWCHK   060020
EWMSG   055606
EXBCNT= 000010
EXPBRE  016356 G
EXPD    002200 G
EXPGOT  004435
EXPGT2  004471
EXPMSG  002270 G
EXPREC  016350 G
EXTA    005236
EXTEND  005234
E$END = 002100
E$LOAD= 000035
FATCHK  020104
FATERR= 000060
FATFLG  002172 G
FAULTM  052527
FERCM   011554
FIFEXP  012022 G
FIF1MS  012074
FIF2MS  012143
FILLME  020376
FLLTSW  002722 G
FNOINT  004117
FORCER  002146 G
FREE    003076 G
FREEHI  003102
FRESIZ  003100 G
FUSI    004021
F$AU  = 000015
F$AUTO= 000020
F$BGN = 000040
F$CLEA= 000007
F$DU  = 000016
F$END = 000041
F$HARD= 000004
F$HW  = 000013
F$INIT= 000006
F$JMP = 000050
F$MOD = 000000
F$MSG = 000011
F$PROT= 000021
F$PWR = 000017
F$RPT = 000012
F$SEG = 000003
F$SOFT= 000005
F$SRV = 000010
F$SUB = 000002
F$SW  = 000014
F$TEST= 000001
GDDAT   003116 G
GERRMA  002144 G
GETPAT  021102 G
GETSEL  021164 G
G$CNTO= 000200
G$DELM= 000372
G$DISP= 000003
G$EXCP= 000400
G$HILI= 000002
G$LOLI= 000001
G$NO  = 000000
G$OFFS= 000400
G$OFSI= 000376
G$PRMA= 000001
G$PRMD= 000002
G$PRML= 000000
G$RADA= 000140
G$RADB= 000000
G$RADD= 000040
G$RADL= 000120
G$RADO= 000020
G$XFER= 000004
G$YES = 000010
HIADDR= 001400
HIMEM = 007776
HOE   = 100000 G
HPM1    060246
HPM2    060275
HPM3    060321
IBE   = 010000 G
IDU   = 000040 G
IER   = 020000 G
IFAULT  004160
INCERK  017746
INTCPC  017024
INTFLA  017021
INTMAS  017020
INTR    017072 G
INTREC  002174 G
INTVEC  017022
INTX    004202
IOKCKI= 000200
IOKSTP= 000001
IPRI    002162 G
ISR   = 000100 G
IVEC    002160 G
IXE   = 004000 G
I$AU  = 000041
I$AUTO= 000041
I$CLN = 000041
I$DU  = 000041
I$HRD = 000041
I$INIT= 000041
I$MOD = 000040
I$MSG = 000041
I$PROT= 000040
I$PTAB= 000041
I$PWR = 000041
I$RPT = 000041
I$SEG = 000041
I$SETU= 000041
I$SFT = 000041
I$SRV = 000041
I$SUB = 000041
I$TST = 000041
J$JMP = 000167
KIPAR0= 172340
KIPAR1= 172342
KIPAR2= 172344
KIPAR3= 172346
KIPAR4= 172350
KIPAR5= 172352
KIPAR6= 172354
KIPAR7= 172356
KIPDR0= 172300
KIPDR1= 172302
```

```
KIPDR2= 172304          L$PROT  021756 G        L10054  047546          O$DU  = 000001          PST32W  003110 G
KIPDR3= 172306          L$PRT   002112 G        L10055  050334          O$ERRT= 000000          PUNIT   022504
KIPDR4= 172310          L$REPP  002062 G        L10056  051122          O$GNSW= 000001          PW.D11= 000021
KIPDR5= 172312          L$REV   002010 G        L10057  060210          O$POIN= 000001          PW.D13= 000022
KIPDR6= 172314          L$RPT   023060 G        L10060  055372          O$SETU= 000001          PW.D22= 000020
KIPDR7= 172316          L$SOFT  060354 G        L10061  060246          PASRPT  022250          PW.NOP= 000000
KTENAB  003106 G        L$SPC   002056 G        L10062  060376          PATCH   060574 G        PW.NO1= 000023
KTFLG   003104 G        L$SPCP  002020 G        L10063  060604          PATDAT  021136          PW.RDE= 000024
KTINIT  021624           L$SPTP  002024 G        L10065  060612          PC.ERA= 002400          PW.RDR= 000001
KTOFF   020250           L$STA   002030 G        MEMADD  013606 G        PC.IER= 002000          PW.RDS= 000005
KTON    020232           L$SW    002134 G        MENASC  021355          PC.NOO= 001000          PW.RFI= 000003
LERRMA  002142 G        L$TEST  002114 G        MENERR  021302          PC.REL= 000000          PW.WCT= 000006
LISTAL= 000001           L$TIML  002014 G        MENRES  021404          PC.REW= 000400          PW.WFI= 000004
LOE   = 040000 G        L$UNIT  002012 G        MESBFA  002720 G        PKBCNT= 000006          PW.WFM= 000007
LOOPCN  002166 G        L10000  002132          MESBFN  014640          PKHI  = 000004          PW.WMI= 000010
LOOPCO  012760           L10001  002146          MESHEA  015023          PKLOW = 000002          PW.WNP= 000011
LOOPFL  003120 G        L10002  005232          MMVEC = 000250          PKTADD  007272          PW.WTR= 000002
LOT   = 000010 G        L10003  011676          MPR   = 174406          PKTFRM  007234          P.ACK = 100000
L$ACP   002110 G        L10004  011726          MSA.FR= 000006          PKTGET  011730 G        P.CMD = 000037
L$APT   002036 G        L10005  011744          MSA.NO= 000000          PKTMES  011754 G        P.CONT= 000012
L$AU    022552 G        L10006  011752          MSA.NR= 000004          PKTNEW  007327           P.CVC = 040000
L$AUT   002070 G        L10007  011770          MSA.VO= 000002          PKTRAM  004647 G        P.FMT = 000140
L$AUTO  022756 G        L10010  012006          MSGEXP  012010 G        PKTSSR  011700 G        P.FORM= 000011
L$CCP   002106 G        L10011  012020          MSGLOO  012716 G        PNT   = 001000 G        P.GETS= 000017
L$CLEA  023032 G        L10012  012072          MSGSTA  012202 G        PRAMPK  013630          P.IE  = 000200
L$CO    002032 G        L10013  012242          MSGSUB  013574 G        PRBEXP  016344          P.INIT= 000013
L$DEPO  002011 G        L10014  012756          MS.ATT= 000006          PRBMSG  016212          P.MODE= 007400
L$DESC  003346 G        L10015  013604          MS.EXT= 000200          PRBREC  016346          P.OPP = 020000
L$DESP  002076 G        L10016  013626          MS.RSD= 000001          PRBTOT  016277          P.POSI= 000010
L$DEVP  002060 G        L10017  016354          MS.RSF= 000020          PRBYTE  015776 G        P.READ= 000001
L$DISP  060562 G        L10020  016362          MS.RST= 000010          PRI   = 002000 G        P.SWB = 010000
L$DLY   002116 G        L10021  016370          NBA   = 002000          PRIADD  007706           P.WRIT= 000005
L$DTP   002040 G        L10022  016402          NEWPAS  022204          PRIAO   007756           P.WRTC= 000004
L$DTYP  002034 G        L10023  016424          NODEV   003066 G        PRIBXO  007340 G        P.WRTS= 000006
L$DU    022650 G        L10024  016452          NOINIT  004237          PRIEQU  007606           QVP     002154 G
L$DUT   002072 G        L10025  016612          NOINTR  004123          PRIPKT  007066 G        RAMASC  013776
L$DVTY  003340 G        L10026  017122          NOITS   002136 G        PRIRAM  007614           RAMDAT  002210 G
L$EF    002052 G        L10030  022502          NOMAN   021442          PRITAD  010022           RAMER   010636 G
L$ENVI  002044 G        L10031  022646          NP.IR = 000200          PRITSS  005270           RAMERR  016364 G
L$ETP   002102 G        L10032  022754          NP.LOO= 000040          PRITO   010072           RAMEXP  016404 G
L$EXP1  002046 G        L10033  023030          NP.OUT= 000100          PRIXOR  007470 G        RAMFHR  014542
L$EXP4  002064 G        L10034  023056          NP.WRP= 000020          PRIO0 = 000000 G        RAMFOR  007644
L$EXP5  002066 G        L10035  023320          NSI     004054          PRIO1 = 000040 G        RAMHLD  011020
L$HARD  060214 G        L10036  032324          NSINIT  004311          PRIO2 = 000100 G        RAMIOP  011024
L$HIME  002120 G        L10037  024306          NUL     004431          PRIO3 = 000140 G        RAMPD   011075
L$HPCP  002016 G        L10040  024762          NULCR   004432          PRIO4 = 000200 G        RAMR5H  011022
L$HPTP  002022 G        L10041  025464          NXM   = 004000          PRIO5 = 000240 G        RAMSIZ  002250 G
L$HW    002124 G        L10042  026330          NXR     003642          PRIO6 = 000300 G        RAMTAD  016372 G
L$ICP   002104 G        L10043  041320          NXRERR  005202 G        PRIO7 = 000340 G        RBPCRA  015135
L$INIT  021766 G        L10044  033722          NXRX    003701          PRMESS  014062          RCVHIA  002252 G
L$LADP  002026 G        L10045  035316          NXTU    022216          PRMNO   002266 G        RCVLOA  002254 G
L$LAST  060600 G        L10046  035670          OFL   = 000100          PRMSGE  015426 G        RDERR   005110
L$LOAD  002100 G        L10047  036332          ONEFIL= 000000          PRMSG0  015606           READ  = 000014
L$LUN   002074 G        L10050  046636          O$APTS= 000000          PRMSG1  015653           READY = 000001
L$MREV  002050 G        L10051  042214          O$AU  = 000001          PRMSG2  015711           RECMSG  002434 G
L$NAME  002000 G        L10052  043004          O$BGNR= 000001          PROASC  014720           RECV    002202 G
L$PRIO  002042 G        L10053  052776          O$BGNS= 000001          PR1ASC  014765           REGSAV  021042
```

```
REWIND  010434 G      S1.I1R= 020000        TTIVEC= 000060 G      T2      032326 G      T29WDF  027421
RMCHBE= 000167        S1.I2R= 040000        TTOBFR= 177566        T2.1    032362        T29WDR  026530
RMCHEN= 000200        S1.PAR= 100000        TTOCSR= 177564        T2.2    033724        T29WNG  026573
RMMSGB= 000104        S2.ATI= 000010        TUV2A   002000 G      T2.3    035320        T29WRT  027714
RMMSGE= 000117        S2.BTI= 000004        T$ARGC= 000001        T2.4    035672        T29WSS  031046
RMPKTB= 000020        S2.DIM= 000200        T$CODE= 002130        T29AM3  030412        T3      041322 G
RMPKTE= 000027        S2.ILW= 000100        T$ERRN= 001021        T29BA   030754        T3.1    041362
RMR   = 010000        S2.INR= 000020        T$EXCP= 000000        T29BFR  026400        T3.2    042216
RWPACK  010530        S2.OUT= 000040        T$FLAG= 000040        T29BF2  026520        T30BFR  036400
SC    = 100000        S2.UND= 000003        T$FREE= 060612        T29BOT  027761        T30BF2  036520
SCE   = 020000        TBLEND= 003034 G      T$GMAN= 000000        T29BSO  026520        T30BOT  037731
SCME    004715        TCOASC  006173        T$HILI= 000007        T29BS1  026521        T30BSO  036520
SDELAY  010330        TCOCOD  006374        T$LAST= 000001        T29CNT  026544        T30BS1  036521
SEEK  = 000006        TEMP1   003070 G      T$LOLI= 000000        T29CON  026532        T30CNT  036540
SELASC  021350        TEMP2   003072 G      T$LSYM= 010000        T29DAT  026370        T30CNU  036542
SELDAT= 000004        TERCLS= 000016        T$LTNO= 000005        T29DLY  026550        T30DAT  036370
SEL2  = 000002        TESTNO= 000005        T$NEST= 000000        T29DTA  030026        T30DLY  036546
SETMAP  020272        TEXASC  006132        T$NSO = 000000        T29EOT  030114        T30DTA  041024
SETU    022302        TFCASC  006234        T$NS1 = 000005        T29LON  031135        T30DTR  040760
SFFMSG  011746 G      TIMEXP  016426 G      T$NS2 = 000002        T29LOO  023702        T30ETM  036376
SFHERR  003607        TIMSGO  016454        T$PCNT= 000000        T29LOP  031217        T30FCN  036544
SFIERR  003554        TINERR  011653        T$PTAB= 010064        T29LOQ  027476        T30IBT  036721
SFIMSG  011666 G      TKB   = 177562        T$PTHV= 000001        T29LOR  027351        T30IBU  036550
SFPTBL  002134 G      TKS   = 177560        T$PTNU= 000001        T29NEF  026700        T30IMV  036526
SIFLAG  003112 G      TMPBFR  002600 G      T$SAVL= 177777        T29NEQ  031455        T30LOO  032362
SIMSG   011620        TNAM    017606        T$SEGL= 177777        T29OFL  026552        T30LOQ  037520
SKIPT   003336        TPB   = 177566        T$SIZE= 000005        T29PAC  026360        T30NEF  040466
SOFINI  016650 G      TPS   = 177564        T$SUBN= 000001        T29PBP  031301        T30OFL  040177
SPACE   010134 G      TRANST  002134 G      T$TAGL= 177777        T29PK2  026470        T30PAC  036360
SPM1    060376        TSBA  = 177776 G      T$TAGN= 010066        T29PK3  026510        T30PK2  036470
SPM4    060442        TSBAH = 177777 G      T$TEMP= 000006        T29RB   026512        T30PK3  036510
SPM6    060472        TSBAL = 177776 G      T$TEST= 000005        T29RDF  026770        T30PTB  037132
SR0   = 177572        TSDB  = 177776 G      T$TSTM= 177777        T29RDG  031553        T30RB   036512
SR1   = 177574        TSDBH = 177777 G      T$TSTS= 000001        T29RES  032140        T30RDF  037303
SR2   = 177576        TSDBL = 177776 G      T$$AU = 010031        T29RIB  031716        T30RDG  037361
SR3   = 172516        TSFCOD  006734        T$$AUT= 010033        T29RN   026526        T30RES  041142
SSR   = 000200        TSREJ = 000006        T$$CLE= 010034        T29RNC  030337        T30RIB  036635
STATCO  012244        TSSDEF  006303        T$$DAT= 010065        T29RRF  027037        T30RN   036526
SVCGBL= 000000        TSSR  = 000000 G      T$$DU = 010032        T29RRG  027153        T30RRM  040545
SVCINS= 000001        TSSRBI  003404 G      T$$HAR= 010061        T29RRN  032016        T30RRN  040623
SVCSUB= 000001        TSSRFO  006112        T$$HW = 010000        T29RSZ  026546        T30RRP  040702
SVCTAG= 000001        TSSRH = 000001 G      T$$INI= 010030        T29RT2  032232        T30RT2  041234
SVCTST= 000001        TSSX    003722        T$$MSG= 010025        T29RT3  032274        T30RT3  041276
S$LSYM= 010000        TSTBLK  002724 G      T$$PC = 000001        T29RWN  030270        T30RWN  040130
S0.IDB= 000010        TSTCNT  002164 G      T$$PRO= 010027        T29SC   027267        T30SKM  037004
S0.IFB= 000002        TSTEND  017622        T$$PTA= 010064        T29SDG  031634        T30SSR  037601
S0.IFP= 000001        TSTFLA  002262 G      T$$RPT= 010035        T29SSR  027557        T30SZ   036516
S0.ILD= 000020        TSTLOO  017360 G      T$$SOF= 010062        T29SZ   026516        T30S2   036522
S0.ION= 000040        TSTPTR  002264 G      T$$SRV= 010026        T29S2   026522        T30S3   036524
S0.IRD= 000100        TSTSET  017412 G      T$$SUB= 010060        T29S3   026524        T30TM   037776
S0.IRW= 000004        TST29I  032111        T$$SW = 010001        T29TM   030212        T30TMK  040404
S0.ISP= 000200        TST30I  041121        T$$TES= 010057        T29TRL  031367        T30TM2  040053
S1.ICE= 002000        TST31I  046413        T1      023642 G      T29VCK  030701        T30TPB  037223
S1.IEO= 010000        TST32I  052470        T1.1    023702        T29WB   026512        T30VCK  040331
S1.IFM= 001000        TST34I  057224        T1.2    024310        T29WDC  030607        T30WB   036512
S1.IHE= 000400        TTIBFR= 177562 G      T1.3    024764        T29WDD  030500        T30WDC  040252
S1.IID= 004000        TTICSR= 177560 G      T1.4    025466        T29WDE  027632        T30WDD  037060
```

```
T30WDE  037652        T31TRL  045622        T34BOT  056232        T4.3    050336        XSONEF= 002000
T30WDF  037443        T31TSA  046106        T34BS0  055570        T5      053000 G      XSOONL= 000100
T31AM3  044666        T31VCK  045153        T34BS1  055571        T5.1    053040        XSOPED= 000010
T31BA   045226        T31WB   043162        T34CNT  055562        UAM   = 000200 G      XSORLL= 010000
T31BFR  043050        T31WDC  045100        T34CON  055602        UNITN   002152 G      XSORLS= 040000
T31BF2  043170        T31WDD  045010        T34DAT  055430        UNREC = 000006        XSOTMK= 100000
T31BOT  044215        T31WDE  044103        T34DLY  055564        USI     004025        XSOVCK= 000020
T31BS0  043170        T31WDF  043711        T34EOT  057146        WAITF   017124 G      XSOWLE= 004000
T31BS1  043171        T31WDR  043200        T34ET   057057        WC.IFA= 000200        XSOWLK= 000004
T31CNT  043206        T31WNG  043341        T34ETC  056155        WC.IFE= 000002        XS1CON  015247
T31CNU  043210        T31WNH  043260        T34ETN  056516        WC.IGO= 000001        XS2CON  015314
T31CON  043202        T31WRF  046213        T34ETO  055764        WC.IRE= 000010        XS3CON  015361
T31DAT  043040        T31WSS  045301        T34ETS  056601        WC.IRW= 000004        XXCOMM  003074 G
T31DLY  043212        T32AM3  051577        T34ETZ  056667        WC.IOT= 000100        X$ALWA= 000000
T31DTA  046316        T32BA   051713        T34ET2  056430        WC.I1T= 000040        X$FALS= 000040
T31EOT  044410        T32BFR  051170        T34LOO  053040        WC.I5R= 000020        X$OFFS= 000400
T31LON  045370        T32BOE  052216        T34PAC  055420        WF.IED= 000010        X$TRUE= 000020
T31LOO  041362        T32BOT  051346        T34PK2  055530        WF.IER= 000004        X1.COR= 020000
T31LOP  045452        T32CMD  051310        T34PK3  055550        WF.IHI= 000200        X1.DLT= 100000
T31LOQ  043766        T32CNT  051340        T34POS  055676        WF.IRE= 000040        X1.MBZ= 017375
T31LOR  043641        T32CNU  051342        T34RB   055552        WF.IWF= 000020        X1.RBP= 000400
T31NEF  045710        T32DAT  051160        T34RES  057726        WF.IWR= 000100        X1.SPA= 040000
T31OFL  044735        T32DLY  051344        T34RRE  056054        WF.I3R= 000002        X1.UNC= 000002
T31PAC  043030        T32ECF  052305        T34RRF  057600        WF.I4R= 000001        X2.BUF= 000100
T31PBP  045534        T32EOT  051441        T34RSZ  055560        WRTCHR  010332 G      X2.EXT= 000200
T31PK2  043140        T32ERA  051646        T34RT2  060122        WRTERR  005015        X2.OPM= 100000
T31PK3  043160        T32LOO  046732        T34RT3  060164        WRTMSG  004760        X2.RCE= 040000
T31RB   043162        T32OPI  052433        T34RWN  057246        XFERAS  016614        X2.REV= 000077
T31RDE  043214        T32PAC  051150        T34STE  057421        XNXM    017300        X2.SPA= 035400
T31RDF  043413        T32PK2  051260        T34STM  057325        XORBFO  007422        X2.UNI= 000007
T31RES  046460        T32PK3  051300        T34SZ   055556        XORFOR  007540        X2.WCF= 002000
T31RN   043176        T32RB   051302        T34S2   055572        XST0  = 000006 G      X3.DCK= 000010
T31RNC  044613        T32RES  052632        T34S3   055574        XST1  = 000010 G      X3.MBZ= 000006
T31RRF  043462        T32RIB  051766        T34TMK  056751        XST2  = 000012 G      X3.MDE= 177400
T31RT2  046552        T32RT2  052724        T34TMN  057515        XST3  = 000014 G      X3.OPI= 000100
T31RT3  046614        T32RT3  052754        T34TRK  055566        XST4  = 000016 G      X3.REV= 000040
T31RWN  044544        T32RWN  051530        T34WB   055552        XSOBOT= 000002        X3.RIB= 000001
T31SC   043557        T32SCF  052064        T34WD   055576        XSOCON  015202        X3.SPA= 000200
T31SCF  046031        T32SZ   051306        T34WDR  055600        XSOEOT= 000001        X3.TRF= 000020
T31SSR  044047        T32TSA  052141        T34WOL  057652        XSOIE = 000040        X4.HSP= 100000
T31SZ   043166        T32WB   051302        T34WTM  056341        XSOILA= 000400        X4.MBZ= 017400
T31S2   043172        T32WDC  052366        T4      046640 G      XSOILC= 001000        X4.RCE= 040000
T31S3   043174        T34BFR  055440        T4.1    046732        XSOLET= 020000        X4.TSM= 020000
T31TIM  044310        T34BF2  055570        T4.2    047550        XSOMOT= 000200        X4.WRC= 000377
T31TM   044467

. ABS.  060612    000   (RW,I,GBL,ABS,OVR)
        000000    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 289
Work  file writes: 276
Size of work file: 31064 Words  ( 122 Pages)
Size of core pool: 19714 Words  ( 75 Pages)
```

```
Operating  system: RSX-11M/PLUS (Under VAX/VMS)

Elapsed time: 00:10:18.25
CZTKHB.BIN,CZTKHB/-SP=SVC/ML,CZTKHB
```