.REM\

IDENTIFICATION

PRODUCT ID: AC-T777A-MC
PRODUCT TITLE: CZTKFA TK25 FRT END FUNC 02
PRODUCT DATE: MARCH, 1984
DEPARTMENT: TAPE DIAGNOSTIC ENGINEERING
AUTHOR: DICE SYSTEMS, INC.

TABLE OF CONTENTS

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

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 (CIQPMAO VERSION 34 OR LATER)
PDP-11 DIAGNOSTIC LOADER/MONITOR (XXDP+)

2.3 PREREQUISITES

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

F1

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 "CIQPMA0 XXDP" PROGRAMMER'S MANUAL NUMBER AC-S296A-AC.

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

```
CHMDLE0 XXDP+ DL MONITOR
BOOTED VIA UNIT 0
28K NON-UNIBUS SYSTEM

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

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

.R CZTKFA
*
CZTKFA.BIC

DRS-E0
CZTKF-A-0
CZTKFA TK-25 FRT END FUNC #2 UNIT IS TK25
RSTRT ADR 147642
DR>START/FLAG:PNT:HOE
```

THE ABOVE COMMAND WILL START THE DIAGNOSTIC. THE COMMAND HAS TWO
SWITCHES ON WHICH ARE "PRINT EACH TEST NUMBER 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.>
```

4.0 OPERATING INSTRUCTIONS - SAMPLE PRINTOUTS

4.1 SUCCESSUL RUN EXAMPLES

4.1.1 SUCCESSFUL RUN EXAMPLE - CZTKFA -

```
TST: 001 INITIALIZATION #2 TEST
TST: 002 OFF-LINE REJECT AND REWIND TEST
TST: 003 BASIC WRITE TEST
TST: 004 BASIC READ DATA (FORWARD AND REVERSE) TEST
TST: 005 STAND-ALONE MANUAL INTERVENTION NOT EXECUTED TEST
TST: 006 STAND-ALONE CONFIGURATION TYPEOUT NOT EXECUTED TEST
TST: 007 STAND-ALONE SCOPE LOOPS NOT EXECUTED TEST
CZTKF EOP    1
    0 TOTAL ERRS
```

NOTE: PROGRAM NOW STARTS OVER AGAIN AT TEST 1

4.2 OPERATING INSTRUCTIONS - SAMPLE ERROR MESSAGE

ERROR MESSAGE EXAMPLE

```
TST: 003 BASIC WRITE TEST
CZTKF HRD ERR 00303 ON UNIT 00 TST 003 SUB 001 PC:026500
TSSR INCORRECT AFTER WRITE COMMAND, MORE BITS SET THAN SSR

TSSR=000000
TERMINATION CLASS CODE = NORMAL TERMINATION
*****REPLACE CONTROLLER*****
PACKET ADDRESS=030510
PACKET WORD #0=140005
PACKET WORD #1=053470
PACKET WORD #2=000000
PACKET WORD #3=000024


MESSAGE BUFFER ADDRESS=030400
MESSAGE BUFFER CONTENTS;
MESSAGE BUFFER HEADER    =100020
DATA FIELD LENGTH        =000012
RESIDUAL BYTE COUNTER    =000000
XSTAT0 CONTENTS          =000300
XSTAT1 CONTENTS          =000000
XSTAT2 CONTENTS          =101000
XSTAT3 CONTENTS          =000000
```

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 - CZTKF

TEST NUMBER	N/I SECS.	DEF SECS.
1	2	2
2	2	2
3	41	41
4	70	70
5	N/A	N/A
6	N/A	N/A
7	N/A	N/A

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

Q.V.	1 MIN 55 SECONDS
DEFAULT	1 MIN 55 SECONDS

6.0 TEST DESCRIPTIONS - CZTKF

6.1 TEST 1 - INITIALIZATION #2

```
**************************************************************
* NOTE: IF THIS TEST DETECTS AN ERROR REPLACE THE TK25'S     *
* CONTROLLER                                                 *
**************************************************************
```

THIS TEST VERIFIES THAT WRITING INTO THE TSSR RETURNS THE CONTROLLER TO ITS INITIALIZED STATE FROM VARIOUS CONDITONS.

6.2 TEST 2 - OFF LINE REJECT AND REWIND TEST

THIS TEST VERIFIES BASIC TAPE MOTION COMMAND DECODING AND BASIC OPERATION OF THE REWIND POSITIONING COMMAND. IT DOES NOT NECESSARILY DEMONSTRATE THAT THE TRANSPORT CAN BE REWOUND FROM AN ARBITRARY POSITION ON THE TAPE. SUBSEQUENT TESTS IMPLICITLY CHECK THE OPERATION OF THE REWIND COMMAND SINCE THEY MUST TYPICALLY REWIND THE TAPE IN IN THE NORMAL COURSE OF THEIR TEST SEQUENCES. THE TEST CONSISTS OF THE FOLLOWING THREE SUBTESTS:

6.2.1 TEST 2, SUBTEST 1: -

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

6.2.2 TEST 2, SUBTEST 2: -

THIS SUBTEST VERIFIES THAT A REWIND COMMAND WITH A CVC=1 CLEARS VCK AND RETURNS PROPER STATUS IN THE MESSAGE BUFFER.

6.3 TEST 3 - BASIC WRITE TEST

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

THIS TEST VERIFIES THAT THE WRITE DATA (NEXT) COMMAND OPERATES CORRECTLY, UP TO THE POINT OF CHECKING THAT THE DATA WAS ACTUALLY WRITTEN ONTO THE TAPE CORRECTLY. THE TESTING IN THIS TEST IS LIMITED TO VERIFYING THAT THE COMMAND WAS TERMINATED CORRECTLY WITH THE CORRECT REGISTER, BUFFER, AND RAM CONTENTS.

6.3.1 TEST 3, SUBTEST 1: -

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

6.3.2 TEST 3, SUBTEST 2: -

THIS SUBTEST VERIFIES THAT WRITE DATA COMMANDS WITH CVC=1 AND THE SWAP BYTES (SWB) BIT CLEAR OPERATES PROPERLY. THE BYTE COUNT (RECORD SIZE) VARIES FROM 20 THROUGH 64K IN VARYING INCREMENTS (DEPENDING ON WHETHER OR NOT THE DIAGNOSTIC IS RUNNING ON THE LONG VERIFICATION MODE). THE TAPE IS NOT REWOUND BETWEEN SUCCESSIVE RECORDS BUT IS REWOUND AFTER THE FINAL RECORD IS WRITTEN. AN INCREMENTING COUNT PATTERN IS SUPPLIED IN THE DATA BUFFER. AFTER EACH BLOCK IS WRITTEN, THE TTSR AND THE TSBA REGISTERS AND THE MESSAGE BUFERS ARE CHECKED.

6.3.3 TEST 3, SUBTEST 3: -

THIS SUBTEST VERIFIES THAT WRITE DATA COMMANDS WITH CVC=1 AND THE SWAP BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2. THE RESULTS SHOULD BE THE SAME.

6.3.4 TEST 3, SUBTEST 4: -

THIS SUBTEST VERIFIES THAT A WRITE COMMAND WITH AN ILLEGAL BUFFER ADDRESS IS REJECTED WITH THE PROPER ERROR STATUS AND THAT TAPE DOES NOT MOVE.

6.3.5 TEST 3, SUBTEST 5: -

THIS SUBTEST VERIFIES THAT A WRITE DATA COMMAND SPECIFYING A DATA BUFFER STARTING IN NONEXISTANT MEMORY TERMINATES WITH THE PROPER ERROR STATUS WITHOUT MOVING TAPE. THIS TEST IS SKIPPED IF NONEXISTANT MEMORY CAN NOT BE ADDRESSED.

6.3.6 TEST 3, SUBTEST 6: -

THIS SUBTEST VERIFIES THAT A WRITE DATA COMMAND SPECIFYING A DATA BUFFER STARTING IN EXISTANT MEMORY BUT RUNNING INTO NONEXISTANT MEMORY TERMINATES WITH THE PROPER ERROR STATUS. A LARGE ENOUGH RECORD SIZE IS SPECIFIED SUCH THAT TAPE IS ACTUALLY MOVED AND WRITTEN.

6.4 TEST 4 - BASIC READ DATA TEST (FORWARD AND REVERSE)

```
**************************************************************
* 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 READ FORWARD AND READ REVERSE COMMANDS OPERATE PROPERLY. VARIOUS COMBINATIONS OF ODD AND EVEN DATA BUFFER BOUNDARIES, RECORD SIZES AND BYTE SWAP CONTROL VARIABLES ARE USED. THE TEST FURTHER VERIFIES THE WRITE DATA COMMAND BY ACTUALLY READING AND VERIFYING WRITTEN DATA. ALSO TESTED ARE PROPER TERMINATIONS ON EXCEPTIONAL OR ERROR CONDITIONS: RECORD LENGTH LONG, RECORD LENGTH SHORT, READ REVERSE AT BOT, ILLEGAL DATA BUFFER ADDRESSES, AND NONEXISTANT DATA BUFFER ADDRESSES.

6.4.1 TEST 4, SUBTEST 1: -

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

6.4.2 TEST 4, SUBTEST 2: -

THIS SUBTEST VERIFIES THAT THE READ DATA COMMANDS WITH CVC=1 AND THE SWAP BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2. THE RESULTS, EXCEPT FOR RAM CONTENTS SHOULD BE THE SAME.

6.4.3 TEST 4, SUBTEST 3: -

THIS SUBTEST VERIFIES THAT A READ FORWARD COMMAND READING A RECORD LONGER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH (RLL) BIT SET.

6.4.4 TEST 4, SUBTEST 4: -

THIS SUBTEST VERIFIES THAT A READ FORWARD COMMAND READING A RECORD SHORTER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH SHORT (RLS) BIT SET. IT IS VERIFIED THAT THE RESIDUAL BYTE COUNT (RBPCR) IN THE MESSAGE BUFFER CONTAINS THE PROPER NONZERO VALUE (E.G. THE DIFFERENCE BETWEEN THE ACTUAL BYTE COUNT AND THE ACTUAL RECORD LENGTH).

6.4.5 TEST 4, SUBTEST 5: -

THIS SUBTEST VERIFIES
THAT A READ REVERSE COMMAND OPERATES PROPERLY. THE
TAPE IS FIRST REWOUND AND THEN WRITTEN WITH A SERIES OF TEST RECORDS
VARYING IN LENGTH AND DATA CONTENT. THE TAPE IS THEN READ IN REVERSE
SEQUENTIALLY AND THE RESULTS (STATUS, DATA, ETC.) VERIFIED. THE BYTE COUNT
ON EACH READ REVERSE COMMAND IS SET TO THE LENGTH OF THE EXPECTED RECORD,
SO NO EXCEPTIONAL CONDITIONS SHOULD OCCUR.

6.4.6 TEST 4, SUBTEST 6: -

THIS SUBTEST VERIFIES THAT THE READ DATA COMMANDS WITH CVC=1 AND THE SWAP
BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST SEQUENCE IS IDENTICAL TO
THAT USED IN SUBTEST 2. THE RESULTS EXCEPT FOR RAM CONTENTS SHOULD BE THE
SAME.

6.4.7 TEST 4, SUBTEST 7: -

THIS SUBTEST VERIFIES THAT A READ REVERSE COMMAND, READING A RECORD LONGER
THAN THE SPECIFIED BYTE COUNT, CAUSES A TAPE STATUS ALERT TERMINATION WITH
THE RECORD LENGTH LONG (RLL) BIT SET.

6.4.8 TEST 4, SUBTEST 8: -

THIS SUBTEST VERIFIES THAT A READ REVERSE COMMAND SPECIFYING A DATA BUFFER
STARTING IN NONEXISTANT MEMORY TERMINATES WITH THE PROPER ERROR STATUS
WITHOUT MOVING THE TAPE.

6.4.9 TEST 4, SUBTEST 9: -

THIS SUBTEST VERIFIES THAT ILLEGAL BUFFER ADDRESSES CAUSE A FUNCTION REJECT
TERMINATION WITH ILLEGAL ADDRESS (ILA) ERROR BIT SET.

6.4.10 TEST 4, SUBTEST 10: -

THIS SUBTEST VERIFIES THAT A DATA BUFFER ADDRESS, REFERENCING NONEXISTANT
MEMORY, CAUSES RECOVERABLE ERROR TERMINATION (TC=4), WITH THE NXM BIT SET
IN THE TSSR, AND THAT THE TAPE IS ULTIMATELY POSITIONED PROPERLY.

6.4.11 TEST 4, SUBTEST 11: -

THIS SUBTEST VERIFIES THAT A READ REVERSE COMMAND ISSUED WHILE THE TAPE IS AT BOT RESULTS IN A FUNCTION REJECT TERMINATION WITH THE NONEXECUTABLE FUNCTION (NEF) ERROR BIT SET.

6.4.12 TEST 4, SUBTEST 12: -

THIS SUBTEST VERIFIES THAT A READ REVERSE COMMAND ISSUED WHILE THE TAPE IS POSITIONED BEFORE THE FIRST RECORD ON TAPE (BUT NOT AT BOT) RESULTS IN TAPE STATUS ALERT.

F2

6.5 TEST 5 - MANUAL INTERVENTION

THIS TEST MUST BE STARTED AS FOLLOWS:

AT THE DIAGNOSTIC SUPERVISOR PROMPT "DR>"
TYPE "START/FLAG:PNT/TEST:5/PASS:1"

THE MANUAL INTERVENTION TEST IS A STANDALONE ROUTINE (NOT REALLY A "TEST") THAT ALLOWS THE OPERATOR TO CHECK OUT VARIOUS ELEMENTS AND FUNCTIONS OF THE SUBSYSTEM THAT CANNOT BE MANIPULATED BY THE PROGRAM ALONE. WHEN THIS ROUTINE IS STARTED, IT FIRST PRINTS OUT A MENU OF SELECTABLE SUBTESTS AND THEN WAITS FOR THE OPERATOR TO TYPE IN A SELECTION CODE. THE ONLY WAYS TO EXIT THIS ROUTINE AND RETURN TO THE DIAGNOSTIC SUPERVISOR ARE BY TYPING <CTRL-C> OR SELECTING CODE 4.
SELECTION CODES AND SUBROUTINES ARE:

CODE	ROUTINE
0	DISPLAY THIS MENU
1	REWIND AND UNLOAD COMMAND TEST
2	WRITE PROTECT TEST
3	FRONT PANEL ON-LINE/OFF-LINE SWITCH TEST
4	RETURN TO THE DIAGNOSTIC SUPERVISOR

ENTER MENU SELECTION: (0) ?

EACH MENU ITEM CORRESPONDS TO A SUBTEST, AS FOLLOWS:

SELECTION 0 - PRINTS OUT THE MENU ON THE CONSOLE TERMINAL.

SELECTION 1 - THIS ROUTINE INSTRUCTS THE OPERATOR TO PLACE THE DRIVE ON-LINE AND AT OR BEYOND BOT. THE TEST WILL THEN ISSUE THE REWIND AND UNLOAD COMMAND. IT WILL ALSO TELL THE OPERATOR IF THE DRIVE ENDED UP ON-LINE OR OFF-LINE.

SELECTION 2 - THIS ROUTINE INSTRUCTS THE OPERATOR TO MOUNT A SCRATCH TAPE REEL THAT DOES NOT HAVE A WRITE-ENABLE RING INSTALLED. THEN WAITS FOR THE OPERATOR TO RESPOND THAT THIS HAS BEEN ACCOMPLISHED. UPON THE RESPONSE, THE PROGRAM VERIFIES THAT THE TRANSPORT SHOWS A WRITE-PROTECTED STATUS, THEN ATTEMPTS TO WRITE DATA ON THE TAPE AND EXPECTS THE APPROPRIATE ERROR TERMINATION INDICATING THAT THE WRITE FUNCTION COULD NOT BE PERFORMED BECAUSE THE REEL IS WRITE-PROTECTED. IF THE APPROPRIATE TERMINATION IS NOT RECEIVED, AN ERROR IS REPORTED.

SELECTION 3 - THIS TEST CHECKS THAT THE PROGRAM CAN READ THE SENSE OF THE FRONT PANEL "ON-LINE" BUTTON/LIGHT. THE PROGRAM CHECKS THE STATE OF THE DRIVE (ON-LINE OR OFF-LINE) AND PRINTS A MESSAGE TO NOTIFY THE OPERATOR I.E. "DRIVE IS NOW OFF-LINE" OR "DRIVE IS NOW ON-LINE".

SELECTION 4 - THIS WILL RETURN THE PROGRAM TO THE DIAGNOSTIC SUPERVISOR PROMPT. NOTE: IF THE OPERATOR FAILED TO SELECT A PASS COUNT OF ONE, THE PROGRAM WILL LOOP UNTIL STOPPED WITH A CONTROL C.

6.6 TEST 6 - CONFIGURATION TYPEOUT

THIS TEST MUST BE STARTED AS FOLLOWS:

AT THE DIAGNOSTIC SUPERVISOR PROMPT "DR>"
TYPE "START/FLAG:PNT/TEST:6/PASS:1"

THIS IS A STANDALONE ROUTINE THAT PRINTS OUT ON THE CONSOLE TERMINAL THE CONFIGURATION OF THE CONTROLLER MODULE AND THE TK25 SUBSYSTEM. SPECIFICALLY, THE FOLLOWING INFORMATION IS PRINTED:

1. MICROCODE REVISION LEVEL OF THE CONTROLLER.
2. NUMBER OF TAPE TRANSPORTS CONNECTED TO THE CONTROLLER.
3. UNIT SELECT CODE AND STATE (ONLINE/OFFLINE, WRITE ENABLED/PROTECTED)OF EACH CONNECTED TRANSPORT.

THE OPERATOR IS EXPECTED TO READ THE PRINTOUT AND VERIFY THAT IT MATCHES THE ACTUAL CONFIGURATION AT HAND. IF, FOR EXAMPLE, THE PROGRAM INDICATES THAT IT "SEES" TWO TRANSPORTS CONNECTED WHEN IN FACT ONLY ONE IS PRESENT, THE OPERATOR MUST INTERPRET THIS AS AN ERROR AND ATTEMPT TO FIND THE CAUSE (BAD CABLE, FAULTY UNIT SELECT DECODING IN THE TRANSPORT, ETC.).[SINCE THE CONTROLLER CAN ONLY ACCESS UNIT 0 IF IT IS IN "STANDARD" MODE, THE PROGRAM WILL FORCE THE MODULE INTO EXTENDED MODE VIA THE WRITE SUBSYSTEM MEMORY COMMAND IN ORDER TO SCAN FOR CONNECTED TRANSPORTS.]

THIS ROUTINE, WHEN ITS ACTIONS ARE COMPLETED, WILL EXIT BACK TO THE DIAGNOSTIC SUPERVISOR SO THAT IF ADDITIONAL UNITS (CONTROLLERS) ARE SELECTED (E.G. FROM THE INITIAL STARTUP DIALOGUE), THE ROUTINE WILL BE REENTERED SO THAT THEIR CONFIGURATIONS CAN BE PRINTED.

6.7 TEST 7 - SCOPE LOOPS

THIS TEST MUST BE STARTED AS FOLLOWS:

AT THE DIAGNOSTIC SUPERVISOR PROMPT "DR>"
TYPE "START/FLAG:PNT/TEST:7/PASS:1"

THIS IS A STANDALONE ROUTINE PROVIDING A NUMBER OF TIGHT "SCOPE LOOPS" USEFUL FOR DEBUGGING BASIC REGISTER ACCESS PROBLEMS WITH THE CONTROLLER MODULE. THESE SCOPE LOOPS CAN BE USED WHEN THE NORMAL "LOOP ON ERROR" OR "LOOP ON TEST (SUBTEST)" FACILITIES DON'T SEEM TO ALLOW THE OPERATOR TO ZERO IN A PROBLEM IN THE EARLY TESTS (I.E. THE HARDWARE MAY NOT BE RESPONDING TO A REGISTER ACCESS, CAUSING A BUS ERROR TRAP, EVEN THOUGH THE DEVICE ADDRESS SELECTED BY THE PROGRAM MATCHES THE THE CONFIGURATION SET UP IN THE HARDWARE DIP SWITCHES). THE FOLLOWING MENU OF SCOPE LOOPS IS AVAILABLE:

CODE	SCOPE LOOP
0	HELP, PRINT THIS MENU
1	TSBA READ ACCESS
2	TSSR READ ACCESS
3	INITIALIZE (TSSR WRITE ACCESS)
4	TSDB HIGH BYTE WRITE ACCESS
5	TSDB LOW BYTE WRITE ACCESS
6	TSDB MAINTENANCE-MODE WORD WRITE ACCESS
7	EXIT (RETURN TO SUPERVISOR)

FOR SCOPE LOOPS THAT WRITE INTO REGISTERS, THE PROGRAM PROMPTS THE OPERATOR FOR THE DATA TO BE WRITTEN. TYPING <RETURN> CAUSES AN EXIT FROM THE SCOPE LOOP BACK TO THE MENU.

```
                        \
                                .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  CZTKF,A,0,655.,0
002000                          L$NAME::                ;DIAGNOSTIC NAME
002000  103                             .ASCII /C/
002001  132                             .ASCII /Z/
002002  124                             .ASCII /T/
002003  113                             .ASCII /K/
002004  106                             .ASCII /F/
002005  000                             .BYTE   0
002006  000                             .BYTE   0
002007  000                             .BYTE   0
002010                          L$REV::                 ;REVISION LEVEL
002010  101                             .ASCII  /A/
002011                          L$DEPO::                ;0
002011  060                             .ASCII  /0/
002012                          L$UNIT::                ;NUMBER OF UNITS
002012  000001                          .WORD   T$PTHV
002014                          L$TIML::                ;LONGEST TEST TIME
002014  001217                          .WORD   655.
002016                          L$HPCP::                ;POINTER TO H.W. QUES.
002016  053066                          .WORD   L$HARD
002020                          L$SPCP::                ;POINTER TO S.W. QUES.
002020  053226                          .WORD   L$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$LADP::                ;DIAG. END ADDRESS
002026  053442                          .WORD   L$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$DTP::                 ;PTR. TO DISPATCH TABLE
002040  053420                          .WORD   L$DISPATCH
```

PROGRAM HEADER

```
002042                          L$PRIO::                ;DIAGNOSTIC RUN PRIORITY
002042  000000                          .WORD   0
002044                          L$ENVI::                ;FLAGS DESCRIBE HOW IT WAS SETUP
002044  000000                          .WORD   0
002046                          L$EXP1::                ;EXPANSION WORD
002046  000000                          .WORD   0
002050                          L$MREV::                ;SVC REV AND EDIT #
002050      003                         .BYTE   C$REVISION
002051      003                         .BYTE   C$EDIT
002052                          L$EF::                  ;DIAG. EVENT FLAGS
002052  000000                          .WORD   0
002054  000000                          .WORD   0
002056                          L$SPC::
002056  000000                          .WORD   0
002060                          L$DEVP::                ; POINTER TO DEVICE TYPE LIST
002060  003334                          .WORD   L$DVTYP
002062                          L$REPP::                ;PTR. TO REPORT CODE
002062  023052                          .WORD   L$RPT
002064                          L$EXP4::
002064  000000                          .WORD   0
002066                          L$EXP5::
002066  000000                          .WORD   0
002070                          L$AUT::                 ;PTR. TO ADD UNIT CODE
002070  022544                          .WORD   L$AU
002072                          L$DUT::                 ;PTR. TO DROP UNIT CODE
002072  022642                          .WORD   L$DU
002074                          L$LUN::                 ;LUN FOR EXERCISERS TO FILL
002074  000000                          .WORD   0
002076                          L$DESP::                ;POINTER TO DIAG. DESCRIPTION
002076  003342                          .WORD   L$DESC
002100                          L$LOAD::                ;GENERATE SPECIAL AUTOLOAD EMT
002100  104035                          EMT     E$LOAD
002102                          L$ETP::                 ;POINTER TO ERRTBL
002102  000000                          .WORD   0
002104                          L$ICP::                 ;PTR. TO INIT CODE
002104  021764                          .WORD   L$INIT
002106                          L$CCP::                 ;PTR. TO CLEAN-UP CODE
002106  023024                          .WORD   L$CLEAN
002110                          L$ACP::                 ;PTR. TO AUTO CODE
002110  022750                          .WORD   L$AUTO
002112                          L$PRT::                 ;PTR. TO PROTECT TABLE
002112  021754                          .WORD   L$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
```

```
716                                     .SBTTL  DEFAULT HARDWARE P-TABLE
717
718                             ;++
719                             ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
720                             ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
721                             ; IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
722                             ;--
723 002122                              BGNHW   DFPTBL          ;DEFAULT HARD-P-TABLE
    002122  000003                      .WORD   L10000-L$HW/2
    002124                      L$HW::
    002124                      DFPTBL::
724
725 002124  172522                      .WORD   172522          ; 2ND (OF 2) REGISTERS.
726 002126  000224                      .WORD   224             ; INTERRUPT VECTOR
727 002130  000240                      .WORD   PRI05           ; INTERRUPT PRIORITY.
728 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  000004                  .WORD   L10001-L$SW/2
002134                  L$SW::
002134                  SFPTBL::

002134  000000          TRANSTST::      .WORD   0       ;ENABLE RAM DUMP
002136  000000          NOITS::         .WORD   0       ; INHIBIT ITERATION OPTION.
                                                        ; ... 0 = ITERATE.
                                                        ; ...NZ = INHIBIT ITERATE.
002140  000031          LERRMAX::       .WORD   25.     ; LOCAL (PER TEST) ERROR LIMIT
002142  000310          GERRMAX::       .WORD   200.    ; GLOBAL (PER UNIT) ERROR LIMIT
002144                          ENDSW
002144                  L10001:
```

```
                                        .SBTTL  GLOBAL EQUATES SECTION


                                        .SBTTL  GLOBAL EQUATES SECTION

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

002144                                  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
                                ;
```

GLOBAL EQUATES SECTION

```
                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
780
781 002144                              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 REGISTERS
                                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
                        ;*KERNEL "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)
00003                           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
```

```
                        ;*
                        ;
                        ;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.N01311       = 23            ;DISABLE MICROTEST 11 AND 13
000024                  PW.RDEXT        = 24            ;READ EXT. TAPE STATUS (NOT SUPPORTED BY ALL TRANSPO
RTS

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

```
                                WC.IOTAD        = BIT6          ;ITAD0  - TRANSPORT ADDRESS BIT 0
                                WC.I1TAD        = BIT5          ;ITAD1  - TRANSPORT ADDRESS BIT 1
                                WC.I5RESV       = BIT4          ;IRESV5 - RESERVED #5
                                WC.IREW         = BIT3          ;IREW   - REWIND
                                WC.IRWU         = BIT2          ;IRWU   - REWIND AND UNLOAD
                                WC.IFEN         = BIT1          ;IFEN   - FORMATTER ENABLE
                                WC.IGO          = BIT0          ;GO

                                ;+
                                ;BSEL1 CODES FOR WRITE FORMAT
                                ;-
                                WF.IHISP        = BIT7          ;IHISP  - HIGH SPEED
                                WF.IWRT         = BIT6          ;IWRT   - WRITE
                                WF.IREV         = BIT5          ;IREV   - REVERSE
                                WF.IWFM         = BIT4          ;IWFM   - WRITE FILE MARK
                                WF.IEDIT        = BIT3          ;IEDIT  - EDIT
                                WF.IERASE       = BIT2          ;IERASE - ERASE
                                WF.I3RESV       = BIT1          ;IRESV3 - RESERVED #3
                                WF.I4RESV       = BIT0          ;IRESV4 - RESERVED #4


                                ;+
                                ;BSEL1 CODES FOR WRITE MISCELLANEOUS SUBCOMMAND
                                ;-
                                MS.EXT          = BIT7          ;INVERT SENSE OF EXTENDED FEATURES SWITCH
                                MS.RSFIFO       = BIT4          ;RESET FIFO AND INPUT PARITY ERRORR
                                MS.RSTAPE       = BIT3          ;RESET TAPE STATUS IN 2 FLIP-FLOPS
                                MS.ATTN         = BIT2!BIT1     ;ATTENTION TRIGGER FIELD
                                MS.RSD          = BIT0          ;RESET TIMER A,B THEN DELAY TIMES IN SEL2
                                ;+
                                ; MS.ATTN SUBCODES
                                ;-
                                        MSA.NOP = 0*2           ;NO-OP (NOTHING TRIGGERED)
                                        MSA.VOL = 1*2           ;SIMULATE ON-LINE/OFF-LINE TRANSISTION
                                        MSA.NRAM= 2*2           ;FORCE NON-FATAL RAM ERROR (FORCES ERRCODE 54)
                                        MSA.FRAM= 3*2           ;FORCE FATAL RAM ERROR (CAUSES SCE TO SET)
                                ;+
                                ; WRITE SUBSYSTEM WRITE NPR BSEL1 BIT DEFINITIONS
                                ;-
                                NP.IR           = BIT7          ;INTERRUPT REQUEST (0-1 TRANSITION)
                                NP.OUT          = BIT6          ;TAPE DATA DIRECTION OUT (0= IN)
                                NP.LOOP         = BIT5          ;ENABLE TRANSPORT LOOPBACK
                                NP.WRP          = BIT4          ;WRITE CORRECT PARITY (SET=0 TO WRITE WRONG)
                                ;+
                                ; READ STATUS MESSAGE BUFFER BIT DEFINITIONS
                                ;-

                                S2.DIM          = BIT7          ;WORD #9 BYTE 2 DATA IN MISS
                                S2.ILW          = BIT6          ;               ILW H
                                S2.OUTRDY       = BIT5          ;               OUT RDY H
                                S2.INRDY        = BIT4          ;               IN RDY H
                                S2.ATIMR        = BIT3          ;               TIMER A FLAG H
                                S2.BTIMR        = BIT2          ;               TIMER B FLAG H
                                S2.UNDEF        = BIT1+BIT0     ;(UNDEFINED)
                                S1.PARIN        = BIT15         ;WORD #8 BYTE 1 PARIN H
                                S1.I2RESV       = BIT14         ;               IRESV2
                                S1.I1RESV       = BIT13         ;               IRESV1
```

```
1071    010000          S1.IEOT         = BIT12                 ;               IEOT L
1072    004000          S1.IIDENT       = BIT11                 ;               IIDENT H
1073    002000          S1.ICER         = BIT10                 ;               ICER H
1074    001000          S1.IFMK         = BIT9                  ;               IFMK H
1075    000400          S1.IHER         = BIT8                  ;               IHER H
1076    000200          S0.ISPEED       = BIT7                  ;WORD #0 BYTE 0 ISPEED H
1077    000100          S0.IRDY         = BIT6                  ;               IRDY L
1078    000040          S0.IONL         = BIT5                  ;               IONL L
1079    000020          S0.ILDP         = BIT4                  ;               ILDP L
1080    000010          S0.IDBY         = BIT3                  ;               IDBY L
1081    000004          S0.IRWD         = BIT2                  ;               IRWD L
1082    000002          S0.IFBY         = BIT1                  ;               IFBY L
1083    000001          S0.IFPT         = BIT0                  ;               IFPT L.
1084                    ;       SPECIAL KEYBORD STUFF FOR MOVER PROGRAM
1085
1086    177560                  TKS     =177560                 ;KEYBOARD STATUS REGISTER
1087    177562                  TKB     =177562                 ;KEYBOARD DATA REGISTER
1088    177564                  TPS     =177564                 ;CONSOLE PRINTER STATUS REGISTER
1089    177566                  TPB     =177566                 ;CONSOLE PRINTER DATA REGISTER
1090    007776                  HIMEM   =007776                 ;HIGH MEMORY MASK VALUE
1091                    ;       CONTROLLER DEFINITIONS
1092                    ;
1093                    ;
1094    174400                  CSR     =174400                 ;STATUS AND CONTROL REGISTER
1095    174402                  BAR     =174402                 ;DL ADDRESS REGISTER
1096    174404                  DAR     =174404                 ;PLATTER ADDRESS
1097    174406                  MPR     =174406                 ;MULTIPURPOSE REGISTER
1098                    ;
1099                    ;
1100                    ;
1101                    ;
1102                    ;
1103                    ;       CONTROLLER COMMANDS
1104                    ;
1105                    ;
1106    000004                  DLGETS  =4                      ;GET STATUS COMMAND
1107    000006                  SEEK    =6                      ;SEEK TRACK AND HEAD SELECT
1108    000010                  DLRDHD  =10                     ;READ SECTOR HEADER
1109    000014                  READ    =14                     ;READ COMMAND
1110    000016                  DLRDNH  =16                     ;READ SECTOR NO HEADER CHECK
1111                    ;
1112                    ;
1113                    ;
1114                    ;
1115                    ;
1116                    ;
1117    000001                  READY   =1                      ;DRIVE READY BIT IN STATUS REG.
1118    000013                  DLSR    =13                     ;STATUS AND RESET
1119    177730                  DLERR   =177730                 ;MASK FOR COVER OPEN
1120    000006                  DLUN    =6                      ;HEADS UNLOADED
1121    000177                  DLCYL   =000177                 ;MASK FOR CYLINDER ADDRESS
1122    100200                  DLDNER  =100200                 ;DONE SET OR ERROR SET BITS
1123                    ;
1124                    ;
1125                    ;
1126                    ;
1127    177560                  TTICSR  =       177560          ;KEYBOARD INPUT STATUS
```

```
1128        177562                      TTIBFR  =       177562          ;KEYBOARD DATA REGISTER
1129        177564                      TTOCSR  =       177564          ;CONSOLE PRINTER STATUS REGISTER
1130        177566                      TTOBFR  =       177566          ;CONSOLE PRINTER DATA REGISTER
1131
```

```
1133                                    .SBTTL  SPECIAL MACROS AND OPDEFS.
1134
1135
1136                                    ;+
1137                                    ;SAVE GENERAL REGS 1 TO 5
1138                                    ;-
1139
1140                                            .MACRO  SAVREG
1141                                            JSR     R5,REGSAV
1142                                            .ENDM
1143
1144                                    ;+
1145                                    ; MACRO TO FORCE AN ERROR
1146                                    ;-
1147                                    .MACRO  FORCERROR       TAG,NOTSSR
1148                                    .NLIST
1149                                    .IIF NDF LISTALL, .NLIST
1150                                    .LIST
1151                                    .IF B NOTSSR
1152                                            MOV     TSSR(R5),R1             ;READ TSSR
1153                                    .ENDC
1154                                            MOV     FORCER,FORCER           ;IS FORCER SET? (LEAVE C BIT ALONE)
1155                                            BNE     TAG                     ;BR IF YES
1156                                    .NLIST
1157                                    .IIF NDF LISTALL, .LIST
1158                                    .LIST
1159                                    .ENDM
1160
1161                                    ;+
1162                                    ; MACRO TO FORCE AN EXIT TO AVOID SECTION ITERATIONS
1163                                    ;       WILL EXIT TO A LABEL IF FORCER IS NEGATIVE
1164                                    ;       SO TO FORCE ERRORS AND EXIT ON 1 ERROR SET
1165                                    ;       FORCER TO 177777
1166                                    ;       TO FORCE ERRORS AND ITERATIONS SET FORCER TO 1.
1167                                    ;-
1168                                    .MACRO  FORCEXIT        TAG
1169                                    .NLIST
1170                                    .IIF NDF LISTALL, .NLIST
1171                                    .LIST
1172                                            MOV     FORCER,FORCER           ;IS FORCER NEGATIVE?
1173                                            BMI     TAG                     ;BR IF YES
1174                                    .NLIST
1175                                    .IIF NDF LISTALL, .LIST
1176                                    .LIST
1177                                    .ENDM
1178                                    ;+
1179                                    ; MACRO TO INCREMENT ERROR COUNTS
1180                                    ;-
1181                                    .MACRO  NEXT.ERRNO
1182                                    .NLIST
1183                                    ;;;.IIF NDF LISTALL, .NLIST
1184                                            ERRNO=ERRNO+1
1185                                    ;;;.IIF NDF LISTALL, .LIST
1186                                    .LIST
1187                                    .ENDM
1188
1189                                    ;+
```

```
                        ;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.
                        ;
002144  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.
```

```
                                        .SBTTL  GLOBAL DATA SECTION

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


                                ;
                                ;THE FOLLOWING DATA ARE SET FOR EACH UNIT AT INIT TIME.
                                ;SINGLE UNIT DEFAULTS (LISTED) ARE IN THE DEFAULT P-TABLE.
                                ;
002146  000000                  EPRTSW::        .WORD   0               ;PRINT SWITCH
002150  000000                  UNITN::         .WORD   0               ;UNIT # UNDER TEST.
002152  000000                  QVP::           .WORD   0               ;QUICK VERIFY FLAG.
002154  000000                  CSRADDR::       .WORD   0               ;ADDRESS OF CSR FOR CURRENT DEVICE
002156  000224                  IVEC::          .WORD   224             ;INTERRUPT VECTOR
002160  000200                  IPRI::          .WORD   PRI04           ;INTERRUPT PRIORITY.
002162  000000                  TSTCNT::        .WORD   0               ;NUMBER OF TESTS RUN IN THIS PASS
002164  000000                  LOOPCNT::       .WORD   0               ;REMAINING ITERATION COUNT FOR TEST
002166  000000                  DEVCNT::        .WORD   0               ;NUMBER OF DEVICE UNDER TEST
002170  000000                  FATFLG::        .WORD   0               ;SET IF FATAL ERROR IS DETECTED IN TEST
002172  000000                  INTRECV::       .WORD   0               ;SET IF TAPE INTERRUPT WAS RECEIVED
002174  000000                  BENBSW::        .WORD   0               ;BUFFER ENABLE SWITCH SW 0=OFF;1=ON
002176  000000                  EXPD::          .WORD   0               ;EXPECTED RAM DATA FOR PRAMPKT ROUTINE
002200  000000                  RECV::          .WORD   0               ;RECEIVED RAM DATA FOR PRAMPKT ROUTINE
002202  000000                  ERRHI::         .WORD   0               ;HIGH ADDRESS MEMORY ERROR
002204  000000                  ERRLO::         .WORD   0               ;LOW ADDRESS MEMORY ERROR
002206                          RAMDATA::       .BLKW   16.             ;DATA READ FROM RAM PACKET OR MESSAGE BUF AREA
002246  000000                  RAMSIZ::        .WORD   0               ;RAM DATA SIZE FOR PRAMPKT ROUTINE
002250  000000                  RCVHIADD::      .WORD   0               ;RECEIVED BUFFER HIGH ADDRESS
002252  000000                  RCVLOADD::      .WORD   0               ;RECEIVED BUFFER LOW ADDRESS
002254  000000                  COUNT::         .WORD   0               ;TEST COUNT PATTERN
002256  000000                  DATA::          .WORD   0               ;TEST DATA
002260  000000                  TSTFLAG::       .WORD   0               ;TEST FLAG WORD
002262  000000                  TSTPTR::        .WORD   0               ;TSTBLK POINTER
002264  000000                  PRMNO::         .WORD   0               ;PRINT ROUTINE TEMP
002266                          EXPMSG::        .BLKB   100.            ;EXPECTED MESSAGE BUFFER DATA
002432                          RECMSG::        .BLKB   100.            ;RECEIVED MESSAGE BUFFER DATA
002576                          TMPBFR::        .BLKB   80.             ;TEMPORARY STORAGE FOR PRINT
002716  000000                  MESBFA::        .WORD   0               ;STORES ADDRESS OF MESSAGE BUFFER FOR ERR PRT
```

```
                                        .SBTTL  TSTBLK  - TEST DATA TABLE

                                ;+
                                ;
                                ;THIS TABLE CONTAINS TEST DATA USED IN SEVERAL TESTS
                                ;
                                ;IN SEQUENCE THE DATA IS:
                                ;
                                ;       ALL ZEROS
                                ;       ALL ONES
                                ;       WALKING ONES
                                ;       WALKING ZEROS
                                ;       ALTERNATING ONES AND ZEROS
                                ;
                                ;-

002720                          TSTBLK::
002720  000000                          .WORD   0                       ;ALL ZEROS
002722  177777                          .WORD   177777                  ;ALL ONES
002724  000001                          .WORD   BIT0                    ;DATA FOR WALKING ONES
002726  000002                          .WORD   BIT1
002730  000004                          .WORD   BIT2
002732  000010                          .WORD   BIT3
002734  000020                          .WORD   BIT4
002736  000040                          .WORD   BIT5
002740  000100                          .WORD   BIT6
002742  000200                          .WORD   BIT7
002744  000400                          .WORD   BIT8
002746  001000                          .WORD   BIT9
002750  002000                          .WORD   BIT10
002752  004000                          .WORD   BIT11
002754  010000                          .WORD   BIT12
002756  020000                          .WORD   BIT13
002760  040000                          .WORD   BIT14
002762  100000                          .WORD   BIT15
002764  177776                          .WORD   ^CBIT0                  ;DATA FOR WALKING ZEROS
002766  177775                          .WORD   ^CBIT1
002770  177773                          .WORD   ^CBIT2
002772  177767                          .WORD   ^CBIT3
002774  177757                          .WORD   ^CBIT4
002776  177737                          .WORD   ^CBIT5
003000  177677                          .WORD   ^CBIT6
003002  177577                          .WORD   ^CBIT7
003004  177377                          .WORD   ^CBIT8
003006  176777                          .WORD   ^CBIT9
003010  175777                          .WORD   ^CBIT10
003012  173777                          .WORD   ^CBIT11
003014  167777                          .WORD   ^CBIT12
003016  157777                          .WORD   ^CBIT13
003020  137777                          .WORD   ^CBIT14
003022  077777                          .WORD   ^CBIT15
003024  125252                          .WORD   125252                  ;ALTERNATING ONES, ZEROS
003026  052525                          .WORD   052525                  ;ALTERNATING ONES, ZERO OPPOSITE FROM ABOVE
003030                          TBLEND**:
```

```
                                        .SBTTL  GLOBAL ENVIRONMENT STORAGE
                                ;
                                ;STORAGE FOR DEVICE REGISTERS
                                ;
003030  000000  100000  000000 DUMMY   0,100000,0,0                    ;DUMMY DEVICE REGISTERS...
003040  000000  000000  000000         0,0,0,0,0,0,0,0
                                                                       ;...FOR MULTI-UNIT CHECKOUT.


003060  000000                 DUFLG::         .WORD   0              ;"DROPPED UNIT" FLAG.
                                                                       ;INHIBITS CODE IN "CLEAN-UP".
003062  000000                 NODEV::         .WORD   0              ;FLAG TO SAY NO DEVICE.

003064  000000                 TEMP1::         .WORD   0              ;SOME TEMP LOCATIONS.
003066  000000                 TEMP2::         .WORD   0
003070  000000                 ?XCOMM::        .WORD   0              ;XXDP+ COMM BLOCK POINTER.
003072  000000                 FREE::          .WORD   0              ;1ST FREE MEMORY ADDRESS...
003074  000000                 FRESIZ::        .WORD   0              ;...AND SIZE (IN WORDS).
003076  000000                 FREEHI:         .WORD   0              ;LAST WORD IN FREE SPACE
003100  000000                 KTFLG::         .WORD   0              ;KT11, MEM AVAIL FLAG -
                                                                       ;- .WORD        0 = <24K OR NO KT -
                                                                       ;- NZ = >24K AND KT.

003102  000000                 KTENABLE::      .WORD   0              ;SET BY TEST ROUTINES TO FLAG >28K UNDER TEST
003104  002000                 PST32W::        .WORD   2000           ;32W BLOCK ADDRESS FOR 32K START
003106  000000                 SIFLAG::        .WORD   0              ;
003110  000000                 BADDAT::        .WORD   0              ;ACTUAL DATA
003112  000000                 GODAT::         .WORD   0              ;EXPECTED DATA
003114  000000                 LOOPFL::        .WORD   0
003116                         CTAB::                                 ;CONFIGURATION TABLES.
003116  000000                 CTABM::         .WORD   0              ;CONFIG WORK.
003120  000000                                 .WORD   0
003122  000000                                 .WORD   0
003124  000000                                 .WORD   0
003126  177777                                 .WORD   -1             ;END OF MEM TABLE.
003130                         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
                               ;
003130                         ERTABL:         .BLKW   64.
003330  000000                 ERTABE:         .WORD   0

003332  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
                                ;-

003334                                  DEVTYP  <TK-25>
003334                          L$DVTYP::
003334  124  113  055                    .ASCIZ  /TK-25/
                                        .EVEN

                                ;+
                                ;TEST DESCRIPTION
                                ;-
003342                                  DESCRIPT <CZTKFA TK-25 FRT END FUNC #2>
003342                          L$DESC::
003342  103  132  124                    .ASCIZ  /CZTKFA TK-25 FRT END FUNC #2/
                                        .EVEN

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

003400  003440  003443  003447  TSSRBIT::        .WORD   1$,2$,3$,4$,5$,6$,7$,8$
003420  003501  003505  003511                  .WORD   9$,10$,11$,12$,13$,14$,15$,16$
003440     123     103     000  1$:      .ASCIZ  'SC'
003443     102     111     105  2$:      .ASCIZ  'BIE'
003447     123     103     105  3$:      .ASCIZ  'SCE'
003453     122     115     122  4$:      .ASCIZ  'RMR'
003457     116     130     115  5$:      .ASCIZ  'NXM'
003463     116     102     101  6$:      .ASCIZ  'NBA'
003467     102     111     124  7$:      .ASCIZ  'BIT9'
003474     102     111     124  8$:      .ASCIZ  'BIT8'
003501     123     123     122  9$:      .ASCIZ  'SSR'
003505     117     106     114  10$:     .ASCIZ  'OFL'
003511     102     111     124  11$:     .ASCIZ  'BIT5'
003516     102     111     124  12$:     .ASCIZ  'BIT4'
003523     102     111     124  13$:     .ASCIZ  'BIT3'
003530     102     111     124  14$:     .ASCIZ  'BIT2'
003535     102     111     124  15$:     .ASCIZ  'BIT1'
003542     102     111     124  16$:     .ASCIZ  'BIT0'
                                .EVEN
003550     124     123     123  SFIERR:  .ASCIZ  'TSSR ERROR AFTER SOFT INIT'
003603     124     123     123  SFHERR:  .ASCIZ  'TSSR ERROR AFTER BUS RESET'
003636     040     040     116  NXR:     .ASCIZ  /  NON-EXISTANT DEVICE REGISTER/
003675     045     101     040  NXRX:    .ASCIZ  /%A  ADDRESS: %O6/
003716     045     101     040  TSSX:    .ASCII  /%A  TSBA,TSSR EXP'D: %O6%A,%O6%N/
003756     045     101     040          .ASCIZ  /%A  TSBA,TSSR REC'D: %O6%A,%O6/
004015     045     116     045  FUSI:    .ASCII  /%N%A/
004021     040     040     125  USI:     .ASCIZ  /  UNEXPECTED INTERRUPT/
004050     040     040     111  NSI:     .ASCIZ  /  INTERRUPT EXPECTED, NOT RECEIVED/
```

```
1411 004113  045  116  045  FNOINTR:        .ASCII  /%N%A/
1412 004117  040  040  116  NOINTR: .ASCIZ  /  NO INTERRUPT WAS GENERATED/
1413 004154  040  040  111  IFAULT: .ASCIZ  /  INTERRUPT FAULT/
1414 004176  045  101  040  INTX:   .ASCIZ  /%A  CPU PC: %O6%A  TSBA: %O6/
1415 004233  040  040  042  NOINIT: .ASCIZ  /  "BUS-INIT" DIDN'T INITIALIZE CONTROLLER/
1416 004305  040  040  042  NSINIT: .ASCIZ  /  "SOFT-INIT" DIDN'T INITIALIZE THE DPU/
1417 004355  040  040  042  BRINIT: .ASCIZ  /  "BUS-RESET" DIDN'T INITIALIZE THE DPU/
1418
1419 004425  000            NUL:    .ASCIZ  //
1420 004426  045  116  000  NULCR:  .ASCIZ  /%N/
1421 004431  045  101  040  EXPGOT: .ASCIZ  /%A EXP'D: %O6%A, REC'D: %O6/
1422 004465  045  116  045  EXPGT2: .ASCIZ  /%N%A EXP'D: %O6%A, %O6%N%A REC'D: %O%A, %O6/
1423 004541  045  101  040  DUAD12: .ASCIZ  /%A REG(W) WRITTEN TO: %O6%A REG(R) READ; EXP'D: %O6%A, REC'D: %O6/
1424 004643  122  101  115  PKTRAM::        .ASCIZ  'RAM Contents Do Not Match Packet Sent'
1425 004711  040  040  103  SCME:   .ASCIZ  /  CONFIG DOESN'T MATCH MFG. MASTER/
1426 004754  127  122  111  WRTMSG: .ASCIZ  'WRITE CHARACTERISTICS Failed'
1427 005011  124  123  123  WRTERR: .ASCIZ  'TSSR Incorrect After WRITE Command, More Bits Set Than SSR'
1428 005104  124  123  123  RDERR:  .ASCIZ  'TSSR Incorrect After READ Command, More Bits Set Than SSR'
1429                         .EVEN
1430
1431
1432
```

```
                                        .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.
                                ;--

005176                                  BGNMSG  NXRERR                  ;NON-EXISTANT DEVICE REGISTER.
005176                          NXRERR::
005176                                  PRINTX  #NXRX,NODEV             ;NODEV = NEXM ADDRESS.
005176  013746  003062                    MOV     NODEV,-(SP)
005202  012746  003675                    MOV     #NXRX,-(SP)
005206  012746  000002                    MOV     #2,-(SP)
005212  010600                            MOV     SP,R0
005214  104415                            TRAP    C$PNTX
005216  062706  000006                    ADD     #6,SP
005222  004737  005230                    JSR     PC,EXTEND               ; PRINT EXTENSION IF REQUIRED.
005226                                  ENDMSG
005226                          L10002:
005226  104423                            TRAP    C$MSG


                                ;
                                ; THIS ROUTINE APPENDS A UNIQUE EXTENSION (IF REQUIRED)
                                ; TO ANY OF THE ABOVE ERROR SIGNATURES.
                                ;
005230  005727                  EXTEND: TST     (PC)+
005232  000000                  EXTA:   0                               ; 0 = NO EXTENSION.
005234  001402                          BEQ     1$
005236  004777  177770                  JSR     PC,@EXTA                ; APPEND EXTENSION TEXT.
005242                          1$:     PRINTX  #NULCR                  ; PRINT A BLANK LINE
005242  012746  004426                    MOV     #NULCR,-(SP)
005246  012746  000001                    MOV     #1,-(SP)
005252  010600                            MOV     SP,R0
005254  104415                            TRAP    C$PNTX
005256  062706  000004                    ADD     #4,SP
005262  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
                        ;
                        ;-

005264                  PRITSSR:
005264                          SAVREG                          ;SAVE GENERAL REGISTERS
005270  010104                  MOV     R1,R4                   ;SAVE THE TSSR CONTENTS
005272                          PRINTB  #TSSRFOR,R4             ;PRINT THE CONTENTS OF TSSR
005272  010446                  MOV     R4,-(SP)
005274  012746  006117          MOV     #TSSRFOR,-(SP)
005300  012746  000002          MOV     #2,-(SP)
005304  010600                  MOV     SP,R0
005306  104414                  TRAP    C$PNTB
005310  062706  000006          ADD     #6,SP
005314  010400                  MOV     R4,R0                   ;GET TSSR BACK FOR CHKAMB
005316  004737  016714          JSR     PC,CHKAMB               ;ARE CONTENTS AMBIGUOUS ?
005322  103410                  BCS     5$                      ;BRANCH IF NOT
005324                          PRINTX  #AMBTSSR                ;SHOW CONTENTS ARE AMBIGUOUS
005324  012746  006337          MOV     #AMBTSSR,-(SP)
005330  012746  000001          MOV     #1,-(SP)
005334  010600                  MOV     SP,R0
005336  104415                  TRAP    C$PNTX
005340  062706  000004          ADD     #4,SP
005344  010403          5$:     MOV     R4,R3                   ;CONTENTS OF TSSR
005346  042703  001476          BIC     #HIADDR!FATERR!TERCLS,R3        ;CLEAR ALL MULTIPLE BIT FIELDS
005352  001434                  BEQ     20$                     ;NO BITS ARE SET
005354  012702  002576          MOV     #TMPBFR,R2              ;TEMPORARY ASCII BUFFER
005360  012701  003400          MOV     #TSSRBIT,R1             ;ASCII EQUIVALENT OF BITS
005364  005703          10$:    TST     R3                      ;REMAINING BITS TO CONVERT
005366  001413                  BEQ     15$                     ;BRANCH WHEN ALL ARE DONE
005370  006101                  CLC                             ;CLEAR CARRY FOR SHIFT
005372  006103                  ROL     R3                      ;SHIFT NEXT BIT TO CARRY
005374  103004                  BCC     13$                     ;BRANCH IF BIT NOT SET
005376  011100                  MOV     (R1),R0                 ;POINTER TO BIT DEFINITION
005400  112202          11$:    MOVB    (R0)+,(R2)+             ;MOVE ASCIZ TO BUFFER
005402  001376                  BNE     11$                     ;MOVE ALL BITS
005404  112742  000054  177777  MOVB    #',,-1(R2)              ;INSERT A COMMA TO TERMINATE
005412  005701          13$:    TST     (R1)+                   ;POINT TO NEXT DESCRIPTION
005414  000765                  BR      10$                     ;GET THE REMAINING BITS
005416  105042          15$:    CLRB    -(R2)                   ;TERMINATE THE LINE
005420                          PRINTX  #TSSDEF,#TMPBFR         ;PRINT THE BIT DEFINITIONS
005420  012746  002576          MOV     #TMPBFR,-(SP)
005424  012746  006310          MOV     #TSSDEF,-(SP)
```

```
        005430  012746  000002                  MOV     #2,-(SP)
        005434  010600                          MOV     SP,R0
        005436  104415                          TRAP    C$PNTX
        005440  062706  000006                  ADD     #6,SP
  1504
  1505  005444  010403                  20$:    MOV     R4,R3                   ;GET THE TSSR CONTENTS
  1506  005446  042703  177761                  BIC     #^CTERCLS,R3            ;CLEAR ALL BUT TERMINATION
  1507  005452  016303  006400                  MOV     TCOCOD(R3),R3           ;GET THE TERMINATION CODE MEANING
  1508  005456                                  PRINTX  #TCOASC,R3              ;PRINT THE TERMINATION CODE
        005456  010346                          MOV     R3,-(SP)
        005460  012746  006200                  MOV     #TCOASC,-(SP)
        005464  012746  000002                  MOV     #2,-(SP)
        005470  010600                          MOV     SP,R0
        005472  104415                          TRAP    C$PNTX
        005474  062706  000006                  ADD     #6,SP
  1509  005500  010403                          MOV     R4,R3                   ;TSSR CONTENTS AGAIN
  1510  005502  042703  177717                  BIC     #^CFATERR,R3            ;CLEAR ALL BUT FATAL TERMINATION
  1511  005506  001421                          BEQ     25$                     ;DON'T PRINT IF ZERO
  1512  005510  006203                          ASR     R3
  1513  005512  006203                          ASR     R3
  1514  005514  006203                          ASR     R3                      ;ALINE TERMINATION CODE FOR INDEX
  1515  005516  016303  006740                  MOV     TSFCOD(R3),R3           ;GET THE FATAL TERMINATION CODE
  1516  005522                                  PRINTX  #TFCASC,R3              ;PRINT THE FATAL TERMINATION CODE
        005522  010346                          MOV     R3,-(SP)
        005524  012746  006241                  MOV     #TFCASC,-(SP)
        005530  012746  000002                  MOV     #2,-(SP)
        005534  010600                          MOV     SP,R0
        005536  104415                          TRAP    C$PNTX
        005540  062706  000006                  ADD     #6,SP
  1517  005544  012737  000031  002170          MOV     #25.,FATFLG             ;DROP THIS UNIT AFTER ERROR
  1518  005552  010403                  25$:    MOV     R4,R3                   ;GET TSSR CONTENTS
  1519  005554  042703  176377                  BIC     #^CHIADDR,R3            ;CLEAR ALL BUT EXTENDED ADDRESS
  1520  005560  001411                          BEQ     30$                     ;DON'T PRINT IF ZERO
  1521  005562                                  PRINTX  #TEXASC,R3              ;PRINT THE EXTENDED ADDRESS BITS
        005562  010346                          MOV     R3,-(SP)
        005564  012746  006137                  MOV     #TEXASC,-(SP)
        005570  012746  000002                  MOV     #2,-(SP)
        005574  010600                          MOV     SP,R0
        005576  104415                          TRAP    C$PNTX
        005600  062706  000006                  ADD     #6,SP
  1522  005604  022704  100210          30$:    CMP     #100210,R4              ;CHECK FOR MEDIA ERROR
  1523  005610  001003                          BNE     31$                     ;BR, IF PROBABLY NOT TAPE ERROR
  1524  005612  012737  006026  002146          MOV     #EPRT3,EPRTSW           ;"PROBABLY MEDIA RELETED ERROR - BAD TAPE"
  1525  005620  005737  002146          31$:    TST     EPRTSW                  ;CHECK FOR THE SWITCH EMPTY
  1526  005624  001003                          BNE     310$                    ;BR, IF SWITCH IS NOT EMPTY
  1527  005626  012737  005762  002146          MOV     #EPRT1,EPRTSW           ;SET SWITCH TO DEFAULT
  1528  005634  013737  002146  005644  310$:   MOV     EPRTSW,32$+2            ;PUT REAL SWITCHABLE MESSAGE IN PLACE
  1529  005642                          32$:    PRINTB  #EPRT1                  ;PRINT THE ERROR MESSAGE
        005642  012746  005762                  MOV     #EPRT1,-(SP)
        005646  012746  002001                  MOV     #1,-(SP)
        005652  010600                          MOV     SP,R0
        005654  104414                          TRAP    C$PNTB
        005656  062706  000004                  ADD     #4,SP
  1530  005662  012737  005762  002146          MOV     #EPRT1,EPRTSW           ;RESET TO NORMAL ERROR POINTER
  1531  005670  000207                          RTS     PC                      ;RETURN TO CALLER
  1532  005672      045     116     045  EPRT2:  .ASCIZ  '%N%A *****CHECK CONTROLLER, CABLES AND TRANSPORT******%S'
  1533  005762      045     116     045  EPRT1:  .ASCIZ  '%N%A *****REPLACE CONTROLLER******%S'
```

```
1534 006026      045      116      045  EPRT3:  .ASCIZ  '%N%A *****POSSIBLE MEDIA RELATED ERROR - BAD TAPE*****%S'
1535
1536 006117      045      116      045  TSSRFOR:        .ASCIZ  '%N%A TSSR = %O6'
1537 006137      045      116      045  TEXASC:         .ASCIZ  '%N%A Extended Address Bits = %O6'
1538 006200      045      116      045  TCOASC:         .ASCIZ  '%N%A Termination Class Code = %T'
1539 006241      045      116      045  TFCASC:         .ASCIZ  '%N%A Fatal Termination Class Code = %T'
1540 006310      045      116      045  TSSDEF:         .ASCIZ  '%N%A TSSR Bits Set: %T'
1541 006337      045      116      045  AMBTSSR:        .ASCIZ  '%N%A TSSR Contents Are Ambiguous'
1542                                    .EVEN
1543 006400  006420  006443  006471  TCOCOD: .WORD   1$,2$,3$,4$,5$,6$,7$,8$
1544 006420      116      157      162  1$:     .ASCIZ  'Normal Termination'
1545 006443      124      145      162  2$:     .ASCIZ  'Termination Condition'
1546 006471      124      141      160  3$:     .ASCIZ  'Tape Status Alert'
1547 006513      106      165      156  4$:     .ASCIZ  'Function Reject'
1548 006533      122      145      143  5$:     .ASCIZ  'Recoverable Error - Tape Position One Record Down'
1549 006615      122      145      143  6$:     .ASCIZ  'Recoverable Error - Tape Was Not Moved'
1550 006664      125      156      162  7$:     .ASCIZ  'Unrecoverable Error'
1551 006710      106      141      164  8$:     .ASCIZ  'Fatal Controller Error'
1552                                    .EVEN
1553
1554 006740  006750  007004  007015  TSFCOD: .WORD   1$,2$,3$,4$
1555 006750      111      156      164  1$:     .ASCIZ  'Internal Diagnostic Failure'
1556 007004      122      145      163  2$:     .ASCIZ  'Reserved'
1557 007015      102      165      163  3$:     .ASCIZ  'Bus Interface or Sanity Check Error'
1558 007061      122      145      163  4$:     .ASCIZ  'Reserved'
1559                                    .EVEN
```

PRIPKT - PRINT THE ADDRESS/CONTENTS OF COMMAND PACKET

```
                                        .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.
                                ;-

007072                          PRIPKT::
007072                                  SAVREG                  ;SAVE THE REGISTERS
007076  010005                          MOV     R0,R5           ;SAVE NO. OF WORDS IN PACKET
007100  005737  003102                  TST     KTENABLE        ;ABOVE 28K UNDER TEST?
007104  001001                          BNE     10$             ;BR IF YES
007106  005003                          CLR     R3              ;SET HIGH ORDER ADDRESS TO 0
007110  010301                  10$:    MOV     R3,R1           ;COPY HIGH ORDER ADDRESS
007112  010400                          MOV     R4,R0           ;GET LOWER ADDRESS
007114  006100                          ROL     R0              ;SHIFT BIT 15 INTO C BIT
007116  006101                          ROL     R1              ;AND INTO HIGH ORDER.
007120                                  PRINTB  #PKTADD,R1,R4   ;PRINT PACKET ADDRESS
007120  010446                          MOV     R4,-(SP)
007122  010146                          MOV     R1,-(SP)
007124  012746  007276                  MOV     #PKTADD,-(SP)
007130  012746  000003                  MOV     #3,-(SP)
007134  010600                          MOV     SP,R0
007136  104414                          TRAP    C$PNTB
007140  062706  000010                  ADD     #10,SP
007144  010300                  15$:    MOV     R3,R0           ;GET HIGH ORDER ADDRESS
007146  001404                          BEQ     20$             ;BR IF NOT ABOVE 28K.
007150  010401                          MOV     R4,R1           ;GET LOW ORDER ADDRESS
007152  004737  020270                  JSR     PC,$TMAP        ;SETUP PAR6 MAPPING FOR 18 BIT ADDRESS
007156  010004                          MOV     R0,R4           ;GET RETURNED PAR6 ADDRESS BIAS
007160  005001                  20$:    CLR     R1              ;SAVE WORD NUMBER
007162  012402                  25$:    MOV     (R4)+,R2        ;GET PACKET CONTENTS
007164                                  PRINTB  #PKTFRM,R1,R2   ;PRINT THE DATA
007164  010246                          MOV     R2,-(SP)
007166  010146                          MOV     R1,-(SP)
007170  012746  007240                  MOV     #PKTFRM,-(SP)
007174  012746  000003                  MOV     #3,-(SP)
007200  010600                          MOV     SP,R0
007202  104414                          TRAP    C$PNTB
007204  062706  000010                  ADD     #10,SP
007210  005201                          INC     R1              ;NEXT WORD NUMBER
007212  020105                          CMP     R1,R5           ;DONE ALL PACKET WORDS?
007214  002762                          BLT     25$             ;LOOP TILL ALL DONE
007216                                  PRINTB  #PKTNEW         ;JUST A COUPLE NEW LINES
007216  012746  007333                  MOV     #PKTNEW,-(SP)
007222  012746  000001                  MOV     #1,-(SP)
007226  010600                          MOV     SP,R0
007230  104414                          TRAP    C$PNTB
007232  062706  000004                  ADD     #4,SP
```

J4


```
1599 007236  000207                          RTS     PC              ;RETURN
1600
1601 007240      045     116     045  PKTFRM: .ASCIZ  '%N%A Packet Word 0=%D1%A = %06'
1602 007276      045     116     045  PKTADD: .ASCIZ  '%N%A Packet Address = %01%05'
1603
1604 007333      045     116     045  PKTNEW: .ASCIZ  '%N%N%A  '
1605                                     .EVEN
1606
```

```
                                        .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
                                ;
                                ;-

007344                          PRIBXOR::
007344                                  SAVREG                          ;SAVE THE REGISTERS
007350  010203                          MOV     R2,R3                   ;EXPECTED DATA
007352                                  XOR     R1,R3                   ;FORM THE EXCLUSIVE OR
007362  012700  177400                  MOV     #^C<377>,R0             ;BYTE MASK
007366  040001                          BIC     R0,R1                   ;SAVE LOW BYTE RECV
007370  040002                          BIC     R0,R2                   ;SAVE LOW BYTE EXPD
007372  040003                          BIC     R0,R3                   ;SAVE LOW BYTE XOR
007374                                  PRINTB  #XORBFOR,R2,R1,R3       ;PRINT THE MESSAGE
007374  010346                          MOV     R3,-(SP)
007376  010146                          MOV     R1,-(SP)
007400  010246                          MOV     R2,-(SP)
007402  012746  007426                  MOV     #XORBFOR,-(SP)
007406  012746  000004                  MOV     #4,-(SP)
007412  010600                          MOV     SP,R0
007414  104414                          TRAP    C$PNTB
007416  062705  000012                  ADD     #12,SP
007422  010300                          MOV     R3,R0                   ;R0 HAS XOR ON RETURN
007424  000207                          RTS     PC                      ;RETURN TO CALLER

007426      045     116     045 XORBFOR:        .ASCIZ  '%NN%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
                                ;
                                ;-

007474                          PRIXOR::
                                        SAVREG                  ;SAVE THE REGISTERS
007474  010203                          MOV     R2,R3           ;EXPECTED DATA
007500                                  XOR     R1,R3           ;FORM THE EXCLUSIVE OR
007502                                  PRINTB  #XORFOR,R2,R1,R3 ;PRINT THE MESSAGE
007512  010346                          MOV     R3,-(SP)
007514  010146                          MOV     R1,-(SP)
007516  010246                          MOV     R2,-(SP)
007520  012746  007544                  MOV     #XORFOR,-(SP)
007524  012746  000004                  MOV     #4,-(SP)
007530  010600                          MOV     SP,R0
007532  104414                          TRAP    C$PNTB
007534  062706  000012                  ADD     #12,SP
007540  010300                          MOV     R3,R0           ;R0 HAS XOR ON RETURN
007542  000207                          RTS     PC              ;RETURN TO CALLER

007544    045     116     045 XORFOR: .ASCIZ  '%N%A EXPD: %O6%A RECV: %O6%A XOR: %O6'
                                    .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
                                ;
                                ;-

007612                          PRIEQU:
007612                                  SAVREG                          ;SAVE THE REGISTERS
007616  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
                                ;
                                ;-
007620                          PRIRAM:
007620                                  SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
007624                                  PRINTB  #RAMFOR,R4              ;PRINT RAM ADDRESS IN ERROR
007624  010446                          MOV     R4,-(SP)
007626  012746  007650                  MOV     #RAMFOR,-(SP)
007632  012746  000002                  MOV     #2,-(SP)
007636  010600                          MOV     SP,R0
007640  104414                          TRAP    C$PNTB
007642  062706  000006                  ADD     #6,SP
007646  000207                          RTS     PC                      ;RETURN

007650      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
```

```
1722                                    ;
1723                                    ;-
1724 007712                             PRIADD:
1725 007712                                     SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
1726 007716  013700  002202                     MOV     ERRHI,R0                ;GET HIGH ADDRESSS
1727 007722  013701  002204                     MOV     ERRLO,R1                ;GET LOW ADDRESS
1728 007726  010102                             MOV     R1,R2                   ;COPY LOW ADDRESS
1729 007730  006101                             ROL     R1                      ;SHIFT BIT 15 TO C BIT
1730 007732  006100                             ROL     R0                      ;SHIFT INTO HIGH ORDER
1731 007734                                     PRINTB  #PRIA0,R0,R2            ;PRINT MEMORY ADDRESS IN ERROR
     007734  010246                             MOV     R2,-(SP)
     007736  010046                             MOV     R0,-(SP)
     007740  012746  007762                     MOV     #PRIA0,-(SP)
     007744  012746  000003                     MOV     #3,-(SP)
     007750  010600                             MOV     SP,R0
     007752  104414                             TRAP    C$PNTB
     007754  062706  000010                     ADD     #10,SP
1732 007760  000207                             RTS     PC                      ;RETURN
1733
1734 007762     045     116     045  PRIA0:     .ASCIZ  '#N#A MEMORY ERROR ADDRESS = #01#05'
1735                                             .EVEN
1736
1737
1738                                             .SBTTL  PRITADD - PRINT MEMORY TEST ADDRESS
1739                                    ;+
1740                                    ;
1741                                    ;PRINT MEMORY ADDRESS
1742                                    ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
1743                                    ;
1744                                    ; IMPLICIT INPUTS
1745                                    ;
1746                                    ;       ERRHI   - HIGH ORDER ADDRESS
1747                                    ;       ERRLO   - LOW ORDER ADDRESS
1748                                    ;
1749                                    ;-
1750 010026                             PRITADD:
1751 010026                                     SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
1752 010032  013700  002202                     MOV     ERRHI,R0                ;GET HIGH ADDRESSS
1753 010036  013701  002204                     MOV     ERRLO,R1                ;GET LOW ADDRESS
1754 010042  010102                             MOV     R1,R2                   ;COPY LOW ADDRESS
1755 010044  006101                             ROL     R1                      ;SHIFT BIT 15 TO C BIT
1756 010046  006100                             ROL     R0                      ;SHIFT INTO HIGH ORDER
1757 010050                                     PRINTB  #PRIT0,R0,R2            ;PRINT MEMORY ADDRESS IN ERROR
     010050  010246                             MOV     R2,-(SP)
     010052  010046                             MOV     R0,-(SP)
     010054  012746  010076                     MOV     #PRIT0,-(SP)
     010060  012746  000003                     MOV     #3,-(SP)
     010064  010600                             MOV     SP,R0
     010066  104414                             TRAP    C$PNTB
     010070  062706  000010                     ADD     #10,SP
1758 010074  000207                             RTS     PC                      ;RETURN
1759
1760 010076     045     116     045  PRIT0:     .ASCIZ  '#N#A MEMORY TEST ADDRESS = #01#05'
1761                                             .EVEN
1762
1763
1764
```

```
                        .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:
                ;
                ;
                ;-

010140                  SPACE::
                        SAVREG                          ;SAVE THE GENERAL REGISTERS
010144  012737  000764  010330          MOV     #500.,SDELAY    ;SET UP DELAY
010152  012737  140010  010320          MOV     #140010,80$     ;SET UP COMMAND, SPACE FORWARD
010160  005703                          TST     R3              ;CHECK FOR DIRECTION
010162  100403                          BMI     5$              ;BR, IF REVERSE INDICATED
010164  010337  010322                  MOV     R3,90$          ;LOAD UP NUMBER OF RECORDS TO SPACE
010170  000407                          BR      10$             ;GO DO COMMAND
010172  042703  100000          5$:     BIC     #BIT15,R3       ;CLEAR DIRECTION BIT
010176  010337  010322                  MOV     R3,90$          ;LOAD UP NUMBER OF RECORDS TO SPACE
010202  052737  000400  010320          BIS     #BIT8,80$       ;SET REVERSE BIT IN COMMAND PACKET
010210  012704  010320          10$:    MOV     #80$,R4         ;SET UP R4 WITH PACKET ADDRESS
010214  010465  177776                  MOV     R4,TSDB(R5)     ;SEND OUT COMMAND
010220  004737  017120          15$:    JSR     PC,WAITF        ;WAIT FOR SSR
010224  103420                          BCS     20$             ;BR, IF SSR IS SET AND OK
010226                                  DELAY   250             ;DELAY ABOUT .25 SECONDS
010226  012727  000250                  MOV     #250,(PC)+
010232  000000                          .WORD   0
010234  013727  002116                  MOV     L$DLY,(PC)+
010240  000000                          .WORD   0
010242  005367  177772                  DEC     -6(PC)
010246  001375                          BNE     .-4
```

SPACE - SPACE RECORDS (FORWARD AND REVERSE) COMMAND

```
        010250  005367  177756                  DEC     -22(PC)
        010254  001367                          BNE     .-20
  1817  010256  005337  010330                  DEC     SDELAY          ;BUMP DELAY COUNTER DOWN
  1818  010262  001356                          BNE     15$             ;BR, IF MORE DELAY
  1819  010264  000411                          BR      60$             ;BR IF TROUBLE CARRY = CLEAR
  1820  010266  016501  000000          20$:    MOV     TSSR(R5),R1     ;READ TSSR
  1821  010272  012702  000200                  MOV     #SSR,R2         ;SET UP EXPECTED
  1822  010276  020201                  25$:    CMP     R2,R1           ;ARE THEY OK
  1823  010300  001401                          BEQ     40$             ;BR, IF EQUAL = OK
  1824  010302  000402                          BR      60$             ;TROUBLE EXIT
  1825  010304  000261                  40$:    SEC                     ;SET CARRY NO TROUBLE
  1826  010306  000401                          BR      70$             ;EXIT
  1827  010310  000241                  60$:    CLC                     ;CARRY CLEAR = ERROR
  1828  010312                          70$:
  1829  010312  010400                          MOV     R4,R0           ;PASS PACKET ADDRESS
  1830  010314  000207                          RTS     PC              ;RETURN
```

```
                                        ;
                                        ;
                                        ;
                                        ;PACKET FOR SPACE COMMAND
                                        ;
                                                .BLKB   10-<.-TUV2A&7>
                                        ;
                                        ;COMMAND WORD
1842 010320  000000                     80$:    .WORD
                                        ;NUMBER OF RECORDS TO BE SPACED OVER WORD
1844 010322  000000                     90$:    .WORD
1845 010324  000000                             .WORD
1846 010326  000000                             .WORD
1847 010330  000000                     SDELAY: .WORD   0               ;DELAY COUNTER
                                                .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  002174                  CLR     BENBSW          ;CLEAR BUFFER ENABLE SWITCH
010342  010465  177776          10$:    MOV     R4,TSDB(R5)     ;SEND OUT COMMAND
010346  004737  017236                  JSR     PC,CHKTSSR      ;WAIT FOR SSR
010352  103401                          BCS     20$             ;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$:    ADD     #8.,R4          ;POINT TO WRT CHARA DATA PACKET
010412  011403                          MOV     (R4),R3         ;GET ADDRESS OF MESSAGE BUFFER
010414  010337  002716                  MOV     R3,MESBFA       ;STORE FOR PRINT ROUTINES
010420  000261                          SEC                     ;SET CARRY NO TROUBLE
010422  000401                          BR      70$             ;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
                                ;
                                ;-
                                REWIND::
                                        SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
                                        MOV     #RWPACK,R4              ;GET PACKET ADDRESS
                                        MOV     R4,TSDB(R5)             ;SEND PACKET ADDRESS TO EXECUTE
                                        MOV     #360.,R3                ;ENOUGH TIME FOR 2400' REEL TO REWIND
                                10$:    JSR     PC,WAITF                ;WAIT FOR SSR TO SET
                                        BCS     20$                     ;LEAVE WHEN SSR IS SET
                                        DELAY   250.                    ;WAIT FOR .25 SECONDS
                                        MOV     #250.,(PC)+
                                        .WORD   0
                                        MOV     L$DLY,(PC)+
                                        .WORD   0
                                        DEC     -6(PC)
                                        BNE     .-4
                                        DEC     -22(PC)
                                        BNE     .-20
                                        DEC     R3                      ;BUMP COUNTER DOWN
                                        BNE     10$                     ;KEEP GOING
                                        CLC                             ;CLEAR CARRY TO SET ERROR
                                20$:    MOV     R4,R0                   ;PASS THE PACKET ADDRESS
                                        RTS     PC                      ;RETURN
                                        .BLKB   10-<.-TUV2A&7>
                                RWPACK:
                                        .WORD   102010                  ;POSTION COMMAND (REWIND)
                                        .WORD   0                       ;NOT USED
```

```
                        .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:
                ;
                ;-

                CKRAM::
                        SAVREG                  ;SAVE THE GENERAL REGISTERS
010540 012701 002206            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 017236            JSR     PC,CHKTSSR      ;WAIT FOR SSR
010556 004737 017236    10$:    JSR     PC,CHKTSSR      ;WAIT FOR SSR TO SET
010562 110265 177777            MOVB    R2,TSD8H(R5)    ;SELECT NEXT RAM ADDRESS
010566 004737 017236            JSR     PC,CHKTSSR      ;WAIT FOR SSR TO SET
010572 116511 177776            MOVB    TSBAL(R5),(R1)  ;READ THE RAM DATA
010576 122124                   CMPB    (R1)+,(R4)+     ;COMPARE TO EXPECTED
010600 001401                   BEQ     20$             ;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$:    SEC                     ;SHOW GOOD COMPARE
010626 012737 000010 002246 50$:MOV     #8.,RAMSIZ      ;SETUP RAMSIZ FOR PRAMPKT ROUTINE
010634 000207                   RTS     PC              ;RETURN
```

```
                        .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:
                ;
                ;
                ;-

                RAMER::
010636                  SAVREG                  ;SAVE THE GENERAL REGISTERS
010636 013705 011022    MOV     RAMR5H,R5       ;RESET R5 TO FIRST DEVICE REGISTER
010642 012701 002206    MOV     #RAMDATA,R1     ;ADDRESS TO SAVE THE RAM DATA
010646 013702 011020    MOV     RAMHLD,R2       ;BYTE ADDRESS OF THE FIRST RAM DATA
010652 013703 002246    MOV     RAMSIZ,R3       ;SET THE SIZE OF THE READ UP
010656 004737 017236    10$:    JSR     PC,CHKTSSR      ;WAIT FOR THE SSR TO SET
010662 110265 177777    MOVB    R2,TSDBH(R5)    ;SELECT NEXT RAM ADDRESS
010666 004737 017236    JSR     PC,CHKTSSR      ;WAIT FOR SSR TO SET
010672 116521 177776    MOVB    TSBAL(R5),(R1)+ ;READ THE RAM DATA
010676 062702 000001    20$:    ADD     #1,R2           ;ADDRESS OF THE NEXT RAM LOCATION
010702 077313           SOB     R3,10$          ;NUMBER OF LOCATIONS COUNTER
010706 013704 002246    MOV     RAMSIZ,R4       ;GET THE RAM SIZE
010712 013702 011020    MOV     RAMHLD,R2       ;GET THE STARTING RAM ADDRESS
010716 060204           ADD     R2,R4           ;CALCULATE THE END ADDRESS
010720 162704 000001    SUB     #1,R4           ;CORRECT VALUE OF PRINTOUT
010724                  PRINTX  #RAMIOP,R2,R4   ;RAM ADDRESS = 10 - 17, ETC.
010724 010446           MOV     R4,-(SP)
010726 010246           MOV     R2,-(SP)
010730 012746 011024    MOV     #RAMIOP,-(SP)
010734 012746 000003    MOV     #3,-(SP)
010740 010600           MOV     SP,R0
010742 104415           TRAP    C$PNTX
010744 062706 000010    ADD     #10,SP
010750 012701 002206    MOV     #RAMDATA,R1     ;ADDRESS OF WHERE RAM DATA IS
010754 013703 002246    MOV     RAMSIZ,R3       ;THE SIZE OF THE RAM FIELD READ
010760 005004           30$:    CLR     R4              ;NO EXTRA DATA LEFT OVER
010762 112104           MOVB    (R1)+,R4        ;PICK UP BYTE OF RAM DATA
010764 042704 177400    BIC     #177400,R4      ;GET RID OF SIGN EXTEND
010770                  PRINTX  #RAMPD,R4       ;"010 211 111 222 377 000 123 134 ETC."
010770 010446           MOV     R4,-(SP)
010772 012746 011073    MOV     #RAMPD,-(SP)
010776 012746 000002    MOV     #2,-(SP)
011002 010600           MOV     SP,R0
011004 104415           TRAP    C$PNTX
011006 062706 000006    ADD     #6,SP
011012 077316           SOB     R3,30$          ;LOOP UNTIL ALL PRINTED
```

```
2052 011016  000207                  50$:    RTS     PC              ;RETURN
2053
2054 011020  000000                  RAMHLD: .WORD   0               ;RAM ADDR HOLDER 1ST ADDRESS
2055 011022  000000                  RAMR5H: .WORD   0               ;HOLDS R5 FOR LATER
2056 011024     045     116     045  RAMIOP: .ASCIZ  '%N%A Ram Address (Octal) = %03%A - %03%N'
2057 011075     045     101     040  RAMPD:  .ASCIZ  '%A %03%A '
2058                                 .EVEN
2059
```

```
                        .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  002206          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  017236          JSR     PC,CHKTSSR      ;WAIT FOR SSR
011132  004737  017236  10$:    JSR     PC,CHKTSSR      ;WAIT FOR SSR TO SET
011136  110265  177777          MOVB    R2,TSDBH(R5)    ;SELECT NEXT RAM ADDRESS
011142  004737  017236          JSR     PC,CHKTSSR      ;WAIT FOR SSR TO SET
011146  116511  177776          MOVB    TSBAL(R5),(R1)  ;READ THE RAM DATA
011152  122124                  CMPB    (R1)+,(R4)+     ;COMPARE TO EXPECTED
011154  001401                  BEQ     20$             ;BRANCH IF OK
011156  005203                  INC     R3              ;SET ERROR FLAG
011160  005202          20$:    INC     R2              ;ADDRESS OF NEXT RAM LOCATION
011162  012737  000010  002246  MOV     #8.,RAMSIZ      ;ASSUME NORMAL NOT SET
011170  020227  000176          CMP     R2,#RMCHEND-2   ;REACHED END YET ?
011174  003756                  BLE     10$             ;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  002250          MOV     R0,RCVHIADD             ;SAVE RECV HIGH ADDRESS
011222  010137  002252          MOV     R1,RCVLOAD              ;SAVE RECV LOW ADDRESS
011226  005737  003102          TST     KTENABLE                ;TESTING ABOVE 28K?
011232  001403                  BEQ     10$                     ;BR IF NO
011234  004737  020270          JSR     PC,SETMAP               ;RETURN ADDRESS BIASED TO PAR6 IN R0
011240  010001                  MOV     R0,R1                   ;GET RETURNED ADDRESS BIASED TO PAR6
011242  005004                  10$:    CLR     R4              ;WORD IN BUFFER
011244  005003                  CLR     R3                      ;CLEAR ERROR SEEN FLAG
011246  010205                  MOV     R2,R5                   ;GET EXPD BUFFER ADDRESS
011250  011264  002266          15$:    MOV     (R2),EXPMSG(R4) ;SAVE EXPD FOR ERROR REPORT
011254  011164  002432          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::
                                SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
020327  000144                  CMP     R3,#RECMSG-EXPMSG;@@D IS COUNT ABOVE MAX ALLOWED?
003412                          BLE     5$                      ;@@D BR IF NO
012703  000144                  MOV     #RECMSG-EXPMSG,R3;@@D
                                PRINTF  #DEBUGMSG               ;@@D
012746  011464                  MOV     #DEBUGMSG,-(SP)
012746  000001                  MOV     #1,-(SP)
010600                          MOV     SP,R0
104417                          TRAP    C$PNTF
062706  000004                  ADD     #4,SP
010037  002250          5$:     MOV     R0,RCVHIADD             ;SAVE RECV HIGH ADDRESS
010137  002252                  MOV     R1,RCVLOAD              ;SAVE RECV LOW ADDRESS
005737  003102                  TST     KTENABLE                ;TESTING ABOVE 28K?
001403                          BEQ     10$                     ;BR IF NO
004737  020270                  JSR     PC,SETMAP               ;RETURN ADDRESS BIASED TO PAR6 IN R0
010001                          MOV     R0,R1                   ;GET RETURNED ADDRESS BIASED TO PAR6
005004          10$:    CLR     R4                              ;WORD IN BUFFER
005005                          CLR     R5                      ;CLEAR ERROR SEEN FLAG
111264  002266          15$:    MOVB    (R2),EXPMSG(R4)         ;SAVE EXPD FOR ERROR REPORT
111164  002432                  MOVB    (R1),RECMSG(R4)         ;SAVE RECV FOR ERROR REPORT
122221                          CMPB    (R2)+,(R1)+             ;EXPD EQUAL RECV?
001401                          BEQ     25$                     ;BR IF YES
005205                          INC     R5                      ;SET ERROR SEEN FLAG
062704  000001          25$:    ADD     #1,R4                   ;POINT TO NEXT BYTE
020403                          CMP     R4,R3                   ;DONE ALL BYTES?
002001                          BGE     50$                     ;BR IF YES
000764                          BR      15$                     ;DO NEXT BYTE
005705          50$:    TST     R5                              ;ANY ERRORS SEEN?
001402                          BEQ     55$                     ;BR IF NO
```

```
2221 011454  000241                        CLC                     ;SET FAILURE
2222 011456  000401                        BR      60$             ;
2223 011460  000261                55$:     SEC                     ;SET SUCCESS
2224 011462  000207                60$:     RTS     PC              ;RETURN
2225
2226 011464     120     122     117  DEBUGMSG:       .ASCIZ  'PROGRAM INTERNAL ERROR -CKMSG2 MESSAGE BUFFER EXCEEDED-';@@D
2227 011554     045     116     045  FERCM:  .ASCII  /%N%A  %%%/
2228 011565     040     040     124  ERCM:   .ASCIZ  /  TSSR ERROR CODE REC'D = /
2229 011620     056     056     056  SIMSG:  .ASCIZ  /.... AFTER DOING SOFT INIT/
2230 011653     124     105     123  TINERR: .ASCIZ  /TEST: .../
2231                                    .EVEN
```

```
2233
2234
2235                                    ;+
2236                                    ;
2237                                    ;PRINT ROUTINE TO FATAL SOFT INIT ERRORS
2238                                    ;
2239                                    ;INPUT:
2240                                    ;
2241                                    ;       R1      CONTENTS OF TSSR AT ERROR
2242                                    ;
2243                                    ;SIDE EFFECTS:
2244                                    ;
2245                                    ;       EXECUTES DROP UNIT TO CEASE TESTING
2246                                    ;
2247                                    ;-
2248
2249 011666                                     BGNMSG  SFIMSG
     011666                             SFIMSG::
2250 011666  004737  005264                     JSR     PC,PRITSSR      ;PRINT CONTENTS OF TSSR REGISTER
2251 011672  004737  020154                     JSR     PC,CKDROP       ;DROP UNIT, IF ALLOWED
2252 011676                                     ENDMSG
     011676                             L10003:
     011676  104423                             TRAP    C$MSG
2253
2254                                    ;+
2255                                    ;PRINT ROUTINE TO PRINT THE CONTENTS OF
2256                                    ;TSSR AND A COMMAND PACKET OTHER THAN GET STATUS COMMAND PACKET.
2257                                    ;
2258                                    ;INPUTS:
2259                                    ;
2260                                    ;       R1      TSSR CONTENTS
2261                                    ;       R4      ADDRESS OF COMMAND PACKET
2262                                    ;
2263                                    ;-
2264
2265 011700                                     BGNMSG  PKTSSR
     011700                             PKTSSR::
2266 011700  004737  005264                     JSR     PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
2267 011704  012700  000004                     MOV     #4,R0           ;NO. OF WORDS IN PACKET
2268 011710  004737  007072                     JSR     PC,PRIPKT       ;PRINT THE CONTENTS OF COMMAND PACKET
2269 011714  013700  002716                     MOV     MESBFA,R0       ;ADDRESS OF MESSAGE BUFFER
2270 011720  005001                             CLR     R1              ;ASSUME NO HIGH MEMORY
2271 011722  004737  014062                     JSR     PC,PRMESS       ;PRINT THE MESSAGE BUFFER ALSO
2272 011726                                     ENDMSG
     011726                             L10004:
     011726  104423                             TRAP    C$MSG
2273
2274                                    ;+
2275                                    ;PRINT ROUTINE TO PRINT THE CONTENTS OF
2276                                    ;TSSR AND A GET STATUS COMMAND PACKET.
2277                                    ;
2278                                    ;INPUTS:
2279                                    ;
2280                                    ;       R1      TSSR CONTENTS
2281                                    ;       R4      ADDRESS OF COMMAND PACKET
2282                                    ;
2283                                    ;-
```

```
2284
2285 011730                                    BGNMSG  PKTGETS
     011730                              PKTGETS::
2286 011730  004737  005264                    JSR     PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
2287 011734  012700  000002                    MOV     #2,R0           ;NO. OF WORDS IN GET STATUS PACKET
2288 011740  004737  007072                    JSR     PC,PRIPKT       ;PRINT THE CONTENTS OF COMMAND PACKET
2289 011744                                    ENDMSG
     011744                              L10005:
     011744  104423                            TRAP    C$MSG
2290
2291
2292                                        ;+
2293                                        ;PRINT TSSR ERRORS FOR INITIALIZATION TESTS
2294                                        ;
2295                                        ;INPUTS:
2296                                        ;
2297                                        ;       R1      TSSR CONTENTS
2298                                        ;       R4      ADDRESS OF COMMAND PACKET
2299                                        ;-
2300
2301 011746                                    BGNMSG  SFFMSG
     011746                              SFFMSG::
2302 011746  004737  005264                    JSR     PC,PRITSSR      ;PRINT CONTENTS OF TSSR REGISTER
2303 011752                                    ENDMSG
     011752                              L10006:
     011752  104423                            TRAP    C$MSG
2304
2305
2306                                            .SBTTL  PKTMES  - PRINT TSSR AND MESSAGE BUFFER
2307                                        ;+
2308                                        ;
2309                                        ;PRINT ROUTINE TO PRINT THE CONTENTS OF TSSR AND MESSAGE
2310                                        ;BUFFER FOR ERROR REPORTS
2311                                        ;
2312                                        ;INPUTS:
2313                                        ;
2314                                        ;       R1      CONTENTS OF TSSR
2315                                        ;       R2      LOW ORDER MESSAGE BUFFER
2316                                        ;       R3      HIGH ORDER MESSAGE BUFFER ADDRESS
2317                                        ;       NOTE: R3 IS IGNORED IF KTENABLE FLAG IS CLEAR
2318                                        ;-
2319 011754                                    BGNMSG  PKTMES
     011754                              PKTMES::
2320 011754  004737  005264                    JSR     PC,PRITSSR      ;PRINT CONTENTS OF TSSR
2321 011760  010200                            MOV     R2,R0           ;LOW ORDER ADDRESS
2322 011762  010301                            MOV     R3,R1           ;HIGH ORDER ADDRESS
2323 011764  004737  014062                    JSR     PC,PRMESS       ;PRINT THE MESSAGE BUFFER
2324 011770                                    ENDMSG
     011770                              L10007:
     011770  104423                            TRAP    C$MSG
2325
```

```
                                        .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
                                ;-
                                        BGNMSG  ADDSSR
                                ADDSSR::
                                        JSR     PC,PRITADD      ;PRINT MEMORY TEST ADDRESS
                                        MOV     TSSR(R5),R1     ;GET CURRENT TSSR
                                        JSR     PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
                                        ENDMSG
                                L10010:
                                        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
                                ;-
                                        BGNMSG  MSGEXP
                                MSGEXP::
                                        MOV     #7,R0           ;ASSUME NO EXT FEATURES
                                5$:     JSR     PC,PRMSGEXP     ;PRINT EXPD/RECV MESSAGE BUFFERS
                                        ENDMSG
                                L10011:
                                        TRAP    C$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
012022  010146                          MOV     R1,-(SP)
012024  012746  012074                  MOV     #FIF1MSG,-(SP)
012030  012746  000002                  MOV     #2,-(SP)
012034  010600                          MOV     SP,R0
012036  104415                          TRAP    C$PNTX
012040  062706  000006                  ADD     #6,SP
                                        PRINTX  #FIF2MSG        ;PRINT HEADER MSG
012044  012746  012143                  MOV     #FIF2MSG,-(SP)
012050  012746  000001                  MOV     #1,-(SP)
012054  010600                          MOV     SP,R0
012056  104415                          TRAP    C$PNTX
012060  062706  000004                  ADD     #4,SP
012064  010100                          MOV     R1,R0           ;GET BYTE COUNT
012066  004737  015772                  JSR     PC,PRBYTEXP     ;PRINT FIFO BYTES IN ERROR
012072                                  ENDMSG
012072                          L10012:
012072  104423                          TRAP    C$MSG
012074     045     116     045  FIF1MSG:        .ASCIZ  '%N%A NUMBER OF BYTES TRANSFERRED = %D2'
012143     045     116     045  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$:    MOV     #10.,R0         ;NUMBER OF WORDS IN A READ STATUS BUFFER
012236  004737  015422                  JSR     PC,PRMSGEXP     ;PRINT EXPD/RECV MESSAGE BUFFERS
012242                                  ENDMSG
012242                          L10013:
012242  104423                          TRAP    C$MSG

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

```
2436 012722  012100                 10$:    MOV     (R1)+,R0        ;DONE ALL MSG LINES?
2437 012724  001410                         BEQ     20$             ;BR IF YES
2438 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$PNTX
     012740  062706  000004                 ADD     #4,SP
2439 012744  000766                         BR      10$             ;DO ANOTHER MSG LINE
2440 012746  012700  000012         20$:    MOV     #10.,R0         ;NUMBER OF WORDS IN A READ STATUS BUFFER
2441 012752  004737  015422                 JSR     PC,PRMSGEXP     ;PRINT EXPD/RECV MESSAGE BUFFERS
2442 012756                                 ENDMSG
     012756                         L10014:
     012756  104423                         TRAP    C$MSG
2443
2444 012760  013000  013053  013152 LOOPCOD:        .WORD   1$,2$,3$,4$,5$,6$,7$,0
2445 013000     045     116     045 1$:.ASCIZ  '<N><A Tape Bus Loopback Signals in Word #8:'
2446 013053     045     116     045 2$:.ASCIZ  '<N><A      PARERR<15>          IRESV2<14>          IRESV1<13>'
2447 013152     045     116     045 3$:.ASCIZ  '<N><A IHISP=>IEOT<12>     IWRT=>IIDENT<11>     IREV  =>ICER  <10>'
2448 013251     045     116     045 4$:.ASCIZ  '<N><A IWFM =>IFMK<09>     IEDIT=>IHER <08>     IFAD  =>ISPEED<07>'
2449 013350     045     116     045 5$:.ASCIZ  '<N><A ITAD0=>IRDY<06>     ITAD1=>IONL <05>     IERASE=>ILDP  <04>'
2450 013447     045     116     045 6$:.ASCIZ  '<N><A IREW =>IDBY<03>     IRWU =>IRWD <02>     IFEN  =>IFBY  <01>'
2451 013546     045     116     045 7$:.ASCIZ  '<N><A IGO  =>IFPT<00>'
2452                                        .EVEN
2453
```

```
                                        .SBTTL  MSGSUB - PRINT WRITE SUBSYSTEM MESSAGE BUFFER
                                ;+
                                ;
                                ;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
                                ;-
013574                                  BGNMSG  MSGSUB
013574                          MSGSUB::
013574  012700  000012                    MOV     #10.,R0         ;SIZE OF WRITE SUBSYSTEM BUFFER
013600  004737  015422                    JSR     PC,PRMSGEXP     ;PRINT EXPD/RECV MESSAGE BUFFERS
013604                                  ENDMSG
013604                          L10015:
013604  104423                          TRAP    C$MSG


                                        .SBTTL  MEMADD - PRINT MEMORY ADDRESS DATA ERROR
                                ;+
                                ;
                                ;PRINT ROUTINE TO PRINT MEMORY ADDRESS DATA COMPARE ERROR
                                ;
                                ;IMPLICIT INPUTS:
                                ;
                                ;       ERRHI   - MEMORY ERROR HIGH ORDER ADDRESS
                                ;       ERRLO   - MEMORY ERROR LOW ORDER ADDRESS
                                ;       EXP     - EXPECTED DATA
                                ;       RECV    - RECEIVED DATA
                                ;-
013606                                  BGNMSG  MEMADD
013606                          MEMADD::
013606  004737  007712                    JSR     PC,PRIADD       ;PRINT MEMORY ADDRESS IN ERROR
013612  013701  002176                    MOV     EXPD,R1         ;GET EXPD DATA
013616  013702  002200                    MOV     RECV,R2         ;GET RECEIVED DATA
013622  004737  007474                    JSR     PC,PRIXOR       ;PRINT EXPD/RECV
013626                                  ENDMSG
013626                          L10016:
013626  104423                          TRAP    C$MSG
```

```
                                        .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:
                                        SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
                                        MOV     #RAMDATA,R1             ;DATA FROM THE RAM
                                        CLR     R2                      ;INIT BYTE NUMBER
                                5$:     CMPB    (R1)+,(R4)+             ;COMPARE EXPECTED, RECEIVED
                                        BNE     7$                      ;BR IF NO MATCH
                                7$:     MOVB    -1(R1),R5               ;GET RECV RAM DATA
                                        MOVB    -1(R4),R3               ;GET EXPD PACKET DATA
                                        XOR     R5,R3                   ;XOR EXPD/RECV
                                        BIC     #177400,R3              ;LOW BYTE ONLY
                                        MOVB    -1(R1),RECV             ;GET RECEIVED RAM DATA
                                        MOVB    -1(R4),EXPD             ;GET EXPECTED RAM DATA
                                        PRINTB  #RAMASC,R2,RECV,EXPD,R3
                                        MOV     R3,-(SP)
                                        MOV     EXPD,-(SP)
                                        MOV     RECV,-(SP)
                                        MOV     R2,-(SP)
                                        MOV     #RAMASC,-(SP)
                                        MOV     #5,-(SP)
                                        MOV     SP,R0
                                        TRAP    C$PNTB
                                        ADD     #14,SP
                                10$:    INC     R2                      ;UPDATE BYTE COUNT
                                        TST     RAMSIZ                  ;DEFAULT TO 8.?
                                        BEQ     15$                     ;BR IF YES
                                        CMP     R2,RAMSIZ               ;DONE ALL BYTES?
                                        BLE     5$                      ;BR IF NO
                                        BR      25$                     ;
                                15$:    CMP     R2,#8.                  ;DONE DEFAULT NUMBER OF BYTES?
                                20$:    BLT     5$                      ;BR IF NO
                                25$:    CLR     RAMSIZ                  ;SET DEFAULT RAMSIZ
                                        RTS     PC                      ;RETURN
                                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  003102                  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$:    MOV     R1,R3           ;SAVE HIGH ORDER ADDRESS
014106  006100                          ROL     R0              ;SHIFT BIT15 TO C BIT
014110  006101                          ROL     R1              ;SHIFT TO HIGH ORDER FOR PRINTOUT
014112                                  PRINTX  #PROASC,R1,R5    ;PRINT MESSAGE BUFFER ADDRESS
014112  010546                          MOV     R5,-(SP)
014114  010146                          MOV     R1,-(SP)
014116  012746  014714                  MOV     #PROASC,-(SP)
014122  012746  000003                  MOV     #3,-(SP)
014126  010600                          MOV     SP,R0
014130  104415                          TRAP    C$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  014634                  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  014761                  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  020270                  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 ="
```

J6


```
        014222  012546                          MOV     (R5)+,-(SP)
        014224  012746  015017                  MOV     #MESHEA,-(SP)
        014230  012746  000002                  MOV     #2,-(SP)
        014234  010600                          MOV     SP,R0
        014236  104415                          TRAP    C$PNTX
        014240  062706  000006                  ADD     #6,SP
   2584 014244                                  PRINTX  #DATAFL,(R5)+   ;PRINT "DATA FIELD LENGTH      ="
        014244  012546                          MOV     (R5)+,-(SP)
        014246  012746  015064                  MOV     #DATAFL,-(SP)
        014252  012746  000002                  MOV     #2,-(SP)
        014256  010600                          MOV     SP,R0
        014260  104415                          TRAP    C$PNTX
        014262  062706  000006                  ADD     #6,SP
   2585 014266                                  PRINTX  #RBPCRA,(R5)+   ;PRINT "RESIDUAL BYTE COUNTER ="
        014266  012546                          MOV     (R5)+,-(SP)
        014270  012746  015131                  MOV     #RBPCRA,-(SP)
        014274  012746  000002                  MOV     #2,-(SP)
        014300  010600                          MOV     SP,R0
        014302  104415                          TRAP    C$PNTX
        014304  062706  000006                  ADD     #6,SP
   2586 014310                                  PRINTX  #XSOCON,(R5)+   ;PRINT "XSTATO CONTENTS        ="
        014310  012546                          MOV     (R5)+,-(SP)
        014312  012746  015176                  MOV     #XSOCON,-(SP)
        014316  012746  000002                  MOV     #2,-(SP)
        014322  010600                          MOV     SP,R0
        014324  104415                          TRAP    C$PNTX
        014326  062706  000006                  ADD     #6,SP
   2587 014332                                  PRINTX  #XS1CON,(R5)+   ;PRINT "XSTAT1 CONTENTS        ="
        014332  012546                          MOV     (R5)+,-(SP)
        014334  012746  015243                  MOV     #XS1CON,-(SP)
        014340  012746  000002                  MOV     #2,-(SP)
        014344  010600                          MOV     SP,R0
        014346  104415                          TRAP    C$PNTX
        014350  062706  000006                  ADD     #6,SP
   2588 014354                                  PRINTX  #XS2CON,(R5)+   ;PRINT "XSTAT2 CONTENTS        ="
        014354  012546                          MOV     (R5)+,-(SP)
        014356  012746  015310                  MOV     #XS2CON,-(SP)
        014362  012746  000002                  MOV     #2,-(SP)
        014366  010600                          MOV     SP,R0
        014370  104415                          TRAP    C$PNTX
        014372  062706  000006                  ADD     #6,SP
   2589 014376                                  PRINTX  #XS3CON,(R5)+   ;PRINT "XSTAT3 CONTENTS        ="
        014376  012546                          MOV     (R5)+,-(SP)
        014400  012746  015355                  MOV     #XS3CON,-(SP)
        014404  012746  000002                  MOV     #2,-(SP)
        014410  010600                          MOV     SP,R0
        014412  104415                          TRAP    C$PNTX
        014414  062706  000006                  ADD     #6,SP
   2590 014420  022737  000001  002134          CMP     #1,TRANSTST     ;CHECK FOR DUMP
   2591 014426  001042                          BNE     50$             ;BR, IF NO DUMP REQUIRED
   2592 014430                                  PRINTX  #RAMFHR
        014430  012746  014536                  MOV     #RAMFHR,-(SP)
        014434  012746  000001                  MOV     #1,-(SP)
        014440  010600                          MOV     SP,R0
        014442  104415                          TRAP    C$PNTX
        014444  062706  000004                  ADD     #4,SP
   2593 014450  012737  000010  002246          MOV     #8.,RAMSIZ      ;RAM FIELD IS 8 BYTES LONG
```

PRMESS - PRINT CONTENTS OF MESSAGE BUFFER

```
2594 014456  012737  000020  011020          MOV     #20,RAMHLD      ;FIELD STARTS AT 20 OCTAL (10 HEX)
2595 014464  004737  010636                  JSR     PC,RAMER        ;READ AND PRINT THEM
2596 014470  012737  000040  011020          MOV     #40,RAMHLD      ;FIELD STARTS AT 40 OCTAL (20 HEX)
2597 014476  004737  010636                  JSR     PC,RAMER        ;READ AND PRINT THEM
2598 014502  012737  000060  011020          MOV     #60,RAMHLD      ;FIELD STARTS AT 60 OCTAL (30 HEX)
2599 014510  004737  010636                  JSR     PC,RAMER        ;READ AND PRINT THEM
2600 014514  012737  000020  002246          MOV     #16.,RAMSIZ     ;RAM FIELD IS SIXTEEN BYTES LONG
2601 014522  012737  000100  011020          MOV     #100,RAMHLD     ;FIELD STARTS AT 100 OCTAL (40 HEX)
2602 014530  004737  010636                  JSR     PC,RAMER        ;READ AND PRINT THEM
2603 014534  000207                  50$:    RTS     PC              ;RETURN
2604 014536     045     116     045  RAMFHR: .ASCIZ  '<N><A ********** SPECIAL CONTROLLER RAM MEMORY DUMP **********'
2605 014634     045     116     045  MESBFN: .ASCIZ  '<N><A MESSAGE BUFFER CONTENTS PROBABLY NOT VALID'
2606 014714     045     116     045  PROASC: .ASCIZ  '<N><A Message Buffer Address = <01><05>'
2607 014761     045     116     045  PR1ASC: .ASCIZ  '<N><A Message Buffer Contents:'
2608
2609 015017     045     116     045  MESHEA: .ASCIZ  '<N><A Message Buffer Header     = <06>'
2610 015064     045     116     045  DATAFL: .ASCIZ  '<N><A Data Field Length         = <06>'
2611 015131     045     116     045  RBPCRA: .ASCIZ  '<N><A Residual Byte Counter     = <06>'
2612 015176     045     116     045  XS0CON: .ASCIZ  '<N><A XSTAT0 Contents           = <06>'
2613 015243     045     116     045  XS1CON: .ASCIZ  '<N><A XSTAT1 Contents           = <06>'
2614 015310     045     116     045  XS2CON: .ASCIZ  '<N><A XSTAT2 Contents           = <06>'
2615 015355     045     116     045  XS3CON: .ASCIZ  '<N><A XSTAT3 Contents           = <06>'
2616                                  .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
                                ;-
015422                          PRMSGEXP::
015422                                  SAVREG                  ;SAVE R1-R5 UNTIL NEXT RETURN
015426  010005                          MOV     R0,R5           ;SAVE NUMBER OF WORDS
015430  013700  002252                  MOV     RCVLOADD,R0     ;GET RECV LOW ADDRESS
015434  010004                          MOV     R0,R4           ;COPY LOW ADDRESS
015436  013701  002250                  MOV     RCVHIADD,R1     ;GET RECV HIGH ADDRESS
015442  006100                          ROL     R0              ;SHIFT BIT15 TO C BIT
015444  006101                          ROL     R1              ;SHIFT TO HIGH ORDER FOR PRINTOUT
015446                                  PRINTX  #PRMSG0,R1,R4   ;PRINT MESSAGE BUFFER ADDRESS
015446  010446                          MOV     R4,-(SP)
015450  010146                          MOV     R1,-(SP)
015452  012746  015602                  MOV     #PRMSG0,-(SP)
015456  012746  000003                  MOV     #3,-(SP)
015462  010600                          MOV     SP,R0
015464  104415                          TRAP    C$PNTX
015466  062706  000010                  ADD     #10,SP
015472                                  PRINTX  #PRMSG1         ;PRINT HEADER FOR CONTENTS
015472  012746  015647                  MOV     #PRMSG1,-(SP)
015476  012746  000001                  MOV     #1,-(SP)
015502  010600                          MOV     SP,R0
015504  104415                          TRAP    C$PNTX
015506  062706  000004                  ADD     #4,SP
015512  005004                          CLR     R4              ;NUMBER OF THE CURRENT WORD
015514  012701  002266                  MOV     #EXPMSG,R1      ;GET EXPD BUFFER ADDRESS
015520  012702  002432                  MOV     #RECMSG,R2      ;GET RECV BUFFER ADDRESS
015524  011100                  20$:    MOV     (R1),R0         ;GET EXPD
015526  011203                          MOV     (R2),R3         ;GET RECV
015530                                  XOR     R0,R3           ;XOR EXPD/RECV
015540                                  PRINTX  #PRMSG2,R4,(R1)+,(R2)+,R3
015540  010346                          MOV     R3,-(SP)
015542  012246                          MOV     (R2)+,-(SP)
015544  012146                          MOV     (R1)+,-(SP)
015546  010446                          MOV     R4,-(SP)
015550  012746  015705                  MOV     #PRMSG2,-(SP)
015554  012746  000005                  MOV     #5,-(SP)
015560  010600                          MOV     SP,R0
015562  104415                          TRAP    C$PNTX
015564  062706  000014                  ADD     #14,SP
015570  005204                          INC     R4              ;NUMBER OF THE NEXT
015572  020405                          CMP     R4,R5           ;DONE ALL YET?
015574  002001                          BGE     50$             ;BR IF YES
015576  000752                          BR      20$             ;DO ANOTHER
015600  000207                  50$:    RTS     PC              ;RETURN
```

M6


```
2654
2655 015602     045     116     045  PRMSG0: .ASCIZ  '%N%A Message Buffer Address = %01%05'
2656 015647     045     116     045  PRMSG1: .ASCIZ  '%N%A Message Buffer Contents:'
2657 015705     045     116     045  PRMSG2: .ASCIZ  '%N%A   WORD 0%D2%A  EXPD: %06%A  RECV: %06%A  XOR: %06'
2658                                          .EVEN
2659
```

```
                                        .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
                                ;-
015772                          PRBYTEXP::
015772                                  SAVREG                          ;SAVE R1-R5 UNTIL NEXT RETURN
015776  010005                          MOV     R0,R5                   ;SAVE NUMBER OF BYTES
016000  005037  002264                  CLR     PRMNO                   ;INIT ERROR COUNT
016004  005004                          CLR     R4                      ;NUMBER OF THE CURRENT BYTE
016006  012701  002266                  MOV     #EXPMSG,R1              ;GET EXPD BUFFER ADDRESS
016012  012702  002432                  MOV     #RECMSG,R2              ;GET RECV BUFFER ADDRESS
016016  111100                  20$:    MOVB    (R1),R0                 ;GET EXPD BYTE
016020  042700  177400                  BIC     #^C<377>,R0             ;CLEAR UPPER BYTE
016024  110037  016340                  MOVB    R0,PRBEXP               ;SAVE FOR ERROR REPORT
016030  111203                          MOVB    (R2),R3                 ;GET RECV BYTE
016032  042703  177400                  BIC     #^C<377>,R3             ;CLEAR UPPER BYTE
016036  110337  016342                  MOVB    R3,PRBREC               ;FOR ERROR REPORT
016042                                  XOR     R0,R3                   ;XOR EXPD/RECV
016052  122122                          CMPB    (R1)+,(R2)+             ;EXPD = RECV?
016054  001431                          BEQ     30$                     ;BR IF YES
016056  005237  002264                  INC     PRMNO                   ;UPDATE ERROR COUNT
016062  023727  002264  000010          CMP     PRMNO,#8.               ;PRINTED 8?
016070  101023                          BHI     30$                     ;BR IF YES
016072                          27$:    PRINTX  #PRBMSG,R4,PRBEXP,PRBREC,R3
016072  010346                          MOV     R3,-(SP)
016074  013746  016342                  MOV     PRBREC,-(SP)
016100  013746  016340                  MOV     PRBEXP,-(SP)
016104  010446                          MOV     R4,-(SP)
016106  012746  016206                  MOV     #PRBMSG,-(SP)
016112  012746  000005                  MOV     #5,-(SP)
016116  010600                          MOV     SP,R0
016120  104415                          TRAP    C$PNTX
016122  062706  000014                  ADD     #14,SP
016126                                  FORCEXIT        50$             ;BUG
016136  000404                          BR      35$                     ;BUG
016140                          30$:
016140                                  FORCERROR  27$,NOTSSR           ;BUG
016150                          35$:                                    ;BUG
016150  005204                          INC     R4                      ;NUMBER OF THE NEXT
016152  020405                          CMP     R4,R5                   ;DONE ALL YET?
016154  002001                          BGE     50$                     ;BR IF YES
016156  000717                          BR      20$                     ;DO ANOTHER
016160                          50$:    PRINTX  #PRBTOT,PRMNO           ;PRINT TOTAL ERROR COUNT
016160  013746  002264                  MOV     PRMNO,-(SP)
016164  012746  016273                  MOV     #PRBTOT,-(SP)
016170  012746  000002                  MOV     #2,-(SP)
016174  010600                          MOV     SP,R0
016176  104415                          TRAP    C$PNTX
```

```
        016200  062706  000006                  ADD     #6,SP
2704    016204  000207                          RTS     PC              ;RETURN
2705
2706    016206    045     116     045   PRBMSG: .ASCIZ  '%N%A   BYTE #%D2%A  EXPD: #03%A  RECV: #03%A  XOR: #03'
2707    016273    045     116     045   PRBTOT: .ASCIZ  '%N%A NUMBER OF BYTES IN ERROR = #D2'
2708                                      .EVEN
2709    016340  000000                  PRBEXP: .WORD   0               ;EXPD
2710    016342  000000                  PRBREC: .WORD   0               ;RECV
2711
```

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

016344                                  BGNMSG  EXPREC
016344                          EXPREC::
016344  004737  007474                  JSR     PC,PRIXOR       ;PRINT THE DATA
016350                                  ENDMSG
016350                          L10017:
016350  104423                          TRAP    C$MSG
```

```
2731                                            .SBTTL  EXPBREC - PRINT EXPD/RECV BYTE DATA
2732                                    ;+
2733                                    ;
2734                                    ;PRINT ROUTINE TO DISPLAY BYTE EXPD/RECV DATA
2735                                    ;
2736                                    ;
2737                                    ;INPUTS:
2738                                    ;
2739                                    ;       R1      RECEIVED DATA BYTE
2740                                    ;       R2      EXPECTED DATA BYTE
2741                                    ;
2742                                    ;-
2743
2744 016352                                     BGNMSG  EXPBREC
     016352                             EXPBREC::
2745 016352  004737  007344                      JSR     PC,PRIBXOR      ;PRINT THE DATA
2746 016356                                     ENDMSG
     016356                             L10020:
     016356  104423                             TRAP    C$MSG
2747
2748
2749
2750
2751                                            .SBTTL  RAMERR  - PRINT RAM AND PACKET DATA
2752                                    ;+
2753                                    ;
2754                                    ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
2755                                    ;
2756                                    ;INPUTS:
2757                                    ;
2758                                    ;       R4      POINTER TO COMMAND PACKET
2759                                    ;
2760                                    ;IMPLICIT INPUTS:
2761                                    ;
2762                                    ;       RAMDATA         DATA AS READ FROM THE RAM
2763                                    ;       RAMSIZ          NUMBER OF BYTES IN PACKET
2764                                    ;                       IF RAMSIZ=0 THEN DEFAULT TO 8.
2765                                    ;
2766                                    ;IMPLICIT OUTPUTS:
2767                                    ;
2768                                    ;       RAMSIZ  SET TO 0
2769                                    ;-
2770
2771 016360                                     BGNMSG  RAMERR
     016360                             RAMERR::
2772 016360  004737  013630                      JSR     PC,PRAMPKT      ;PRINT RAM/PACKET DATA
2773 016364                                     ENDMSG
     016364                             L10021:
     016364  104423                             TRAP    C$MSG
2774
2775
2776                                            .SBTTL  RAMTADD - PRINT TEST ADDRESS, RAM AND PACKET DATA
2777                                    ;+
2778                                    ;
2779                                    ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
2780                                    ;
2781                                    ;INPUTS:
```

```
2782                                    ;
2783                                    ;       R4      POINTER TO COMMAND PACKET
2784                                    ;
2785                                    ;IMPLICIT INPUTS:
2786                                    ;
2787                                    ;       RAMDATA         DATA AS READ FROM THE RAM
2788                                    ;       RAMSIZ          NUMBER OF BYTES IN PACKET
2789                                    ;                       IF RAMSIZ=0 THEN DEFAULT TO 8.
2790                                    ;       ERRHI           HIGH ORDER TEST ADDRESS
2791                                    ;       ERRLO           LOW ORDER TEST ADDRESS
2792                                    ;
2793                                    ;IMPLICIT OUTPUTS:
2794                                    ;
2795                                    ;       RAMSIZ  SET TO 0
2796                                    ;-
2797
2798 016366                                      BGNMSG  RAMTADD
       016366                           RAMTADD::
2799 016366  004737  010026                       JSR     PC,PRITADD      ;PRINT TEST ADDRESS
2800 016372  004737  013630                       JSR     PC,PRAMPKT      ;PRINT RAM/PACKET DATA
2801 016376                                       ENDMSG
       016376                           L10022:
       016376  104423                           TRAP    C$MSG
2802
2803
2804                                            .SBTTL  RAMEXP  - PRINT RAM EXPD/RECV DATA
2805                                    ;+
2806                                    ;
2807                                    ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
2808                                    ;
2809                                    ;INPUTS:
2810                                    ;
2811                                    ;       R1      RECEIVED DATA
2812                                    ;       R2      EXPECTED DATA
2813                                    ;       R4      CONTROLLER RAM ADDRESS
2814                                    ;-
2815
2816 016400                                      BGNMSG  RAMEXP
       016400                           RAMEXP::
2817 016400  042701  177400                       BIC     #^C<377>,R1     ;SAVE EXPD RAM DATA BYTE
2818 016404  042702  177400                       BIC     #^C<377>,R2     ;SAVE EXPD RAM DATA BYTE
2819 016410  004737  007620                       JSR     PC,PRIRAM       ;PRINT THE RAM ADDRESS
2820 016414  004737  007474                       JSR     PC,PRIXOR       ;PRINT THE DATA
2821 016420                                       ENDMSG
       016420                           L10023:
       016420  104423                           TRAP    C$MSG
2822
2823                                            .SBTTL  TIMEXP  - PRINT TIMER A,B AND EXP/REC
2824                                    ;+
2825                                    ;
2826                                    ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
2827                                    ;AND TIMER A,B HEADER MESSAGE
2828                                    ;
2829                                    ;INPUTS:
2830                                    ;
2831                                    ;       R1      RECEIVED DATA
2832                                    ;       R2      EXPECTED DATA
```

```
2833                                    ;-              ,
2834
2835 016422                                             BGNMSG  TIMEXP
     016422                             TIMEXP::
2836 016422                                             PRINTX  #TIMSG0         ;PRINT HEADER
     016422  012746  016450                             MOV     #TIMSG0,-(SP)
     016426  012746  000001                             MOV     #1,-(SP)
     016432  010600                                     MOV     SP,R0
     016434  104415                                     TRAP    C$PNTX
     016436  062706  000004                             ADD     #4,SP
2837 016442  004737  007474                             JSR     PC,PRIXOR       ;PRINT THE DATA
2838 016446                                             ENDMSG
     016446                             L10024:
     016446  104423                                     TRAP    C$MSG
2839
2840
2841 016450     045     116     045    TIMSG0: .ASCIZ  '%N%A TIMER A STATUS IS IN BIT 3%N%A TIMER B STATUS IS IN BIT 2'
2842                                    .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)
                                ;
                                ;-

016550                                  BGNMSG  BADSSR
016550                          BADSSR::
016550  010246                          MOV     R2,-(SP)        ;SAVE DATA TRANSFERRED
016552  042702  177400                  BIC     #177400,R2      ;GET JUST ONE BYTE
016556                                  PRINTB  #XFERASC,R2
016556  010246                          MOV     R2,-(SP)
016560  012746  016610                  MOV     #XFERASC,-(SP)
016564  012746  000002                  MOV     #2,-(SP)
016570  010600                          MOV     SP,R0
016572  104414                          TRAP    C$PNTB
016574  062706  000006                  ADD     #6,SP
016600  012602                          MOV     (SP)+,R2        ;RESTORE R2
016602  004737  005264                  JSR     PC,PRITSSR      ;DECODE TSSR CONTENTS
016606                                  ENDMSG
016606                          L10025:
016606  104423                          TRAP    C$MSG
016610     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
                                ;
                                ;-

016644                          SOFINIT::
016644                                  SAVREG                  ; SAVE THE REGISTERS
016650  012765  000000  000000          MOV     #0,TSSR(R5)     ; DO THE INIT.
016656  004737  017120                  JSR     PC,WAITF        ; WAIT FOR SSR
016662  016500  000000                  MOV     TSSR(R5),R0     ;GET THE TSSR REGISTER
016666  010004                          MOV     R0,R4           ;START SETUP OF EXPECTED TSSR
016670  042704  176277                  BIC     #^C<HIADDR!OFL>,R4 ;CLEAR OUT UNUSED BITS
016674  052704  002200                  BIS     #SSR!NBA,R4     ;R4 HAS EXPECTED CONTENTS
016700  020400                          CMP     R4,R0           ;ONLY EXPECTED BITS SET ?
016702  001402                          BEQ     5$              ;BRANCH IF OKAY
016704  000241                          CLC                     ;CLEAR THE CARRY FOR ERROR
016706  000401                          BR      10$             ;GO TO EXIT
016710  000261                  5$:     SEC                     ;SET THE CARRY BIT
016712  000207                  10$:    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
                                ;
                                ;-

016714                          CHKAMB:
016714                                  SAVREG                  ;SAVE THE GENERAL REGISTERS
016720  010004                        MOV     R0,R4           ;CONTENTS OF TSSR
016722  032700  100000                BIT     #SC,R0          ;IS BIT 15 SET ?
016726  001004                        BNE     5$              ;BRANCH IF YES
016730  032700  174077                BIT     #^C<NBA!OFL!SSR!HIADDR>,R0      ;ANY OTHER BITS SET ?
016734  001023                        BNE     40$             ;MUST BE AN ERROR
016736  000424                        BR      45$             ;RETURN WITH SUCCESS
016740  032700  000200        5$:     BIT     #SSR,R0         ;IS READY BIT SET ?
016744  001011                        BNE     10$             ;BRANCH IF READY BIT IS SET.
016746  032700  000040                BIT     #BIT5,R0        ;IS FATAL ERROR BIT SET ?
016752  001414                        BEQ     40$             ;ERROR IF NOT
016754  042704  177761                BIC     #^CTERCLS,R4    ;CLEAR ALL BUT TERMINATION CODE
016760  020427  000016                CMP     R4,#16          ;ALL THREE BITS MUST BE SET
016764  001007                        BNE     40$             ;ERROR IF NOT SET
016766  000410                        BR      45$             ;OK IF ALL ARE SET
016770  032700  000040        10$:    BIT     #BIT5,R0        ;IS FATAL ERROR BIT SET ?
016774  001405                        BEQ     45$             ;ERROR IF BIT IS SET WITH SSR
016776  032700  000006                BIT     #BIT2!BIT1,R0   ;IS THIS A FUNCTION REJECT
017002  001002                        BNE     45$             ;BR, IF TSSR IS OK
017004  000241        40$:    CLC                             ;AMBIGUOUS CONTENTS
017006  000401                        BR      50$
017010  000261        45$:    SEC                             ;SHOW SUCCESS - NO AMBIGUITY
017012  000207        50$:    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
017014     000                  INTMASK:        .BYTE   0
                                ;INTERRUPT FLAG -- SAYS WE GOT ONE (IF POSITIVE)
017015     000                  INTFLAG:        .BYTE   0

                                ;SAVED INTERRUPT VECTOR:
017016  000000                  INTVEC: .WORD   0
                                ;SAVE CPU PC
017020  000000                  INTCPC: .WORD   0

                                ;SUBROUTINE TO ENABLE INTERRUPTS:
017022  010046                  ENAINT: MOV     R0,-(SP)        ;SAVE R0
017024  013700  002156                  MOV     IVEC,R0         ;GET POINTER TO VECTORS
017030  012720  017066                  MOV     #INTR,(R0)+     ;SET UP INTERRUPT VECTOR
017034  012720  000340                  MOV     #PRI07,(R0)+
017040  012600                          MOV     (SP)+,R0        ;RESTORE R0
017042  011646                          MOV     (SP),-(SP)
017044  012766  000000  000002          MOV     #0,2(SP)        ;SET CPU TO LEVEL 0
017052  000002                          RTI
                                ;
                                ;SUBROUTINE TO DISABLE INTERRUPTS (RAISE PRIORITY TO LEVEL 7)
017054  011646                  DSBINT: MOV     (SP),-(SP)
017056  012766  000340  000002          MOV     #PRI07,2(SP)
017064  000002                          RTI
                                ;
```

```
3001                                            .SBTTL  INTR    - INTERRUPT HANDLERS
3002
3003 017066                                     BGNSRV  INTR            ;DEFINE INTERRUPT ENTRY
     017066                             INTR::
3004 017066  012737  000001  002172            MOV     #1,INTRECV      ;SET FLAG TO SHOW INTERRUPT RECEIVED
3005 017074  105037  017015                    CLRB    INTFLAG         ;CLEAR FLAG TO SAY WE GOT INTERRUPT
3006 017100  132737  000001  017014            BITB    #IOKSTP,INTMASK ;EXPECTING STOP INTERRUPT?
3007 017106  001003                            BNE     1$              ;BR IF YES
3008 017110  152737  000001  017015            BISB    #IOKSTP,INTFLAG ;NO. SET THE ERROR FLAG.
3009
3010                                    ;SAVE REGISTERS, MSG BUFFER, ETC.
3011 017116                             1$:
3012 017116                                     ENDSRV
     017116                             L10026:
     017116  000002                             RTI
3013
3014
```

```
                                        .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
                                ;
017120                          WAITF:: BREAK                   ; DO A SUPVSR BREAK FIRST.
017120  104422                          TRAP    C$BRK
017122  012746  177776                  MOV     #177776,-(SP)   ;BIG MSEC TIMER
017126                                  DELAY   1               ;DELAY 100US
017126  012727  000001                  MOV     #1,(PC)+
017132  000000                          .WORD   0
017134  013727  002116                  MOV     L$DLY,(PC)+
017140  000000                          .WORD   0
017142  005367  177772                  DEC     -6(PC)
017146  001375                          BNE     .-4
017150  005367  177756                  DEC     -22(PC)
017154  001367                          BNE     .-20
017156  016500  000000          2$:     MOV     TSSR(R5),R0     ;READ THE TSSR REGISTER
017162  105700                          TSTB    R0              ;TEST FOR READY BIT SET
                                ;
017164  100421                          BMI     3$              ; EXIT ON STOP FLAG.
017166                                  DELAY   1               ; WAIT 100 USEC
017166  012727  000001                  MOV     #1,(PC)+
017172  000000                          .WORD   0
017174  013727  002116                  MOV     L$DLY,(PC)+
017200  000000                          .WORD   0
017202  005367  177772                  DEC     -6(PC)
017206  001375                          BNE     .-4
017210  005367  177756                  DEC     -22(PC)
017214  001367                          BNE     .-20
017216                                  BREAK                   ; DO A SUPVSR BREAK FIRST.
017216  104422                          TRAP    C$BRK
017220  005316                          DEC     (SP)            ;REDUCE DELAY COUNT
017222  001355                          BNE     2$              ;RETRY UNTIL TIMER EXPIRES
017224  000241                          CLC                     ; C = 0, CONTROLLER STILL RUNNING...
017226  000401                          BR      4$              ;...OR HUNG-UP AFTER 300 MSEC.
017230  000261                  3$:     SEC                     ; C = 1, CONTROLLER IS STOPPED.
017232  005326                  4$:     DEC     (SP)+           ;RESTORE STACK WITHOUT CHANGING CARRY BIT
017234  000207                          RTS     PC
```

CHKTSSR - CHECK TSSR FOR READY

```
                                        .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
                                ;
                                ;-

017236                          CHKTSSR:
017236  004737  017120                  JSR     PC,WAITF        ;WAIT FOR READY
017242  103014                          BCC     20$             ;BRANCH IF TIME OUT
017244  004737  016714                  JSR     PC,CHKAMB       ;TSSR AMBIGOUS?
017250  103006                          BCC     10$             ;BR IF YES
017252  032700  100000                  BIT     #SC,R0          ;SPECIAL CONDITION SET?
017256  001405                          BEQ     15$             ;BR IF NO
017260  032700  074000                  BIT     #<SCE!BIE!RMR!NXM>,R0   ;ANY ERROR BITS SET?
017264  001402                          BEQ     15$             ;BR IF NO
017266  000241                  10$:    CLC                     ;SET FAILURE
017270  000401                          BR      20$             ;
017272  000261                  15$:    SEC                     ;SET SUCCESS
017274  000207                  20$:    RTS     PC              ;RETURN TO CALLER
```

```
3080                                            .SBTTL  XNXM    - CHECK FOR NONEXISTENT MEMORY
3081                                    ;+
3082                                    ; ROUTINE TO TEST FOR A NEXM IN THE RANGE (R1) THRU (R2).
3083                                    ; ON RETURN, IF "C" = 1, (R1) = NEXM ADDRESS.
3084                                    ;               "C" = 0, ALL ADDRESSES OK.
3085                                    ;
3086                                    ;CALL:  MOV ADR1,R1
3087                                    ;       MOV ADR2,R2
3088                                    ;       JSR PC,NXM
3089                                    ;       RETURN                  ;TEST "C" AND PROCEED.
3090                                    ;
3091 017276  012737  017330  000004    XNXM:   MOV     #2$,@#4         ; SET BUSERR VECTOR.
3092 017304  012737  000200  000006            MOV     #PRI04,@#6
3093 017312  005003                            CLR     R3              ;FLAG.
3094 017314  005711                    1$:     TST     (R1)            ;TEST THE ADDRESS(ES).
3095                                                                    ;IF ANY TRAP, CONTINUE AT 2$.
3096 017316  020102                            CMP     R1,R2           ;OTHERWISE, CONTINUE HERE.
3097 017320  001407                            BEQ     3$              ;BR IF FINISHED (NO NEXM'S).
3098 017322  062701  000002                    ADD     #2,R1           ;SET NEXT ADDRESS...
3099 017326  000772                            BR      1$              ;...AND CONTINUE.
3100
3101 017330  005103                    2$:     COM     R3              ;GOT ONE, SET FLAG...
3102 017332  012716  017340                    MOV     #3$,(SP)
3103 017336  000002                            RTI                     ;...AND DISMISS INTERRUPT...
3104 017340                            3$:     CLRVEC  #4              ;...AND GIVE BACK THE VECTOR.
     017340  012700  000004                    MOV     #4,R0
     017344  104436                            TRAP    C$CVEC
3105 017346  005703                            TST     R3              ;DID WE CATCH ONE ??
3106 017350  001401                            BEQ     .+4             ;NO, "C" = 0, SKIP NEXT.
3107 017352  000261                            SEC                     ;YES, "C" = 1, (R1) = NEXM ADDR.
3108 017354  000207                            RTS     PC
3109
3110
3111
3112
3113                                            .SBTTL  TSTLOOP - CHECK ITERATION COUNT
3114                                    ;+
3115                                    ; SUBROUTINE TO EXECUTE TEST ITERATIONS.
3116                                    ; EXIT WITH "C" SET IF LOOPS ALLOWED AND LOOP COUNT NON-ZERO.
3117                                    ; LOOP COUNTER IS SET BY "BEGIN.TEST" MACRO.
3118                                    ;
3119                                    ; CALL: LOOPTO  ARG
3120                                    ;
3121 017356                            TSTLOOP::
3122 017356  005737  002136                    TST     NOITS           ; ITERATIONS INHIBITED?
3123 017362  001006                            BNE     1$              ; YES.
3124 017364  005737  002152                    TST     QVP             ; NO.
3125 017370  100403                            BMI     1$              ;LOOPS DISALLOWED IN QUICK PASS.
3126 017372  005337  002164                    DEC     LOOPCNT         ; BUMP LOOP COUNTER.
3127 017376  001002                            BNE     2$
3128 017400  000241                    1$:     CLC                     ;LOOP DISALLOWED, OR DONE.
3129 017402  000401                            BR      3$
3130 017404  000261                    2$:     SEC                     ;LOOP ENABLED.
3131 017406  000207                    3$:     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 RAISED TO LEVEL OF
                ;       THE DEVICE UNDER TEST
                ;
                ;-

017410                  TSTSETUP::
017410  010046          MOV     R0,-(SP)        ;SAVE THE TEST ID MESSAGE
017412  005037  003106  CLR     SIFLAG          ; CLEAR "SOFT INIT" FLAG
017416  005037  017656  CLR     ERRK            ; CLEAR LOCAL ERROR COUNTER.
017422  005037  005232  CLR     EXTA            ; CLEAR ERROR EXTENSION FLAG.
017426  105037  017014  CLRB    INTMASK         ; CLEAR INTERRUPT MASK (CHECK ERROR)
017432  013700  002150  MOV     UNITN,R0        ; GET THE UNIT NUMBER.
017436  006300          ASL     R0              ; ... AND MAKE IT A WORD OFFSET.
017440  005737  003062  TST     NODEV           ; DID STARTUP FIND THE DEVICE?
017444  001430          BEQ     4$              ; BR IF YES
017446  100010          BPL     3$              ; BR IF NOT IDLE
017450  052760  160000  003130  BIS     #160000,ERTABL(R0) ; FLAG ERROR IN THE ERROR TABLE
017456                  ERRDF   1,NXR,NXRERR    ; NO DEVICE HERE -- PRINT IT
017456  104455          TRAP    C$ERDF
017460  000001          .WORD   1
017462  003636          .WORD   NXR
017464  005176          .WORD   NXRERR
017466  000407          BR      2$
017470  052760  160001  003130  3$:     BIS     #160001,ERTABL(R0) ; FLAG ERROR IN THE ERROR TABLE
017476                  ERRDF   2,NOINIT        ; DEVICE NOT IDLE
017476  104455          TRAP    C$ERDF
017500  000002          .WORD   2
017502  004233          .WORD   NOINIT
017504  000000          .WORD   0
017506  012737  177777  003060  2$:     MOV     #-1,DUFLG       ; DROP THE UNIT
017514                  DODU    UNITN
017514  013700  002150  MOV     UNITN,R0
017520  104451          TRAP    C$DODU
017522                  DOCLN                   ; ABORT THE PASS
```

```
     017522  104444                          TRAP    C$DCLN
3180 017524  000423                          BR      5$
3181
3182 017526                          4$:     RFLAGS  R0              ; GET THE OPERATOR FLAGS.
     017526  104421                          TRAP    C$RFLA
3183 017530  032700  001000                  BIT     #PNT,R0         ; PRINT THE TEST NUMBERS?
3184 017534  001412                          BEQ     1$              ; BR IF NO
3185 017536  011600                          MOV     (SP),R0         ;GET THE ID MESSAGE
3186 017540                                  PRINTF  #TNAM,R0        ;DISPLAY THE TEST ID
     017540  010046                          MOV     R0,-(SP)
     017542  012746  017604                  MOV     #TNAM,-(SP)
     017546  012746  000002                  MOV     #2,-(SP)
     017552  010600                          MOV     SP,R0
     017554  104417                          TRAP    C$PNTF
     017556  062706  000006                  ADD     #6,SP
3187 017562  005237  002162          1$:     INC     TSTCNT          ; BUMP TEST COUNTER.
3188 017566                                  SETPRI  IPRI            ;PRIORITY THAT OF DEVICE
     017566  013700  002160                  MOV     IPRI,R0
     017572  104441                          TRAP    C$SPRI
3189 017574  005726                  5$:     TST     (SP)+           ;FIX UP THE STACK
3190 017576  013705  002154                  MOV     CSRADDR,R5      ; ADDRESS OF TSV REGISTERS ON UNIBUS
3191 017602  000207                          RTS     PC
3192 017604     045     123     045  TNAM:   .ASCIZ  '%S%T%A Test'
3193                                          .EVEN
```

```
                                        .SBTTL  TSTEND  - PRINT ERRORS RECEIVED
                                ;
                                ; AT END OF EACH TEST, PRINT THE NUMBER OF ERRORS RECEIVED
                                ; IF NORMAL ERROR REPORTING IS DISABLED (FLA:IER).
                                ;
017620                          TSTEND: RFLAGS  R0
017620  104421                         TRAP    C$RFLA
017622  030027  020000                 BIT     R0,#IER
017626  001412                         BEQ     1$                      ; BR IF "IER" NOT SET.
017630                                  PRINTF  #ESUM,ERRK              ; PRINT ERROR COUNT.
017630  013746  017656                 MOV     ERRK,-(SP)
017634  012746  017660                 MOV     #ESUM,-(SP)
017640  012746  000002                 MOV     #2,-(SP)
017644  010600                         MOV     SP,R0
017646  104417                         TRAP    C$PNTF
017650  062706  000006                 ADD     #6,SP
017654  000207                  1$:     RTS     PC

017656  000000                  ERRK:   0                               ; LOCAL ERROR COUNT.
017660     045     101     040  ESUM:   .ASCIZ  /%A %D%A ERRORS/
017677     105     122     122  EMAXDU: .ASCIZ  /ERROR LIMIT REACHED -- DROPPING UNIT/
                                        .EVEN

                                        .SBTTL  INCERK  - INCREMENT LOCAL ERROR COUNT
                                ;+
                                ; ROUTINES TO INCREMENT LOCAL ERROR COUNT AND CHECK FOR LIMIT:
                                ;-
017744  005237  017656          INCERK: INC     ERRK                    ; INCREMENT LOCAL ERROR COUNT
017750  010046                          MOV     R0,-(SP)                ; SAVE R0
017752  013700  002150                 MOV     UNITN,R0                ; GET UNIT NUMBER,
017756  006300                         ASL     R0                      ; ... AND MAKE IT A WORD OFFSET.
017760  062700  003130                 ADD     #ERTABL,R0              ; R0 GETS ADDRESS OF ERROR TABLE ENTRY.
017764  005210                         INC     (R0)                    ; INCREMENT THE DEVICE ERROR COUNT
017766  032710  007777                 BIT     #7777,(R0)              ; DID WE OVERFLOW THE FIELD?
017772  001001                         BNE     1$                      ; BR IF NO.
017774  005310                         DEC     (R0)                    ; YES -- BACK IT UP TO 7777.
017776  012600                  1$:     MOV     (SP)+,R0                ; RESTORE R0
020000  000207                         RTS     PC                      ; RETURN TO CALLER.

020002  010046                  CKEMAX: MOV     R0,-(SP)                ; SAVE R0
020004  013700  002150                 MOV     UNITN,R0                ; GET UNIT NUMBER
020010  006300                         ASL     R0                      ; ... AND MAKE IT A WORD OFFSET
020012  016000  003130                 MOV     ERTABL(R0),R0           ; GET ERROR TABLE ENTRY
020016  042700  170000                 BIC     #170000,R0              ; EXTRACT ERROR COUNT FIELD
020022  020037  002142                 CMP     R0,GERRMAX              ; IS GLOBAL LIMIT EXCEEDED FOR THIS UNIT?
020026  103004                         BHIS    1$                      ; BR IF YES
020030  023737  017656  002140         CMP     ERRK,LERRMAX            ; IS LOCAL LIMIT EXCEEDED FOR THIS TEST?
020036  103417                         BLO     2$                      ; BR IF NO
020040                          1$:     RFLAGS  R0                      ; GET OPERATOR FLAGS
020040  104421                         TRAP    C$RFLA
020042  032700  000040                 BIT     #IDU,R0                 ; IS DROPPING INHIBITED?
020046  001013                         BNE     2$                      ; BR IF YES.
020050  012737  177777  003060         MOV     #-1,DUFLG               ; NO -- DROP THE UNIT
020056                                  ERRDF   4,EMAXDU
020056  104455                         TRAP    C$ERDF
020060  000004                         .WORD   4
020062  017677                         .WORD   EMAXDU
```

```
      020064  000000                          .WORD   0
 3241 020066                                  DODU    UNITN
      020066  013700  002150                  MOV     UNITN,R0
      020072  104451                          TRAP    C$DODU
 3242 020074                                  DOCLN
      020074  104444                          TRAP    C$DCLN
 3243 020076  012600                  2$:     MOV     (SP)+,R0        ; RESTORE R0
 3244 020100  000207                          RTS     PC              ; RETURN TO CALLER
 3245                                         .SBTTL  FATCHK - INC FATAL ERRORS AND CHECK FOR LIMIT
 3246                                 ;+
 3247                                 ;
 3248                                 ;       CHECK FATAL COUNTER, AFTER INC, FOR MORE THAN 25
 3249                                 ;       ERRORS AND IF OVER CALL UNIT DROP ROUTINE
 3250                                 ;
 3251                                 ;-
 3252 020102                          FATCHK:
 3253 020102                                  SAVREG                          ;BETTER SAVE THE REGISTERS
 3254 020106  013701  002150                  MOV     UNITN,R1                ;PICK UP THE UNIT NUMBER
 3255 020112  006301                          ASL     R1                      ;MAKE IT INTO A BYTE OFFSET
 3256 020114  062761  000001  003130          ADD     #1,ERTABL(R1)           ;ADD 1 TO THE PROPER UNIT'S ERROR COUNTER
 3257 020122  005237  002170                  INC     FATFLG                  ;BUMP FATAL ERROR COUNTER
 3258 020126  023727  002170  000031          CMP     FATFLG,#25.             ;CHECK AGAINST 25
 3259 020134  002406                          BLT     9$                      ;BR. IF LESS THAN 25 ERRORS
 3260 020136                                  RFLAGS  R0                      ;READ THE FLAGS INTO R0
      020136  104421                          TRAP    C$RFLA
 3261 020140  032700  040000                  BIT     #BIT14,R0               ;BR. IF LOOP ON ERROR IS SET
 3262 020144  001002                          BNE     9$                      ;OTHERWISE NEVER BE ABLE TO SCOPE ETC.
 3263 020146  004737  020154                  JSR     PC,CKDROP               ;DROP UNIT IF ALLOWED
 3264 020152  000207                  9$:     RTS     PC                      ;RETURN ETC.
 3265                                 ;
 3266                                 ;
 3267                                 ;
```

```
                                        .SBTTL  CKDROP  - CHECK IF UNIT SHOULD BE DROPPED
                                ;+
                                ; CHECK IF UNIT SHOULD BE DROPPED
                                ;-
020154  010046                 CKDROP: MOV     R0,-(SP)
020156                                  FORCERROR       1$,NOTSSR
020166                                  RFLAGS  R0
020166  104421                         TRAP    C$RFLA
020170  032700  000040                 BIT     #IDU,R0
020174  001010                         BNE     1$
020176  011600                         MOV     (SP),R0
020200  012737  177777  003060         MOV     #-1,DUFLG
020206                                  DODU    UNITN
020206  013700  002150                 MOV     UNITN,R0
020212  104451                         TRAP    C$DODU
020214                                  DOCLN                   ;ABORT THE PASS
020214  104444                         TRAP    C$DCLN
020216  012600                 1$:     MOV     (SP)+,R0
020220  000207                         RTS     PC


                                        .SBTTL  CONFIG  - DETERMINE CONFIGURATION OF SYSTEM
                                ;
                                ; SUBROUTINE - DETERMINE CONFIGURATION OF TK-25 SYSTEM.
                                ;
020222                          CONFIG:
020222  004737  016644                 JSR     PC,SOFINIT
020226  000207                         RTS     PC
```

```
                                        .SBTTL  KTON,KTOFF      - ENABLE/DISABLE MEMORY MANAGEMENT
                                ;
                                ; SUBROUTINE - ENABLE MEM MGT.
                                ;
020230  005737  003100          KTON:   TST     KTFLG           ; GOT KT?
020234  001403                          BEQ     1$              ; NO.
020236  012737  000001  177572          MOV     #1,SR0          ; YES. ENABLE KT11.
020244  000207                  1$:     RTS     PC
                                ;
                                ; SUBROUTINE - DISABLE MEM MGT.
                                ;
020246  005737  003100          KTOFF:  TST     KTFLG           ; GOT KT11?
020252  001405                          BEQ     1$              ; NO.
020254  000240                          NOP
020256  000240                          NOP
020260  012737  000000  177572          MOV     #0,SR0          ; DISABLE KT.
020266  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
                                ;-
020270                          SETMAP:
020270                                  SAVREG                  ;SAVE R1-R4 UNTIL NEXT RETURN
020274  005737  003100                  TST     KTFLG           ;SYSTEM HAVE ABOVE 28K?
020300  001433                          BEQ     10$             ;BR IF NO
020302  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
020334  042701  000177                  BIC     #177,R1         ;ALINE FOR LOWER 4K BOUNDARY
020340  020137  003100                  CMP     R1,KTFLG        ;HIGHER THAN EXISTING MEMORY?
020344  103011                          BHIS    10$             ;BR IF YES
020346  010137  172354                  MOV     R1,@#KIPAR6     ;SETUP MAPPING REGISTER PAR6
020352  042702  160000                  BIC     #160000,R2      ;SETUP DISPLACEMENT IN PAGE
020356  062702  140000                  ADD     #140000,R2      ;ADD IN PAR6 BIAS
020362  010200                          MOV     R2,R0           ;RETURN IN R0
020364  000261                          SEC                     ;SET SUCCESS
020366  000401                          BR      15$             ;
020370  000241                  10$:    CLC                     ;SET FAILURE
020372  000207                  15$:    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
                                ;-
                                ;
020374                          FILLMEM:
020374                                  SAVREG                  ;SAVE R1-R5 UNTIL NEXT RETURN
020400  004737  020246                  JSR     PC,KTOFF        ;DISABLE KT.
020404  010003                          MOV     R0,R3           ;COPY TEST PATTERN
020406  013701  003072                  MOV     FREE,R1         ;GET FIRST FREE LOCATION
020412  013702  003074                  MOV     FRESIZ,R2       ;SIZE OF FREE SPACE BELOW 28K.
020416  010321                  10$:    MOV     R3,(R1)+        ;STORE A BACKGROUND WORD
020420  005302                          DEC     R2              ;DONE ALL MEMORY IN FREE SPACE?
020422  003375                          BGT     10$             ;BR IF NO
020424  005737  003100                  TST     KTFLG           ; GOT KT?
020430  001452                          BEQ     55$             ; NO. GET OUT.
020432  004737  020230                  JSR     PC,KTON         ; YES. ENABLE KT.
020436  005000                          CLR     R0              ;HIGH ORDER ADDRESS START
020440  013701  003104                  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
020510  004737  020270                  JSR     PC,SETMAP       ;SETUP PAR6 MAPPING REGISTER
020514  010320                  30$:    MOV     R3,(R0)+        ;STORE TEST PATTERN IN >28K ADDRESS
020516  020027  160000                  CMP     R0,#160000      ;END OF PAR6 MAPPING AREA?
020522  103774                          BLO     30$             ;BR IF NO
020524  162700  020000                  SUB     #20000,R0       ;BACKUP INTO PAR6 MAPPING BEGIN
020530  062737  000200  172354          ADD     #200,@#KIPAR6   ;POINT TO NEXT 4K BLOCK >28K.
020536  023737  172354  003100          CMP     @#KIPAR6,KTFLG  ;END OF MEMORY?
020544  001402                          BEQ     50$             ;BR IF YES
020546  000137  020514                  JMP     30$             ;KEEP GOING ON ETC.
020552  004737  020246          50$:    JSR     PC,KTOFF        ; DISABLE KT.
020556  000207                  55$:    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
                                ;-
020560                          CMPMEM:
020560                                  SAVREG                  ;SAVE R1-R5 UNTIL NEXT RETURN
020564  010003                          MOV     R0,R3           ;COPY TEST PATTERN
020566  004737  020246                  JSR     PC,KTOFF        ;DISABLE KT.
020572  013701  003072                  MOV     FREE,R1         ;GET FIRST FREE LOCATION
020576  013702  003074                  MOV     FRESIZ,R2       ;SIZE OF FREE SPACE BELOW 28K.
020602  020311                  10$:    CMP     R3,(R1)         ;FREE SPACE LOCATION EQUAL TO EXPD?
020604  001411                          BEQ     15$             ;BR IF YES
020606  010137  002204                  MOV     R1,ERRLO        ;SAVE ADDRESS IN ERROR
020612  005037  002202                  CLR     ERRHI           ;NO HIGH ADDRESS
020616  010337  002176                  MOV     R3,EXPD         ;SAVE EXPD FOR ERROR REPORT
020622  011137  002200                  MOV     (R1),RECV       ;SAVE RECV FOR ERROR REPORT
020626  000474                          BR      50$             ;
020630  005721                  15$:    TST     (R1)+           ;POINT TO NEXT ADDRESS
020632  005302                          DEC     R2              ;DONE ALL MEMORY IN FREE SPACE?
020634  003362                          BGT     10$             ;BR IF NO
020636  005737  003100                  TST     KTFLG           ; GOT KT?
020642  001472                          BEQ     55$             ; NO. GET OUT.
020644  004737  020230                  JSR     PC,KTON         ; YES. ENABLE KT.
020650  005000                          CLR     R0              ;HIGH ORDER ADDRESS START
020652  013701  003104                  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
020706  042701  000177                  BIC     #177,R1         ;ALINE 4K BOUNDARY
020712  010046                          MOV     R0,-(SP)        ;SAVE HIGH ORDER
020714  010146                          MOV     R1,-(SP)        ;SAVE LOW ORDER
020716  004737  020270                  JSR     PC,SETMAP       ;SETUP PAR6 MAPPING REGISTER
020722  010004                          MOV     R0,R4           ;COPY ADDRESS BIASED TO PAR6
020724  012601                          MOV     (SP)+,R1        ;RESTORE LOW ORDER IN NON PAR6 FORMAT
020726  012600                          MOV     (SP)+,R0        ;RESTORE HIGH ORDER IN NON PAR6 FORMAT
020730  020314                  30$:    CMP     R3,(R4)         ;ABOVE 28K LOCATION EQUAL EXPD?
020732  001411                          BEQ     32$             ;BR IF YES
020734  010037  002202                  MOV     R0,ERRHI        ;SAVE HIGH ORDER IN ERROR
```

```
3468 020740  010137  002204                MOV     R1,ERRLO        ;SAVE LOW ORDER IN ERROR
3469 020744  010337  002176                MOV     R3,EXPD         ;SAVE EXPD FOR ERROR REPORT
3470 020750  011437  002200                MOV     (R4),RECV       ;SAVE RECV FOR ERROR REPORT
3471 020754  000421                        BR      50$             ;
3472 020756  062701  000002          32$:  ADD     #2,R1           ;UPDATE NON PAR6 ADDRESS
3473 020762  005500                        ADC     R0              ;MAKE IT DOUBLE PRECISION ADD
3474 020764  062704  000002                ADD     #2,R4           ;UPDATE PAR FORMAT ADDRESS
3475 020770  020427  160000                CMP     R4,#160000      ;END OF PAR6 MAPPING AREA?
3476 020774  103755                        BLO     30$             ;BR IF NO
3477 020776  162704  020000                SUB     #20000,R4       ;BACKUP INTO PAR6 MAPPING BEGIN
3478 021002  062737  000200  172354        ADD     #200,@#KIPAR6   ;POINT TO NEXT 4K BLOCK >28K.
3479 021010  023737  172354  003100        CMP     @#KIPAR6,KTFLG  ;END OF MEMORY?
3480 021016  101744                        BLOS    30$             ;BR IF NO
3481 021020  004737  020246          50$:  JSR     PC,KTOFF        ;TURN OFF MEMORY MAPPING
3482 021024  000241                        CLC                     ;SET FAILURE
3483 021026  000403                        BR      60$             ;
3484 021030  004737  020246          55$:  JSR     PC,KTOFF        ;TURN OFF MEMORY MAPPING
3485 021034  000261                        SEC                     ;SET SUCCESS
3486 021036  000207                60$:  RTS     PC
3487
```

```
                                .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
                        ;
                        ;-

021040                  REGSAV:
021040                          BREAK                   ;LOOK FOR CNTL C
021040  104422                  TRAP    C$BRK
021042  010446                  MOV     R4,-(SP)
021044  010346                  MOV     R3,-(SP)
021046  010246                  MOV     R2,-(SP)
021050  010146                  MOV     R1,-(SP)
021052  010546                  MOV     R5,-(SP)
021054  016605  000012          MOV     10.(SP),R5
021060  004736                  JSR     PC,@(SP)+
021062  012601                  MOV     (SP)+,R1
021064  012602                  MOV     (SP)+,R2
021066  012603                  MOV     (SP)+,R3
021070  012604                  MOV     (SP)+,R4
021072  012605                  MOV     (SP)+,R5
021074                          BREAK                   ;LOOK FOR CNTL C
021074  104422                  TRAP    C$BRK
021076  000207                  RTS     PC
```

```
3527                                            .SBTTL  GETPAT  - GET 8 BIT PATTERN FROM OPERATOR
3528                                    ;+
3529                                    ;
3530                                    ;ROUTINE TO REQUEST AN 8 BIT DATA PATTERN FROM THE OPERATOR
3531                                    ;
3532                                    ;INPUTS:
3533                                    ;
3534                                    ;       NONE.
3535                                    ;
3536                                    ;OUTPUTS:
3537                                    ;
3538                                    ;       R0      OCTAL NUMBER FROM THE OPERATOR
3539                                    ;
3540                                    ;CALLING SEQUENCE:
3541                                    ;
3542                                    ;       JSR     PC,GETPAT
3543                                    ;
3544                                    ;-
3545
3546 021100                             GETPAT::
3547 021100                                     SAVREG                          ;SAVE THE GENERAL REGISTERS
3548 021104                             1$:     GMANID  DATASC,PATDAT,0,377,0,377,NO
     021104  104443                            TRAP    C$GMAN
     021106  000406                            BR      10000$
     021110  021134                            .WORD   PATDAT
     021112  000022                            .WORD   T$CODE
     021114  021136                            .WORD   DATASC
     021116  000377                            .WORD   377
     021120  000000                            .WORD   T$LOLIM
     021122  000377                            .WORD   T$HILIM
     021124                                 10000$:
3549 021124                                     BNCOMPLETE      1$              ;RETRY IF ERROR
     021124  103367                            BCC     1$
3550 021126  013700  021134                    MOV     PATDAT,R0               ;DATA PATTERN FROM OPERATOR
3551 021132  000207                            RTS     PC                      ;RETURN TO CALLER
3552
3553                                    ;+
3554                                    ;LOCAL DATA AREA
3555                                    ;-
3556
3557 021134  000000                     PATDAT: .WORD   0                       ;TEMPORARY STORAGE FOR DATA
3558 021136    105    116    124       DATASC: .ASCIZ  'ENTER DATA PATTERN'
3559                                            .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
                                ;
                                ;-
021162                          GETSEL::
                                        SAVREG                  ;SAVE GENERAL REGISTERS
021162  010002                  1$:   MOV     R0,R2             ;SAVE THE MENU ADDRESS
021166  010203                  2$:   MOV     R2,R3             ;START OF MENU STRING
021170  005713                        TST     (R3)              ;END OF ASCII ?
021172  001412                        BEQ     3$                ;BRANCH IF ALL LINES DISPLAYED
021174                                PRINTF  #SELASC,(R3)+     ;DISPLAY THE MENU
021176  012346                        MOV     (R3)+,-(SP)
021200  012746  021346                MOV     #SELASC,-(SP)
021204  012746  000002                MOV     #2,-(SP)
021210  010600                        MOV     SP,R0
021212  104417                        TRAP    C$PNTF
021214  062706  000006                ADD     #6,SP
021220  000764                        BR      2$
021222                          3$:   GMANID  MENASC,MENRES,D,-1,0,-1,NO
021222  104443                        TRAP    C$GMAN
021224  000406                        BR      10001$
021226  021402                        .WORD   MENRES
021230  000042                        .WORD   T$CODE
021232  021353                        .WORD   MENASC
021234  177777                        .WORD   -1
021236  000000                        .WORD   T$LOLIM
021240  177777                        .WORD   T$HILIM
021242                          10001$:
021242                                BNCOMPLETE      1$        ;RETRY IF ERROR
021242  103352                        BCC     1$
021244  013700  021402                MOV     MENRES,R0         ;GET THE OPERATOR'S REPLY
021250  020001                        CMP     R0,R1             ;COMPARE TO MAXIMUM ALLOWED
021252  101411                        BLOS    5$                ;BRANCH IF OK
021254                                PRINTF  #MENERR           ;DISPLAY ERROR MESSAGE
021254  012746  021300                MOV     #MENERR,-(SP)
021260  012746  000001                MOV     #1,-(SP)
021264  010600                        MOV     SP,R0
021266  104417                        TRAP    C$PNTF
021270  062706  000004                ADD     #4,SP
021274  000735                        BR      1$                ;RETRY
021276  000207                  5$:   RTS     PC                ;RETURN TO CALLER
021300     045     116     045  MENERR: .ASCIZ  '%N%A *** Menu Selection Too Large ***'
021346     045     116     045  SELASC: .ASCIZ  '%N%T'
021353     105     156     164  MENASC: .ASCIZ  'Enter Menu Selection: '
                                        .EVEN
021402  000000                  MENRES:       .WORD   0
```

```
                                        .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.
                                ;
                                ;-

021404                          CHKMAN::
021404                                  SAVREG                  ;SAVE THE REGISTERS
021410                                  MANUAL                  ;SEE IF MANUAL INTERVENTION OK
021410  104450                          TRAP    C$MANI
021412                                  BCOMPLETE 1$            ;BRANCH IF ALLOWED
021412  103411                          BCS     1$
021414                                  PRINTF  #NOMAN          ;PRINT THE WARNING MESSAGE
021414  012746  021440                  MOV     #NOMAN,-(SP)
021420  012746  000001                  MOV     #1,-(SP)
021424  010600                          MOV     SP,R0
021426  104417                          TRAP    C$PNTF
021430  062706  000004                  ADD     #4,SP
021434  000241                          CLC                     ;CLEAR CARRY FOR ERROR
021436  000207                  1$:     RTS     PC              ;RETURN

021440     045     116     045  NOMAN:  .ASCIZ  '%N%A *** Manual Intervention not Allowed - Test Aborted ***'
                                        .even
```

```
3631                                            .SBTTL  ENVIRN  - SETUP FREE DIAGNOSTIC SPACE
3632                                    ;
3633                                    ; SUBROUTINE TO SET-UP VARIOUS ENVIRONMENTAL PARAMETERS.
3634                                    ;
3635 021534                            ENVIRN: MEMORY  R0
     021534  104431                             TRAP    C$MEM
3636 021536  010037  003072                    MOV     R0,FREE         ; GET 1ST FREE ADDRESS...
3637 021542  062737  000002  003072            ADD     #2,FREE
3638 021550  011037  003074                    MOV     (R0),FRESIZ     ;...AND WORD COUNT.
3639 021554  162737  000004  003074            SUB     #4,FRESIZ
3640 021562  013702  002012                    MOV     L$UNIT,R2       ; GET NUMBER OF UNITS
3641 021566  162737  000007  003074  10$:      SUB     #7,FRESIZ       ; TAKE AWAY 7 WORDS PER UNIT
3642 021574  005302                            DEC     R2
3643 021576  001373                            BNE     10$
3644 021600  013700  003072                    MOV     FREE,R0         ;GET FIRST FREE ADDRESS
3645 021604  063700  003074                    ADD     FRESIZ,R0       ;POINT TO LAST FREE ADDRESS
3646 021610  162700  000002                    SUB     #2,R0           ;BACKUP 1 WORD
3647 021614  010037  003076                    MOV     R0,FREEHI       ;STORE LAST FREE ADDRESS
3648 021620  000207                            RTS     PC              ;RETURN
3649
```

```
3651                                    .SBTTL  KTINIT  - SETUP KT11 MEMORY MANAGEMENT REGISTERS
3652                                ;*
3653                                ;
3654                                ;ROUTINE TO INIT KT-11
3655                                ;
3656                                ;-
3657
3658 021622                        KTINIT:
3659 021622  005037  003100                    CLR     KTFLG           ; INIT >28K MEMORY FLAG
3660 021626  005037  003102                    CLR     KTENABLE        ; INIT TEST >28K FLAG
3661 021632  023727  002120  001577            CMP     L$HIME,#1577    ; GOT ENOUGH MEMORY (>28K)?
3662 021640  101444                           BLOS    9$              ; NO.
3663 021642  013700  000004                    MOV     @#ERRVEC,R0     ; SAVE OLD ERR VEC PTR.
3664 021646  012737  021740  000004            MOV     #2$,@#ERRVEC    ; SET ERR VEC PTR.
3665 021654  005737  177572                    TST     @#SR0           ; GOT KT11?
3666 021660  000240                           NOP                     ;  (TRAP IF NO).
3667 021662  013737  002120  003100            MOV     L$HIME,KTFLG    ; YES. SET KT FLAG.
3668 021670  042737  000177  003100            BIC     #177,KTFLG      ;
3669 021676  010037  000004                    MOV     R0,@#ERRVEC     ; RESTORE OLD ERR VEC PTR.
3670 021702  005000                           CLR     R0              ; R0 = AR DATA.
3671 021704  012701  172340                    MOV     #KIPAR0,R1      ; R1 = KI REGS PTR.
3672 021710  012761  077406  177740  1$:      MOV     #77406,-40(R1)  ; SET DESCRIPTOR REG.
3673 021716  010021                           MOV     R0,(R1)+        ; SET KIPAR REG.
3674 021720  062700  000200                    ADD     #200,R0         ; BUMP AR DATA BY "4K".
3675 021724  020027  002000                    CMP     R0,#2000        ; AT "I/O"?
3676 021730  001367                           BNE     1$              ; NO.
3677 021732  012741  177600                    MOV     #177600,-(R1)   ; YES. SET KTPAR7 FOR I/O.
3678 021736  000405                           BR      9$
3679
3680 021740  012716  021746          2$:      MOV     #6$,(SP)        ; SET UP RETURN
3681 021744  000002                           RTI                     ; RTI TO NEXT LOCATION
3682
3683 021746  010037  000004          6$:      MOV     R0,@#ERRVEC     ; RESTORE OLD ERR VEC PTR.
3684
3685 021752  000207                  9$:      RTS     PC
3694
3695
3701
```

```
3703                                            .SBTTL  PROTECTION TABLE
3704 021754                                     BGNPROT
     021754                             L$PROT::
3705 021754  177777  177777  177777             .WORD   -1, -1, -1, -1  ;NO DEVICE PROTECTION REQUIRED.
3706 021764                                     ENDPROT
3707
```

```
                                        .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.
                                ;
                                ;--
                                ;+
                                ;INSERT TEMPORARY JUMP TO ODT
                                ;-
                                        BGNINIT
                                L$INIT::
                                40$:
021764                                  MOV     #EPRT1,EPRTSW   ;SET UP PRIMARY MESSAGE FOR REPLACEMENT
021772  005037  003106                   CLR     SIFLAG          ;CLEAR "SOFT INIT" FLAG
021776  005037  003102                   CLR     KTENABLE        ;CLEAR TEST ABOVE 28K FLAG
022002  005037  002246                   CLR     RAMSIZ          ;CLEAR RAM SIZE FOR RAMERR ROUTINE
022006                                  READEF  #EF.CONTINUE
022006  012700  000036                   MOV     #EF.CONTINUE,R0
022012  104447                          TRAP    C$REFG
022014                                  BNCOMPLETE 1$
022014  103023                          BCC     1$
022016  023737  002150  002012           CMP     UNITN,L$UNIT    ;UNIT IN RANGE?
022024  103064                          BHIS    4$              ;BR IF NO.
022026  005737  003060                   TST     DUFLG           ;DROPPED UNIT?
022032  100466                          BMI     NXTU            ;BR IF YES
022034  013701  002150                   MOV     UNITN,R1
022040  006301                          ASL     R1
022042  005761  003130                   TST     ERTABL(R1)
022046  001512                          BEQ     SETU
022050  032761  010000  003130           BIT     #BIT14,ERTABL(R1) ;DROPPED?
022056  001054                          BNE     NXTU
022060                                  EXIT    INIT            ;DO NOTHING IF "CONTINUE".
022060  104432                          TRAP    C$EXIT
022062  000412                          .WORD   L10030-.
022064                          1$:     READEF  #EF.NEW
022064  012700  000035                   MOV     #EF.NEW,R0
022070  104447                          TRAP    C$REFG
022072                                  BNCOMPLETE NXTU         ;TAKE NEXT UNIT IF NOT NEW PASS.
022072  103046                          BCC     NXTU
022074                                  READEF  #EF.START
022074  012700  000040                   MOV     #EF.START,R0
022100  104447                          TRAP    C$REFG
022102                                  BCOMPLETE 2$
022102  103404                          BCS     2$
022104                                  READEF  #EF.RESTART
022104  012700  000037                   MOV     #EF.RESTART,R0
022110  104447                          TRAP    C$REFG
022112                                  BNCOMPLETE 31$
022112  103025                          BCC     31$
022114                          2$:                             ;1ST PASS, BUS-INIT...
022114                                  BRESET                  ;BUS RESET.
022114  104433                          TRAP    C$RESET
022116  005037  002162                   CLR     TSTCNT          ;NUMBER OF TESTS RUN IN PASS
```

```
3750 022122  005037  002170                    CLR     FATFLG          ;RESET FLAG TO ZERO "FATAL ERRORS"
3751 022126  005037  003332          19$:      CLR     SKIPT           ;CLEAR THE SUBTEST "SKIPPER"
3752 022132                          20$:
3753 022132  012737  177777  002152            MOV     #-1,QVP         ;...QUICK VERIFY...
3754 022140  004737  021534                    JSR     PC,ENVIRN       ;SET ENVIRONMENT.
3755 022144  004737  021622                    JSR     PC,KTINIT       ;INITIALIZE KT MEMORY MANAGEMENT
3756 022150  012700  003130                    MOV     #ERTABL,R0
3757 022154  005020                  30$:      CLR     (R0)+           ;CLEAR THE ERROR TABLE
3758 022156  020027  003330                    CMP     R0,#ERTABE
3759 022162  103774                          BLO     30$
3760 022164  000404                          BR      4$
3761 022166  005037  002152          31$:      CLR     QVP
3762 022172  000137  022242                    JMP     PASRPT          ;GO REPORT THE STATUS
3763
3764 022176                          4$:
3765 022176  012737  177777  002150  NEWPAS:   MOV     #-1,UNITN       ;INIT UNIT NUMBER...
3766 022204  005037  002166                    CLR     DEVCNT          ;CLEAR COUNT OF DEVICES RUNNING
3767 022210                          NXTU:     BREAK
     022210  104422                          TRAP    C$BRK
3768 022212  005237  002150                    INC     UNITN           ;...AND SET NEXT UNIT NUMBER.
3769 022216  023737  002150  002012          CMP     UNITN,L$UNIT
3770 022224  103423                          BLO     SETU
3771 022226  012737  177777  003060          MOV     #-1,DUFLG
3772 022234  000401                          BR      11$
3773 022236                                   DOCLN                   ;ABORT, NO MORE UNITS.
     022236  104444                          TRAP    C$DCLN
3774 022240  000240                  11$:      NOP
3775 022242                          PASRPT:
3776 022242  023727  002012  000001          CMP     L$UNIT,#1       ;HOW MANY UNITS SELECTED?
3777 022250  101752                          BLOS    NEWPAS          ;BR IF ONLY 1
3778 022252  005737  002166                    TST     DEVCNT          ;ARE ANY STILL RUNNING?
3779 022256  001747                          BEQ     NEWPAS          ;BR IF NO
3780 022260                                   RFLAGS  R0
     022260  104421                          TRAP    C$RFLA
3781 022262  032700  000100                    BIT     #ISR,R0         ;SHOULD WE PRINT STATISTICS
3782 022266  001343                          BNE     NEWPAS          ;BR IF NO
3783
3784 022270                                   DORPT
     022270  104424                          TRAP    C$DRPT
3785 022272  000741                          BR      NEWPAS
3786 022274                          10$:
3787
3788 022274                          SETU:     GPHARD  UNITN,R0        ;GET UNIT N P-TABLE POINTER.
     022274  013700  002150                    MOV     UNITN,R0
     022300  104442                          TRAP    C$GPHRD
3789 022302                                   BNCOMPLETE NXTU         ;BR IF UNIT NOT AVAILABLE.
     022302  103342                          BCC     NXTU
3790 022304  005037  003060                    CLR     DUFLG           ;CLEAR "DROPPED" FLAG.
3791 022310  005237  002166                    INC     DEVCNT
3792 022314  012001                          MOV     (R0)+,R1        ;GET 1ST REGISTER ADDRESS.
3793 022316  010137  002154                    MOV     R1,CSRADDR      ;ADDRESS OF REGISTERS OF UNIT UNDER TEST
3794
3795 022322  012001                          MOV     (R0)+,R1        ;GET VECTOR ADDRESS.
3796 022324  011002                          MOV     (R0),R2         ;GET INTERRUPT PRIORITY
3797 022326  010237  002160                    MOV     R2,IPRI         ;SET INTERRUPT PRIORITY.
3798 022332  010137  002156                    MOV     R1,IVEC         ;SET INTERRUPT VECTOR POINTER...
3799 022336  012721  017066                    MOV     #INTR,(R1)+     ;...VECTOR...
```

INITIALIZE SECTION

```
3800 022342  010221                        MOV     R2,(R1)+        ;...AND PRIORITY.
3801
3802 022344                          1$:
3803                                 ;       TST     QVP             ;1ST PASS ??
3804                                 ;       BEQ     5$              ;NO, SKIP THE PASS 1 STUFF.
3805
3806                                 ;
3807                                 ;1ST PASS, CHECK THAT DEVICE ADDRESSES ARE VALID, AND
3808                                 ;THAT THE DISPLAY STATUS IS PROPERLY INITIALIZED.
3809                                 ;
3810 022344  013701  002150                 MOV     UNITN,R1
3811 022350  006301                         ASL     R1
3812 022352  052761  100000  003130         BIS     #BIT15,ERTABL(R1) ;SAY DEVICE RUNNING
3813 022360  005037  005232                 CLR     EXTA            ;CLEAR ERROR EXTENSION FLAG.
3814 022364  023727  002012  000001         CMP     L$UNIT,#1       ;ARE WE TESTING MULTIPLE UNITS?
3815 022372  101416                         BLOS    10$             ;BR IF NO.
3816 022374                                 RFLAGS  R0              ;YES -- GET OPERATOR FLAGS.
     022374  104421                         TRAP    C$RFLA
3817 022376  032700  001000                 BIT     #PNT,R0         ;SHOULD WE PRINT UNIT #?
3818 022402  001412                         BEQ     10$             ;BR IF NOT.
3819 022404                                 PRINTF  #PUNIT,UNITN    ;PRINT THE UNIT #
     022404  013746  002150                 MOV     UNITN,-(SP)
     022410  012746  022476                 MOV     #PUNIT,-(SP)
     022414  012746  000002                 MOV     #2,-(SP)
     022420  010600                         MOV     SP,R0
     022422  104417                         TRAP    C$PNTF
     022424  062706  000006                 ADD     #6,SP
3820 022430                          10$:
3821 022430  005037  003062                 CLR     NODEV
3822 022434  013701  002154                 MOV     CSRADDR,R1      ;ADDRESS OF FIRST REGISTER
3823 022440  010102                         MOV     R1,R2           ;START OF REGISTERS
3824 022442  062702  000000                 ADD     #TSSR,R2        ;ADDRESS OF TSSR REGISTER
3825 022446  004737  017276                 JSR     PC,XNXM         ;TEST BOTH CONTROLLER REGISTERS...
3826 022452  103005                         BCC     2$              ;...AND BR IF ALL OK.
3827 022454  010137  003062                 MOV     R1,NODEV        ;FLAG DEVICE AS NON-EXISTENT
3828 022460  012737  177777  003060         MOV     #-1,DUFLG       ;DROP THIS UNIT.
3829 022466                          2$:
3830                                 ;
3831                                 ;FINALLY, SET CPU PRIORITY AND WE'RE DONE.
3832                                 ;
3833 022466                          5$:    SETPRI  #PRI00          ;ENABLE INTERRUPTS.
     022466  012700  000000                 MOV     #PRI00,R0
     022472  104441                         TRAP    C$SPRI
3834 022474                                 ENDINIT
     022474                          L10030:
     022474  104411                         TRAP    C$INIT
3835
3836 022476     045     116     045  PUNIT: .ASCIZ  /%N%N%A***** TESTING UNIT %D2%A *****/
3837                                         .EVEN
```

```
                                        .SBTTL  ADD AND DROP UNITS SECTIONS

                                ;++
                                ; THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                ; TO BE (A) ADDED TO THE TEST LIST FOR THE FIRST TIME,
                                ;    OR (B) RE-INSERTED IF IT HAD BEEN PREVIOUSLY DROPPED.
                                ;--
022544                                  BGNAU
022544                          L$AU::
022544  010001                          MOV     R0,R1           ; GET UNIT TO BE ADDED (R0)
022546  006301                          ASL     R1              ; MAKE IT A WORD INDEX
022550  052761  100000  003130          BIS     #100000,ERTABL(R1) ; SET THE "ACTIVE" BIT
022556  042761  040000  003130          BIC     #40000,ERTABL(R1)  ; CLEAR THE "DROPPED" BIT
022564                                  PRINTF  #1$,R0
022564  010046                          MOV     R0,-(SP)
022566  012746  022612                  MOV     #1$,-(SP)
022572  012746  000002                  MOV     #2,-(SP)
022576  010600                          MOV     SP,R0
022600  104417                          TRAP    C$PNTF
022602  062706  000006                  ADD     #6,SP
022606                                  EXIT    AU
022606  000167                          .WORD   J$JMP
022610  000026                          .WORD   L10031-2-.
022612     045     116     045  1$:     .ASCIZ  /%N%A  UNIT %D%A ADDED/
                                        .EVEN

022640                                  ENDAU                   ; UNUSED.
022640                          L10031:
022640  104452                          TRAP    C$AU

                                ;++
                                ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                ; TO BE REMOVED FROM THE TEST LIST.
                                ;
                                ; SUPVSR DOES THE "DROPPING".  THIS IS JUST TO TELL THE MAN.
                                ; "DROPPED" UNITS ARE RE-SELECTED ON OPERATOR "STA" OR "ADD"
                                ; COMMAND, OTHERWISE REMAIN INACTIVE.  THE "DISPLAY" COMMAND
                                ; WILL PRINT ALL DROPPED UNITS, AND THE P-TABLES OF THOSE
                                ; WHICH ARE STILL ACTIVE.
                                ; UPON ENTRY, R0 CONTAINS THE UNIT TO BE DROPPED.

022642                                  BGNDU
022642                          L$DU::
022642  012737  177777  003060          MOV     #-1,DUFLG
022650  010001                          MOV     R0,R1
022652  006301                          ASL     R1
022654  052761  140000  003130          BIS     #140000,ERTABL(R1)      ; SAY DROPPED
022662  000240  000240  000240          240,240,240                     ; ?????????
022670                                  PRINTF  #1$,R0
022670  010046                          MOV     R0,-(SP)
022672  012746  022716                  MOV     #1$,-(SP)
022676  012746  000002                  MOV     #2,-(SP)
022702  010600                          MOV     SP,R0
022704  104417                          TRAP    C$PNTF
022706  062706  000006                  ADD     #6,SP
022712                                  EXIT    DU
022712  000167                          .WORD   J$JMP
022714  000030                          .WORD   L10032-2-.
```

```
3876 022716     045     116     045  1$:     .ASCIZ  /%N%A  UNIT %D%A DROPPED/
3877                                         .EVEN
3878 022746                                  ENDDU
     022746                          L10032:
     022746  104453                         TRAP    C$DU
3879                                 ;++
3880                                 ; AUTO-DROP CODE SECTION.
3881                                 ;--
3882 022750                                  BGNAUTO
     022750                          L$AUTO::
3883 022750  012703  000550                 MOV     #360.,R3        ;ENOUGH TIME FOR 2400' REEL TO REWIND
3884 022754  004737  017120          10$:    JSR     PC,WAITF        ;WAIT FOR SSR TO SET
3885 022760  103420                         BCS     20$             ;LEAVE WHEN SSR IS SET
3886 022762                                  DELAY   250.            ;WAIT FOR .25 SECONDS
     022762  012727  000372                 MOV     #250.,(PC)+
     022766  000000                         .WORD   0
     022770  013727  002116                 MOV     L$DLY,(PC)+
     022774  000000                         .WORD   0
     022776  005367  177772                 DEC     -6(PC)
     023002  001375                         BNE     .-4
     023004  005367  177756                 DEC     -22(PC)
     023010  001367                         BNE     .-20
3887 023012  005303                         DEC     R3              ;BUMP COUNTER DOWN
3888 023014  001357                         BNE     10$             ;KEEP GOING
3889 023016  004737  020154                 JSR     PC,CKDROP       ;TRY AND DROP UNIT
3890 023022                          20$:
3891 023022                                  ENDAUTO                 ; UNUSED.
     023022                          L10033:
     023022  104461                         TRAP    C$AUTO
```

```
                                        .SBTTL  CLEAN-UP AND REPORT CODING SECTIONS

                                ;++
                                ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS
                                ; EXECUTED AT THE END OF EACH PASS (OR SUB-PASS).
                                ; USE TO RETURN DEVICE UNDER TEST TO A NEUTRAL STATE.
                                ;--
023024                                  BGNCLN
023024                          L$CLEAN::
023024  005737  003060                  TST     DUFLG           ;"DROPPED" FLAG IS SET ON...
023030  100407                          BMI     1$              ;...AND GROSS CONTROLLER FAULT...
                                                                ;...DON'T TRY TO XCT CLEANUP CODE.
023032  013705  002154                  MOV     CSRADDR,R5      ; ADDRESS OF TSV REGISTERS ON UNIBUS
023036  012765  000000  000000          MOV     #0,TSSR(R5)     ;DO SOFT INIT
023044  004737  017120                  JSR     PC,WAITF
023050                          1$:
023050                          2$:     ENDCLN
023050                          L10034:
023050  104412                          TRAP    C$CLEAN

                                ;++
                                ; THE REPORT CODING SECTION CONTAINS THE
                                ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
                                ;--
023052                                  BGNRPT
023052                          L$RPT::
023052                                  PRINTS  #DEVSUM
023052  012746  023314                  MOV     #DEVSUM,-(SP)
023056  012746  000001                  MOV     #1,-(SP)
023062  010600                          MOV     SP,R0
023064  104416                          TRAP    C$PNTS
023066  062706  000004                  ADD     #4,SP
023072  010246                          MOV     R2,-(SP)
023074  010346                          MOV     R3,-(SP)
023076  010446                          MOV     R4,-(SP)
023100  012704  003130                  MOV     #ERTABL,R4      ; GET START OF ERROR TABLE.
023104  005003                          CLR     R3              ; CLEAR UNIT NUMBER
023106  011402                  1$:     MOV     (R4),R2         ; GET ERROR TABLE ENTRY & TEST IT.
023110  001467                          BEQ     4$              ; ZERO IF UNIT NOT RUN
023112  100066                          BPL     4$
023114  032702  040000                  BIT     #BIT14,R2       ; WAS UNIT DROPPED?
023120  001015                          BNE     2$              ; BR IF YES
023122  042702  170000                  BIC     #^C7777,R2      ; GET ERROR COUNT FIELD
023126                                  PRINTS  #DEVONL,R3,R2   ; PRINT
023126  010246                          MOV     R2,-(SP)
023130  010346                          MOV     R3,-(SP)
023132  012746  023351                  MOV     #DEVONL,-(SP)
023136  012746  000003                  MOV     #3,-(SP)
023142  010600                          MOV     SP,R0
023144  104416                          TRAP    C$PNTS
023146  062706  000010                  ADD     #10,SP
023152  000446                          BR      4$
023154  020227  160000          2$:     CMP     R2,#160000      ; WAS UNIT NON-EXISTENT?
023160  001012                          BNE     3$              ; BR IF NO
023162                                  PRINTS  #DEVNXR,R3
023162  010346                          MOV     R3,-(SP)
023164  012746  023421                  MOV     #DEVNXR,-(SP)
```

```
                023170  012746  000002                  MOV     #2,-(SP)
                023174  010600                          MOV     SP,R0
                023176  104416                          TRAP    C$PNTS
                023200  062706  000006                  ADD     #6,SP
3932            023204  000431                          BR      4$
3933            023206  020227  160001          3$:     CMP     R2,#160001      ; WAS UNIT NOT READY AT STARTUP?
3934            023212  001012                          BNE     30$             ; BR IF NO.
3935            023214                                  PRINTS  #DEVNRD,R3
                023214  010346                          MOV     R3,-(SP)
                023216  012746  023503                  MOV     #DEVNRD,-(SP)
                023222  012746  000002                  MOV     #2,-(SP)
                023226  010600                          MOV     SP,R0
                023230  104416                          TRAP    C$PNTS
                023232  062706  000006                  ADD     #6,SP
3936            023236  000414                          BR      4$
3937            023240  042702  170000          30$:    BIC     #^C7777,R2
3938            023244                                  PRINTS  #DEVDRO,R3,R2
                023244  010246                          MOV     R2,-(SP)
                023246  010346                          MOV     R3,-(SP)
                023250  012746  023564                  MOV     #DEVDRO,-(SP)
                023254  012746  000003                  MOV     #3,-(SP)
                023260  010600                          MOV     SP,R0
                023262  104416                          TRAP    C$PNTS
                023264  062706  000010                  ADD     #10,SP
3939            023270  062704  000002          4$:     ADD     #2,R4
3940            023274  005203                          INC     R3
3941            023276  020427  003330                  CMP     R4,#ERTABE
3942            023302  103701                          BLO     1$
3943            023304  012604                          MOV     (SP)+,R4
3944            023306  012603                          MOV     (SP)+,R3
3945            023310  012602                          MOV     (SP)+,R2
3946            023312                                  ENDRPT                  ; UNUSED.
                023312                          L10035:
                023312  104425                          TRAP    C$RPT
3947
3948
3949            023314      045     116     045 DEVSUM: .ASCIZ  /%N%ADEVICE STATUS SUMMARY:%N/
3950            023351      045     101     040 DEVONL: .ASCIZ  /%A  UNIT %D3%A   ONLINE,  ERRORS = %D%N/
3951            023421      045     101     040 DEVNXR: .ASCIZ  /%A  UNIT %D3%A   DROPPED, NON-EXISTENT REGISTER%N/
3952            023503      045     101     040 DEVNRD: .ASCIZ  /%A  UNIT %D3%A   DROPPED, NOT READY AT STARTUP%N/
3953            023564      045     101     040 DEVDRO: .ASCIZ  /%A  UNIT %D3%A   DROPPED, ERRORS = %D%N/
3954                                                .EVEN
3957
3958
3959
3966
3972
3980
```

```
                                        .SBTTL  TEST  1: INITIALIZE #2 TEST
                                ;+
                                ;
                                ;THIS TEST VERIFIES THAT WRITING INTO THE TSSR RETURNS THE
                                ;CONTROLLER TO ITS INITIALIZED STATE FROM VARIOUS CONDITIONS
                                ;
                                ;-
023634                                  BGNTST
023634                                                                          T1::
023634  005037  002170                  CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
023640  005037  003100                  CLR     KTFLG           ;HOLD OFF KT11
023644  012737  005762  002146          MOV     #EPRT1,EPRTSW           ;SET UP PRIMARY ERROR MESSAGE

                                ;
                                ;
                                ;TEST 1
                                ;
                                ;
                                ;-

023652  004737  017054                  JSR     PC,DSBINT               ;DISABLE INTERRUPTS
023656  012700  024516                  MOV     #T$T21ID,R0             ;ASCII MESSAGE TO IDENTIFY TEST
023662  004737  017410                  JSR     PC,TSTSETUP             ;DO INITIAL TEST SETUP
023666  012737  000002  002164          MOV     #2,LOOPCNT              ;PERFORM 2 ITERATIONS
023674                          T21LOOP:
023674  004737  024540                  JSR     PC,T21REST              ;SET COMMAND PACKET
023700  004737  024630                  JSR     PC,T21RT2               ;SET UP OTHER COMMAND PACKET

                                ;*************************************************************
                                ;
                                ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                ;
                                ;*************************************************************

023704  012737  176750  024276          MOV     #65000.,T21DLY          ;SET DELAY ROUTINE
023712  004737  016644          11$:    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
023716  103426                  BCS     20$                     ;BR IF INIT WAS OK
023720                          DELAY   250                     ;DELAY FOR A REWIND TO FINISH
023720  012727  000250                                                          MOV     #250.,(PC)+
023724  000000                                                                  .WORD   0
023726  013727  002116                                                          MOV     L$DLY,(PC)+
023732  000000                                                                  .WORD   0
023734  005367  177772                                                          DEC     -6(PC)
023740  001375                                                                  BNE     .-4
023742  005367  177756                                                          DEC     -22(PC)
023746  001367                                                                  BNE     .-20
023750  005337  024276                  DEC     T21DLY                  ;BUMP COUNTER DOWN
023754  001356                  BNE     11$                     ;BR, IF MORE TIME TO GO
023756  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
023762  010001                  MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
023764                          ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
023764  104455                                                                  TRAP    C$ERDF
023766  000145                                                                  .WORD   101
023770  003550                                                                  .WORD   SFIERR
023772  011666                                                                  .WORD   SFIMSG
023774                          20$:
023774  012704  024150                  MOV     #T21PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
```

```
4034
4035                                    ;*****************************************************************
4036                                    ;
4037                                    ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4038                                    ;
4039                                    ;*****************************************************************
4040
4041 024000  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
4042 024004  103407                            BCS     23$                     ;BR, IF COMMAND ISSUED OK
4043 024006  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4047 024012  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
4048 024014                                    ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     024014  104456                                                                    TRAP    C$ERHRD
     024016  000146                                                                    .WORD   102
     024020  004754                                                                    .WORD   WRTMSG
     024022  011666                                                                    .WORD   SFIMSG
4049 024024                            23$:
4050 024024  012765  000000  000000                MOV     #0,TSSR(R5)             ;ISSUE A SOFT INITIALIZE
4051 024032  004737  017120                    JSR     PC,WAITF                ;WAIT FOR JUST THE SSR BIT TO SET
4052 024036  016501  000000                    MOV     TSSR(R5),R1             ;READ THE TSSR BACK
4053 024042  005002                            CLR     R2                      ;SET UP EXPECTED CONTENTS IN R2
4054 024044  032701  000100                    BIT     #OFL,R1                 ;CHECK FOR OFF LINE SET (NOT ERROR)
4055 024050  001402                            BEQ     24$                     ;BR, IF OFL IS NOT SET
4056 024052  052702  000100                    BIS     #OFL,R2                 ;IT WAS SET SO SET IN EXPECTED
4057 024056  052702  002200            24$:    BIS     #SSR!NBA,R2             ;R2 HAS EXPECTED CONTENTS
4058 024062  032701  000100                    BIT     #OFL,R1                 ;IS OFF LINE BIT SET
4059 024066  001012                            BNE     38$                     ;BR, IF DRIVE IS OFF LINE
4060 024070  020102                    35$:    CMP     R1,R2                   ;EXPECTED (R2) = RECEIVED (R1)
4061 024072  001406                            BEQ     37$                     ;BR, IF THEY ARE EQUAL (OK)
4062 024074  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4066 024100                                    ERRHRD  ERRNO,T21AM3,EXPREC     ;"ERROR TRYING TO INIT AFTER WRITE MISC.
     024100  104456                                                                    TRAP    C$ERHRD
     024102  000147                                                                    .WORD   103
     024104  024375                                                                    .WORD   T21AM3
     024106  016344                                                                    .WORD   EXPREC
4067 024110                            37$:    CKLOOP                          ;LOOP IF SELECTED
     024110  104406                                                                    TRAP    C$CLP1
4068 024112  000406                            BR      40$                     ;SKIP OVER OFF-LINE STUFF
4069 024114                            38$:
4073 024114                                    ERRDF   ERRNO,T21OFL,EXPREC     ;DRIVE IS OFF LINE
     024114  104455                                                                    TRAP    C$ERDF
     024116  000150                                                                    .WORD   104
     024120  024475                                                                    .WORD   T21OFL
     024122  016344                                                                    .WORD   EXPREC
4074 024124  004737  020154                    JSR     PC,CKDROP               ;TRY AND DROP UNIT
4075 024130  004737  017356            40$:    JSR     PC,TSTLOOP              ;DO WE NEED TO ITERATE TEST
4076 024134  103002                            BCC     63$                     ;BR, IF NO LOOP REQUIRED
4077 024136  000137  023674                    JMP     T21LOOP                 ;EXECUTE AGAIN
4078 024142                            63$:    EXIT    TST                     ;ALL DONE THIS TEST
     024142  104432                                                                    TRAP    C$EXIT
     024144  000526                                                                    .WORD   L10036-.
```

```
                                ;*
                                ;LOCAL STORAGE FOR THIS TEST
                                ;-
                                        .BLKB   10-<.-TUV2A&7>
024146                          T21PACKET:
024150                                  .WORD   100004          ;COMMAND PACKET FOR TEST
024150  100004                          .WORD   T21DATA         ;WRITE CHARACTERISTICS COMMAND, WITH, ACK
024152  024160                          .WORD   0               ;ADDRESS OF CHARACTERISTICS BLOCK
024154  000000                          .WORD   10.             ;STARTING VALUE OF BLOCK SIZE
024156  000012                  T21DATA:                        ;CHARACTERISTICS DATA BLOCK
024160                                  .WORD   T21BFR          ;ADDRESS OF MESSAGE BUFFER
024160  024170                          .WORD   0
024162  000000                          .WORD   20.             ;LENGTH OF MESSAGE BUFFER
024164  000024                          .WORD   0
024166  000000                  T21BFR: .BLKW   25.             ;MESSAGE BUFFER
024170                          ;
                                ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
                                ;
                                        .BLKB   10-<.-TUV2A&7>
024252                          T21PK2:
024260                                  .WORD   100206          ;WRITE SUB SYS MEM COMMAND, IE AND ACK
024260  100206                          .WORD   T21BF2          ;ADDRESS OF SELECT BLOCK DATA
024262  024270                          .WORD   0
024264  000000                          .WORD   6.              ;SIZE OF DATA PACKET
024266  000006
                                        .EVEN
024270                          T21BF2:
024270      000                 T21BS0: .BYTE   0               ;BSEL0 AREA --- "COMMAND" BYTE
024271      000                 T21BS1: .BYTE   0               ;BSEL1 AREA
024272  000000                  T2152:  .WORD   0               ;SEL 2 AREA
024274  000000                  T2153:  .WORD   0               ;DATA AREA
024276  000000                  T21DLY: .WORD   0               ;DELAY COUNTER
```

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

024300     127     122     111  T21SSR: .ASCIZ  'WRITE MISCELLANEOUS CONTROL/READ STATUS Command Not Accepted'
024375     124     123     123  T21AM3: .ASCIZ  'TSSR Init. Failed After WRITE MISCELLANEOUS CONRTOL/READ STATUS'
024475     104     162     151  T210FL: .ASCIZ  'Drive is OFFLINE'
024516     111     156     151  TST21ID:        .ASCIZ  'Initialization #2'
                                        .EVEN
                                        ;+
                                        ;
                                        ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
                                        ;WRITE SUBSYSTEM MEMORY COMMAND
                                        ;
                                        ;-

024540                                  T21REST:
024540                                          SAVREG                          ;SAVE THE REGISTERS
024544  012701  024150                          MOV     #T21PACKET,R1           ;START OF THE PACKET
024550  012721  100004                          MOV     #100004,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK.
024554  012721  024160                          MOV     #T21DATA,(R1)+          ;ADDRESS OF CHARAISTICS DATA BLOCK
024560  005021                                  CLR     (R1)+                   ;EXTENDED ADDRESS
024562  012721  000010                          MOV     #8.,(R1)+               ;SIZE OF DATA BLOCK IN BYTES
024566  012721  024170                          MOV     #T21BFR,(R1)+           ;ADDRESS OF MESSAGE BUFFER
024572  005021                                  CLR     (R1)+
024574  012721  000024                          MOV     #20.,(R1)+              ;LENGTH OF MESSAGE BUFFER
024600  005021                                  CLR     (R1)+
024602  005011                                  CLR     (R1)
024604  012702  000020                          MOV     #20,R2                  ;NUMBER OF LOCATIONS TO BE CLEARED
024610  012762  177777  024170  64$:           MOV     #177777,T21BFR(R2)      ;ALL ONES TO MESSAGE BUFFER
024616  005742                                  TST     -(R2)                   ;NEXT LOCATION
024620  020227  000000                          CMP     R2,#0                   ;CHECK R2 FOR ZERO
024624  001371                                  BNE     64$                     ;BR. IF NOT AT ZERO YET
024626  000207                                  RTS     PC                      ;RETURN

024630                                  T21RT2:
024630                                          SAVREG                          ;SAVE THE REGISTERS
024634  012701  024260                          MOV     #T21PK2,R1              ;START OF THE PACKET
024640  012721  100206                          MOV     #100206,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK, IE
024644  012721  024270                          MOV     #T21BF2,(R1)+           ;ADDRESS OF DATA BLOCK
024650  005021                                  CLR     (R1)+                   ;EXTENDED ADDRESS
024652  012721  000006                          MOV     #6.,(R1)+               ;SIZE OF DATA BLOCK IN BYTES
024656  005021                                  CLR     (R1)+
024660  012701  024270                          MOV     #T21BF2,R1              ;ADDRESS OF DATA FOR WRT SUB SYS MEM
024664  005021                                  CLR     (R1)+
024666  005011                                  CLR     (R1)
024670  000207                                  RTS     PC                      ;RETURN
024672                                          ENDTST
024672                                                                  L10036:
024672  104401                                                                  TRAP    C$ETST
```

```
                                        .SBTTL  TEST  2: OFF-LINE AND REJECT REWIND
                                ;*
                                ;
                                ;THIS TEST VERIFIES BASIC TAPE-MOTION COMMAND DECODING AND BASIC
                                ;OPERATION OF THE REWIND POSITIONING COMMAND. IT DOES NOT
                                ;NECESSARILY DEMONSTRATE THAT THE TRANSPORT CAN BE REWOUND FROM AN
                                ;ARBITRARY POSITION ON THE TAPE. SUBSEQUENT TESTS IMPLICITLY
                                ;CHECK THE OPERATION OF THE REWIND COMMAND SINCE THEY MUST
                                ;TYPICALLY REWIND THE TAPE IN THE NORMAL COURSE OF THEIR TEST
                                ;SEQUENCES. THE TEST CONSISTS OF THE FOLLOWING ONE SUBTEST
                                ;
                                ;
                                ;-
024674                          BGNTST
024674                                                                  T2::
024674  005037  002170          CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
024700  005037  003100          CLR     KTFLG           ;HOLD OFF KT11
024704  012737  005762  002146  MOV     #EPRT1,EPRTSW           ;SET UP PRIMARY ERROR MESSAGE
024712  004737  017054          JSR     PC,DSBINT               ;DISABLE INTERRUPTS
024716  012700  026032          MOV     #TST22ID,R0             ;ASCII MESSAGE TO IDENTIFY TEST
024722  004737  017410          JSR     PC,TSTSETUP             ;DO INITIAL TEST SETUP
024726  012737  000002  002164  MOV     #2,LOOPCNT              ;PERFORM 2 ITERATIONS
                                ;*
                                ;
```

```
 4199 024734                            T22LOOP:
 4200                                   ;+
 4201                                   ;
 4202                                   ;TEST 2, SUBTEST 1
 4203                                   ;
 4204                                   ;VERIFIES THAT A REWIND COMMAND WITH CVC=1 CLEARS VCK
 4205                                   ;AND RETURNS PROPER STATUS IN THE MESSAGE BUFFER.
 4206                                   ;
 4207                                   ;
 4208                                   ;-
 4209 024734                                    BGNSUB                          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
      024734                                                                                    T2.1:
      024734  104402                                                                                TRAP    C$BSUB
 4210 024736  004737  026066                    JSR     PC,T22REST              ;SET COMMAND PACKET
 4211 024742  004737  026160                    JSR     PC,T22RT2               ;SET UP OTHER COMMAND PACKET
 4212
 4213                                   ;*************************************************************************
 4214                                   ;
 4215                                   ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
 4216                                   ;
 4217                                   ;*************************************************************************
 4218
 4219 024746  004737  016644                    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
 4220 024752  103407                            BCS     20$                     ;BR IF INIT WAS OK
 4221 024754  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4225 024760  010001                            MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
 4226 024762                                    ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      024762  104455                                                                           TRAP    C$ERDF
      024764  000311                                                                          .WORD   201
      024766  003550                                                                          .WORD   SFIERR
      024770  011666                                                                          .WORD   SFIMSG
 4227 024772                            20$:
 4228 024772  012704  025200                    MOV     #T22PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
 4229
 4230                                   ;*************************************************************************
 4231                                   ;
 4232                                   ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
 4233                                   ;
 4234                                   ;*************************************************************************
 4235
 4236 024776  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
 4237 025002  103407                            BCS     65$                     ;BR. IF COMMAND ISSUED OK
 4238 025004  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4242 025010  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
 4243 025012                                    ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      025012  104456                                                                           TRAP    C$ERHRD
      025014  000312                                                                          .WORD   202
      025016  004754                                                                          .WORD   WRTMSG
      025020  011666                                                                          .WORD   SFIMSG
 4244 025022  012737  142010  025310    65$:    MOV     #142010,T22PK2          ;POSITION COMMAND (REWIND MODE) CVC=1
 4245 025030  012704  025310                    MOV     #T22PK2,R4              ;R4 = POINTER TO PACKET
 4246 025034  010465  177776                    MOV     R4,TSDB(R5)             ;ISSUE COMMAND
 4247 025040  004737  017120                    JSR     PC,WAITF                ;WAIT FOR SSR TO SET
 4248 025044  016501  000000                    MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
 4249 025050  012702  000200                    MOV     #SSR,R2                 ;SET UP EXPECTED
 4250 025054  020102                            CMP     R1,R2                   ;ARE THEY EQUAL
 4251 025056  001406                            BEQ     80$                     ;BR. IF OK ESP. FUNCTION REJECT
```

```
4252 025060  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4256 025064                                ERRHRD  ERRNO,T22RWJ,EXPREC     ;TSSR INCORRECT AFTER TAPE MOTION CMD
     025064  104456                                                            TRAP    C$ERHRD
     025066  000313                                                            .WORD   203
     025070  025704                                                            .WORD   T22RWJ
     025072  016344                                                            .WORD   EXPREC
4257 025074                          80$:   CKLOOP                          ;LOOP IF SELECTED
     025074  104406                                                            TRAP    C$CLP1
4258 025076  012703  025222                MOV     #T22BFR,R3              ;POINTER TO MESSAGE BUFFER
4259 025102  016301  000006                MOV     XSTO(R3),R1             ;PICK UP XSTO FROM MESSAGE BUFFER
4260 025106  010102                        MOV     R1,R2                   ;SET UP EXPECTED
4261 025110  042702  000020                BIC     #BIT4,R2                ;VCK SHOULD BE CLEAR
4262 025114  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
4263 025116  001406                        BEQ     90$                     ;BR, IF OK (GOOD)
4264 025120  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4268 025124                                ERRHRD  ERRNO,T22VCK,EXPREC     ;VCK WASN'T CLEAR (BAD)
     025124  104456                                                            TRAP    C$ERHRD
     025126  000314                                                            .WORD   204
     025130  025757                                                            .WORD   T22VCK
     025132  016344                                                            .WORD   EXPREC
4269 025134                          90$:
4270 025134                                ENDSUB                          ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
     025134                                                                     L10040:
     025134  104403                                                            TRAP    C$ESUB
4271 025136  023727  002170  000031              CMP     FATFLG,#25.             ;IS ERROR COUNT AT 25
4272 025144  002402                              BLT     999$                    ;BR, IF LESS THAN 25
4273 025146  004737  020154                      JSR     PC,CKDROP               ;TRY TO DROP THE UNIT
4274 025152                          999$:
4275 025152  004737  017356                JSR     PC,TSTLOOP              ;DO WE NEED TO ITERATE TEST
4276 025156  103002                        BCC     163$                    ;BR, IF NO LOOP REQUIRED
4277 025160  000137  024734                JMP     T22LOOP                 ;EXECUTE AGAIN
4278 025164                          163$:  EXIT    TST                     ;ALL DONE THIS TEST
     025164  104432                                                            TRAP    C$EXIT
     025166  001036                                                            .WORD   L10037-.
```

```
4280                                    ;+
4281                                    ;LOCAL STORAGE FOR THIS TEST
4282                                    ;-
4284 025170                                     .BLKB   10-<.-TUV2A&7>
4286 025200                             T22PACKET:                      ;COMMAND PACKET FOR TEST
4287 025200  100204                             .WORD   100204          ;WRITE CHARACTERISTICS COMMAND, WITH IE, ACK
4288 025202  025210                             .WORD   T22DATA         ;ADDRESS OF CHARACTERISTICS BLOCK
4289 025204  000000                             .WORD   0
4290 025206  000012                             .WORD   10.             ;STARTING VALUE OF BLOCK SIZE
4291 025210                             T22DATA:                        ;CHARACTERISTICS DATA BLOCK
4292 025210  025222                             .WORD   T22BFR          ;ADDRESS OF MESSAGE BUFFER
4293 025212  000000                             .WORD   0
4294 025214  000024                             .WORD   20.             ;LENGTH OF MESSAGE BUFFER
4295 025216  000000                             .WORD   0
4296 025220  000007                             .WORD   7               ;SELECT DRIVE 7
4297 025222                             T22BFR: .BLKW   25.             ;MESSAGE BUFFER
4298                                    ;
4299                                    ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
4300                                    ;
4302 025304                                     .BLKB   10-<.-TUV2A&7>
4304 025310                             T22PK2:
4305 025310  100206                             .WORD   100206          ;WRITE SUB SYS MEM COMMAND, IE AND ACK
4306 025312  025320                             .WORD   T22BF2          ;ADDRESS OF SELECT BLOCK DATA
4307 025314  000000                             .WORD   0
4308 025316  000006                             .WORD   6.              ;SIZE OF DATA PACKET
4309
4310                                            .EVEN
4311 025320                             T22BF2:
4312 025320     000                     T22BS0: .BYTE   0               ;BSEL0 AREA
4313 025321     000                     T22BS1: .BYTE   0               ;BSEL1 AREA
4314 025322  000000                     T22S2:  .WORD   0               ;SEL 2 AREA
4315 025324  000000                     T22S3:  .WORD   0               ;DATA AREA
4316                                    ;
4317                                    ;
4318                                            .EVEN
4319                                    ;TAPE MOTION PACKET COMMAND VALUES
4320 025326  100201                     T22RD:  .WORD   100201          ;READ TAPE FORWARD
4321 025330  100205                     T22WRT: .WORD   100205          ;WRITE TAPE FORWARD
4322 025332  100210                     T22POS: .WORD   100210          ;POSITION TAPE
4323 025334  100211                     T22FOR: .WORD   100211          ;FORMAT TAPE
4324 025336  177777                             .WORD   177777          ;END OF DATA
4325
4326
```

```
4328
4329                                    ;+
4330                                    ;LOCAL TEXT MESSAGES FOR TEST
4331                                    ;-
4332
4333 025340     127     122     111    T22SSR: .ASCIZ  'WRITE MISCELLANEOUS CONTROL/READ STATUS Command Not Accepted'
4334 025435     124     123     123    T22AM3: .ASCIZ  'TSSR Init. Failed After WRITE MISCELLANEOUS CONRTOL/READ STATUS'
4335 025535     104     162     151    T22OFL: .ASCIZ  'Drive 7 Select Failed To Set "OFL" In TSSR'
4336 025610     124     123     123    T22TM:  .ASCIZ  'TSSR Incorrect After Tape Motion Command To Off-Line Device'
4337 025704     124     123     123    T22RWJ: .ASCIZ  'TSSR Not Correct After REWIND With VCK Set'
4338 025757     103     126     103    T22VCK: .ASCIZ  'CVC Set, Didn''t Reset VCK In Message Buffer'
4339 026032     117     146     146    TST22ID:        .ASCIZ  'Off-Line And Reject Rewind'
4340                                            .EVEN
4341                                    ;+
4342                                    ;
4343                                    ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
4344                                    ;WRITE SUBSYSTEM MEMORY COMMAND
4345                                    ;
4346                                    ;-
4347
4348 026066                             T22REST:
4349 026066                                     SAVREG                          ;SAVE THE REGISTERS
4350 026072  012701  025200                     MOV     #T22PACKET,R1           ;START OF THE PACKET
4351 026076  012721  100204                     MOV     #100204,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK, IE
4352 026102  012721  025210                     MOV     #T22DATA,(R1)+          ;ADDRESS OF CHARAISTICS DATA BLOCK
4353 026106  005021                             CLR     (R1)+                   ;EXTENDED ADDRESS
4354 026110  012721  000012                     MOV     #10.,(R1)+              ;SIZE OF DATA BLOCK IN BYTES
4355 026114  012721  025222                     MOV     #T22BFR,(R1)+           ;ADDRESS OF MESSAGE BUFFER
4356 026120  005021                             CLR     (R1)+
4357 026122  012721  000024                     MOV     #20.,(R1)+              ;LENGTH OF MESSAGE BUFFER
4358 026126  005021                             CLR     (R1)+
4359 026130  012711  000007                     MOV     #7,(R1)                 ;SELECT DRIVE SEVEN
4360 026134  012702  000020                     MOV     #20,R2                  ;NUMBER OF LOCATIONS TO BE CLEARED
4361 026140  012762  177777  025222  64$:      MOV     #177777,T22BFR(R2)      ;ALL ONES TO MESSAGE BUFFER
4362 026146  005742                             TST     -(R2)                   ;BUMP R2 DOWN
4363 026150  020227  000000                     CMP     R2,#0                   ;IS R2 AT ZERO YET
4364 026154  001371                             BNE     64$                     ;KEEP GOING UNTIL DONE
4365 026156  000207                             RTS     PC                      ;RETURN
4366
4367
4368 026160                             T22RT2:
4369 026160                                     SAVREG                          ;SAVE THE REGISTERS
4370 026164  012701  025310                     MOV     #T22PK2,R1              ;START OF THE PACKET
4371 026170  012721  100206                     MOV     #100206,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK, IE
4372 026174  012721  025320                     MOV     #T22BF2,(R1)+           ;ADDRESS OF DATA BLOCK
4373 026200  005021                             CLR     (R1)+                   ;EXTENDED ADDRESS
4374 026202  012721  000006                     MOV     #6.,(R1)+               ;SIZE OF DATA BLOCK IN BYTES
4375 026206  005021                             CLR     (R1)+
4376 026210  012701  025320                     MOV     #T22BF2,R1              ;POINT TO DATA SEL AREA
4377 026214  005021                             CLR     (R1)+
4378 026216  005011                             CLR     (R1)
4379 026220  005011                             CLR     (R1)                    ;LAST LOC TO BE CLEARED
4380 026222  000207                             RTS     PC                      ;RETURN
4381 026224                                     ENDTST
     026224                                                             L10037:
     026224  104401                                                             TRAP    C$ETST
```

```
                                        .SBTTL  TEST  3: BASIC WRITE
                                ;*
                                ;
                                ;THIS TEST VERIFIES THAT THE WRITE DATA (NEXT) COMMAND OPERATES
                                ;PROPERLY, UP TO THE POINT OF CHECKING THAT THE DATA WAS ACTUALLY
                                ;WRITTEN ONTO THE TAPE CORRECTLY. CHECKING IN THIS TEST IS
                                ;LIMITED TO VERIFYING THAT THE COMMAND TERMINATED CORRECTLY WITH
                                ;THE CORRECT REGISTER, MESSAGE BUFFER AND RAM CONTENTS.
                                ;
                                ;THE TEST CONSISTS OF THE FOLLOWING 5 SUBTESTS
                                ;
                                ;
                                ;
                                ;-
026226                                  BGNTST
026226                                                                  T3::
026226  005037  002170                    CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
026232  005037  003100                    CLR     KTFLG           ;HOLD OFF KT11
026236  012737  005762  002146            MOV     #EPRT1,EPRTSW           ;SET UP PRIMARY ERROR MESSAGE
026244  005037  003102                    CLR     KTENABLE                ;TURN OFF KT11
026250  004737  020246                    JSR     PC,KTOFF                ;TURN OFF KT11
026254  004737  017054                    JSR     PC,DSBINT               ;DISABLE INTERRUPTS
026260  012700  031761                    MOV     #TST23ID,R0             ;ASCII MESSAGE TO IDENTIFY TEST
026264  004737  017410                    JSR     PC,TSTSETUP             ;DO INITIAL TEST SETUP
026270  012737  000001  002164            MOV     #1,LOOPCNT              ;PERFORM 1 ITERATIONS
                                ;*
                                ;
```

```
4414 026276                          T23LOOP:
4415                                 ;+
4416                                 ;
4417                                 ;TEST 3, SUBTEST 1
4418                                 ;
4419                                 ;VERIFIES THAT WRITE DATA COMMANDS WITH CVC=1 AND THE
4420                                 ;SWAP BYTES (SWB) BIT CLEAR OPERATES PROPERLY. THE
4421                                 ;BYTE COUNT (RECORD SIZE) VARIES FROM 20 THROUGH 64K
4422                                 ;IN VARYING INCREMENTS (DEPENDING UPON WHETHER OR NOT
4423                                 ;THE DIAGNOSTIC IS RUNNING IN THE LONG VERIFICATION
4424                                 ;MODE). THE TAPE IS NOT REWOUND BETWEEN SUCCESSIVE
4425                                 ;RECORDS BUT IS REWOUND AFTER THE FINAL RECORD IS
4426                                 ;WRITTEN. AN INCREMENTING COUNT PATTERN IS SUPPLIED
4427                                 ;IN THE DATA BUFFER. AFTER EACH BLOCK IS WRITTEN, THE
4428                                 ;TSSR AND TSBA REGISTERS, AND THE MESSAGE BUFFER.
4429                                 ;
4430                                 ;
4431                                 ;
4432                                 ;-
4433 026276                                 BGNSUB                  ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
      026276                                                                     T3.1:
      026276  104402                                                                TRAP    C$BSUB
4434 026300  013701  003074                 MOV     FRESIZ,R1       ;GET SIZE OF AVAILIABLE MEMORY
4435 026304  000241                         CLC                     ;CLEAR THE CARRY BIT
4436 026306  006101                         ROL     R1              ;MAKE INTO BYTES
4437 026310  010137  030520                 MOV     R1,T23RSZ       ;STORE IN TEST FOR USE LATER
4438 026314  004737  031776                 JSR     PC,T23REST      ;SET COMMAND PACKET
4439 026320  004737  032132                 JSR     PC,T23RT3       ;RESTORE PACKET
4440 026324  004737  032070                 JSR     PC,T23RT2       ;SET UP OTHER COMMAND PACKET
4441
4442                                 ;*****************************************************************
4443                                 ;
4444                                 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
4445                                 ;
4446                                 ;*****************************************************************
4447
4448 026330  004737  016644                 JSR     PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
4449 026334  103407                         BCS     20$             ;BR IF INIT WAS OK
4450 026336  004737  020102                 JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
4454 026342  010001                         MOV     R0,R1           ;CONTENTS OF TSSR REGISTER
4455 026344                                 ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      026344  104455                                                                TRAP    C$ERDF
      026346  000455                                                                .WORD   301
      026350  003550                                                                .WORD   SFIERR
      026352  011666                                                                .WORD   SFIMSG
4456 026354                          20$:
4457 026354  012704  030360                 MOV     #T23PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
4458
4459                                 ;*****************************************************************
4460                                 ;
4461                                 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4462                                 ;
4463                                 ;*****************************************************************
4464
4465 026360  004737  010332                 JSR     PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
4466 026364  103407                         BCS     23$             ;BR, IF COMMAND ISSUED OK
4467 026366  004737  020102                 JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
```

```
4471 026372  010001                        MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
4472 026374                                ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     026374  104456                                                        TRAP    C$ERHRD
     026376  000456                                                        .WORD   302
     026400  004754                                                        .WORD   WRTMSG
     026402  011666                                                        .WORD   SFIMSG
4473 026404                        23$:    CKLOOP                          ;LOOP IF SELECTED
     026404  104406                                                        TRAP    C$CLP1
4474
4475                               ;*****************************************************************
4476                               ;
4477                               ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4478                               ;
4479                               ;*****************************************************************
4480
4481 026406  004737  010434                MOV     PC,REWIND               ;CALL THE TAPE REWIND
4482 026412  012703  000024                MOV     #20.,R3                 ;STARTING RECORD SIZE
4483 026416  013737  003072  030512  65$:   MOV     FREE,T23WB              ;STARTING WRITE BUFFER ADDRESS
4484
4485                               ;*****************************************************************
4486                               ;
4487                               ;WRITE DATA,CVC=1,ACK COMMAND
4488                               ;
4489                               ;*****************************************************************
4490
4491 026424  012737  140005  030510          MOV     #140005,T23PK3          ;WRITE DATA,CVC=1,ACK COMMAND
4492 026432  012704  030510                MOV     #T23PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
4493 026436  010300                        MOV     R3,R0                   ;SET PATTERN IN CORRECT REGISTER
4494 026440  004737  020374                JSR     PC,FILLMEM              ;FILL MEMORY WITH RECORD SIZE
4495 026444  010337  030516                MOV     R3,T23SZ                ;SET UP RECORD SIZE IN PACKET
4496 026450  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
4497 026454  004737  017120                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
4498 026460  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
4499 026464  012702  000200                MOV     #SSR,R2                 ;SET UP EXPECTED
4500 026470  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
4501 026472  001406                        BEQ     80$                     ;BR. IF OK
4502 026474  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4507 026500                                ERRHRD  ERRNO,WRTERR,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     026500  104456                                                        TRAP    C$ERHRD
     026502  000457                                                        .WORD   303
     026504  005011                                                        .WORD   WRTERR
     026506  011700                                                        .WORD   PKTSSR
4508 026510                        80$:    CKLOOP                          ;LOOP IF SELECTED
     026510  104406                                                        TRAP    C$CLP1
4509 026512  016501  177776                MOV     TSBA(R5),R1             ;GET TSBA CONTENTS
4510 026516  012702  030510                MOV     #T23PK3,R2              ;SET UP EXPECTED
4511 026522  020102                85$:    CMP     R1,R2                   ;ARE THEY EQUAL
4512 026524  001406                        BEQ     90$                     ;BR. IF TSBA IS CORRECT
4513 026526  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4517 026532                                ERRHRD  ERRNO,T23BA,EXPREC      ;TSBA WAS NOT CORRECT AFTER WRITE DATA
     026532  104456                                                        TRAP    C$ERHRD
     026534  000460                                                        .WORD   304
     026536  031620                                                        .WORD   T23BA
     026540  016344                                                        .WORD   EXPREC
4518 026542                        90$:    CKLOOP                          ;LOOP IF SELECTED
     026542  104406                                                        TRAP    C$CLP1
4519 026544  062703  001750        115$:   ADD     #1000.,R3               ;NEXT RECORD SIZE/DATA PATTERN
```

```
4520 026550  020337  030520                CMP     R3,T23RSZ               ;IS R3 OVER MAX RECORD SIZE
4521 026554  002005                        BGE     120$                    ;IF RECORD SIZE IS TOO BIG QUIT
4522 026556  020327  177776                CMP     R3,#65534.              ;END OF SUBTEST MAX RECORD SIZE
4523 026562  001402                        BEQ     120$                    ;BR, IF COMPLETED
4524 026564  000137  026416                JMP     65$                     ;DO MORE RECORDS
4525 026570                          120$:
4526 026570  004737  032132                JSR     PC,T23RT3               ;RESTORE PACKET
4527 026574  004737  032070                JSR     PC,T23RT2               ;CLEAN UP PACKET
4528 026600  004737  010434                JSR     PC,REWIND               ;ISSUE REWIND COMMAND WITH WAIT
4529 026604  103407                        BCS     130$                    ;BR, IF TSSR IS OK (GOOD)
4530 026606  010001                        MOV     R0,R1                   ;SAVE TSSR CONTENTS
4531 026610  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4535 026614                                ERRHRD  ERRNO,T23RWN,PKTSSR     ;TSSR IS INCORRECT AFTER REWIND
     026614  104456                                                                TRAP    C$ERHRD
     026616  000461                                                                .WORD   305
     026620  031111                                                                .WORD   T23RWN
     026622  011700                                                                .WORD   PKTSSR
4536 026624                          130$:
4537 026624                                ENDSUB                          ;>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>>
     026624                                                                         L10042:
     026624  104403                                                                TRAP    C$ESUB
4538 026626  023727  002170  000031        CMP     FATFLG,#25.                     ;IS ERROR COUNT AT 25
4539 026634  002402                        BLT     999$                            ;BR, IF LESS THAN 25
4540 026636  004737  020154                JSR     PC,CKDROP                       ;TRY TO DROP THE UNIT
4541 026642                          999$:
```

```
4543                                    ;+
4544                                    ;
4545                                    ;TEST 3, SUBTEST 2
4546                                    ;
4547                                    ;VERIFIES THAT WRITE DATA COMMANDS WITH CVC=1 AND THE
4548                                    ;SWAP BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST
4549                                    ;SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2.
4550                                    ;THE RESULTS, EXCEPT FOR RAM CONTENTS, SHOULD BE THE SAME.
4551                                    ;
4552                                    ;
4553                                    ;
4554                                    ;-
4555 026642                                    BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
     026642                                                                                   T3.2:
     026642  104402                                                                              TRAP    C$BSUB
4556 026644  004737  031776                    JSR     PC,T23REST              ;SET COMMAND PACKET
4557 026650  004737  032070                    JSR     PC,T23RT2               ;SET UP OTHER COMMAND PACKET
4558
4559                                    ;*****************************************************************
4560                                    ;
4561                                    ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
4562                                    ;
4563                                    ;*****************************************************  ************
4564
4565 026654  004737  016644                    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
4566 026660  103407                            BCS     20$                     ;BR IF INIT WAS OK
4567 026662  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4571 026666  010001                            MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
4572 026670                                    ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
     026670  104455                                                                           TRAP    C$ERDF
     026672  000462                                                                           .WORD   306
     026674  003550                                                                           .WORD   SFIERR
     026676  011666                                                                           .WORD   SFIMSG
4573 026700                                20$:
4574 026700  012704  030360                    MOV     #T23PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
4575
4576                                    ;*****************************************************************
4577                                    ;
4578                                    ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4579                                    ;
4580                                    ;*****************************************************************
4581
4582 026704  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
4583 026710  103407                            BCS     23$                     ;BR, IF COMMAND ISSUED OK
4584 026712  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4588 026716  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
4589 026720                                    ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     026720  104456                                                                           TRAP    C$ERHRD
     026722  000463                                                                           .WORD   307
     026724  004754                                                                           .WORD   WRTMSG
     026726  011666                                                                           .WORD   SFIMSG
4590 026730                                23$:
4591 026730  012703  000024                    MOV     #20.,R3                 ;STARTING RECORD SIZE
4592 026734  013737  003072  030512    65$:    MOV     FREE,T23WB              ;STARTING WRITE BUFFER ADDRESS
4593
4594                                    ;*****************************************************************
4595                                    ;
```

```
4596                                    ;WRITE DATA,CVC=1,ACK,SWB COMMAND
4597                                    ;
4598                                    ;*****************************************************************
4599
4600 026742  012737  150005  030510          MOV     #150005,T23PK3          ;WRITE DATA,CVC=1,ACK,SWB COMMAND
4601 026750  012704  030510                  MOV     #T23PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
4602 026754  010300                          MOV     R3,R0                   ;SET PATTERN IN CORRECT REGISTER
4603 026756  004737  020374                  JSR     PC,FILLMEM              ;FILL MEMORY WITH RECORD SIZE
4604 026762  010337  030516                  MOV     R3,T23SZ                ;SET UP RECORD SIZE IN PACKET
4605 026766  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
4606 026772  004737  017120                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
4607 026776  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
4608 027002  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
4609 027006  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
4610 027010  001406                          BEQ     80$                     ;BR, IF OK
4611 027012  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4615 027016                                  ERRHRD  ERRNO,WRTERR,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     027016  104456                                                                  TRAP    C$ERHRD
     027020  000464                                                                  .WORD   308
     027022  005011                                                                  .WORD   WRTERR
     027024  011700                                                                   WORD   PKTSSR
4616 027026                          80$:    CKLOOP                          ;LOOP IF SELECTED
     027026  104406                                                                  TRAP    C$CLP1
4617 027030  016501  177776                  MOV     TSBA(R5),R1             ;GET TSBA CONTENTS
4618 027034  012702  030510                  MOV     #T23PK3,R2              ;SET UP EXPECTED
4619 027040  020102                  85$:    CMP     R1,R2                   ;ARE THEY EQUAL
4620 027042  001406                          BEQ     90$                     ;BR, IF TSBA IS CORRECT
4621 027044  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4625 027050                                  ERRHRD  ERRNO,T23BA,EXPREC      ;TSBA WAS NOT CORRECT AFTER WRITE DATA
     027050  104456                                                                  TRAP    C$ERHRD
     027052  000465                                                                  .WORD   309
     027054  031620                                                                  .WORD   T23BA
     027056  016344                                                                  .WORD   EXPREC
4626 027060                          90$:    CKLOOP                          ;LOOP IF SELECTED
     027060  104406                                                                  TRAP    C$CLP1
4627 027062  020327  007376                  CMP     R3,#7376                ;ONLY CHECK RAM UNTIL ITS FULL
4628 027066  002057                          BGE     115$                    ;IT WRAPS AROUND ETC.
4629 027070  004737  032070                  JSR     PC,T23RT2               ;MAKE SURE PACKET AND DATA ARE CLEAN
4630 027074  012737  000400  030524          MOV     #256.,T23S2             ;STARTING RAM ADDRESS
4631 027102  112737  000000  030522          MOVB    #0,T23BS0               ;STOP INTERNAL TUV05 DIAGNOSTICS
4632 027110  112737  000000  030523          MOVB    #0,T23BS1               ;SIZE OF RAM READ
4633 027116  012704  030470                  MOV     #T23PK2,R4              ;SET R4 WITH PACKET ADDRESS
4634 027122  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE WRITE SUB SYS MEM COMMAND
4635 027126  004737  017236                  JSR     PC,CHKTSSR              ;CHECK TSSR AND WAIT FOR SSR TO SET
4636 027132  103407                          BCS     92$                     ;BR, IF NO ERRORS IN TSSR
4637 027134  010001                          MOV     R0,R1                   ;SAVE TSSR
4638 027136  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4642 027142                                  ERRHRD  ERRNO,T23WSS,PKTSSR     ;TSSR BAD AFTER WRITE SUB SYS MEM
     027142  104456                                                                  TRAP    C$ERHRD
     027144  000466                                                                  .WORD   310
     027146  031672                                                                  .WORD   T23WSS
     027150  011700                                                                  .WORD   PKTSSR
4643 027152                          92$:    CKLOOP                          ;LOOP IF SELECTED
     027152  104406                                                                  TRAP    C$CLP1
4644 027154  004737  032070                  JSR     PC,T23RT2               ;MAKE SURE PACKET AND DATA ARE CLEAN
4645 027160  012737  000400  030524          MOV     #256.,T23S2             ;STARTING RAM ADDRESS
4646 027166  112737  000001  030522          MOVB    #1,T23BS0               ;READ RAM COMMAND FOR WRITE SUB SYS M.
```

```
4647 027174  112737  000002  030523          MOVB    #2,T23BS1               ;SIZE OF RAM READ
4648 027202  012704  030470                  MOV     #T23PK2,R4              ;SET R4 WITH PACKET ADDRESS
4649 027206  010465  177776          95$:    MOV     R4,TSDB(R5)             ;ISSUE WRITE SUB SYS MEM CMD (READ RAM)
4650 027212  004737  017236                  JSR     PC,CHKTSSR              ;CHECK TSSR AND WAIT FOR SSR TO SET
4651 027216  103403                          BCS     115$                    ;BR, IF NO ERRORS IN TSSR
4652 027220  010001                          MOV     R0,R1                   ;SAVE TSSR
4653 027222  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4657 027226  062703  001750          115$:   ADD     #1000.,R3               ;NEXT RECORD SIZE/DATA PATTERN
4658 027232  020337  030520                  CMP     R3,T23RSZ               ;IS R3 OVER MAX RECORD SIZE
4659 027236  002005                          BGE     120$                    ;IF RECORD SIZE IS TOO BIG QUIT
4660 027240  020327  177776                  CMP     R3,#65534.              ;END OF SUBTEST MAX RECORD SIZE
4661 027244  001402                          BEQ     120$                    ;BR, IF COMPLETED
4662 027246  000137  026734                  JMP     65$                     ;DO MORE RECORDS
4663 027252                          120$:
4664 027252  004737  032070                  JSR     PC,T23RT2               ;CLEAN UP PACKET
4665 027256  004737  010434                  JSR     PC,REWIND               ;ISSUE REWIND COMMAND WITH WAIT
4666 027262  103407                          BCS     130$                    ;BR, IF TSSR IS OK (GOOD)
4667 027264  010001                          MOV     R0,R1                   ;SAVE TSSR CONTENTS
4668 027266  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4672 027272                                  ERRHRD  ERRNO,T23RWN,PKTSSR     ;TSSR IS INCORRECT AFTER REWIND
     027272  104456                                                                      TRAP    C$ERHRD
     027274  000470                                                                      .WORD   312
     027276  031111                                                                      .WORD   T23RWN
     027300  011700                                                                      .WORD   PKTSSR
4673 027302                          130$:
4674 027302                                  ENDSUB                          ;>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
     027302                                                                              L10043:
     027302  104405                                                                      TRAP    C$ESUB
4675 027304  023727  002170  000031          CMP     FATFLG,#25.                     ;IS ERROR COUNT AT 25
4676 027312  002402                          BLT     999$                            ;BR, IF LESS THAN 25
4677 027314  004737  020154                  JSR     PC,CKDROP                       ;TRY TO DROP THE UNIT
4678 027320                          999$:
```

```
                                ;+
                                ;
                                ;TEST 3, SUBTEST 3
                                ;
                                ;VERIFIES THAT A WRITE COMMAND WITH AN ILLEGAL MODE
                                ;FIELD OR AN ILLEGAL BUFFER ADDRESS IS REJECTED WITH
                                ;THE PROPER ERROR STATUS AND THAT TAPE DOES NOT MOVE
                                ;
                                ;
                                ;-

027320                                  BGNSUB                          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
027320                                                                                          T3.3:
027320  104402                                                                                  TRAP    C$BSUB
027322  004737  031776                    JSR     PC,T23REST              ;SET COMMAND PACKET
027326  004737  032132                    JSR     PC,T23RT3               ;RESTORE PACKET
027332  004737  032070                    JSR     PC,T23RT2               ;SET UP OTHER COMMAND PACKET

                                ;*************************************************************************
                                ;
                                ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                ;
                                ;*************************************************************************

027336  004737  016644                    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
027342  103407                            BCS     20$                     ;BR IF INIT WAS OK
027344  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
027350  010001                            MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
027352                                  ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
027352  104455                                                                                  TRAP    C$ERDF
027354  000471                                                                                  .WORD   313
027356  003550                                                                                  .WORD   SFIERR
027360  011666                                                                                  .WORD   SFIMSG
027362                          20$:
027362  012704  030360                    MOV     #T23PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS

                                ;*************************************************************************
                                ;
                                ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
                                ;
                                ;*************************************************************************

027366  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
027372  103407                            BCS     23$                     ;BR, IF COMMAND ISSUED OK
027374  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
027400  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
027402                                  ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
027402  104456                                                                                  TRAP    C$ERHRD
027404  000472                                                                                  .WORD   314
027406  004754                                                                                  .WORD   WRTMSG
027410  011666                                                                                  .WORD   SFIMSG

                                ;*************************************************************************
                                ;
                                ;WRITE DATA, ACK, ILLEGAL BITS
                                ;
                                ;*************************************************************************
```

```
 4733
 4734 027412  012737  104405  030510  23$:    MOV     #104405,T23PK3          ;WRITE DATA, ACK, ILLEGAL BITS
 4735 027420  013737  003072  030512          MOV     FREE,T23WB              ;SET UP WRITE BUFFER ADDRESS
 4736 027426  062737  000001  030512          ADD     #1,T23WB                ;MAKE ADDRESS ODD (ILLEGAL)
 4737 027434  012737  000400  030516          MOV     #256.,T23SZ             ;SET UP BUFFER SIZE
 4738 027442  012704  030510                  MOV     #T23PK3,R4              ;R4 = POINTER TO PACKET
 4739 027446  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
 4740 027452  004737  017120                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
 4741 027456  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
 4742 027462  012702  100206                  MOV     #SSR!SC!BIT1!BIT2,R2    ;SET UP EXPECTED
 4743 027466  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
 4744 027470  001406                          BEQ     80$                     ;BR, IF OK ESP, FUNCTION REJECT
 4745 027472  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 4749 027476                                  ERRHRD  ERRNO,T23TM,PKTSSR      ;TSSR INCORRECT AFTER WRITE COMMAND
      027476  104456                                                                  TRAP    C$ERHRD
      027500  000473                                                                  .WORD   315
      027502  030736                                                                  .WORD   T23TM
      027504  011700                                                                  .WORD   PKTSSR
 4750 027506                          80$:    CKLOOP                          ;LOOP IF SELECTED
      027506  104406                                                                  TRAP    C$CLP1
 4751 027510                                  ENDSUB                          ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
      027510                                                                       L10044:
      027510  104403                                                                  TRAP    C$ESUB
 4752 027512  023727  002170  000031          CMP     FATFLG,#25.                     ;IS ERROR COUNT AT 25
 4753 027520  002402                          BLT     999$                            ;BR, IF LESS THAN 25
 4754 027522  004737  020154                  JSR     PC,CKDROP                       ;TRY TO DROP THE UNIT
 4755 027526                          999$:
```

```
                                        ;*
                                        ;
                                        ;TEST 3, SUBTEST 4
                                        ;
                                        ;VERIFIES THAT A WRITE DATA COMMAND SPECIFYING A DATA
                                        ;BUFFER STARTING IN NONEXISTANT MEMORY TERMINATES WITH
                                        ;THE PROPER ERROR STATUS WITHOUT MOVING TAPE
                                        ;
                                        ;
                                        ;
                                        ;-

                                                BGNSUB                          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
                                                                                                        T3.4:
027526                                                                                                          TRAP    C$BSUB
027526  104402
027530  005737  003100                        TST     KTFLG                   ;CHECK FOR KT-11
                                        ;       BEQ     10$                     ;BR, IF NO KT-11
                                        ;
                                        ;       DIALOG PATCH FOR NXM
                                        ;
                                                NOP                             ;REMOVE WHEN NXM IS FIXED
027534  000240
027536  000137  027776                        JMP     90$                     ;SKIP THIS TEST IF KT-11
027542  004737  031776                10$:    JSR     PC,T23REST              ;SET COMMAND PACKET
027546  004737  032132                        JSR     PC,T23RT3               ;RESTORE PACKET
027552  004737  032070                        JSR     PC,T23RT2               ;SET UP OTHER COMMAND PACKET

                                        ;*******************************************************************
                                        ;
                                        ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                        ;
                                        ;*******************************************************************

027556  004737  016644                        JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
027562  103407                                BCS     20$                     ;BR IF INIT WAS OK
027564  004737  020102                        JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
027570  010001                                MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
027572                                        ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
027572  104455                                                                                          TRAP    C$ERDF
027574  000474                                                                                          .WORD   316
027576  003550                                                                                          .WORD   SFIERR
027600  011666                                                                                          .WORD   SFIMSG
027602                                  20$:
027602  012704  030360                        MOV     #T23PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS

                                        ;*******************************************************************
                                        ;
                                        ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
                                        ;
                                        ;*******************************************************************

027606  004737  010332                        JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
027612  103407                                BCS     23$                     ;BR, IF COMMAND ISSUED OK
027614  004737  020102                        JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
027620  010001                                MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
027622                                        ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
027622  104456                                                                                          TRAP    C$ERHRD
027624  000475                                                                                          .WORD   317
```

```
        027626  004754                                                          .WORD   WRTMSG
        027630  011666                                                          .WORD   SFIMSG
   4812
   4813                                 ;*****************************************************************
   4814                                 ;
   4815                                 ;WRITE DATA, ACK, CVC=1
   4816                                 ;
   4817                                 ;*****************************************************************
   4818
   4819 027632  005037  040726          23$:    CLR     T24DLY                  ;SET EXTENDED ADDRESS BITS TO 0
   4820 027636  012737  140005  030510  25$:    MOV     #140005,T23PK3          ;WRITE DATA, ACK, CVC=1
   4821 027644  012701  160000                  MOV     #160000,R1              ;START POSSIBLE NXM ADDRESS
   4822 027650  012702  177776                  MOV     #177776,R2              ;END POSSIBLE NXM ADDRESS
   4823 027654  004737  017276                  JSR     PC,XNXM                 ;CALL NXM FINDER ROUTINE
   4824 027660  103402                          BCS     76$                     ;BR IF NXM ADDRESS FOUND
   4825 027662  000137  027776                  JMP     90$                     ;JMP OVER CAN'T FIND NXM
   4826 027666  010137  030512          76$:    MOV     R1,T23WB                ;SET UP WRITE BUFFER ADDRESS
   4827 027672  013737  040726  030514          MOV     T24DLY,T23WB+2          ;HIGH ORDER ADDRESS BITS
   4828 027700  012737  000100  030516          MOV     #64.,T23SZ              ;SET UP BUFFER SIZE
   4829 027706  012704  030510                  MOV     #T23PK3,R4              ;R4 = POINTER TO PACKET
   4830 027712  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
   4831 027716  004737  017120                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
   4832 027722  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
   4833 027726  012702  104210                  MOV     #SC!NXM!SSR!BIT3,R2     ;SET UP EXPECTED
   4834 027732  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
   4835 027734  001417                          BEQ     80$                     ;BR, IF OK ESP. FUNCTION REJECT
   4836 027736  062737  000001  040726          ADD     #1,T24DLY               ;LOOK AT NEXT EXTENDED BITS WORTH OF MEM
   4837 027744  022737  000100  040726          CMP     #100,T24DLY             ;TOO MUCH MEMORY YET
   4838 027752  001402                          BEQ     168$                    ;BR, IF OVER 18 BIT ADDRESS
   4839 027754  000137  027636                  JMP     25$                     ;TRY AGAIN (NEXT BUNCH OF MEMORY)
   4840 027760  004737  020102          168$:   JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
   4844 027764                          ERRHRD  ERRNO,T23TM,PKTSSR              ;TSSR INCORRECT AFTER WRITE COMMAND
        027764  104456                                                                  TRAP    C$ERHRD
        027766  000476                                                                  .WORD   318
        027770  030736                                                                  .WORD   T23TM
        027772  011700                                                                  .WORD   PKTSSR
   4845 027774                          80$:    CKLOOP                          ;LOOP IF SELECTED
        027774  104406                                                                  TRAP    C$CLP1
   4846 027776                          90$:
   4847 027776                          ENDSUB                                  ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
        027776                                                                       L10045:
        027776  104403                                                                  TRAP    C$ESUB
   4848 030000  023727  002170  000031          CMP     FATFLG,#25.                     ;IS ERROR COUNT AT 25
   4849 030006  002402                          BLT     999$                            ;BR, IF LESS THAN 25
   4850 030010  004737  020154                  JSR     PC,CKDROP                       ;TRY TO DROP THE UNIT
   4851 030014                          999$:
```

```
                                ;*
                                ;
                                ;TEST 3, SUBTEST 5
                                ;
                                ;VERIFIES THAT A WRITE DATA COMMAND SPECIFYING A DATA
                                ;BUFFER STARTING IN EXISTANT MEMORY BUT RUNNING INTO
                                ;NONEXISTENT MEMORY TERMINATES WITH THE PROPER ERROR
                                ;STATUS. A LARGE ENOUGH RECORD SIZE IS SPECIFIED SUCH
                                ;THAT TAPE IS ACTUALLY MOVED AND WRITTEN.
                                ;
                                ;
                                ;
                                ;-

030014                                  BGNSUB                          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
030014                                                                                  T3.5:
030014  104402                                                                                  TRAP    C$BSUB
030016  005737  003100        9$:     TST     KTFLG                   ;CHECK FOR KT11
                                ;       BEQ     10$                     ;BR, IF NO KT11
                                ;
                                ;       DIALOG PATCH FOR NXM
                                ;
030022  000240                          NOP                             ;REMOVE WHEN NXM IS FIXED
030024  000137  030320                  JMP     85$                     ;SKIP THIS TEST
030030  004737  032132        10$:    JSR     PC,T23RT3               ;RESTORE PACKET
030034  004737  031776                JSR     PC,T23REST              ;SET COMMAND PACKET
030040  004737  032070                JSR     PC,T23RT2               ;SET UP OTHER COMMAND PACKET

                                ;*****************************************************************
                                ;
                                ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                ;
                                ;*****************************************************************

030044  004737  016644                  JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
030050  103407                          BCS     20$                     ;BR IF INIT WAS OK
030052  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
030056  010001                          MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
030060                                  ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
030060  104455                                                                          TRAP    C$ERDF
030062  000477                                                                          .WORD   319
030064  003550                                                                          .WORD   SFIERR
030066  011666                                                                          .WORD   SFIMSG
030070                          20$:
030070  012704  030360                  MOV     #T23PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS

                                ;*****************************************************************
                                ;
                                ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
                                ;
                                ;*****************************************************************

030074  004737  010332                  JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
030100  103407                          BCS     23$                     ;BR, IF COMMAND ISSUED OK
030102  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
030106  010001                          MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
```

```
4910 030110                                ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     030110  104456                                                          TRAP    C$ERHRD
     030112  000500                                                          .WORD   320
     030114  004754                                                          .WORD   WRTMSG
     030116  011666                                                          .WORD   SFIMSG
4911
4912                                    ;*************************************************************************
4913                                    ;
4914                                    ;WRITE DATA, ACK,CVC=1
4915                                    ;
4916                                    ;*************************************************************************
4917
4918 030120  012737  000000  030514  23$:   MOV     #0,T23WB+2              ;HIGH ORDER ADDRESS BITS ETC.
4919 030126  012737  140005  030510  24$:   MOV     #140005,T23PK3          ;WRITE DATA, ACK,CVC=1
4920 030134  013701  003076                MOV     FREEHI,R1               ;HIGHEST ADDRESS
4921 030140  162701  000100                SUB     #100,R1                 ;SET ADDRESS A LITTLE LOWER
4922 030144  010137  030512                MOV     R1,T23WB                ;LOAD INTO THE PACKET
4923 030150  012737  175000  030516        MOV     #64000.,T23SZ           ;SET UP BUFFER SIZE (64K BYTES)
4924 030156  012704  030510                MOV     #T23PK3,R4              ;R4 - POINTER TO PACKET
4925 030162  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
4926 030166  004737  017120                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
4927 030172  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
4928 030176  012702  104210                MOV     #SC!NXM!SSR!BIT3,R2     ;SET UP EXPECTED
4929 030202  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
4930 030204  001415                        BEQ     80$                     ;BR, IF OK ESP. FUNCTION REJECT
4931 030206  062737  000001  030514        ADD     #1,T23WB+2              ;START CUTTING THE HIGH ADDRESS BITS DOWN
4932 030214  022737  000100  030514        CMP     #100,T23WB+2
4933 030222  001341                        BNE     24$                     ;IF NOT AT ZERO, KEEP TRYING
4934 030224  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4938 030230                                ERRHRD  ERRNO,T23TMA,PKTSSR     ;TSSR INCORRECT AFTER WRITE COMMAND
     030230  104456                                                          TRAP    C$ERHRD
     030232  000501                                                          .WORD   321
     030234  031025                                                          .WORD   T23TMA
     030236  011700                                                          .WORD   PKTSSR
4939 030240                            80$:    CKLOOP                      ;LOOP IF SELECTED
     030240  104406                                                          TRAP    C$CLP1
4940 030242  004737  032070                JSR     PC,T23RT2               ;CLEAN UP PACKET
4941 030246  004737  032132                JSR     PC,T23RT3               ;RESTORE PACKET
4942
4943                                    ;*************************************************************************
4944                                    ;
4945                                    ;       ISSUE A CALL TO REWIND ROUTINE WHICH WILL WAIT QUITE SOME TIME FOR SSR
4946                                    ;
4947                                    ;*************************************************************************
4948 030252  004737  010434                JSR     PC,REWIND               ;CALL THE TAPE REWIND
4949
4950
4951
4952 030256  012737  102010  030470        MOV     #102010,T23PK2          ;REWIND (POSITION) COMMAND
4953 030264  012704  030470                MOV     #T23PK2,R4              ;LOAD R4 WITH PACKET ADDRESS
4954 030270  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE REWIND COMMAND
4955 030274  004737  017236                JSR     PC,CHKTSSR              ;WAIT FOR SSR TO SET
4956 030300  103407                        BCS     85$                     ;BR, IF TSSR IS OK (GOOD)
4957 030302  010001                        MOV     R0,R1                   ;SAVE TSSR CONTENTS
4958 030304  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
4962 030310                                ERRHRD  ERRNO,T23RWN,PKTSSR     ;TSSR IS INCORRECT AFTER REWIND
     030310  104456                                                          TRAP    C$ERHRD
```

```
        030312  000502                                                                  .WORD   322
        030314  031111                                                                  .WORD   T23RWN
        030316  011700                                                                  .WORD   PKTSSR
   4963 030320                          85$:
   4964 030320                          SKIP:
   4965 030320                          130$:
   4966 030320                                  ENDSUB                  ;>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>>
        030320                                                                  L10046:
        030320  104403                                                                  TRAP    C$ESUB
   4967 030322  023727  002170  000031          CMP     FATFLG,#25.             ;IS ERROR COUNT AT 25
   4968 030330  002402                          BLT     999$                    ;BR, IF LESS THAN 25
   4969 030332  004737  020154                  JSR     PC,CKDROP               ;TRY TO DROP THE UNIT
   4970 030336                          999$:
   4971 030336  004737  017356                  JSR     PC,TSTLOOP              ;DO WE NEED TO ITERATE TEST
   4972 030342  103002                          BCC     163$                    ;BR, IF NO LOOP REQUIRED
   4973 030344  000137  026276                  JMP     T23LOOP                 ;EXECUTE AGAIN
   4974 030350                          163$:
   4975 030350                                  EXIT    TST                     ;ALL DONE THIS TEST
        030350  104432                                                                  TRAP    C$EXIT
        030352  001604                                                                  .WORD   L10041-.
```

```
4977                                    ;*
4978                                    ;LOCAL STORAGE FOR THIS TEST
4979                                    ;-
4981 030354                                     .BLKB   10-<.-TUV2A&7>
4983 030360                             T23PACKET:                      ;COMMAND PACKET FOR TEST
4984 030360  100004                             .WORD   100004          ;WRITE CHARACTERISTICS COMMAND, WITH ACK
4985 030362  030370                             .WORD   T23DATA         ;ADDRESS OF CHARACTERISTICS BLOCK
4986 030364  000000                             .WORD   0
4987 030366  000010                             .WORD   8.              ;STARTING VALUE OF BLOCK SIZE
4988 030370                             T23DATA:                        ;CHARACTERISTICS DATA BLOCK
4989 030370  030400                             .WORD   T23BFR          ;ADDRESS OF MESSAGE BUFFER
4990 030372  000000                             .WORD   0
4991 030374  000012                             .WORD   10.             ;LENGTH OF MESSAGE BUFFER
4992 030376  000000                             .WORD   0
4993 030400                             T23BFR: .BLKW   25.             ;MESSAGE BUFFER
4994                                    ;
4995                                    ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
4996                                    ;
4998 030462                                     .BLKB   10-<.-TUV2A&7>
5000 030470                             T23PK2:
5001 030470  100006                             .WORD   100006          ;WRITE SUB SYS MEM COMMAND, AND ACK
5002 030472  030522                             .WORD   T23BF2          ;ADDRESS OF SELECT BLOCK DATA
5003 030474  000000                             .WORD   0
5004 030476  000006                             .WORD   6.              ;SIZE OF DATA PACKET
5005
5007 030500                                     .BLKB   10-<.-TUV2A&7>
5009 030510                             T23PK3:
5010 030510  100005                             .WORD   100005          ;WRITE COMMAND,  AND ACK
5011 030512  000000                     T23WB:  .WORD   0               ;ADDRESS OF WRITE BUFFER
5012 030514  000000                             .WORD   0
5013 030516  000000                     T23SZ:  .WORD   0               ;SIZE OF BUFFER (EXTENT)
5014                                            .EVEN
5015                                    ;
5016 030520  000000                     T23RSZ: .WORD   0               ;LARGEST TAPE RECORD IN BYTES
5017                                    ;
5018                                    ;
5019 030522                             T23BF2:
5020 030522     010                     T23BS0: .BYTE   10              ;BSEL0 AREA
5021 030523     200                     T23BS1: .BYTE   200             ;BSEL1 AREA
5022 030524  000000                     T23S2:  .WORD   0               ;SEL 2 AREA
5023 030526  000000                     T23S3:  .WORD   0               ;DATA AREA
5024                                    ;
5025                                    ;
5026                                            .EVEN
5027                                    ;TAPE MOTION PACKET COMMAND VALUES
5028
5029 030530  100005                     T23WD:  .WORD   100005          ;WRITE DATA (NEXT)
5030 030532  100405                     T23WDR: .WORD   100405          ;WRITE DATA RETRY
5031 030534  102005                     T23CON: .WORD   102005          ;WRITE CONTINOUS
5032 030536  177777                             .WORD   177777          ;END OF DATA
5033
5034
```

```
5036
5037                                    ;*
5038                                    ;LOCAL TEXT MESSAGES FOR TEST
5039                                    ;-
5040
5041 030540      127     122     111    T23SSR: .ASCIZ  'WRITE Command Not Accepted'
5042 030573      105     117     124    T23ET:  .ASCIZ  'EOT Not Found In 12000 4k Writes, (Use Shorter Tape)'
5043 030660      127     122     111    T23EOT: .ASCIZ  'WRITE DATA OVER EOT GAVE NO TAPE STATUS ALERT'
5044 030736      124     123     123    T23TM:  .ASCIZ  'TSSR Not Correct After WRITE Command Reject Due To NXM'
5045 031025      124     123     123    T23TMA: .ASCIZ  'TSSR Not Correct After WRITE To Non-Existent Memory'
5046 031111      122     145     167    T23RWN: .ASCIZ  'Rewind (POSITION) Command Not Accepted'
5047 031160      122     101     115    T23RNC: .ASCIZ  'RAM Error, Correct Data Pattern Not In Ram'
5048 031233      124     123     123    T23AM3: .ASCIZ  'TSSR Init. Failed After WRITE Command'
5049 031301      104     162     151    T23OFL: .ASCIZ  'Drive 7 Select Failed To Set "OFL" In TSSR'
5050 031354      124     123     123    T23WDD: .ASCIZ  'TSSR Not Correct After WRITE DATA Command, SWB Bit Set'
5051 031443      124     123     123    T23WDC: .ASCIZ  'TSSR Not Correct After WRITE DATA Command, Check For Tape Offline'
5052 031545      103     126     103    T23VCK: .ASCIZ  'CVC Set, Didn''t Reset VCK In Message Buffer'
5053 031620      124     123     102    T23BA:  .ASCIZ  'TSBA Not Correct After WRITE DATA Command'
5054 031672      127     122     111    T23WSS: .ASCIZ  'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
5055 031761      102     141     163    TST23ID:        .ASCIZ  'Basic Write'
5056                                            .EVEN
5057                                    ;*
5058                                    ;
5059                                    ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
5060                                    ;WRITE SUBSYSTEM MEMORY COMMAND
5061                                    ;
5062                                    ;-
5063
5064 031776                      T23REST:
5065 031776                              SAVREG                          ;SAVE THE REGISTERS
5066 032002  012701  030360              MOV     #T23PACKET,R1           ;START OF THE PACKET
5067 032006  012721  100004              MOV     #100004,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK
5068 032012  012721  030370              MOV     #T23DATA,(R1)+          ;ADDRESS OF CHARAISTICS DATA BLOCK
5069 032016  005021                      CLR     (R1)+                   ;EXTENDED ADDRESS
5070 032020  012721  000012              MOV     #10.,(R1)+              ;SIZE OF DATA BLOCK IN BYTES
5071 032024  012721  030400              MOV     #T23BFR,(R1)+           ;ADDRESS OF MESSAGE BUFFER
5072 032030  005021                      CLR     (R1)+
5073 032032  012721  000024              MOV     #20.,(R1)+              ;LENGTH OF MESSAGE BUFFER
5074 032036  005021                      CLR     (R1)+
5075 032040  012711  000000              MOV     #0,(R1)                 ;SELECT DRIVE ZERO
5076 032044  012702  000030              MOV     #24.,R2                 ;NUMBER OF LOCATIONS TO BE CLEARED
5077 032050  012762  177777  030400  64$:  MOV     #177777,T23BFR(R2)      ;ALL ONES TO MESSAGE BUFFER
5078 032056  005742                      TST     -(R2)                   ;BUMP DOWN TO NEXT LOCATION
5079 032060  020227  000000              CMP     R2,#0                   ;R2 AT ZERO YET
5080 032064  001371                      BNE     64$                     ;KEEP GOING UNTIL DONE
5081 032066  000207                      RTS     PC                      ;RETURN
5082
5083
5084 032070                      T23RT2:
5085 032070                              SAVREG                          ;SAVE THE REGISTERS
5086 032074  012701  030470              MOV     #T23PK2,R1              ;START OF THE PACKET
5087 032100  012721  100006              MOV     #100006,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK
5088 032104  012721  030522              MOV     #T23BF2,(R1)+           ;ADDRESS OF DATA BLOCK
5089 032110  005021                      CLR     (R1)+                   ;EXTENDED ADDRESS
5090 032112  012721  000006              MOV     #6.,(R1)+               ;SIZE OF DATA BLOCK IN BYTES
5091 032116  012701  030522              MOV     #T23BF2,R1              ;POINT TO DATA SEL AREA
5092 032122  005021                      CLR     (R1)+
```

```
5093 032124  005021                        CLR     (R1)+
5094 032126  005011                        CLR     (R1)
5095 032130  000207                        RTS     PC                      ;RETURN
5096 032132                        T23RT3:
5097 032132                                SAVREG                          ;SAVE THE REGISTERS
5098 032136  012701  030510                MOV     #T23PK3,R1              ;START OF THE PACKET
5099 032142  012721  100005                MOV     #100005,(R1)+           ;WRITE  TAPE, WITH ACK
5100 032146  005021                        CLR     (R1)+                   ;ADDRESS OF DATA BLOCK
5101 032150  005021                        CLR     (R1)+                   ;EXTENDED ADDRESS
5102 032152  005011                        CLR     (R1)                    ;SIZE OF DATA BLOCK
5103 032154  000207                        RTS     PC                              ;RETURN
5104 032156                                ENDTST
     032156                                                          L10041:
     032156  104401                                                          TRAP    C$ETST
```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```
                                        .SBTTL  TEST  4: BASIC READ DATA (FORWARD AND REVERSE)
                                ;+
                                ;
                                ;THIS TEST VERIFIES THAT THE READ FORWARD AND READ REVERSE
                                ;COMMANDS OPERATE PROPERLY. VARIOUS COMBINATIONS OF ODD AND EVEN
                                ;DATA BUFFER BOUNDARIES, RECORD SIZES (UP TO 64K BYTES IF MEMORY
                                ;SPACE IS AVAILIABLE), AND BYTE-SWAP CONTROL ARE USED. THIS TEST
                                ;OF COURSE, FURTHER VERIFIES THE WRITE DATA COMMAND BY ACTUALLY
                                ;READING AND VERIFYING WRITTEN DATA. ALSO TESTED ARE PROPER
                                ;TERMINATIONS ON EXCEPTIONAL OR ERROR CONDITIONS: RECORD LENGTH
                                ;LONG, RECORD LENGTH SHORT, READ REVERSE AT BOT, ILLEGAL DATA
                                ;BUFFER ADDRESSES, ILLEGAL CODES IN THE MODE FIELD OF THE BASIC
                                ;READ COMMAND, AND DATA BUFFERS IN NON-EXISTANT MEMORY. THE TEST
                                ;
                                ;
                                ;THE TEST CONSISTS OF THE FOLLOWING 12 SUBTESTS
                                ;
                                ;
                                ;
                                ;-
032160                                  BGNTST
032160                                                                  T4::
032160  005037  002170                    CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
032164  005037  003100                    CLR     KTFLG           ;HOLD OFF KT11
032170  012737  005762  002146            MOV     #EPRT1,EPRTSW           ;SET UP PRIMARY ERROR MESSAGE
032176  005037  003102                    CLR     KTENABLE                ;TURN OFF KT11
032202  004737  020246                    JSR     PC,KTOFF                ;TURN KT11 OFF
032206  012700  043172                    MOV     #TST24ID,R0             ;ASCII MESSAGE TO IDENTIFY TEST
032212  004737  017410                    JSR     PC,TSTSETUP             ;DO INITIAL TEST SETUP
032216  012737  000001  002164            MOV     #1,LOOPCNT              ;PERFORM 1 ITERATIONS
                                ;+
032224                          T24LOOP:
                                ;
```

```
                                        ;+
                                        ;
                                        ;TEST 4, SUBTEST 1
                                        ;
                                        ;VERIFIES THAT READ FORWARD COMMANDS WITH SWB=0
                                        ;OPERATES PROPERLY. THE TAPE IS FIRST REWOUND AND THEN
                                        ;WRITTEN WITH A SERIES OF TEST RECORDS VARING IN
                                        ;LENGTH AND DATA CONTENT. THE TAPE IS THEN REWOUND
                                        ;AGAIN AND THE RECORD READ SEQUENTIALLY AND RESULTS
                                        ;(STATUS, DATA, ETC.) VERIFIED. THE BYTE COUNT ON
                                        ;EACH READ FORWARD COMMAND IS SET TO THE LENGTH OF THE
                                        ;EXPECTED RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD
                                        ;OCCUR.
                                        ;
                                        ;
                                        ;-
032224                                          BGNSUB                  ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
032224                                                                                  T4.1:
032224  104402                                                                          TRAP    C$BSUB
032226  004737  043374                        JSR     PC,T24RT3       ;SET UP OTHER COMMAND PACKET
032232  004737  043240                        JSR     PC,T24REST      ;SET COMMAND PACKET
032236  004737  043332                        JSR     PC,T24RT2       ;SET UP OTHER COMMAND PACKET
032242                                          RFLAGS  R0              ;READ SUPERVISOR FLAGS
032242  104421                                                                          TRAP    C$RFLA
032244  032700  000100                        BIT     #BIT6,R0        ;CHECK FOR THE ILS BIT SET
032250  001002                                BNE     9$              ;IF SET SKIP THE PATCH
032252  000137  032776                        JMP     192$            ;PATCH IT OUT
032256                                  9$:
                                        ;*****************************************************************
                                        ;
                                        ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                        ;
                                        ;*****************************************************************

032256  004737  016644                        JSR     PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
032262  103407                                BCS     20$             ;BR IF INIT WAS OK
032264  004737  020102                        JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
032270  010001                                MOV     R0,R1           ;CONTENTS OF TSSR REGISTER
032272                                          ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
032272  104455                                                                          TRAP    C$ERDF
032274  000621                                                                          .WORD   401
032276  003550                                                                          .WORD   SFIERR
032300  011666                                                                          .WORD   SFIMSG
032302                                  20$:
032302  012704  040550                        MOV     #T24PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
                                        ;*****************************************************************
                                        ;
                                        ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
                                        ;
                                        ;*****************************************************************

032306  004737  010332                        JSR     PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
032312  103407                                BCS     24$             ;BR, IF COMMAND ISSUED OK
032314  004737  020102                        JSR     PC,FATCHK       ;INC AND CHECK FOR MORE THAN 25 ERRORS
032320  010001                                MOV     R0,R1           ;SAVE CONTENTS OF TSSR
```

```
5200 032322                                ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     032322  104456                                                          TRAP    C$ERHRD
     032324  000622                                                          .WORD   402
     032326  004754                                                          .WORD   WRTMSG
     032330  011666                                                          .WORD   SFIMSG
5201 032332                          24$:  CKLOOP                           ;LOOP IF SELECTED
     032332  104406                                                          TRAP    C$CLP1
5202
5203                                 ;*****************************************************************
5204                                 ;
5205                                 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5206                                 ;
5207                                 ;*****************************************************************
5208
5209 032334  004737  010434                JSR     PC,REWIND              ;CALL TAPE REWIND COMMAND
5210 032340  103407                        BCS     30$                    ;BR, IF NO PROBLEM
5211 032342  010001                        MOV     R0,R1                  ;SAVE TSSR
5212 032344  004737  020102                JSR     PC,FATCHK              ;INC AND CHECK FOR MORE THAN 25 ERRORS
5216 032350                                ERRHRD  ERRNO,T24RWN,PKTSSR    ;REWIND NOT ACCEPTED
     032350  104456                                                          TRAP    C$ERHRD
     032352  000623                                                          .WORD   403
     032354  042036                                                          .WORD   T24RWN
     032356  011700                                                          .WORD   PKTSSR
5217 032360                          30$:  CKLOOP                           ;LOOP IF SELECTED
     032360  104406                                                          TRAP    C$CLP1
5218
5219                                 ;*****************************************************************
5220                                 ;
5221                                 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
5222                                 ;
5223                                 ;*****************************************************************
5224
5225 032362  013701  040576                MOV     T24BFR+6,R1            ;PICK UP XST0
5226 032366  010102                        MOV     R1,R2                  ;SET UP EXPECTED
5227 032370  052702  000002                BIS     #BIT1,R2               ;SET BOT BIT IN EXPECTED
5228 032374  020102                        CMP     R1,R2                  ;DOES EXP - REC D
5229 032376  001406                        BEQ     40$                    ;BR, IF EQUAL (OK)
5230 032400  004737  020102                JSR     PC,FATCHK              ;INC AND CHECK FOR MORE THAN 25 ERRORS
5234 032404                                ERRHRD  ERRNO,T24BOT,EXPREC    ;TAPE NOT AT BOT AFTER REWIND
     032404  104456                                                          TRAP    C$ERHRD
     032406  000624                                                          .WORD   404
     032410  041553                                                          .WORD   T24BOT
     032412  015344                                                          .WORD   EXPREC
5235 032414                          40$:  CKLOOP                           ;LOOP IF SELECTED
     032414  104406                                                          TRAP    C$CLP1
5236 032416  012703  000400                MOV     #256.,R3               ;RECORD SIZE
5237 032422  013737  003072  040702        MOV     FREE,T24RB             ;STARTING WRITE BUFFER ADDRESS
5238
5239                                 ;*****************************************************************
5240                                 ;
5241                                 ;WRITE DATA,CVC=1,ACK COMMAND
5242                                 ;
5243                                 ;*****************************************************************
5244
5245 032430  012737  140005  040700        MOV     #140005,T24PK3         ;WRITE DATA,CVC=1,ACK COMMAND
5246 032436  012704  040700                MOV     #T24PK3,R4             ;SET UP R4 WITH PACKET ADDRESS
5247 032442                          65$:
```

```
5248 032442  010300                    MOV     R3,R0                   ;SET PATTERN IN CORRECT REGISTER
5249 032444  004737  020374             JSR     PC,FILLMEM              ;FILL MEMORY WITH RECORD SIZE
5250 032450  010337  040706             MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
5251 032454  010465  177776             MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5252 032460  004737  017120             JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5253 032464  016501  000000             MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5254 032470  012702  000200             MOV     #SSR,R2                 ;SET UP EXPECTED
5255 032474  020102                    CMP     R1,R2                   ;ARE THEY EQUAL
5256 032476  001406                    BEQ     75$                     ;BR, IF OK
5257 032500  004737  020102             JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5261                                                                    ;SOFT ERROR, REALLY CHECKING THE
5262                                                                    ;READ COMMAND
5263 032504                            ERRSOFT ERRNO,WRTERR,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     032504  104457                                                                    TRAP    C$ERSOFT
     032506  000625                                                                    .WORD   405
     032510  005011                                                                    .WORD   WRTERR
     032512  011700                                                                    .WORD   PKTSSR
5264 032514                    75$:    CKLOOP                          ;LOOP IF SELECTED
     032514  104406                                                                    TRAP    C$CLP1
5265 032516  005723                    TST     (R3)+           ;BUMP RECORD SIZE
5266 032520  022703  000414             CMP     #268.,R3                ;END OF RECORD YET
5267 032524  001346                    BNE     65$                     ;BR, IF MORE RECORDS TO WRITE
5268 032526                    80$:    CKLOOP                          ;LOOP IF SELECTED
     032526  104406                                                                    TRAP    C$CLP1
5269 032530                    120$:
5270
5271                                   ;*************************************************************************
5272                                   ;
5273                                   ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5274                                   ;
5275                                   ;*************************************************************************
5276
5277 032530  012703  000012                    MOV     #10.,R3                 ;SPECIAL MULTIPLE REWIND
5278 032534  004737  010434             125$:  JSR     PC,REWIND               ;ISSUE REWIND COMMAND
5279 032540  103407                    BCS     130$                    ;BR, IF NO PROBLEM
5280 032542  010001                    MOV     R0,R1                   ;SAVE TSSR
5281 032544  004737  020102             JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5285 032550                            ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
     032550  104456                                                                    TRAP    C$ERHRD
     032552  000626                                                                    .WORD   406
     032554  042036                                                                    .WORD   T24RWN
     032556  011700                                                                    .WORD   PKTSSR
5286 032560                    130$:   CKLOOP                          ;LOOP IF SELECTED
     032560  104406                                                                    TRAP    C$CLP1
5287
5288                                   ;*************************************************************************
5289                                   ;
5290                                   ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
5291                                   ;
5292                                   ;*************************************************************************
5293
5294 032562  013701  040576                    MOV     T24BFR+6,R1             ;PICK UP XST0
5295 032566  010102                    MOV     R1,R2                   ;SET UP EXPECTED
5296 032570  052702  000002             BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
5297 032574  020102                    CMP     R1,R2                   ;DOES EXP = REC'D
5298 032576  001407                    BEQ     140$                    ;BR, IF EQUAL (OK)
5299 032600  077323                    SOB     R3,125$                 ;DO ANOTHER REWIND BEFORE REPORTING ERROR
```

```
5300 032602  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5304 032606                                    ERRHRD  ERRNO,T24BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     032606  104456                                                                    TRAP    C$ERHRD
     032610  000627                                                                    .WORD   407
     032612  041553                                                                    .WORD   T24BOT
     032614  016344                                                                    .WORD   EXPREC
5305 032616                            140$:   CKLOOP                          ;LOOP IF SELECTED
     032616  104406                                                                    TRAP    C$CLP1
5306 032620  012703  000400                    MOV     #256.,R3                ;RECORD SIZE
5307 032624  013737  003072  040702            MOV     FREE,T24RB              ;STARTING READ BUFFER ADDRESS
5308
5309                                   ;*****************************************************************
5310                                   ;
5311                                   ;READ DATA,CVC=1,ACK COMMAND
5312                                   ;
5313                                   ;*****************************************************************
5314
5315 032632  012737  140001  040700            MOV     #140001,T24PK3          ;READ DATA,CVC=1,ACK COMMAND
5316 032640  012704  040700            165$:   MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5317 032644  010337  040706                    MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
5318 032650  010465  177776                    MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5319 032654  004737  017120                    JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5320 032660  016501  000000                    MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5321 032664  012702  000200                    MOV     #SSR,R2                 ;SET UP EXPECTED
5322 032670  020102                            CMP     R1,R2                   ;ARE THEY EQUAL
5323 032672  001406                            BEQ     170$                    ;BR, IF OK
5324 032674  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5328 032700                                    ERRHRD  ERRNO,RDERR,PKTSSR      ;TSSR INCORRECT AFTER READ DATA
     032700  104456                                                                    TRAP    C$ERHRD
     032702  000630                                                                    .WORD   408
     032704  005104                                                                    .WORD   RDERR
     032706  011700                                                                    .WORD   PKTSSR
5329 032710                            170$:   CKLOOP                          ;LOOP IF SELECTED
     032710  104406                                                                    TRAP    C$CLP1
5330 032712  013702  003072                    MOV     FREE,R2                 ;GET BUFFER ADDRESS
5331 032716  010304                            MOV     R3,R4                   ;CURRENT RECORD SIZE
5332 032720  162704  000400                    SUB     #256.,R4                ;FIRST LOCATION IN BUFFER
5333 032724  060204                    173$:   ADD     R2,R4                   ;GET LOCATION IN BUFFER (ADDRESS)
5334 032726  021403                            CMP     (R4),R3                 ;CHECK DATA READ (R3=DATA ALSO)
5335 032730  001410                            BEQ     180$                    ;BR, IF ALL IS WELL
5336 032732  011401                            MOV     (R4),R1                 ;RECD DATA
5337 032734  010302                            MOV     R3,R2                   ;EXPECTED DATA
5338 032736  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5342 032742                                    ERRHRD  ERRNO,T24DTA,EXPREC     ;DATA READ NOT = WRITTEN
     032742  104456                                                                    TRAP    C$ERHRD
     032744  000631                                                                    .WORD   409
     032746  041620                                                                    .WORD   T24DTA
     032750  016344                                                                    .WORD   EXPREC
5343 032752                            180$:   CKLOOP                          ;LOOP IF SELECTED
     032752  104406                                                                    TRAP    C$CLP1
5344 032754  005724                            TST     (R4)+           ;BUMP TO NEXT LOCATION
5345 032756  160204                            SUB     R2,R4           ;GET BACK TO CORRECT SIZE
5346 032760  020403                            CMP     R4,R3                   ;END OF RECORD YET
5347 032762  001360                            BNE     173$                    ;BR, IF NOT AT END OF RECORD
5348 032764  005723                            TST     (R3)+                   ;BUMP RECORD SIZE
5349 032766  022703  000412                    CMP     #266.,R3                ;END OF RECORD YET
5350 032772  001322                            BNE     165$                    ;BR, IF MORE RECORDS TO WRITE
```

```
5351 032774                          190$:   CKLOOP                          ;LOOP IF SELECTED
     032774  104406                                                                  TRAP    C$CLP1
5352 032776                          192$:
5353 032776                                  ENDSUB                          ;>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>>
     032776                                                                        L10050:
     032776  104403                                                                  TRAP    C$ESUB
5354 033000  023727  002170  000031          CMP     FATFLG,#25.                     ;IS ERROR COUNT AT 25
5355 033006  002402                          BLT     999$                            ;BR, IF LESS THAN 25
5356 033010  004737  020154                  JSR     PC,CKDROP                       ;TRY TO DROP THE UNIT
5357 033014                          999$:
```

```
                                        ;*
                                        ;
                                        ;TEST 4, SUBTEST 2
                                        ;
                                        ;VERIFIES THAT READ DATA COMMANDS WITH CVC=1 AND THE
                                        ;SWAP BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST
                                        ;SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2.
                                        ;THE RESULTS, EXCEPT FOR RAM CONTENTS, SHOULD BE THE SAME.
                                        ;
                                        ;
                                        ;
                                        ;-
033014                                          BGNSUB                          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
033014                                                                                  T4.2:
033014  104402                                                                          TRAP    C$BSUB
033016  004737  043374                          JSR     PC,T24RT3               ;SET UP OTHER COMMAND PACKET
033022  004737  043240                          JSR     PC,T24REST              ;SET COMMAND PACKET
033026  004737  043332                          JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET

                                        ;*************************************************************************
                                        ;
                                        ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                        ;
                                        ;*************************************************************************

033032  004737  016644                          JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
033036  103407                                  BCS     20$                     ;BR IF INIT WAS OK
033040  004737  020102                          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
033044  010001                                  MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
033046                                          ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
033046  104455                                                                          TRAP    C$ERDF
033050  000632                                                                          .WORD   410
033052  003550                                                                          .WORD   SFIERR
033054  011666                                                                          .WORD   SFIMSG
033056                                  20$:
033056  012704  040550                          MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS

                                        ;*************************************************************************
                                        ;
                                        ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
                                        ;
                                        ;*************************************************************************

033062  004737  010332                          JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
033066  103407                                  BCS     24$                     ;BR, IF COMMAND ISSUED OK
033070  004737  020102                          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
033074  010001                                  MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
033076                                          ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
033076  104456                                                                          TRAP    C$ERHRD
033100  000633                                                                          .WORD   411
033102  004754                                                                          .WORD   WRTMSG
033104  011666                                                                          .WORD   SFIMSG
033106                                  24$:    CKLOOP                          ;LOOP IF SELECTED
033106  104406                                                                          TRAP    C$CLP1

                                        ;*************************************************************************
                                        ;
```

```
                                ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
                                ;
                                ;*************************************************************
033110  004737  010434                JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
033114  103407                        BCS     30$                     ;BR, IF NO PROBLEM
033116  010001                        MOV     R0,R1                   ;SAVE TSSR
033120  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
033124                                ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
033124  104456                                                                TRAP    C$ERHRD
033126  000634                                                                .WORD   412
033130  042036                                                                .WORD   T24RWN
033132  011700                                                                .WORD   PKTSSR
033134                          30$:    CKLOOP                          ;LOOP IF SELECTED
033134  104406                                                                TRAP    C$CLP1
                                ;
                                ;*************************************************************
                                ;
                                ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
                                ;
                                ;*************************************************************
033136  013701  040576                MOV     T24BFR+6,R1             ;PICK UP XST0
033142  010102                        MOV     R1,R2                   ;SET UP EXPECTED
033144  052702  000002                BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
033150  020102                        CMP     R1,R2                   ;DOES EXP = REC'D
033152  001406                        BEQ     40$                     ;BR, IF EQUAL (OK)
033154  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
033160                                ERRHRD  ERRNO,T24BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
033160  104456                                                                TRAP    C$ERHRD
033162  000635                                                                .WORD   413
033164  041553                                                                .WORD   T24BOT
033166  016344                                                                .WORD   EXPREC
033170                          40$:    CKLOOP                          ;LOOP IF SELECTED
033170  104406                                                                TRAP    C$CLP1
033172  012703  000400                MOV     #256.,R3                ;RECORD SIZE
033176  013737  003072  040702        MOV     FREE,T24RB              ;STARTING WRITE BUFFER ADDRESS
                                ;
                                ;*************************************************************
                                ;
                                ;WRITE DATA,ACK,SWB,CVC=1 COMMAND
                                ;
                                ;*************************************************************
033204  012737  150005  040700        MOV     #150005,T24PK3          ;WRITE DATA,ACK,SWB,CVC=1 COMMAND
033212  012704  040700                MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
033216                          65$:
033216  010300                        MOV     R3,R0                   ;SET PATTERN IN CORRECT REGISTER
033220  004737  020374                JSR     PC,FILLMEM              ;FILL MEMORY WITH RECORD SIZE
033224  010337  040706                MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
033230  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
033234  004737  017120                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
033240  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
033244  012702  000200                MOV     #SSR,R2                 ;SET UP EXPECTED
033250  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
033252  001406                        BEQ     75$                     ;BR, IF OK
033254  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
```

```
5467                                                            ;SOFT ERROR, REALLY CHECKING THE
5468                                                            ;READ COMMAND
5469 033260                        ERRSOFT ERRNO,WRTERR,PKTSSR  ;TSSR INCORRECT AFTER WRITE DATA
     033260  104457                                                     TRAP    C$ERSOFT
     033262  000636                                                     .WORD   414
     033264  005011                                                     .WORD   WRTERR
     033266  011700                                                     .WORD   PKTSSR
5470 033270                    75$:    CKLOOP                          ;LOOP IF SELECTED
     033270  104406                                                     TRAP    C$CLP1
5471 033272  005723                    TST     (R3)+                   ;BUMP RECORD SIZE
5472 033274  022703  000414            CMP     #268.,R3                ;END OF RECORD YET
5473 033300  001346                    BNE     65$                     ;BR, IF MORE RECORDS TO WRITE
5474 033302                    80$:    CKLOOP                          ;LOOP IF SELECTED
     033302  104406                                                     TRAP    C$CLP1
5475 033304                    120$:
5476
5477                           ;*****************************************************************
5478                           ;
5479                           ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5480                           ;
5481                           ;*****************************************************************
5482
5483 033304  004737  010434            JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
5484 033310  103407                    BCS     130$                    ;BR, IF NO PROBLEM
5485 033312  010001                    MOV     R0,R1                   ;SAVE TSSR
5486 033314  004737  020102            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5490 033320                            ERRHRD  ERRNO,T24RWN,EXPREC     ;REWIND NOT ACCEPTED
     033320  104456                                                     TRAP    C$ERHRD
     033322  000637                                                     .WORD   415
     033324  042036                                                     .WORD   T24RWN
     033326  016344                                                     .WORD   EXPREC
5491 033330                    130$:   CKLOOP                          ;LOOP IF SELECTED
     033330  104406                                                     TRAP    C$CLP1
5492
5493                           ;*****************************************************************
5494                           ;
5495                           ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
5496                           ;
5497                           ;*****************************************************************
5498
5499 033332  013701  040576            MOV     T24BFR+6,R1             ;PICK UP XST0
5500 033336  010102                    MOV     R1,R2                   ;SET UP EXPECTED
5501 033340  052702  000002            BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
5502 033344  020102                    CMP     R1,R2                   ;DOES EXP = REC'D
5503 033346  001406                    BEQ     140$                    ;BR, IF EQUAL (OK)
5504 033350  004737  020102            JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5508 033354                            ERRHRD  ERRNO,T24BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     033354  104456                                                     TRAP    C$ERHRD
     033356  000640                                                     .WORD   416
     033360  041553                                                     .WORD   T24BOT
     033362  016344                                                     .WORD   EXPREC
5509 033364                    140$:   CKLOOP                          ;LOOP IF SELECTED
     033364  104406                                                     TRAP    C$CLP1
5510 033366  012703  000400            MOV     #256.,R3                ;RECORD SIZE
5511 033372  013737  003072  040702    MOV     FREE,T24RB              ;STARTING READ BUFFER ADDRESS
5512
5513                           ;*****************************************************************
```

```
5514                                    ;
5515                                    ;READ DATA,IE,ACK,SWB COMMAND
5516                                    ;
5517                                    ;*********************************************************************
5518
5519 033400  012737  110001  040700            MOV     #110001,T24PK3          ;READ DATA,IE,ACK,SWB COMMAND
5520 033406  012704  040700          165$:     MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5521 033412  010337  040706                    MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
5522 033416  010465  177776                    MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5523 033422  004737  017120                    JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5524 033426  016501  000000                    MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5525 033432  012702  000200                    MOV     #SSR,R2                 ;SET UP EXPECTED
5526 033436  020102                            CMP     R1,R2                   ;ARE THEY EQUAL
5527 033440  001406                            BEQ     170$                    ;BR, IF OK
5528 033442  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5532 033446                                    ERRHRD  ERRNO,RDERR,PKTSSR      ;TSSR INCORRECT AFTER READ DATA
     033446  104456                                                                    TRAP    C$ERHRD
     033450  000641                                                                    .WORD   417
     033452  005104                                                                    .WORD   RDERR
     033454  011700                                                                    .WORD   PKTSSR
5533 033456                          170$:     CKLOOP                          ;LOOP IF SELECTED
     033456  104406                                                                    TRAP    C$CLP1
5534 033460  013702  003072                    MOV     FREE,R2                 ;GET BUFFER ADDRESS
5535 033464  010304                            MOV     R3,R4                   ;CURRENT RECORD SIZE
5536 033466  162704  000400                    SUB     #256.,R4                ;FIRST LOCATION IN BUFFER
5537 033472  060204                  173$:     ADD     R2,R4                   ;GET LOCATION IN BUFFER (ADDRESS)
5538 033474  021403                            CMP     (R4),R3                 ;CHECK DATA READ (R3=DATA ALSO)
5539 033476  001410                            BEQ     180$                    ;BR, IF ALL IS WELL
5540 033500  011401                            MOV     (R4),R1                 ;RECD DATA
5541 033502  010302                            MOV     R3,R2                   ;EXPECTED DATA
5542 033504  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5546 033510                                    ERRHRD  ERRNO,T24DTA,EXPREC     ;DATA READ NOT = WRITTEN
     033510  104456                                                                    TRAP    C$ERHRD
     033512  000642                                                                    .WORD   418
     033514  041620                                                                    .WORD   T24DTA
     033516  016344                                                                    .WORD   EXPREC
5547 033520                          180$:     CKLOOP                          ;LOOP IF SELECTED
     033520  104406                                                                    TRAP    C$CLP1
5548 033522  005724                            TST     (R4)+                   ;BUMP TO NEXT LOCATION
5549 033524  160204                            SUB     R2,R4                   ;SET SIZE TO CORRECT VALUE
5550 033526  020403                            CMP     R4,R3                   ;END OF RECORD YET
5551 033530  001360                            BNE     173$                    ;BR, IF NOT AT END OF RECORD
5552 033532  005723                            TST     (R3)+                   ;BUMP RECORD SIZE
5553 033534  022703  000412                    CMP     #266.,R3                ;END OF RECORD YET
5554 033540  001322                            BNE     165$                    ;BR, IF MORE RECORDS TO WRITE
5555 033542                          190$:     CKLOOP                          ;LOOP IF SELECTED
     033542  104406                                                                    TRAP    C$CLP1
5556 033544                                    ENDSUB                          ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
     033544                                                                         L10051:
     033544  104403                                                                    TRAP    C$ESUB
5557 033546  023727  002170  000031                CMP     FATFLG,#25.                     ;IS ERROR COUNT AT 25
5558 033554  002402                            BLT     999$                            ;BR, IF LESS THAN 25
5559 033556  004737  020154                    JSR     PC,CKDROP                       ;TRY TO DROP THE UNIT
5560 033562                          999$:
```

```
5562                                    ;+
5563                                    ;
5564                                    ;TEST 4, SUBTEST 3
5565                                    ;
5566                                    ;VERIFIES THAT A READ FORWARD COMMAND READING A RECORD
5567                                    ;LONGER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE
5568                                    ;STATUS ALERT TERMINATION WITH THE RECORD LENGTH LONG
5569                                    ;(RLL) BIT SET.
5570                                    ;
5571                                    ;
5572                                    ;-
5573
5574 033562                                    BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
      033562                                                                                    T4.3:
      033562  104402                                                                              TRAP    C$BSUB
5575 033564  004737  043374                    JSR     PC,T24RT3               ;SET UP OTHER COMMAND PACKET
5576 033570  004737  043240                    JSR     PC,T24REST              ;SET COMMAND PACKET
5577 033574  004737  043332                    JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET
5578
5579                                    ;*****************************************************************
5580                                    ;
5581                                    ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
5582                                    ;
5583                                    ;*****************************************************************
5584
5585 033600  004737  016644                    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
5586 033604  103407                            BCS     20$                     ;BR IF INIT WAS OK
5587 033606  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5591 033612  010001                            MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
5592 033614                                    ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      033614  104455                                                                            TRAP    C$ERDF
      033616  000643                                                                            .WORD   419
      033620  003550                                                                            .WORD   SFIERR
      033622  011666                                                                            .WORD   SFIMSG
5593 033624                                    20$:
5594 033624  012704  040550                    MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
5595
5596                                    ;*****************************************************************
5597                                    ;
5598                                    ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
5599                                    ;
5600                                    ;*****************************************************************
5601
5602 033630  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
5603 033634  103407                            BCS     24$                     ;BR, IF COMMAND ISSUED OK
5604 033636  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5608 033642  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
5609 033644                                    ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      033644  104456                                                                            TRAP    C$ERHRD
      033646  000644                                                                            .WORD   420
      033650  004754                                                                            .WORD   WRTMSG
      033652  011666                                                                            .WORD   SFIMSG
5610 033654                                    24$:    CKLOOP                  ;LOOP IF SELECTED
      033654  104406                                                                            TRAP    C$CLP1
5611
5612                                    ;*****************************************************************
5613                                    ;
```

```
5614                                    ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5615                                    ;
5616                                    ;*****************************************************************
5617
5618 033656  004737  010434                    JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
5619 033662  103407                            BCS     30$                     ;BR, IF NO PROBLEM
5620 033664  010001                            MOV     R0,R1                   ;SAVE TSSR
5621 033666  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5625 033672                                    ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
      033672  104456                                                                    TRAP    C$ERHRD
      033674  000645                                                                    .WORD   421
      033676  042036                                                                    .WORD   T24RWN
      033700  011700                                                                    .WORD   PKTSSR
5626 033702                             30$:    CKLOOP                          ;LOOP IF SELECTED
      033702  104406                                                                    TRAP    C$CLP1
5627
5628                                    ;*****************************************************************
5629                                    ;
5630                                    ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
5631                                    ;
5632                                    ;*****************************************************************
5633
5634 033704  013701  040576                    MOV     T24BFR+6,R1             ;PICK UP XST0
5635 033710  010102                            MOV     R1,R2                   ;SET UP EXPECTED
5636 033712  052702  000002                    BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
5637 033716  020102                            CMP     R1,R2                   ;DOES EXP = REC'D
5638 033720  001406                            BEQ     40$                     ;BR, IF EQUAL (OK)
5639 033722  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5643 033726                                    ERRHRD  ERRNO,T24BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
      033726  104456                                                                    TRAP    C$ERHRD
      033730  000646                                                                    .WORD   422
      033732  041553                                                                    .WORD   T24BOT
      033734  016344                                                                    .WORD   EXPREC
5644 033736                             40$:    CKLOOP                          ;LOOP IF SELECTED
      033736  104406                                                                    TRAP    C$CLP1
5645 033740  012703  001000                    MOV     #512.,R3                ;RECORD SIZE
5646 033744  013737  003072  040702            MOV     FREE,T24RB              ;STARTING WRITE BUFFER ADDRESS
5647
5648                                    ;*****************************************************************
5649                                    ;
5650                                    ;WRITE DATA,ACK,CVC=1 COMMAND
5651                                    ;
5652                                    ;*****************************************************************
5653
5654 033752  012737  140005  040700            MOV     #140005,T24PK3          ;WRITE DATA,ACK,CVC=1 COMMAND
5655 033760  012704  040700                    MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5656 033764                             65$:
5657 033764  010337  040706                    MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
5658 033770  010465  177776                    MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5659 033774  004737  017120                    JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5660 034000  016501  000000                    MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5661 034004  012702  000200                    MOV     #SSR,R2                 ;SET UP EXPECTED
5662 034010  020102                            CMP     R1,R2                   ;ARE THEY EQUAL
5663 034012  001406                            BEQ     75$                     ;BR, IF OK
5664 034014  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5668                                                                           ;SOFT ERROR, REALLY CHECKING THE
5669                                                                           ;READ DATA COMMAND
```

```
5670 034020                          ERRSOFT ERRNO,WRTERR,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     034020  104457                                                          TRAP    C$ERSOFT
     034022  000647                                                          .WORD   423
     034024  005011                                                          .WORD   WRTERR
     034026  011700                                                          .WORD   PKTSSR
5671 034030                    75$:  CKLOOP                          ;LOOP IF SELECTED
     034030  104406                                                          TRAP    C$CLP1
5672 034032                    120$:
5673
5674                           ;*************************************************************
5675                           ;
5676                           ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5677                           ;
5678                           ;*************************************************************
5679
5680 034032  004737  010434          JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
5681 034036  103407                  BCS     130$                    ;BR, IF NO PROBLEM
5682 034040  010001                  MOV     R0,R1                   ;SAVE TSSR
5683 034042  004737  020102          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5687 034046                          ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
     034046  104456                                                          TRAP    C$ERHRD
     034050  000650                                                          .WORD   424
     034052  042036                                                          .WORD   T24RWN
     034054  011700                                                          .WORD   PKTSSR
5688 034056                    130$: CKLOOP                          ;LOOP IF SELECTED
     034056  104406                                                          TRAP    C$CLP1
5689
5690                           ;*************************************************************
5691                           ;
5692                           ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
5693                           ;
5694                           ;*************************************************************
5695
5696 034060  013701  040576          MOV     T24BFR+6,R1             ;PICK UP XST0
5697 034064  010102                  MOV     R1,R2                   ;SET UP EXPECTED
5698 034066  052702  000002          BIS     #BIT1,R2                ;SET BOT BIT IN EXPECTED
5699 034072  020102                  CMP     R1,R2                   ;DOES EXP = REC'D
5700 034074  001406                  BEQ     140$                    ;BR, IF EQUAL (OK)
5701 034076  004737  020102          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5705 034102                          ERRHRD  ERRNO,T24BOT,EXPREC     ;TAPE NOT AT BOT AFTER REWIND
     034102  104456                                                          TRAP    C$ERHRD
     034104  000651                                                          .WORD   425
     034106  041553                                                          .WORD   T24BOT
     034110  016344                                                          .WORD   EXPREC
5706 034112                    140$: CKLOOP                          ;LOOP IF SELECTED
     034112  104406                                                          TRAP    C$CLP1
5707 034114  012703  000400          MOV     #256.,R3                ;RECORD SIZE
5708 034120  013737  003072  040702  MOV     FREE,T24RB              ;STARTING READ BUFFER ADDRESS
5709
5710                           ;*************************************************************
5711                           ;
5712                           ;READ DATA,ACK,CVC+1 COMMAND
5713                           ;
5714                           ;*************************************************************
5715
5716 034126  012737  140001  040700  MOV     #140001,T24PK3          ;READ DATA,ACK,CVC+1 COMMAND
5717 034134  012704  040700          1[illegible]$:  MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
```

```
5718 034140  010337  040706                MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
5719 034144  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5720 034150  004737  017120                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5721 034154  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5722 034160  012702  100204                MOV     #SSR!SC!BIT2,R2         ;SET UP EXPECTED
5723 034164  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
5724 034166  001406                        BEQ     170$                    ;BR, IF OK
5725 034170  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5729 034174                                ERRHRD  ERRNO,T24TRL,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
     034174  104456                                                                TRAP    C$ERHRD
     034176  000652                                                                .WORD   426
     034200  043104                                                                .WORD   T24TRL
     034202  011700                                                                .WORD   PKTSSR
5730 034204                        170$:   CKLOOP                          ;LOOP IF SELECTED
     034204  104406                                                                TRAP    C$CLP1
5731
5732                               ;*****************************************************************
5733                               ;
5734                               ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
5735                               ;
5736                               ;*****************************************************************
5737
5738 034206  013701  040576                MOV     T24BFR+6,R1             ;GET MESSAGE BUFFER
5739 034212  010102                        MOV     R1,R2                   ;SET UP EXPECTED
5740 034214  052702  010000                BIS     #BIT12,R2               ;SET THE RLL BIT IN EXPECTED
5741 034220  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
5742 034222  001406                        BEQ     180$                    ;BR, IF EQUAL (ALL IS WELL)
5743 034224  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5747 034230                                ERRHRD  ERRNO,T24LON,EXPREC     ;THE RLL BIT WAS NOT SET IN XST0
     034230  104456                                                                TRAP    C$ERHRD
     034232  000653                                                                .WORD   427
     034234  042652                                                                .WORD   T24LON
     034236  016344                                                                .WORD   EXPREC
5748 034240                        180$:
5749 034240                                ENDSUB                          ;>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>>
     034240                                                                     L10052:
     034240  104403                                                                TRAP    C$ESUB
5750 034242  023727  002170  000031        CMP     FATFLG,#25.                     ;IS ERROR COUNT AT 25
5751 034250  002402                        BLT     999$                            ;BR, IF LESS THAN 25
5752 034252  004737  020154                JSR     PC,CKDROP                       ;TRY TO DROP THE UNIT
5753 034256                        999$:
```

```
5755                                    ;+
5756                                    ;
5757                                    ;TEST 4, SUBTEST 4
5758                                    ;
5759                                    ;VERIFIES THAT A READ FORWARD COMMAND READING A RECORD
5760                                    ;SHORTER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE
5761                                    ;STATUS ALERT TERMINATION WITH THE RECORD LENGTH SHORT
5762                                    ;(RLS) BITS SET. IT IS VERIFIED THAT THE RESIDUAL BYTE
5763                                    ;COUNT (RBPCR) IN THE MESSAGE BUFFER CONTAINS THE
5764                                    ;PROPER NONZERO VALUE (E.G. THE DIFFERENCE BETWEEN
5765                                    ;THE ORIGINAL BYTE COUNT AND THE ACTUAL RECORD
5766                                    ;LENGTH).
5767                                    ;
5768                                    ;
5769                                    ;
5770                                    ;-
5771
5772 034256                                     BGNSUB                          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
      034256                                                                                    T4.4:
      034256  104402                                                                              TRAP    C$BSUB
5773 034260  004737  043374                     JSR     PC,T24RT3               ;SET UP OTHER COMMAND PACKET
5774 034264  004737  043240                     JSR     PC,T24REST              ;SET COMMAND PACKET
5775 034270  004737  043332                     JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET
5776
5777                                    ;*************************************************************************
5778                                    ;
5779                                    ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
5780                                    ;
5781                                    ;*************************************************************************
5782
5783 034274  004737  016644                     JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
5784 034300  103407                             BCS     20$                     ;BR IF INIT WAS OK
5785 034302  004737  020102                     JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5789 034306  010001                             MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
5790 034310                                     ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      034310  104455                                                                           TRAP    C$ERDF
      034312  000654                                                                          .WORD   428
      034314  003550                                                                          .WORD   SFIERR
      034316  011666                                                                          .WORD   SFIMSG
5791 034320                                     20$:
5792 034320  012704  040550                     MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
5793
5794                                    ;*************************************************************************
5795                                    ;
5796                                    ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
5797                                    ;
5798                                    ;*************************************************************************
5799
5800 034324  004737  010332                     JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
5801 034330  103407                             BCS     24$                     ;BR, IF COMMAND ISSUED OK
5802 034332  004737  020102                     JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5806 034336  010001                             MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
5807 034340                                     ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      034340  104456                                                                           TRAP    C$ERHRD
      034342  000655                                                                          .WORD   429
      034344  004754                                                                          .WORD   WRTMSG
      034346  011666                                                                          .WORD   SFIMSG
```

```
5808 034350                          24$:    CKLOOP                          ;LOOP IF SELECTED
     034350  104406                                                                  TRAP    C$CLP1
5809
5810                                 ;*****************************************************************
5811                                 ;
5812                                 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5813                                 ;
5814                                 ;*****************************************************************
5815
5816 034352  004737  010434                  JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
5817 034356  103407                          BCS     30$                     ;BR, IF NO PROBLEM
5818 034360  010001                          MOV     R0,R1                   ;SAVE TSSR
5819 034362  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5823 034366                                  ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
     034366  104456                                                                  TRAP    C$ERHRD
     034370  000656                                                                  .WORD   430
     034372  042036                                                                  .WORD   T24RWN
     034374  011700                                                                  .WORD   PKTSSR
5824 034376                          30$:    CKLOOP                          ;LOOP IF SELECTED
     034376  104406                                                                  TRAP    C$CLP1
5825 034400  012703  000400                  MOV     #256.,R3                ;RECORD SIZE
5826 034404  013737  003072  040702          MOV     FREE,T24RB              ;STARTING WRITE BUFFER ADDRESS
5827
5828                                 ;*****************************************************************
5829                                 ;
5830                                 ;WRITE DATA,ACK,CVC-1 COMMAND
5831                                 ;
5832                                 ;*****************************************************************
5833
5834 034412  012737  140005  040700          MOV     #140005,T24PK3          ;WRITE DATA,ACK,CVC-1 COMMAND
5835 034420  012704  040700                  MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5836 034424                          65$:
5837 034424  010337  040706                  MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
5838 034430  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5839 034434  004737  017120                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5840 034440  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5841 034444  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
5842 034450  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
5843 034452  001406                          BEQ     75$                     ;BR, IF OK
5844 034454  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5848                                                                         ;SOFT ERROR, REALLY CHECKING THE
5849                                                                         ;READ DATA COMMAND
5850 034460                                  ERRSOFT ERRNO,WRTERR,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     034460  104457                                                                  TRAP    C$ERSOFT
     034462  000657                                                                  .WORD   431
     034464  005011                                                                  .WORD   WRTERR
     034466  011700                                                                  .WORD   PKTSSR
5851 034470                          75$:    CKLOOP                          ;LOOP IF SELECTED
     034470  104406                                                                  TRAP    C$CLP1
5852 034472                          120$:
5853
5854                                 ;*****************************************************************
5855                                 ;
5856                                 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5857                                 ;
5858                                 ;*****************************************************************
5859
```

```
5860 034472  004737  010434                JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
5861 034476  103407                        BCS     130$                    ;BR, IF NO PROBLEM
5862 034500  010001                        MOV     R0,R1                   ;SAVE TSSR
5863 034502  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5867 034506                                ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
     034506  104456                                                                TRAP    C$ERHRD
     034510  000660                                                                .WORD   432
     034512  042036                                                                .WORD   T24RWN
     034514  011700                                                                .WORD   PKTSSR
5868 034516                        130$:   CKLOOP                          ;LOOP IF SELECTED
     034516  104406                                                                TRAP    C$CLP1
5869 034520  012703  001000                MOV     #512.,R3                ;RECORD SIZE
5870 034524  013737  003072  040702        MOV     FREE,T24RB              ;STARTING READ BUFFER ADDRESS
5871
5872                               ;*****************************************************************
5873                               ;
5874                               ;READ DATA,ACK,CVC=1 COMMAND
5875                               ;
5876                               ;*****************************************************************
5877
5878 034532  012737  140001  040700        MOV     #140001,T24PK3          ;READ DATA,ACK,CVC=1 COMMAND
5879 034540  012704  040700        165$:   MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
5880 034544  010337  040706                MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
5881 034550  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
5882 034554  004737  017120                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
5883 034560  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
5884 034564  012702  100204                MOV     #SSR!SC!BIT2,R2         ;SET UP EXPECTED
5885 034570  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
5886 034572  001406                        BEQ     170$                    ;BR, IF OK
5887 034574  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5391 034600                                ERRHRD  ERRNO,T24TRL,EXPREC     ;TSSR INCORRECT AFTER READ DATA
     034600  104456                                                                TRAP    C$ERHRD
     034602  000661                                                                .WORD   433
     034604  043104                                                                .WORD   T24TRL
     034606  016344                                                                .WORD   EXPREC
5892 034610                        170$:   CKLOOP                          ;LOOP IF SELECTED
     034610  104406                                                                TRAP    C$CLP1
5893
5894                               ;*****************************************************************
5895                               ;
5896                               ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
5897                               ;
5898                               ;*****************************************************************
5899
5900 034612  013701  040576                MOV     T24BFR+6,R1             ;GET MESSAGE BUFFER
5901 034616  010102                        MOV     R1,R2                   ;SET UP EXPECTED
5902 034620  052702  040000                BIS     #BIT14,R2               ;SET THE RLS BIT IN EXPECTED
5903 034624  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
5904 034626  001406                        BEQ     180$                    ;OR, IF EQUAL (ALL IS WELL)
5905 034630  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5909 034634                                ERRHRD  ERRNO,T24LOP,EXPREC     ;THE RLL BIT WAS NOT SET IN XST0
     034634  104456                                                                TRAP    C$ERHRD
     034636  000662                                                                .WORD   434
     034640  042734                                                                .WORD   T24LOP
     034642  016344                                                                .WORD   EXPREC
5910 034644                        180$:
5911 034644  013701  040574                MOV     T24BFR+4,R1             ;PICK UP RESIDUAL BYTE COUNTER
```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```
5912 034650  012702  000400                MOV     #256.,R2              ;THIS SHOULD BE THE DIFFERENCE
5913 034654  020102                        CMP     R1,R2                 ;IS THE DIFFERENCE CORRECT
5914 034656  001406                        BEQ     190$                  ;BR, IF CORRECT
5915 034660  004737  020102                JSR     PC,FATCHK             ;INC AND CHECK FOR MORE THAN 25 ERRORS
5919 034664                                ERRHRD  ERRNO,T24PBP,EXPREC   ;RBPCR NOT CORRECT
     034664  104456                                                              TRAP    C$ERHRD
     034666  000663                                                              .WORD   435
     034670  043016                                                              .WORD   T24PBP
     034672  016344                                                              .WORD   EXPREC
5920 034674                        190$:   CKLOOP                        ;LOOP IF SELECTED
     034674  104406                                                              TRAP    C$CLP1
5921 034676                                ENDSUB                        ;>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>>
     034676                                                                  L10053:
     034676  104403                                                              TRAP    C$ESUB
5922 034700  023727  002170  000031        CMP     FATFLG,#25.                   ;IS ERROR COUNT AT 25
5923 034706  002402                        BLT     999$                          ;BR, IF LESS THAN 25
5924 034710  004737  020154                JSR     PC,CKDROP                     ;TRY TO DROP THE UNIT
5925 034714                        999$:
```

```
                                ;*
                                ;
                                ;TEST 4, SUBTEST 5
                                ;
                                ;VERIFIES THAT READ REVERSE COMMANDS WITH SWB=0
                                ;OPERATES PROPERLY. THE TAPE IS FIRST REWOUND AND THEN
                                ;WRITTEN WITH A SERIES OF TEST RECORDS VARING IN
                                ;LENGTH AND DATA CONTENT. THE TAPE IS THEN READ IN REVERSE
                                ;SEQUENTIALLY AND THE RESULTS
                                ;(STATUS, DATA, ETC.) VERIFIED. THE BYTE COUNT ON
                                ;EACH READ REVERSE COMMAND IS SET TO THE LENGTH OF THE
                                ;EXPECTED RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD
                                ;OCCUR.
                                ;
                                ;
                                ;-
034714                                  BGNSUB                          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
034714                                                                                      T4.5:
034714  104402                                                                              TRAP    C$BSUB
034716  004737  043374                  JSR     PC,T24RT3               ;SET UP OTHER COMMAND PACKET
034722  004737  043240                  JSR     PC,T24REST              ;SET COMMAND PACKET
034726  004737  043332                  JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET

                                ;*************************************************************************
                                ;
                                ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                ;
                                ;*************************************************************************

034732  004737  016644                  JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
034736  103407                          BCS     20$                     ;BR IF INIT WAS OK
034740  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
034744  010001                          MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
034746                                  ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
034746  104455                                                                              TRAP    C$ERDF
034750  000664                                                                              .WORD   436
034752  003550                                                                              .WORD   SFIERR
034754  011666                                                                              .WORD   SFIMSG
034756                          20$:
034756  012704  040550                  MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS

                                ;*************************************************************************
                                ;
                                ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
                                ;
                                ;*************************************************************************

034762  004737  010332                  JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
034766  103407                          BCS     24$                     ;BR, IF COMMAND ISSUED OK
034770  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
034774  010001                          MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
034776                                  ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
034776  104456                                                                              TRAP    C$ERHRD
035000  000665                                                                              .WORD   437
035002  004754                                                                              .WORD   WRTMSG
035004  011666                                                                              .WORD   SFIMSG
```

```
5980 035006                          24$:    CKLOOP                          ;LOOP IF SELECTED
     035006  104406                                                                  TRAP    C$CLP1
5981
5982                                 ;*************************************************************************
5983                                 ;
5984                                 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5985                                 ;
5986                                 ;*************************************************************************
5987
5988 035010  004737  010434                  JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
5989 035014  103407                          BCS     30$                     ;BR, IF NO PROBLEM
5990 035016  010001                          MOV     R0,R1                   ;SAVE TSSR
5991 035020  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
5995 035024                                  ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
     035024  104456                                                                  TRAP    C$ERHRD
     035026  000666                                                                  .WORD   438
     035030  042036                                                                  .WORD   T24RWN
     035032  011700                                                                  .WORD   PKTSSR
5996 035034                          30$:    CKLOOP                          ;LOOP IF SELECTED
     035034  104406                                                                  TRAP    C$CLP1
5997 035036  012703  000400                  MOV     #256.,R3                ;RECORD SIZE
5998 035042  013737  003072  040702          MOV     FREE,T24RB              ;STARTING WRITE BUFFER ADDRESS
5999
6000                                 ;*************************************************************************
6001                                 ;
6002                                 ;WRITE DATA,ACK,CVC-1 COMMAND
6003                                 ;
6004                                 ;*************************************************************************
6005
6006 035050  012737  140005  040700          MOV     #140005,T24PK3          ;WRITE DATA,ACK,CVC-1 COMMAND
6007 035056  012704  040700                  MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6008 035062                          65$:
6009 035062  010300                          MOV     R3,R0                   ;SET PATTERN IN CORRECT REGISTER
6010 035064  004737  020374                  JSR     PC,FILLMEM              ;FILL MEMORY WITH RECORD SIZE
6011 035070  010337  040706                  MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
6012 035074  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6013 035100  004737  017120                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6014 035104  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6015 035110  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
6016 035114  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
6017 035116  001406                          BEQ     75$                     ;BR, IF OK
6018 035120  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6022                                                                         ;SOFT ERROR, REALLY CHECKING THE
6023                                                                         ;READ DATA COMMAND
6024 035124                                  ERRSOFT ERRNO,WRTERR,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     035124  104457                                                                  TRAP    C$ERSOFT
     035126  000667                                                                  .WORD   439
     035130  005011                                                                  .WORD   WRTERR
     035132  011700                                                                  .WORD   PKTSSR
6025 035134                          75$:    CKLOOP                          ;LOOP IF SELECTED
     035134  104406                                                                  TRAP    C$CLP1
6026 035136  005723                          TST     (R3)+                   ;BUMP RECORD SIZE
6027 035140  022703  000414                  CMP     #268.,R3                ;END OF RECORD YET
6028 035144  001346                          BNE     65$                     ;BR, IF MORE RECORDS TO WRITE
6029 035146                          80$:    CKLOOP                          ;LOOP IF SELECTED
     035146  104406                                                                  TRAP    C$CLP1
6030 035150  005743                          TST     -(R3)                   ;SET BACK TO 512.
```

```
 6031 035152  013737  003072  040702          MOV     FREE,T24RB              ;STARTING READ BUFFER ADDRESS
 6032
 6033                                 ;*************************************************************************
 6034                                 ;
 6035                                 ;READ REVERSE DATA,ACK COMMAND
 6036                                 ;
 6037                                 ;*************************************************************************
 6038
 6039 035160  012737  100401  040700          MOV     #100401,T24PK3          ;READ REVERSE DATA,ACK COMMAND
 6040 035166  012704  040700        165$:     MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
 6041 035172  010337  040706                  MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
 6042 035176  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
 6043 035202  004737  017120                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
 6044 035206  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
 6045 035212  012702  000200                  MOV     #SSR,R2                 ;SET UP EXPECTED
 6046 035216  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
 6047 035220  001406                          BEQ     170$                    ;BR, IF OK
 6048 035222  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 6052 035226                                  ERRHRD  ERRNO,T24WDC,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
      035226  104456                                                                      TRAP    C$ERHRD
      035230  000670                                                                      .WORD   440
      035232  042366                                                                      .WORD   T24WDC
      035234  011700                                                                      .WORD   PKTSSR
 6053 035236                        170$:     CKLOOP                          ;LOOP IF SELECTED
      035236  104406                                                                      TRAP    C$CLP1
 6054 035240  013702  003072                  MOV     FREE,R2                 ;GET BUFFER ADDRESS
 6055 035244  010304                          MOV     R3,R4                   ;CURRENT RECORD SIZE
 6056 035246  162704  000400                  SUB     #256.,R4                ;FIRST LOCATION IN BUFFER
 6057 035252  060204                173$:     ADD     R2,R4                   ;SET POINTER TO FRAME (WORD)
 6058 035254  021403                          CMP     (R4),R3                 ;CHECK DATA READ (R3=DATA ALSO)
 6059 035256  001410                          BEQ     180$                    ;BR, IF ALL IS WELL
 6060 035260  011401                          MOV     (R4),R1                 ;RECD DATA
 6061 035262  010302                          MOV     R3,R2                   ;EXPECTED DATA
 6062 035264  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
 6066 035270                                  ERRHRD  ERRNO,T24DTA,EXPREC     ;DATA READ NOT = WRITTEN
      035270  104456                                                                      TRAP    C$ERHRD
      035272  000671                                                                      .WORD   441
      035274  041620                                                                      .WORD   T24DTA
      035276  016344                                                                      .WORD   EXPREC
 6067 035300                        180$:     CKLOOP                          ;LOOP IF SELECTED
      035300  104406                                                                      TRAP    C$CLP1
 6068 035302  005724                          TST     (R4)+                   ;BUMP TO NEXT LOCATION
 6069 035304  160204                          SUB     R2,R4                   ;GET RID OF BASE ADDRESS
 6070 035306  020403                          CMP     R4,R3                   ;END OF RECORD YET
 6071 035310  001360                          BNE     173$                    ;BR, IF NOT AT END OF RECORD
 6072 035312  005743                          TST     -(R3)                   ;BUMP RECORD SIZE
 6073 035314  022703  000400                  CMP     #256.,R3                ;END OF RECORD YET
 6074 035320  001322                          BNE     165$                    ;BR, IF MORE RECORDS TO WRITE
 6075 035322                        190$:     CKLOOP                          ;LOOP IF SELECTED
      035322  104406                                                                      TRAP    C$CLP1
 6076 035324                                  ENDSUB                          ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
      035324                                                                          L10054:
      035324  104403                                                                      TRAP    C$ESUB
 6077 035326  023727  002170  000031                  CMP     FATFLG,#25.             ;IS ERROR COUNT AT 25
 6078 035334  002402                                  BLT     999$                    ;BR, IF LESS THAN 25
 6079 035336  004737  020154                          JSR     PC,CKDROP               ;TRY TO DROP THE UNIT
 6080 035342                        999$:
```

```
                                ;*
                                ;
                                ;TEST 4, SUBTEST 6
                                ;
                                ;VERIFIES THAT READ DATA COMMANDS WITH CVC=1 AND THE
                                ;SWAP BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST
                                ;SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2.
                                ;THE RESULTS, EXCEPT FOR RAM CONTENTS, SHOULD BE THE SAME.
                                ;
                                ;
                                ;-
035342                                  BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
035342                                                                                  T4.6:
035342  104402                                                                          TRAP    C$BSUB
035344  004737  043374                    JSR     PC,T24RT3               ;SET UP OTHER COMMAND PACKET
035350  004737  043240                    JSR     PC,T24REST              ;SET COMMAND PACKET
035354  004737  043332                    JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET

                                ;*****************************************************************
                                ;
                                ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                ;
                                ;*****************************************************************

035360  004737  016644                    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
035364  103407                    BCS     20$                     ;BR IF INIT WAS OK
035366  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
035372  010001                    MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
035374                                  ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
035374  104455                                                                          TRAP    C$ERDF
035376  000672                                                                          .WORD   442
035400  003550                                                                          .WORD   SFIERR
035402  011666                                                                          .WORD   SFIMSG
035404                          20$:
035404  012704  040550                    MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS

                                ;*****************************************************************
                                ;
                                ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
                                ;
                                ;*****************************************************************

035410  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
035414  103407                    BCS     24$                     ;BR. IF COMMAND ISSUED OK
035416  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
035422  010001                    MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
035424                                  ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
035424  104456                                                                          TRAP    C$ERHRD
035426  000673                                                                          .WORD   443
035430  004754                                                                          .WORD   WRTMSG
035432  011666                                                                          .WORD   SFIMSG
035434                          24$:    CKLOOP                          ;LOOP IF SELECTED
035434  104406                                                                          TRAP    C$CLP1

                                ;*****************************************************************
                                ;
```

```
6134                                    ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6135                                    ;
6136                                    ;*****************************************************************
6137
6138 035436  004737  010434             JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
6139 035442  103407                     BCS     30$                     ;BR, IF NO PROBLEM
6140 035444  010001                     MOV     R0,R1                   ;SAVE TSSR
6141 035446  004737  020102             JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6145 035452                             ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
     035452  104456                                                                    TRAP    C$ERHRD
     035454  000674                                                                    .WORD   444
     035456  042036                                                                    .WORD   T24RWN
     035460  011700                                                                    .WORD   PKTSSR
6146 035462                     30$:    CKLOOP                          ;LOOP IF SELECTED
     035462  104406                                                                    TRAP    C$CLP1
6147 035464  012703  000400             MOV     #256.,R3                ;RECORD SIZE
6148 035470  013737  003072  040702     MOV     FREE,T24RB              ;STARTING WRITE BUFFER ADDRESS
6149
6150                                    ;*****************************************************************
6151                                    ;
6152                                    ;WRITE DATA,ACK,CVC=1,SWB COMMAND
6153                                    ;
6154                                    ;*****************************************************************
6155
6156 035476  012737  150005  040700     MOV     #150005,T24PK3          ;WRITE DATA,ACK,CVC=1,SWB COMMAND
6157 035504  012704  040700             MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6158 035510                     65$:
6159 035510  010300                     MOV     R3,R0                   ;SET PATTERN IN CORRECT REGISTER
6160 035512  004737  020374             JSR     PC,FILLMEM              ;FILL MEMORY WITH RECORD SIZE
6161 035516  010337  040706             MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
6162 035522  010465  177776             MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6163 035526  004737  017120             JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6164 035532  016501  000000             MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6165 035536  012702  000200             MOV     #SSR,R2                 ;SET UP EXPECTED
6166 035542  020102                     CMP     R1,R2                   ;ARE THEY EQUAL
6167 035544  001406                     BEQ     75$                     ;BR, IF OK
6168 035546  004737  020102             JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6172                                                                    ;SOFT ERROR, REALLY CHECKING THE
6173                                                                    ;READ DATA COMMAND
6174 035552                             ERRSOFT ERRNO,WRTERR,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     035552  104457                                                                    TRAP    C$ERSOFT
     035554  000675                                                                    .WORD   445
     035556  005011                                                                    .WORD   WRTERR
     035560  011700                                                                    .WORD   PKTSSR
6175 035562                     75$:    CKLOOP                          ;LOOP IF SELECTED
     035562  104406                                                                    TRAP    C$CLP1
6176 035564  005723                     TST     (R3)+                   ;BUMP RECORD SIZE
6177 035566  022703  000414             CMP     #268.,R3                ;END OF RECORD YET
6178 035572  001346                     BNE     65$                     ;BR, IF MORE RECORDS TO WRITE
6179 035574                     80$:    CKLOOP                          ;LOOP IF SELECTED
     035574  104406                                                                    TRAP    C$CLP1
6180 035576  005743                     TST     -(R3)                   ;SET RECORD SIZE BACK TO 512.
6181 035600  013737  003072  040702     MOV     FREE,T24RB              ;STARTING READ BUFFER ADDRESS
6182
6183                                    ;*****************************************************************
6184                                    ;
6185                                    ;READ REVERSE DATA,ACK,SWB COMMAND
```

```
6186                                        ;
6187                                        ;*************************************************************************
6188
6189 035606  012737  110401  040700        MOV     #110401,T24PK3          ;READ REVERSE DATA,ACK,SWB COMMAND
6190 035614  012704  040700        165$:    MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6191 035620  010337  040706                 MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
6192 035624  010465  177776                 MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6193 035630  004737  017120                 JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6194 035634  016501  000000                 MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6195 035640  012702  000200                 MOV     #SSR,R2                 ;SET UP EXPECTED
6196 035644  020102                         CMP     R1,R2                   ;ARE THEY EQUAL
6197 035646  001406                         BEQ     170$                    ;BR, IF OK
6198 035650  004737  020102                 JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6202 035654                                 ERRHRD  ERRNO,T24WDC,EXPREC     ;TSSR INCORRECT AFTER READ DATA
     035654  104456                                                                TRAP    C$ERHRD
     035656  000676                                                                .WORD   446
     035660  042366                                                                .WORD   T24WDC
     035662  016344                                                                .WORD   EXPREC
6203 035664                         170$:   CKLOOP                          ;LOOP IF SELECTED
     035664  104406                                                                TRAP    C$CLP1
6204 035666  013702  003072                 MOV     FREE,R2                 ;GET BUFFER ADDRESS
6205 035672  010304                         MOV     R3,R4                   ;CURRENT RECORD SIZE
6206 035674  162704  000400                 SUB     #256.,R4                ;FIRST LOCATION IN BUFFER
6207 035700  060204                 173$:   ADD     R2,R4                   ;SET POINTER TO FRAME (WORD)
6208 035702  021403                         CMP     (R4),R3                 ;CHECK DATA READ (R3=DATA ALSO)
6209 035704  001410                         BEQ     180$                    ;BR, IF ALL IS WELL
6210 035706  011401                         MOV     (R4),R1                 ;RECD DATA
6211 035710  010302                         MOV     R3,R2                   ;EXPECTED DATA
6212 035712  004737  020102                 JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6216 035716                                 ERRHRD  ERRNO,T24DTA,EXPREC     ;DATA READ NOT = WRITTEN
     035716  104456                                                                TRAP    C$ERHRD
     035720  000677                                                                .WORD   447
     035722  041620                                                                .WORD   T24DTA
     035724  016344                                                                .WORD   EXPREC
6217 035726                         180$:   CKLOOP                          ;LOOP IF SELECTED
     035726  104406                                                                TRAP    C$CLP1
6218 035730  005724                         TST     (R4)+                   ;BUMP TO NEXT LOCATION
6219 035732  160204                         SUB     R2,R4                   ;GET RID OF BASE ADDRESS
6220 035734  020403                         CMP     R4,R3                   ;END OF RECORD YET
6221 035736  001360                         BNE     173$                    ;BR, IF NOT AT END OF RECORD
6222 035740  005743                         TST     -(R3)                   ;BUMP RECORD SIZE
6223 035742  022703  000400                 CMP     #256.,R3                ;END OF RECORD YET
6224 035746  001322                         BNE     165$                    ;BR, IF MORE RECORDS TO WRITE
6225 035750                         190$:   CKLOOP                          ;LOOP IF SELECTED
     035750  104406                                                                TRAP    C$CLP1
6226 035752                                 ENDSUB                          ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
     035752                                                                     L10055:
     035752  104403                                                                TRAP    C$ESUB
6227 035754  023727  002170  000031         CMP     FATFLG,#25.                     ;IS ERROR COUNT AT 25
6228 035762  002402                         BLT     999$                            ;BR, IF LESS THAN 25
6229 035764  004737  020154                 JSR     PC,CKDROP                       ;TRY TO DROP THE UNIT
6230 035770                         999$:
```

```
6232                                    ;+
6233                                    ;
6234                                    ;TEST 4, SUBTEST 7
6235                                    ;
6236                                    ;VERIFIES THAT A READ REVERSE COMMAND READING A RECORD
6237                                    ;LONGER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE
6238                                    ;STATUS ALERT TERMINATION WITH THE RECORD LENGTH LONG
6239                                    ;(RLL) BIT SET.
6240                                    ;
6241                                    ;
6242                                    ;-
6243
6244 035770                                    BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
    035770                                                                                  T4.7:
    035770  104402                                                                          TRAP    C$BSUB
6245 035772  004737  043374                    JSR     PC,T24RT3               ;SET UP OTHER COMMAND PACKET
6246 035776  004737  043240                    JSR     PC,T24REST              ;SET COMMAND PACKET
6247 036002  004737  043332                    JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET
6248
6249                                    ;*********************************************************************
6250                                    ;
6251                                    ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6252                                    ;
6253                                    ;*********************************************************************
6254
6255 036006  004737  016644                    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
6256 036012  103407                            BCS     20$                     ;BR IF INIT WAS OK
6257 036014  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6261 036020  010001                            MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
6262 036022                                    ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
    036022  104455                                                                          TRAP    C$ERDF
    036024  000700                                                                          .WORD   448
    036026  003550                                                                          .WORD   SFIERR
    036030  011666                                                                          .WORD   SFIMSG
6263 036032                                20$:
6264 036032  012704  040550                    MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
6265
6266                                    ;*********************************************************************
6267                                    ;
6268                                    ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6269                                    ;
6270                                    ;*********************************************************************
6271
6272 036036  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
6273 036042  103407                            BCS     24$                     ;BR, IF COMMAND ISSUED OK
6274 036044  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6278 036050  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
6279 036052                                    ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
    036052  104456                                                                          TRAP    C$ERHRD
    036054  000701                                                                          .WORD   449
    036056  004754                                                                          .WORD   WRTMSG
    036060  011666                                                                          .WORD   SFIMSG
6280 036062                                24$:    CKLOOP                          ;LOOP IF SELECTED
    036062  104406                                                                          TRAP    C$CLP1
6281
6282                                    ;*********************************************************************
6283                                    ;
```

```
6284                                    ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6285                                    ;
6286                                    ;*****************************************************************
6287
6288 036064  004737  010434             JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
6289 036070  004737  017236             JSR     PC,CHKTSSR              ;SEE HOW TSSR IS
6290 036074  103407                     BCS     30$                     ;BR, IF NO PROBLEM
6291 036076  010001                     MOV     R0,R1                   ;SAVE TSSR
6292 036100  004737  020102             JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6296 036104                             ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
     036104  104456                                                             TRAP    C$ERHRD
     036106  000702                                                             .WORD   450
     036110  042036                                                             .WORD   T24RWN
     036112  011700                                                             .WORD   PKTSSR
6297 036114                     30$:    CKLOOP                          ;LOOP IF SELECTED
     036114  104406                                                             TRAP    C$CLP1
6298 036116  012703  001000             MOV     #512.,R3                ;RECORD SIZE
6299 036122  013737  003072  040702     MOV     FREE,T24RB              ;STARTING WRITE BUFFER ADDRESS
6300
6301                                    ;*****************************************************************
6302                                    ;
6303                                    ;WRITE DATA,ACK,CVC=1 COMMAND
6304                                    ;
6305                                    ;*****************************************************************
6306
6307 036130  012737  140005  040700     MOV     #140005,T24PK3          ;WRITE DATA,ACK,CVC=1 COMMAND
6308 036136  012704  040700             MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6309 036142                     65$:
6310 036142  010337  040706             MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
6311 036146  010465  177776             MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6312 036152  004737  017120             JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6313 036156  016501  000000             MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6314 036162  012702  000200             MOV     #SSR,R2                 ;SET UP EXPECTED
6315 036166  020102                     CMP     R1,R2                   ;ARE THEY EQUAL
6316 036170  001406                     BEQ     75$                     ;BR, IF OK
6317 036172  004737  020102             JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6321                                                                    ;SOFT ERROR, REALLY CHECKING THE
6322                                                                    ;READ DATA COMMAND
6323 036176                             ERRSOFT ERRNO,WRTERR,PKTSSR     ;TSSR INCORRECT AFTER WRITE DATA
     036176  104457                                                             TRAP    C$ERSOFT
     036200  000703                                                             .WORD   451
     036202  005011                                                             .WORD   WRTERR
     036204  011700                                                             .WORD   PKTSSR
6324 036206                     75$:    CKLOOP                          ;LOOP IF SELECTED
     036206  104406                                                             TRAP    C$CLP1
6325 036210  012703  000400             MOV     #256.,R3                ;SIZE OF RECORD
6326 036214  013737  003072  040702     MOV     FREE,T24RB              ;STARTING READ BUFFER ADDRESS
6327
6328                                    ;*****************************************************************
6329                                    ;
6330                                    ;READ DATA,ACK COMMAND
6331                                    ;
6332                                    ;*****************************************************************
6333
6334 036222  012737  100401  040700     MOV     #100401,T24PK3          ;READ DATA,ACK COMMAND
6335 036230  012704  040700     165$:   MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6336 036234  010337  040706             MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
```

```
6337 036240  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6338 036244  004737  017120                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6339 036250  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6340 036254  012702  100204                MOV     #SSR!SC!BIT2,R2         ;SET UP EXPECTED
6341 036260  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
6342 036262  001406                        BEQ     170$                    ;BR, IF OK
6343 036264  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6347 036270                                ERRHRD  ERRNO,T24TRL,EXPREC     ;TSSR INCORRECT AFTER READ DATA
     036270  104456                                                                TRAP    C$ERHRD
     036272  000704                                                                .WORD   452
     036274  043104                                                                .WORD   T24TRL
     036276  016344                                                                .WORD   EXPREC
6348 036300                        170$:   CKLOOP                          ;LOOP IF SELECTED
     036300  104406                                                                TRAP    C$CLP1
6349
6350                               ;*****************************************************************
6351                               ;
6352                               ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
6353                               ;
6354                               ;*****************************************************************
6355
6356 036302  013701  040576                MOV     T24BFR+6,R1             ;GET MESSAGE BUFFER (XST0)
6357 036306  010102                        MOV     R1,R2                   ;SET UP EXPECTED
6358 036310  052702  010000                BIS     #BIT12,R2               ;SET THE RLL BIT IN EXPECTED
6359 036314  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
6360 036316  001406                        BEQ     180$                    ;BR, IF EQUAL (ALL IS WELL)
6361 036320  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6365 036324                                ERRHRD  ERRNO,T24LON,EXPREC     ;THE RLL BIT WAS NOT SET IN XST0
     036324  104456                                                                TRAP    C$ERHRD
     036326  000705                                                                .WORD   453
     036330  042652                                                                .WORD   T24LON
     036332  016344                                                                .WORD   EXPREC
6366 036334                        180$:
6367 036334                                ENDSUB                          ;>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>>
     036334                                                                     L10056:
     036334  104403                                                                TRAP    C$ESUB
6368 036336  023727  002170  000031            CMP     FATFLG,#25.             ;IS ERROR COUNT AT 25
6369 036344  002402                            BLT     999$                    ;BR, IF LESS THAN 25
6370 036346  004737  020154                    JSR     PC,CKDROP               ;TRY TO DROP THE UNIT
6371 036352                        999$:
6372                               ;*
```

```
6374                                    ;
6375                                    ;TEST 4, SUBTEST 8
6376                                    ;
6377                                    ;VERIFIES THAT A READ REVERSE COMMAND SPECIFYING A DATA
6378                                    ;BUFFER STARTING IN NONEXISTANT MEMORY TERMINATES WITH
6379                                    ;THE PROPER ERROR STATUS WITHOUT MOVING TAPE
6380                                    ;
6381                                    ;
6382                                    ;
6383                                    ;-
6384
6385 036352                                    BGNSUB                          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
      036352                                                                                    T4.8:
      036352  104402                                                                              TRAP    C$BSUB
6386 036354  012737  000000  040726            MOV     #0,T24DLY               ;ADDRESS BITS FOR LATER
6387 036362  005737  003100              9$:   TST     KTFLG                   ;CHECK FOR KT11
6388                                    ;       BEQ     10$                     ;BR. IF NO KT11
6389                                    ;
6390                                    ;       DIALOG NXM PATCH
6391                                    ;
6392 036366  000240                            NOP                             ;REMOVE WHEN NXM IS FIXED
6393 036370  000137  037010                    JMP     180$                    ;JUMP IF KT11
6394 036374  004737  043374            10$:   JSR     PC,T24RT3               ;SET UP OTHER COMMAND PACKET
6395 036400  004737  043240                    JSR     PC,T24REST              ;SET COMMAND PACKET
6396 036404  004737  043332                    JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET
6397
6398                                    ;*****************************************************************
6399                                    ;
6400                                    ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6401                                    ;
6402                                    ;*****************************************************************
6403
6404 036410  004737  016644                    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
6405 036414  103407                            BCS     20$                     ;OR IF INIT WAS OK
6406 036416  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6410 036422  010001                            MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
6411 036424                                    ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      036424  104455                                                                            TRAP    C$ERDF
      036426  000706                                                                            .WORD   454
      036430  003550                                                                            .WORD   SFIERR
      036432  011666                                                                            .WORD   SFIMSG
6412 036434                                20$:
6413 036434  012704  040550                    MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
6414
6415                                    ;*****************************************************************
6416                                    ;
6417                                    ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6418                                    ;
6419                                    ;*****************************************************************
6420
6421 036440  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
6422 036444  103407                            BCS     24$                     ;BR. IF COMMAND ISSUED OK
6423 036446  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6427 036452  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
6428 036454                                    ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      036454  104456                                                                            TRAP    C$ERHRD
      036456  000707                                                                            .WORD   455
```

```
      036460  004754                                                          .WORD   WRTMSG
      036462  011666                                                          .WORD   SFIMSG
 6429 036464                          24$:    CKLOOP                         ;LOOP IF SELECTED
      036464  104406                                                          TRAP    C$CLP1
 6430
 6431                                 ;*****************************************************************
 6432                                 ;
 6433                                 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
 6434                                 ;
 6435                                 ;*****************************************************************
 6436
 6437 036466  004737  010434                  JSR     PC,REWIND              ;CALL TAPE REWIND COMMAND
 6438 036472  004737  017236                  JSR     PC,CHKTSSR             ;SEE HOW TSSR IS
 6439 036476  103407                          BCS     30$                    ;BR, IF NO PROBLEM
 6440 036500  010001                          MOV     R0,R1                  ;SAVE TSSR
 6441 036502  004737  020102                  JSR     PC,FATCHK              ;INC AND CHECK FOR MORE THAN 25 ERRORS
 6445 036506                                  ERRHRD  ERRNO,T24RWN,PKTSSR    ;REWIND NOT ACCEPTED
      036506  104456                                                          TRAP    C$ERHRD
      036510  000710                                                          .WORD   456
      036512  042036                                                          .WORD   T24RWN
      036514  011700                                                          .WORD   PKTSSR
 6446 036516                          30$:    CKLOOP                         ;LOOP IF SELECTED
      036516  104406                                                          TRAP    C$CLP1
 6447 036520  012703  000400                  MOV     #256.,R3               ;RECORD SIZE
 6448 036524  013737  003072  040702          MOV     FREE,T24RB             ;STARTING WRITE BUFFER ADDRESS
 6449
 6450                                 ;*****************************************************************
 6451                                 ;
 6452                                 ;WRITE DATA,ACK,CVC=1 COMMAND
 6453                                 ;
 6454                                 ;*****************************************************************
 6455
 6456 036532  012737  140005  040700          MOV     #140005,T24PK3         ;WRITE DATA,ACK,CVC=1 COMMAND
 6457 036540  012704  040700                  MOV     #T24PK3,R4             ;SET UP R4 WITH PACKET ADDRESS
 6458 036544                          65$:
 6459 036544  010337  040706                  MOV     R3,T24SZ               ;SET UP RECORD SIZE IN PACKET
 6460 036550  010465  177776                  MOV     R4,TSDB(R5)            ;ISSUE COMMAND
 6461 036554  004737  017120                  JSR     PC,WAITF               ;WAIT FOR SSR TO SET
 6462 036560  016501  000000                  MOV     TSSR(R5),R1            ;GET TSSR CONTENTS
 6463 036564  012702  000200                  MOV     #SSR,R2                ;SET UP EXPECTED
 6464 036570  020102                          CMP     R1,R2                  ;ARE THEY EQUAL
 6465 036572  001406                          BEQ     75$                    ;BR, IF OK
 6466 036574  004737  020102                  JSR     PC,FATCHK              ;INC AND CHECK FOR MORE THAN 25 ERRORS
 6470 036600                                  ERRSOFT ERRNO,WRTERR,PKTSSR    ;TSSR INCORRECT AFTER WRITE DATA
      036600  104457                                                          TRAP    C$ERSOFT
      036602  000711                                                          .WORD   457
      036604  005011                                                          .WORD   WRTERR
      036606  011700                                                          .WORD   PKTSSR
 6471 036610                          75$:    CKLOOP                         ;LOOP IF SELECTED
      036610  104406                                                          TRAP    C$CLP1
 6472 036612  012703  000400                  MOV     #256.,R3               ;RECORD SIZE
 6473 036616  012701  160000                  MOV     #160000,R1             ;START POSSIBLE NXM ADDRESS
 6474 036622  012702  177776                  MOV     #177776,R2             ;END POSSIBLE NXM ADDRESS
 6475 036626  004737  017276                  JSR     PC,XNXM                ;CALL NXM FINDER ROUTINE
 6476 036632  103402                          BCS     76$                    ;BR IF NXM ADDRESS FOUND
 6477 036634  000137  037010                  JMP     180$                   ;JMP OVER CAN'T FIND NXM
 6478 036640  010137  040702          76$:    MOV     R1,T24RB               ;STARTING READ REVERSE BUFFER ADDRESS
```

```
6479
6480                                    ;*****************************************************************
6481                                    ;
6482                                    ;READ REVERSE DATA,ACK COMMAND
6483                                    ;
6484                                    ;*****************************************************************
6485
6486 036644  013737  040726  040704          MOV     T24DLY,T24RB+2          ;GET BITS 16 AND 17
6487 036652  012737  100401  040700          MOV     #100401,T24PK3          ;READ REVERSE DATA,ACK COMMAND
6488 036660  012704  040700              165$: MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6489 036664  012737  000400  040706          MOV     #256.,T24SZ             ;SET UP RECORD SIZE IN PACKET
6490 036672  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6491 036676  004737  017120                  JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6492 036702  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6493 036706  012702  104210                  MOV     #SSR!NXM!SC!BIT3,R2     ;SET UP EXPECTED
6494 036712  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
6495 036714  001417                          BEQ     170$                    ;BR, IF OK
6496 036716  062737  000001  040726          ADD     #1,T24DLY               ;NEXT BUNCH OF MEMORY
6497 036724  022737  000100  040726          CMP     #100,T24DLY             ;TOO MUCH MEMORY
6498 036732  001402                          BEQ     168$                    ;BR IF OVER
6499 036734  000137  036374                  JMP     10$                     ;TRY AGAIN
6500 036740  004737  020102              168$: JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6504 036744                                  ERRHRD  ERRNO,T24NXM,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
     036744  104456                                                                      TRAP    C$ERHRD
     036746  000712                                                                      .WORD   458
     036750  041141                                                                      .WORD   T24NXM
     036752  011700                                                                      .WORD   PKTSSR
6505 036754                              170$:   CKLOOP                          ;LOOP IF SELECTED
     036754  104406                                                                      TRAP    C$CLP1
6506
6507                                    ;*****************************************************************
6508                                    ;
6509                                    ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6510                                    ;
6511                                    ;*****************************************************************
6512
6513 036756  013701  040576                  MOV     T24BFR+6,R1             ;GET MESSAGE BUFFER
6514 036762  010102                          MOV     R1,R2                   ;SET UP EXPECTED
6515 036764  052702  040000                  BIS     #BIT14,R2               ;SET THE RLS BIT IN EXPECTED
6516 036770  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
6517 036772  001406                          BEQ     180$                    ;BR, IF EQUAL (ALL IS WELL)
6518 036774  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6522 037000                                  ERRHRD  ERRNO,T24LOP,EXPREC     ;THE RLL BIT WAS NOT SET IN XSTO
     037000  104456                                                                      TRAP    C$ERHRD
     037002  000713                                                                      .WORD   459
     037004  042734                                                                      .WORD   T24LOP
     037006  016344                                                                      .WORD   EXPREC
6523 037010                              180$:
6524 037010                                  ENDSUB                          ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
     037010                                                                          L10057:
     037010  104403                                                                      TRAP    C$ESUB
6525 037012  023727  002170  000031              CMP     FATFLG,#25.                     ;IS ERROR COUNT AT 25
6526 037020  002402                              BLT     999$                            ;BR, IF LESS THAN 25
6527 037022  004737  020154                      JSR     PC,CKDROP                       ;TRY TO DROP THE UNIT
6528 037026                              999$:
```

```
6530                                    ;*
6531                                    ;
6532                                    ;TEST 4, SUBTEST 9
6533                                    ;
6534                                    ;VERIFIES THAT ILLEGAL BUFFER ADDRESSES CAUSE A
6535                                    ;FUNCTION REJECT TERMINATION WITH ILLEGAL ADDRESS
6536                                    ;(ILA) ERROR BIT SET.
6537                                    ;
6538                                    ;
6539                                    ;
6540                                    ;-
6541
6542 037026                                    BGNSUB                          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
      037026                                                                                    T4.9:
      037026  104402                                                                              TRAP    C$BSUB
6543 037030  004737  043374                    JSR     PC,T24RT3               ;SET COMMAND PACKET UP CLEAR
6544 037034  004737  043240                    JSR     PC,T24REST              ;SET COMMAND PACKET
6545 037040  004737  043332                    JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET
6546
6547                                    ;*****************************************************************
6548                                    ;
6549                                    ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6550                                    ;
6551                                    ;*****************************************************************
6552
6553 037044  004737  016644                    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
6554 037050  103407                            BCS     20$                     ;BR IF INIT WAS OK
6555 037052  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6559 037056  010001                            MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
6560 037060                                    ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      037060  104455                                                                            TRAP    C$ERDF
      037062  000714                                                                            .WORD   460
      037064  003550                                                                            .WORD   SFIERR
      037066  011666                                                                            .WORD   SFIMSG
6561 037070                                20$:
6562 037070  012704  040550                    MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
6563
6564                                    ;*****************************************************************
6565                                    ;
6566                                    ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6567                                    ;
6568                                    ;*****************************************************************
6569
6570 037074  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
6571 037100  103407                            BCS     24$                     ;BR, IF COMMAND ISSUED OK
6572 037102  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6576 037106  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
6577 037110                                    ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      037110  104456                                                                            TRAP    C$ERHRD
      037112  000715                                                                            .WORD   461
      037114  004754                                                                            .WORD   WRTMSG
      037116  011666                                                                            .WORD   SFIMSG
6578 037120                                24$:    CKLOOP                          ;LOOP IF SELECTED
      037120  104406                                                                            TRAP    C$CLP1
6579 037122  013737  003072  040702              MOV     FREE,T24RB              ;ILLEGAL STARTING READ BUFFER ADDRESS
6580 037130  012737  177700  040704              MOV     #177700,T24RB+2         ;CREATE ILLEGAL ADDRESS
6581
```

```
6582                                    ;*****************************************************************
6583                                    ;
6584                                    ;LEGAL MODE,ACK,CVC=1,READ COMMAND
6585                                    ;
6586                                    ;*****************************************************************
6587
6588 037136  012737  140001  040700          MOV     #140001,T24PK3          ;LEGAL MODE,ACK,CVC=1,READ COMMAND
6589 037144  012704  040700                  MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6590 037150  012737  000400  040706          MOV     #256.,T24SZ             ;SET UP RECORD SIZE IN PACKET
6591 037156  010465  177776                  MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6592 037162  004737  017120                  JSR     PC,WAITF                ;WAIT FOR SSR!BIT1!BIT2 TO SET
6593 037166  016501  000000                  MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6594 037172  012702  100206                  MOV     #SSR!SC!BIT1!BIT2,R2    ;SET UP EXPECTED
6595 037176  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
6596 037200  001406                          BEQ     75$                     ;BR, IF OK
6597 037202  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6601 037206                                  ERRHRD  ERRNO,T24WDG,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
     037206  104456                                                                  TRAP    C$ERHRD
     037210  000716                                                                  .WORD   462
     037212  041052                                                                  .WORD   T24WDG
     037214  011700                                                                  .WORD   PKTSSR
6602 037216                          75$:    CKLOOP                          ;LOOP IF SELECTED
     037216  104406                                                                  TRAP    C$CLP1
6603
6604                                    ;*****************************************************************
6605                                    ;
6606                                    ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
6607                                    ;
6608                                    ;*****************************************************************
6609
6610 037220  013701  040576                  MOV     T24BFR+6,R1             ;GET MESSAGE BUFFER
6611 037224  010102                          MOV     R1,R2                   ;SET UP EXPECTED
6612 037226  052702  000400                  BIS     #BIT8,R2                ;SET THE ILA BIT IN EXPECTED
6613 037232  020102                          CMP     R1,R2                   ;ARE THEY EQUAL
6614 037234  001406                          BEQ     180$                    ;BR, IF EQUAL (ALL IS WELL)
6615 037236  004737  020102                  JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6619 037242                                  ERRHRD  ERRNO,T24ILA,EXPREC     ;THE ILA BIT WAS NOT SET IN XST0
     037242  104456                                                                  TRAP    C$ERHRD
     037244  000717                                                                  .WORD   463
     037246  041302                                                                  .WORD   T24ILA
     037250  016344                                                                  .WORD   EXPREC
6620 037252                          180$:
6621 037252                                  ENDSUB                          ;>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
     037252                                                                       L10060:
     037252  104403                                                                  TRAP    C$ESUB
6622 037254  023727  002170  000031                  CMP     FATFLG,#25.             ;IS ERROR COUNT AT 25
6623 037262  002402                          BLT     999$                    ;BR, IF LESS THAN 25
6624 037264  004737  020154                  JSR     PC,CKDROP               ;TRY TO DROP THE UNIT
6625 037270                          999$:
```

```
6627                                    ;*
6628                                    ;
6629                                    ;TEST 4, SUBTEST 10
6630                                    ;
6631                                    ;VERIFIES THAT A DATA BUFFER ADDRESS REFERENCING
6632                                    ;NONEXISTANT MEMORY CAUSES RECOVERABLE ERROR
6633                                    ;TERMINATION (TC=4), WITH THE NXM BIT SET IN THE TSSR, AND
6634                                    ;THAT THE TAPE IS ULTIMATELY POSITIONED PROPERLY.
6635                                    ;
6636                                    ;
6637                                    ;
6638                                    ;-
6639
6640 037270                                    BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
      037270                                                                                T4.10:
      037270  104402                                                                          TRAP    C$BSUB
6641 037272  012737  000000  040726            MOV     #0,T24DLY               ;SET BITS 16 AND 17
6642 037300  005737  003100          9$:       TST     KTFLG                   ;CHECK FOR KT11
6643                                    ;       BEQ     10$                     ;BR, IF NO KT11
6644                                    ;
6645                                    ;       DIALOG NXM PATCH
6646                                    ;
6647 037304  000240                            NOP                             ;REMOVE WHEN NXM IS FIXED
6648 037306  000137  037544                    JMP     80$                     ;JUMP IF KT11 (SKIP TEST)
6649 037312  004737  043374          10$:      JSR     PC,T24RT3               ;SET COMMAND PACKET UP CLEAR
6650 037316  004737  043240                    JSR     PC,T24REST              ;SEY COMMAND PACKET
6651 037322  004737  043332                    JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET
6652
6653                                    ;*********************************************************************
6654                                    ;
6655                                    ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6656                                    ;
6657                                    ;*********************************************************************
6658
6659 037326  004737  016644                    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
6660 037332  103407                            BCS     20$                     ;BR IF INIT WAS OK
6661 037334  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6665 037340  010001                            MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
6666 037342                                    ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
      037342  104455                                                                       TRAP    C$ERDF
      037344  000720                                                                       .WORD   464
      037346  003550                                                                       .WORD   SFIERR
      037350  011666                                                                       .WORD   SFIMSG
6667 037352                             20$:
6668 037352  012704  040550                    MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS
6669
6670                                    ;*********************************************************************
6671                                    ;
6672                                    ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6673                                    ;
6674                                    ;*********************************************************************
6675
6676 037356  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
6677 037362  103407                            BCS     24$                     ;BR, IF COMMAND ISSUED OK
6678 037364  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6682 037370  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
6683 037372                                    ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
```

```
        037372  104456                                                          TRAP    C$ERHRD
        037374  000721                                                          .WORD   465
        037376  004754                                                          .WORD   WRTMSG
        037400  011666                                                          .WORD   SFTMSG
  6684  037402                          24$:    CKLOOP                        ;LOOP IF SELECTED
        037402  104406                                                          TRAP    C$CLP1
  6685  037404  012701  160000                  MOV     #160000,R1            ;START POSSIBLE NXM ADDRESS
  6686  037410  012702  177776                  MOV     #177776,R2            ;END POSSIBLE NXM ADDRESS
  6687  037414  004737  017276                  JSR     PC,XNXM               ;CALL NXM FINDER ROUTINE
  6688  037420  103402                          BCS     76$                   ;BR IF WE FOUND NXM
  6689  037422  000137  037544                  JMP     180$                  ;JUMP OVER, CAN'T FIND NXM
  6690  037426  010137  040702          76$:    MOV     R1,T24RB              ;STARTING READ BUFFER ADDRESS
  6691                                  ;*************************************************************
  6692                                  ;
  6693                                  ;READ, ACK, CVC=1, COMMAND
  6694                                  ;
  6695                                  ;*************************************************************
  6696
  6697  037432  013737  040726  040704          MOV     T24DLY,T24RB+2        ;MEMORY BITS 16 AND 17
  6698  037440  012737  140001  040700          MOV     #140001,T24PK3        ;READ, ACK, CVC=1, COMMAND
  6699  037446  012704  040700                  MOV     #T24PK3,R4            ;SET UP R4 WITH PACKET ADDRESS
  6700  037452  012737  000400  040706          MOV     #256.,T24SZ           ;SET UP RECORD SIZE IN PACKET
  6701  037460  010465  177776                  MOV     R4,TSDB(R5)           ;ISSUE COMMAND
  6702  037464  004737  017120                  JSR     PC,WAITF              ;WAIT FOR SSR!BIT1!BIT2 TO SET
  6703  037470  016501  000000                  MOV     TSSR(R5),R1           ;GET TSSR CONTENTS
  6704  037474  012702  104210                  MOV     #SSR!SC!NXM!BIT3,R2   ;SET UP EXPECTED
  6705  037500  020102                          CMP     R1,R2                 ;ARE THEY EQUAL
  6706  037502  001417                          BEQ     75$                   ;BR, IF OK
  6707  037504  062737  000001  040726          ADD     #1,T24DLY             ;NEXT BUNCH OF MEMORY
  6708  037512  022737  000100  040726          CMP     #100,T24DLY           ;TOO MUCH MEMORY
  6709  037520  001402                          BEQ     50$                   ;END OF LINE
  6710  037522  000137  037312                  JMP     10$                   ;TRY AGAIN
  6711  037526  004737  020102          50$:    JSR     PC,FATCHK             ;INC AND CHECK FOR MORE THAN 25 ERRORS
  6715  037532                                  ERRHRD  ERRNO,T24NXM,PKTSSR   ;TSSR INCORRECT AFTER READ DATA
        037532  104456                                                                  TRAP    C$ERHRD
        037534  000722                                                                  .WORD   466
        037536  041141                                                                  .WORD   T24NXM
        037540  011700                                                                  .WORD   PKTSSR
  6716  037542                          75$:    CKLOOP                        ;LOOP IF SELECTED
        037542  104406                                                                  TRAP    C$CLP1
  6717  037544                          80$:
  6718  037544                          180$:   ENDSUB                        ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
        037544                                                                        L10061:
        037544  104403                                                                  TRAP    C$ESUB
  6719  037546  023727  002170  000031          CMP     FATFLG,#25.                   ;IS ERROR COUNT AT 25
  6720  037554  002402                          BLT     999$                          ;BR, IF LESS THAN 25
  6721  037556  004737  020154                  JSR     PC,CKDROP                     ;TRY TO DROP THE UNIT
  6722  037562                          999$:
```

```
                                ;+
                                ;
                                ;TEST 4, SUBTEST 11
                                ;
                                ;VERIFIES THAT A READ REVERSE COMMAND ISSUED WHILE THE
                                ;TAPE IS AT BOT RESULTS IN FUNCTION REJECT TERMINATION
                                ;WITH THE NONEXECUTABLE FUNCTION (NEF) ERROR BIT SET
                                ;
                                ;
                                ;
                                ;-

037562                                  BGNSUB                          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
037562                                                                                      T4.11:
037562  104402                                                                              TRAP    C$BSUB
037564  004737  043374                    JSR     PC,T24RT3               ;SET COMMAND PACKET UP CLEAR
037570  004737  043240                    JSR     PC,T24REST              ;SET COMMAND PACKET
037574  004737  043332                    JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET

                                ;*****************************************************************
                                ;
                                ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                ;
                                ;*****************************************************************

037600  004737  016644                    JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
037604  103407                            BCS     20$                     ;BR IF INIT WAS OK
037606  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
037612  010001                            MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
037614                                  ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
037614  104455                                                                              TRAP    C$ERDF
037616  000723                                                                              .WORD   467
037620  003550                                                                              .WORD   SFIERR
037622  011666                                                                              .WORD   SFIMSG
037624                          20$:
037624  012704  040550                    MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS

                                ;*****************************************************************
                                ;
                                ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
                                ;
                                ;*****************************************************************

037630  004737  010332                    JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
037634  103407                            BCS     24$                     ;BR, IF COMMAND ISSUED OK
037636  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
037642  010001                            MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
037644                                  ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
037644  104456                                                                              TRAP    C$ERHRD
037646  000724                                                                              .WORD   468
037650  004754                                                                              .WORD   WRTMSG
037652  011666                                                                              .WORD   SFIMSG
037654                          24$:    CKLOOP                          ;LOOP IF SELECTED
037654  104406                                                                              TRAP    C$CLP1

                                ;*****************************************************************
                                ;
```

```
6776                                    ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6777                                    ;
6778                                    ;*****************************************************************
6779
6780 037656  004737  010434             JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
6781 037662  004737  017236             JSR     PC,CHKTSSR              ;SEE HOW TSSR IS
6782 037666  103407                     BCS     30$                     ;BR, IF NO PROBLEM
6783 037670  010001                     MOV     R0,R1                   ;SAVE TSSR
6784 037672  004737  020102             JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6788 037676                             ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
      037676  104456                                                            TRAP    C$ERHRD
      037700  000725                                                            .WORD   469
      037702  042036                                                            .WORD   T24RWN
      037704  011700                                                            .WORD   PKTSSR
6789 037706                     30$:    CKLOOP                          ;LOOP IF SELECTED
      037706  104406                                                            TRAP    C$CLP1
6790 037710  012703  000400             MOV     #256.,R3                ;RECORD SIZE
6791 037714  013737  003072  040702     MOV     FREE,T24RB              ;STARTING WRITE BUFFER ADDRESS
6792
6793                                    ;*****************************************************************
6794                                    ;
6795                                    ;READ REVERSE DATA,ACK COMMAND
6796                                    ;
6797                                    ;*****************************************************************
6798
6799 037722  012737  100401  040700     MOV     #100401,T24PK3          ;READ REVERSE DATA,ACK COMMAND
6800 037730  012704  040700             MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6801 037734                     65$:
6802 037734  010337  040706             MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
6803 037740  010465  177776             MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6804 037744  004737  017120             JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6805 037750  016501  000000             MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6806 037754  012702  100206             MOV     #SSR!SC!BIT1!BIT2,R2    ;SET UP EXPECTED
6807 037760  020102                     CMP     R1,R2                   ;ARE THEY EQUAL
6808 037762  001406                     BEQ     75$                     ;BR, IF OK
6809 037764  004737  020102             JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6813 037770                             ERRHRD  ERRNO,T24WDE,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
      037770  104456                                                            TRAP    C$ERHRD
      037772  000726                                                            .WORD   470
      037774  041501                                                            .WORD   T24WDE
      037776  011700                                                            .WORD   PKTSSR
6814 040000                     75$:    CKLOOP                          ;LOOP IF SELECTED
      040000  104406                                                            TRAP    C$CLP1
6815
6816                                    ;*****************************************************************
6817                                    ;
6818                                    ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
6819                                    ;
6820                                    ;*****************************************************************
6821
6822 040002  013701  040576             MOV     T24BFR+6,R1             ;GET MESSAGE BUFFER
6823 040006  010102                     MOV     R1,R2                   ;SET UP EXPECTED
6824 040010  052702  002000             BIS     #BIT10,R2               ;SET THE NEF BIT IN EXPECTED
6825 040014  020102                     CMP     R1,R2                   ;ARE THEY EQUAL
6826 040016  001406                     BEQ     180$                    ;BR, IF EQUAL (ALL IS WELL)
6827 040020  004737  020102             JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6831 040024                             ERRHRD  ERRNO,T24NEF,EXPREC     ;THE RLL BIT WAS NOT SET IN XST0
```

J14


```
        040024  104456                                                                    TRAP    C$ERHRD
        040026  000727                                                                    .WORD   471
        040030  040730                                                                    .WORD   T24NEF
        040032  016344                                                                    .WORD   EXPREC
  6832  040034                                180$:
  6833  040034                                        ENDSUB                  ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
        040034                                                                          L10062:
        040034  104403                                                                  TRAP    C$ESUB
  6834  040036  023727  002170  000031                CMP     FATFLG,#25.             ;IS ERROR COUNT AT 25
  6835  040044  002402                                BLT     999$                    ;BR, IF LESS THAN 25
  6836  040046  004737  020154                        JSR     PC,CKDROP               ;TRY TO DROP THE UNIT
  6837  040052                                999$:
```

```
                                        ;+
                                        ;
                                        ;TEST 4, SUBTEST 12
                                        ;
                                        ;VERIFIES THAT A READ REVERSE COMMAND ISSUED WHILE THE
                                        ;TAPE IS POSITIONED BEFORE THE FIRST RECORD ON TAPE
                                        ;(BUT NOT AT BOT) RESULTS IN TAPE STATUS ALERT.
                                        ;
                                        ;
                                        ;
                                        ;-

040052                                          BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
040052                                                                                      T4.12:
040052  104402                                                                          TRAP    C$BSUB
040054  004737  043374                          JSR     PC,T24RT3               ;SET COMMAND PACKET UP CLEAR
040060  004737  043240                          JSR     PC,T24REST              ;SET COMMAND PACKET
040064  004737  043332                          JSR     PC,T24RT2               ;SET UP OTHER COMMAND PACKET

                                        ;*************************************************************************
                                        ;
                                        ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                        ;
                                        ;*************************************************************************

040070  004737  016644                          JSR     PC,SOFINIT              ;DO INITIALIZE ON CONTROLLER
040074  103407                                  BCS     20$                     ;BR IF INIT WAS OK
040076  004737  020102                          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
040102  010001                                  MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
040104                                          ERRDF   ERRNO,SFIERR,SFIMSG     ;FATAL ERROR TSSR WAS NOT OK
040104  104455                                                                          TRAP    C$ERDF
040106  000730                                                                          .WORD   472
040110  003550                                                                          .WORD   SFIERR
040112  011666                                                                          .WORD   SFIMSG
040114                                  20$:
040114  012704  040550                          MOV     #T24PACKET,R4           ;SUBROUTINE NEEDS PACKET ADDRESS

                                        ;*************************************************************************
                                        ;
                                        ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
                                        ;
                                        ;*************************************************************************

040120  004737  010332                          JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
040124  103407                                  BCS     24$                     ;BR, IF COMMAND ISSUED OK
040126  004737  020102                          JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
040132  010001                                  MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
040134                                          ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
040134  104456                                                                          TRAP    C$ERHRD
040136  000731                                                                          .WORD   473
040140  004754                                                                          .WORD   WRTMSG
040142  011666                                                                          .WORD   SFIMSG
040144                                  24$:    CKLOOP                          ;LOOP IF SELECTED
040144  104406                                                                          TRAP    C$CLP1

                                        ;*************************************************************************
                                        ;
```

```
6891                                    ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6892                                    ;
6893                                    ;*****************************************************************
6894
6895 040146  004737  010434                    JSR     PC,REWIND               ;CALL TAPE REWIND COMMAND
6896 040152  004737  017236                    JSR     PC,CHKTSSR              ;SEE HOW TSSR IS
6897 040156  103407                            BCS     30$                     ;BR, IF NO PROBLEM
6898 040160  010001                            MOV     R0,R1                   ;SAVE TSSR
6899 040162  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6903 040166                                    ERRHRD  ERRNO,T24RWN,PKTSSR     ;REWIND NOT ACCEPTED
      040166  104456                                                                    TRAP    C$ERHRD
      040170  000732                                                                    .WORD   474
      040172  042036                                                                    .WORD   T24RWN
      040174  011700                                                                    .WORD   PKTSSR
6904 040176                            30$:    CKLOOP                          ;LOOP IF SELECTED
      040176  104406                                                                    TRAP    C$CLP1
6905 040200  012703  000400                    MOV     #256.,R3                ;RECORD SIZE
6906 040204  013737  003072  040702            MOV     FREE,T24RB              ;STARTING WRITE BUFFER ADDRESS
6907
6908                                    ;*****************************************************************
6909                                    ;
6910                                    ;WRITE DATA,ACK,CVC=1 COMMAND
6911                                    ;
6912                                    ;*****************************************************************
6913
6914 040212  012737  140005  040700            MOV     #140005,T24PK3          ;WRITE DATA,ACK,CVC=1 COMMAND
6915 040220  012704  040700                    MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6916 040224                            65$:
6917 040224  010337  040706                    MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
6918 040230  010465  177776                    MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6919 040234  004737  017120                    JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6920 040240  016501  000000                    MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6921 040244  012702  000200                    MOV     #SSR,R2                 ;SET UP EXPECTED
6922 040250  020102                            CMP     R1,R2                   ;ARE THEY EQUAL
6923 040252  001406                            BEQ     75$                     ;BR, IF OK
6924 040254  004737  020102                    JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6928                                                                           ;SOFT ERROR, REALLY CHECKING THE
6929                                                                           ;READ REVERSE DATA COMMAND
6930 040260                                    ERRSOFT ERRNO,WRTERR,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
      040260  104457                                                                    TRAP    C$ERSOFT
      040262  000733                                                                    .WORD   475
      040264  005011                                                                    .WORD   WRTERR
      040266  011700                                                                    .WORD   PKTSSR
6931 040270                            75$:    CKLOOP                          ;LOOP IF SELECTED
      040270  104406                                                                    TRAP    C$CLP1
6932 040272  012703  000400                    MOV     #256.,R3                ;RECORD SIZE
6933 040276  013737  003072  040702            MOV     FREE,T24RB              ;STARTING READ BUFFER ADDRESS
6934
6935                                    ;*****************************************************************
6936                                    ;
6937                                    ;READ REVERSE DATA,ACK COMMAND
6938                                    ;
6939                                    ;*****************************************************************
6940
6941 040304  012737  100401  040700            MOV     #100401,T24PK3          ;READ REVERSE DATA,ACK COMMAND
6942 040312  012704  040700            165$:   MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6943 040316  010337  040706                    MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
```

```
6944 040322  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6945 040326  004737  017120                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6946 040332  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6947 040336  012702  0C0200                MOV     #SSR,R2                 ;SET UP EXPECTED
6948 040342  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
6949 040344  001406                        BEQ     170$                    ;BR, IF OK
6950 040346  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6954 040352                                ERRHRD  ERRNO,T24TRL,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
     040352  104456                                                                TRAP    C$ERHRD
     040354  000734                                                                .WORD   476
     040356  043104                                                                .WORD   T24TRL
     040360  011700                                                                .WORD   PKTSSR
6955 040362                        170$:   CKLOOP                          ;LOOP IF SELECTED
     040362  104406                                                                TRAP    C$CLP1
6956 040364  012703  000400                MOV     #256.,R3                ;RECORD SIZE
6957 040370  013737  003072  040702        MOV     FREE,T24RB              ;STARTING READ BUFFER ADDRESS
6958
6959                               ;*************************************************************************
6960                               ;
6961                               ;READ REVERSE DATA,ACK COMMAND
6962                               ;
6963                               ;*************************************************************************
6964
6965 040376  012737  100401  040700        MOV     #100401,T24PK3          ;READ REVERSE DATA,ACK COMMAND
6966 040404  012704  040700        195$:   MOV     #T24PK3,R4              ;SET UP R4 WITH PACKET ADDRESS
6967 040410  010337  040706                MOV     R3,T24SZ                ;SET UP RECORD SIZE IN PACKET
6968 040414  010465  177776                MOV     R4,TSDB(R5)             ;ISSUE COMMAND
6969 040420  004737  017120                JSR     PC,WAITF                ;WAIT FOR SSR TO SET
6970 040424  016501  000000                MOV     TSSR(R5),R1             ;GET TSSR CONTENTS
6971 040430  012702  100204                MOV     #SSR!SC!BIT2,R2                 ;SET UP EXPECTED
6972 040434  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
6973 040436  001406                        BEQ     200$                    ;BR, IF OK
6974 040440  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6978 040444                                ERRHRD  ERRNO,T24TRL,PKTSSR     ;TSSR INCORRECT AFTER READ DATA
     040444  104456                                                                TRAP    C$ERHRD
     040446  000735                                                                .WORD   477
     040450  043104                                                                .WORD   T24TRL
     040452  011700                                                                .WORD   PKTSSR
6979 040454                        200$:   CKLOOP                          ;LOOP IF SELECTED
     040454  104406                                                                TRAP    C$CLP1
6980 040456  013701  040604                MOV     T24BFR+14,R1            ;GET MESSAGE BUFFER (XST3)
6981 040462  010102                        MOV     R1,R2                   ;SET UP EXPECTED
6982 040464  052702  000001                BIS     #BIT0,R2                ;SET THE RIB BIT IN EXPECTED
6983 040470  020102                        CMP     R1,R2                   ;ARE THEY EQUAL
6984 040472  001406                        BEQ     210$                    ;BR, IF EQUAL (ALL IS WELL)
6985 040474  004737  020102                JSR     PC,FATCHK               ;INC AND CHECK FOR MORE THAN 25 ERRORS
6989 040500                                ERRHRD  ERRNO,T24LOR,EXPREC     ;THE RIB BIT WAS NOT SET IN XST0
     040500  104456                                                                TRAP    C$ERHRD
     040502  000736                                                                .WORD   478
     040504  041002                                                                .WORD   T24LOR
     040506  016344                                                                .WORD   EXPREC
6990 040510                        210$:
6991 040510                                ENDSUB                          ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>
     040510                                                                     L10063:
     040510  104403                                                                TRAP    C$ESUB
6992 040512  023727  002170  000031                CMP     FATFLG,#25.                     ;IS ERROR COUNT AT 25
6993 040520  002402                                BLT     999$                            ;BR, IF LESS THAN 25
```

```
6995 040522  004737  020154                JSR     PC,CKDROP               ;TRY TO DROP THE UNIT
6996 040526                          999$:
6997                                 ;
6998                                 ;
6999                                 ;
7000 040526  004737  017356                JSR     PC,TSTLOOP              ;DO WE NEED TO ITERATE TEST
7001 040532  103002                        BCC     163$                    ;BR. IF NO LOOP REQUIRED
7002 040534  000137  032224                JMP     T24LOOP                 ;EXECUTE AGAIN
7003 040540                          163$:
7004 040540                                EXIT    TST                     ;ALL DONE THIS TEST
     040540  104432                                                        TRAP    C$EXIT
     040542  002662                                                        .WORD   L10047-.
```

```
7006                                    ;
7007                                    ;LOCAL STORAGE FOR THIS TEST
7008                                    ;
7010 040544                             .BLKB   10.*2.+TUV2A&7>
7012 040550                             T24PACKET:                      ;COMMAND PACKET FOR TEST
7013 040550  100204                     .WORD   100204                  ;WRITE CHARACTERISTICS COMMAND, WITH IE, ACK
7014 040552  040560                     .WORD   T24DATA                 ;ADDRESS OF CHARACTERISTICS BLOCK
7015 040554  000000                     .WORD   0
7016 040556  000012                     .WORD   10.                     ;STARTING VALUE OF BLOCK SIZE
7017 040560                             T24DATA:                        ;CHARACTERISTICS DATA BLOCK
7018 040560  040570                     .WORD   T24BFR                  ;ADDRESS OF MESSAGE BUFFER
7019 040562  000000                     .WORD   0
7020 040564  000024                     .WORD   20.                     ;LENGTH OF MESSAGE BUFFER
7021 040566  000000                     .WORD   0
7022 040570                             T24BFR: .BLKW   25.             ;MESSAGE BUFFER
7023                                    ;
7024                                    ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
7025                                    ;
7027 040652                             .BLKB   10.*2.+TUV2A&7>
7029 040660                             T24PK2:
7030 040660  100206                     .WORD   100206                  ;WRITE SUB SYS MEM COMMAND, IE AND ACK
7031 040662  040710                     .WORD   T24BF2                  ;ADDRESS OF SELECT BLOCK DATA
7032 040664  000000                     .WORD   0
7033 040666  000006                     .WORD   6.                      ;SIZE OF DATA PACKET
7034
7036 040670                             .BLKB   10.*2.+TUV2A&7>
7038 040700                             T24PK3:
7039 040700  100205                     .WORD   100205                  ;READ COMMAND, IE AND ACK
7040 040702                             T24RB:
7041 040702  003072                     T24WB:  .WORD   FREE            ;ADDRESS OF WRITE BUFFER
7042 040704  000000                             .WORD   0
7043 040706  000000                     T24SZ:  .WORD   0               ;SIZE OF BUFFER (EXTENT)
7044                                            .EVEN
7045                                    ;
7046                                    ;
7047                                    ;
7048 040710                             T24BF2:
7049 040710     010                     T24BS0: .BYTE   10              ;BSEL0 AREA
7050 040711     200                     T24BS1: .BYTE   200             ;BSEL1 AREA
7051 040712  000000                     T24S2:  .WORD   0               ;SEL 2 AREA
7052 040714  000000                     T24S3:  .WORD   0               ;DATA AREA
7053                                    ;
7054                                    ;
7055                                            .EVEN
7056                                    ;TAPE MOTION PACKET COMMAND VALUES
7057
7058 040716  100005                     T24RN:  .WORD   100005          ;READ DATA (NEXT)
7059 040720  100405                     T24WDR: .WORD   100405          ;READ DATA RETRY
7060 040722  102005                     T24CON: .WORD   102005          ;WRITE CONTINOUS
7061 040724  177777                             .WORD   177777          ;END OF DATA
7062 040726  000000                     T24DLY: .WORD   0               ;DELAY STORAGE AREA
7063
7064
```

```
7066
7067                                    ;*
7068                                    ;LOCAL TEXT MESSAGES FOR TEST
7069                                    ;*
7070
7071 040730     116     105     106    T24NEF: .ASCIZ  'NEF Not Set After NON-EXECUTABLE FUNCTION'
7072 041002     122     111     102    T24LOR: .ASCIZ  'RIB Not Set After READ REVERSE Into BOT'
7073 041052     124     123     123    T24WDG: .ASCIZ  'TSSR Not Correct After Illegal Buffer Address Bits Set'
7074 041141     124     123     123    T24NXM: .ASCIZ  'TSSR Not Correct After NXM Memory Address In Packet'
7075 041225     124     123     123    T24WDF: .ASCIZ  'TSSR Not Correct After Illegal Mode Bits Set'
7076 041302     111     154     154    T24ILA: .ASCIZ  'Illegal Address Bits, Failed To Set ILA Bit In XSTO'
7077 041366     111     154     154    T24LOQ: .ASCIZ  'Illegal Mode Bits, Failed To Set ILC Bit In XSTO'
7078 041447     122     105     101    T24SSR: .ASCIZ  'READ COMMAND Not Accepted'
7079 041501     124     123     123    T24WDE: .ASCIZ  'TSSR Not Correct After WRITE DATA Command'
7080 041553     124     141     160    T24BOT: .ASCIZ  'Tape Not At BOT After REWIND Command'
7081 041620     104     141     164    T24DTA: .ASCIZ  'Data Written To Tape Not Equal To Data Read From Tape'
7082 041706     122     105     101    T24EOT: .ASCIZ  'READ DATA OVER EOT GAVE NO TAPE STATUS ALERT'
7083 041763     124     123     123    T24TM:  .ASCIZ  'TSSR Not Correct After READ COMMAND Reject'
7084 042036     122     145     167    T24RWN: .ASCIZ  'Rewind (POSITION) Command Not Accepted'
7085 042105     122     101     115    T24RNC: .ASCIZ  'RAM Error, Correct Data Pattern Not In Ram'
7086 042160     124     123     123    T24AM3: .ASCIZ  'TSSR Init. Failed After READ COMMAND'
7087 042225     104     162     151    T24OFL: .ASCIZ  'Drive 7 Select Failed To Set "OFL" In TSSR'
7088 042300     124     123     123    T24WDD: .ASCIZ  'TSSR Not Correct After READ DATA Command, SWB Bit Set'
7089 042366     124     123     123    T24WDC: .ASCIZ  'TSSR Not Correct After READ DATA Command'
7090 042437     103     126     103    T24VCK: .ASCIZ  'CVC Set, Didn't Reset VCK In Message Buffer'
7091 042512     124     123     102    T24BA:  .ASCIZ  'TSBA Not Correct After READ DATA Command'
7092 042563     127     122     111    T24WSS: .ASCIZ  'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
7093 042652     122     145     141    T24LON: .ASCIZ  'Reading Long Record Failed To Set RLL Bit In XSTO'
7094 042734     122     145     141    T24LOP: .ASCIZ  'Reading Long Record Failed To Set RLS Bit In XSTO'
7095 043016     122     145     163    T24PBP: .ASCIZ  'Residual Byte Count Incorrect After Short Record Read'
7096 043104     122     145     141    T24TRL: .ASCIZ  'Reading Long Record  Failed To Give Tape Status Alert'
7097 043172     102     141     163    TST24ID:        .ASCIZ  'Basic Read Data (Forward and Reverse)'
7098                                            .EVEN
7099                                    ;*
7100                                    ;
7101                                    ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
7102                                    ;WRITE SUBSYSTEM MEMORY COMMAND)
7103                                    ;
7104                                    ;*
7105
7106 043240                             T24REST:
7107 043240                                     SAVREG                          ;SAVE THE REGISTERS
7108 043244     012701  040550                    MOV     #T24PACKET,R1           ;START OF THE PACKET
7109 043250     012721  100004                    MOV     #100004,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK.
7110 043254     012721  040560                    MOV     #T24DATA,(R1)+          ;ADDRESS OF CHARAISTICS DATA BLOCK
7111 043260     005021                            CLR     (R1)+                   ;EXTENDED ADDRESS
7112 043262     012721  000012                    MOV     #10.,(R1)+              ;SIZE OF DATA BLOCK IN BYTES
7113 043266     012721  040570                    MOV     #T24BFR,(R1)+           ;ADDRESS OF MESSAGE BUFFER
7114 043272     005021                            CLR     (R1)+
7115 043274     012721  000024                    MOV     #20.,(R1)+              ;LENGTH OF MESSAGE BUFFER
7116 043300     005021                            CLR     (R1)+
7117 043302     012711  000000                    MOV     #0,(R1)                 ;SELECT DRIVE ZERO
7118 043306     012702  000030                    MOV     #24.,R2                 ;NUMBER OF LOCATIONS TO BE CLEARED
7119 043312     012762  177777  040570  64$:      MOV     #177777,T24BFR(R2)      ;ALL ONES TO MESSAGE BUFFER
7120 043320     005742                            TST     -(R2)                   ;NEXT LOCATION
7121 043322     022702  000000                    CMP     #0,R2                   ;CHECK FOR END OF LOOP
7122 043326     001371                            BNE     64$                     ;KEEP GOING UNTIL DONE
```

```
7123 043330  000207                    RTS     PC                      ;RETURN
7124
7125
7126 043332                        T24RT2:
7127 043332                            SAVREG                          ;SAVE THE REGISTERS
7128 043336  012701  040660            MOV     #T24PK2,R1              ;START OF THE PACKET
7129 043342  012721  100206            MOV     #100206,(R1)+           ;WRITE SUBSYSTEM MEM. WITH ACK, IE
7130 043346  012721  040710            MOV     #T24BF2,(R1)+           ;ADDRESS OF DATA BLOCK
7131 043352  005021                    CLR     (R1)+                   ;EXTENDED ADDRESS
7132 043354  012721  000006            MOV     #6.,(R1)+               ;SIZE OF DATA BLOCK IN BYTES
7133 043360  005021                    CLR     (R1)+
7134 043362  012701  040710            MOV     #T24BF2,R1              ;POINT TO DATA SEL AREA
7135 043366  005021                    CLR     (R1)+
7136 043370  005011                    CLR     (R1)
7137 043372  000207                    RTS     PC                      ;RETURN
7138 043374                        T24RT3:
7139 043374                            SAVREG                          ;SAVE THE REGISTERS
7140 043400  012701  040700            MOV     #T24PK3,R1              ;START OF THE PACKET
7141 043404  012721  000000            MOV     #0,(R1)+                ;CLEAR AREA OUT
7142 043410  012721  000000            MOV     #0,(R1)+                ;ADDRESS OF DATA BLOCK
7143 043414  005021                    CLR     (R1)+                   ;EXTENDED ADDRESS
7144 043416  012711  000000            MOV     #0,(R1)                 ;SIZE OF DATA BLOCK IN BYTES
7145 043422  000207                    RTS     PC                      ;RETURN
7146 043424                            ENDTST
     043424                                                        L10047:
     043424  104401                                                    TRAP    C$ETST
```

```
                        .SBTTL  TEST 5: MANUAL INTERVENTION

;THIS TEST MUST BE STARTED AS FOLLOWS:
;
;AT THE DIAGNOSTIC SUPERVISOR PROMPT "DR>"
;TYPE "START/FLAG:PNT/TEST:5/PASS:1"
;       ----------------------------
;
;
;THE MANUAL INTERVENTION TEST IS A STANDALONE ROUTINE (NOT REALLY A "TEST")
;THAT ALLOWS THE OPERATOR TO CHECK OUT VARIOUS ELEMENTS AND FUNCTIONS OF
;THE SUBSYSTEM THAT CANNOT BE MANIPULATED BY THE PROGRAM ALONE. WHEN
;THIS ROUTINE IS STARTED, IT FIRST PRINTS OUT A MENU OF SELECTABLE
;SUBTESTS AND THEN WAITS FOR THE OPERATOR TO TYPE IN A SELECTION CODE.
;THE ONLY WAYS TO EXIT THIS ROUTINE AND RETURN TO THE DIAGNOSTIC SUPERVISOR
;ARE BY TYPING <CTRL-C> OR SELECTING CODE 4.
;SELECTION CODES AND SUBROUTINES ARE:
;
;
;       CODE    ROUTINE
;
;         0     HELP, PRINTS THIS MENU.
;         1     REWIND AND UNLOAD COMMAND TEST
;         2     WRITE-PROTECT TEST
;         3     FRONT PANEL ON-LINE SWITCH TEST
;         4     EXIT (RETURN TO SUPERVISOR)
;
;
;EACH MENU ITEM CORRESPONDS TO A SUBTEST, AS FOLLOWS:
;
;
;
;SELECTION 0 - PRINTS OUT THE MENU ON THE CONSOLE TERMINAL.
;
;
;
;
;
;
;SELECTION 1 - THIS ROUTINE INSTRUCTS THE OPERATOR TO PLACE THE DRIVE ON-LINE
;AND AT OR BEYOND BOT. THE TEST WILL THEN ISSUE THE REWIND AND
;UNLOAD COMMAND. IT WILL ALSO TELL THE OPERATOR IF THE DRIVE
;ENDED UP ON-LINE OR OFF-LINE.
;
;
;SELECTION 2 - THIS ROUTINE INSTRUCTS THE OPERATOR TO MOUNT A SCRATCH
;TAPE REEL THAT DOES NOT HAVE A WRITE-ENABLE RING INSTALLED, THEN
;WAITS FOR THE OPERATOR TO RESPOND THAT THIS HAS BEEN ACCOMPLISHED.
;UPON THE RESPONSE, THE PROGRAM VERIFIES THAT THE TRANSPORT SHOWS
;A WRITE-PROTECTED STATUS, THEN ATTEMPTS TO WRITE DATA ON THE
;TAPE AND EXPECTS THE APPROPRIATE ERROR TERMINATION INDICATING THAT
;THE WRITE FUNCTION COULD NOT BE PERFORMED BECAUSE THE REEL IS
;WRITE-PROTECTED.  IF THE APPROPRIATE TERMINATION IS NOT RECEIVED,
;AN ERROR IS REPORTED.
;
;SELECTION 3 - THIS TEST CHECKS THAT THE PROGRAM CAN READ THE SENSE
;OF THE FRONT PANEL "ON-LINE" BUTTON/LIGHT. THE PROGRAM CHECKS THE
```

```
                                        ;STATE OF THE DRIVE (ON-LINE OR OFF-LINE) AND PRINTS A MESSAGE TO
                                        ;NOTIFY THE OPERATOR I.E. "DRIVE IS NOW OFF-LINE" OR "DRIVE IS NOW ON-LINE".
                                        ;
                                        ;
                                        ;SELECTION 4 - THIS WILL RETURN THE PROGRAM TO THE DIAGNOSTIC
                                        ;SUPERVISOR PROMPT. NOTE: IF THE OPERATOR FAILED TO SELECT A PASS
                                        ;COUNT OF ONE, THE PROGRAM WILL LOOP UNTIL STOPPED WITH A CONTROL C.
                                        ;
                                        ;
                                        ;
                                        ;
043426                                          BGNTST
043426                                                                                  T5::
043426  005037  002170                          CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
043432  005037  003100                          CLR     KTFLG                   ;HOLD OFF KTJ11
043436  005737  002162                          TST     TSTCNT                  ;IS THIS THE FIRST TEST
043442  001403                                  BEQ     21$                     ;BR, IF FIRST TEST
043444  012700  045400                          MOV     #T38NE,R0               ;"TEST NOT EXECUTED"
043450  000402                                  BR      3$                      ;JUMP IF NOT FIRST TEST
043452                                  21$:
043452  012700  046466                          MOV     #T38ID,R0               ;TEST ID MESSAGE
043456  004737  017410                  3$:     JSR     PC,TSTSETUP             ;DO THE COMMON SETUP
043462  004737  021404                          JSR     PC,CHKMAN               ;IS MANUAL INTERVENTION ALLOWED?
043466  103402                                  BCS     19$                     ;BR, IF MANUAL INTER ALLOWED
043470  000137  044600                          JMP     64$                     ;JUMP IF NOT ALLOWED
043474  022737  000001  002162          19$:    CMP     #1,TSTCNT               ;CHECK MIGHT HAVE TO LEAVE
043502  001402                                  BEQ     22$                     ;BR, IF YOU DON'T HAVE TO
043504  000137  044600                          JMP     64$                     ;WASN'T FIRST TEST IN SEQUENCE
043510                                  22$:
043510  005037  002170                  2$:     CLR     FATFLG                  ;CLEAR THE FATAL ERROR FLAG
043514  012737  176750  044612                  MOV     #65000.,T38DLY          ;SET UP DELAY COUNTER
043522  004737  016644                  5$:     JSR     PC,SOFINIT              ;DO A SOFT INIT
043526  103427                                  BCS     23$                     ;BRANCH IF OK
043530  010001                                  MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
043532  032701  000200                          BIT     #SSR,R1                 ;CHECK FOR TSSR SET
043536  001023                                  BNE     23$                     ;KEEP GOING IF NOT SET
043540                                          DELAY   250                     ;CALL DELAY ROUTINE
043540  012727  000250                                                                  MOV     #250.,(PC)+
043544  000000                                                                          .WORD   0
043546  013727  002116                                                                  MOV     L$DLY,(PC)+
043552  000000                                                                          .WORD   0
043554  005367  177772                                                                  DEC     -6(PC)
043560  001375                                                                          BNE     .-4
043562  005367  177756                                                                  DEC     -22(PC)
043566  001367                                                                          BNE     .-20
043570  005337  044612                          DEC     T38DLY                  ;BUMP COUNTER DOWN
043574  001352                                  BNE     5$                      ;BR, IF MORE TIME LEFT
043576                                          ERRDF   ERRNO,SFIERR,SFIMSG     ;REPORT FATAL ERROR
043576  104455                                                                          TRAP    C$ERDF
043600  000765                                                                          .WORD   501
043602  003550                                                                          .WORD   SFIERR
043604  011666                                                                          .WORD   SFIMSG
043606  012700  046512                  23$:    MOV     #MIMENU,R0              ;MENU OF MANUAL INTERVENTIONS
043612  012701  000004                          MOV     #4,R1                   ;MAXIMUM ALLOWED SELECTION
043616  004737  021162                          JSR     PC,GETSEL               ;GO GET THE OPERATORS SELECTION
043622  010004                                  MOV     R0,R4                   ;GET NUMBER FROM ROUTINE
043624  006304                                  ASL     R4                      ;CONVERT TO WORD OFFSET
```

```
 7256 043626  000174  043632            JMP     @6$(R4)                 ;JUMP TO PROPER LOOP
 7257 043632  043510             6$:    .WORD   2$                      ;RETYPE THE MENU
 7258 043634  043644                    .WORD   20$                     ; 1 REWIND AND UNLOAD COMMAND TEST
 7259 043636  044176                    .WORD   25$                     ; 2 WRITE PROTECT
 7260 043640  044456                    .WORD   500$                    ; 3 FRONT PANEL ON-LINE SWITCH TEST
 7261 043642  044600                    .WORD   63$                     ; 4 LEAVE THE TEST
 7262
 7263
 7264 043644                     20$:   PRINTF  #T38MS4                 ;TELL'EM WHAT TO DO
      043644  012746  046026                                                        MOV     #T38MS4,-(SP)
      043650  012746  000001                                                        MOV     #1,-(SP)
      043654  010600                                                               MOV     SP,R0
      043656  104417                                                               TRAP    C$PNTF
      043660  062706  000004                                                        ADD     #4,SP
 7265 043664  004737  016644             222$:  JSR     PC,$OFINIT              ;DO SOFT INIT OF CONTROLLER
 7266 043670  103405                    BCS     300$                    ;BR IF SOFT INIT = OK
 7270 043672  010001                    MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
 7271 043674                            ERRDF   ERRNO,SFIERR,SFIMSG     ;DEVICE FATAL ERROR DURING INIT
      043674  104455                                                               TRAP    C$ERDF
      043676  000766                                                               .WORD   502
      043700  003550                                                               .WORD   SFIERR
      043702  011666                                                               .WORD   SFIMSG
 7272 043704                     300$:
 7273 043704  012704  045320            MOV     #T38PK2,R4              ;SUBROUTINE NEEDS PACKET ADDRESS
 7274 043710  004737  010332            JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
 7275 043714  103405                    BCS     310$                    ;BR, IF COMMAND ISSUED OK
 7279 043716  010001                    MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
 7280 043720                            ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
      043720  104456                                                               TRAP    C$ERHRD
      043722  000767                                                               .WORD   503
      043724  004754                                                               .WORD   WRTMSG
      043726  011666                                                               .WORD   SFIMSG
 7281 043730                     310$:
 7282 043730  012737  000031  044612            MOV     #25.,T38DLY             ;SET UP FOR A LONG WAIT
 7283 043736  036527  000000  000100  311$:     BIT     TSSR(R5),#OFL           ;IS DRIVE OFF-LINE
 7284 043744  001431                    BEQ     315$                    ;BR, IF DRIVE IS ON-LINE
 7285 043746                            DELAY   250                     ;DELAY ABOUT .25 SEC
      043746  012727  000250                                                        MOV     #250.,(PC)+
      043752  000000                                                               .WORD   0
      043754  013727  002116                                                        MOV     L$DLY,(PC)+
      043760  000000                                                               .WORD   0
      043762  005367  177772                                                        DEC     -6(PC)
      043766  001375                                                               BNE     .-4
      043770  005367  177756                                                        DEC     -22(PC)
      043774  001367                                                               BNE     .-20
 7286 043776  005337  044612            DEC     T38DLY                  ;BUMP LONG DELAY COUNTER DOWN
 7287 044002  001355                    BNE     311$                    ;BR, IF MORE LONG DELAY TO GO
 7288 044004                            PRINTF  #T38OFL                 ;"DRIVE IS NOW OFF-LINE"
      044004  012746  045772                                                        MOV     #T38OFL,-(SP)
      044010  012746  000001                                                        MOV     #1,-(SP)
      044014  010600                                                               MOV     SP,R0
      044016  104417                                                               TRAP    C$PNTF
      044020  062706  000004                                                        ADD     #4,SP
 7289 044024  000137  043664            JMP     222$                    ;STAY HERE FOREVER, WITH MESSAGE
 7290 044030                     315$:  PRINTF  #T38MS5                 ;"DRIVE SHOULD NOW REWIND AND GO OFL
      044030  012746  046105                                                        MOV     #T38MS5,-(SP)
      044034  012746  000001                                                        MOV     #1,-(SP)
```

```
                044040  010600                                                          MOV     SP,R0
                044042  104417                                                          TRAP    C$PNTF
                044044  062706  000004                                                  ADD     #4,SP
   7291         044050  012704  045350          MOV     #T38PK3,R4          ;SET UP NEW PACKET FOR REWIND/RELEASE
   7292         044054  010465  177776          MOV     R4,TSDB(R5)         ;REWIND RELEASE,ACK,CVC=1 CMD
   7293         044060  004737  017120          JSR     PC,WAITF            ;WAIT FOR SSR TO SET
   7294         044064  016501  000000          MOV     TSSR(R5),R1         ;GET TSSR STATUS
   7295         044070  012702  000300  320$:    MOV     #SSR!OFL,R2         ;SET UP EXPECTED
   7296         044074  020201                  CMP     R2,R1               ;IS EVERYTHING OK
   7297         044076  001404                  BEQ     350$                ;BR, IF ALL IS WELL
   7301         044100                          ERRHRD  ERRNO,T38SST,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                044100  104456                                                          TRAP    C$ERHRD
                044102  000770                                                          .WORD   504
                044104  046240                                                          .WORD   T38SST
                044106  011700                                                          .WORD   PKTSSR
   7302         044110                  350$:   CKLOOP                      ;LOOP ON ERROR, IF FLAG SET
                044110  104406                                                          TRAP    C$CLP1
   7303         044112  016501  000000          MOV     TSSR(R5),R1         ;READ TSSR STATUS
   7304         044116  032701  000100          BIT     #OFL,R1             ;CHECK THE OFF-LINE BIT
   7305         044122  001011                  BNE     380$                ;BR, IF DRIVE IS OFF-LINE
   7306         044124                          PRINTF  #T380NL             ;"DRIVE IS NOW ON-LINE"
                044124  012746  045676                                                  MOV     #T380NL,-(SP)
                044130  012746  000001                                                  MOV     #1,-(SP)
                044134  010600                                                          MOV     SP,R0
                044136  104417                                                          TRAP    C$PNTF
                044140  062706  000004                                                  ADD     #4,SP
   7307         044144  000410                  BR      390$                ;ALMOST DONE
   7308         044146                  380$:   PRINTF  #T380FL             ;"DRIVE IS NOW OFF-LINE"
                044146  012746  045772                                                  MOV     #T380FL,-(SP)
                044152  012746  000001                                                  MOV     #1,-(SP)
                044156  010600                                                          MOV     SP,R0
                044160  104417                                                          TRAP    C$PNTF
                044162  062706  000004                                                  ADD     #4,SP
   7309         044166  005037  002172  390$:    CLR     INTRECV             ;CLEAR INTERRUPT FLAG
   7310         044172  000137  043510          JMP     2$                  ;TRY AGAIN
   7311         044176                  25$:    GMANIL  T38MSG,T38DAT,-1,NO ;WAIT FOR OPERATOR TO MOUNT TAPE
                044176  104443                                                          TRAP    C$GMAN
                044200  000404                                                          BR      10000$
                044202  047042                                                          .WORD   T38DAT
                044204  000120                                                          .WORD   T$CODE
                044206  046370                                                          .WORD   T38MSG
                044210  177777                                                          .WORD   -1
                044212                                                          10000$:
   7312         044212                          BNCOMPLETE 25$              ;RETRY IF ERROR
                044212  103371                                                          BCC     25$
   7313         044214  005737  047042          TST     T38DAT              ;DID OPERATOR SAY 'YES' ?
   7314         044220  001002                  BNE     27$                 ;BRANCH IF YES
   7315         044222  000137  043510          JMP     2$                  ;RETURN TO MAIN MENU
   7316         044226                  27$:
   7317         044226  004737  016644          JSR     PC,SOFINIT          ;DO SOFT INIT OF CONTROLLER
   7318         044232  103405                  BCS     400$                ;BR IF SOFT INIT = OK
   7322         044234  010001                  MOV     R0,R1               ;SAVE CONTENTS OF TSSR
   7323         044236                          ERRDF   ERRNO,SFIERR,SFIMSG ;DEVICE FATAL ERROR DURING INIT
                044236  104455                                                          TRAP    C$ERDF
                044240  000771                                                          .WORD   505
                044242  003550                                                          .WORD   SFIERR
                044244  011666                                                          .WORD   SFIMSG
```

```
7324 044246                    400$:  CKLOOP                          ;LOOP IF SELECTED
     044246  104406                                                                TRAP    C$CLP1
7325 044250  012704  045320           MOV     #T38PK2,R4              ;SUBROUTINE NEEDS PACKET ADDRESS
7326 044254  004737  010332           JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
7327 044260  103405                   BCS     410$                    ;BR, IF COMMAND ISSUED OK
7331 044262  010001                   MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
7332 044264                           ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     044264  104456                                                                TRAP    C$ERHRD
     044266  000772                                                                .WORD   506
     044270  004754                                                                .WORD   WRTMSG
     044272  011666                                                                .WORD   SFIMSG
7333 044274                    410$:  CKLOOP                          ;LOOP IF SELECTED
     044274  104406                                                                TRAP    C$CLP1
7334 044276  013701  044614           MOV     T38BFR+6,R1             ;PICK UP XSTO CONTENTS
7335 044302  010102                   MOV     R1,R2                   ;SET UP EXPECTED
7336 044304  052702  000004           BIS     #BIT2,R2                ;SET UP THE WRITE LOCKED BIT
7337 044310  020102                   CMP     R1,R2                   ;ARE THEY CORRECT
7338 044312  001406                   BEQ     430$                    ;BR, IF ALL IS WELL (OK)
7342 044314                           ERRHRD  ERRNO,T38WRL,EXPREC     ;"WRITE LOCKED BIT IS NOT SET ETC."
     044314  104456                                                                TRAP    C$ERHRD
     044316  000773                                                                .WORD   507
     044320  045541                                                                .WORD   T38WRL
     044322  016344                                                                .WORD   EXPREC
7343 044324  000137  043510           JMP     2$                      ;BECAUSE OF ERROR GO BACK TO MENU
7344 044330                    430$:  CKLOOP                          ;LOOP IF SELECTED
     044330  104406                                                                TRAP    C$CLP1
7345 044332                           PRINTF  #T38WOK                 ;PRINT "DRIVE IS WRITE PROTECTED"
     044332  012746  046430                                                        MOV     #T38WOK,-(SP)
     044336  012746  000001                                                        MOV     #1,-(SP)
     044342  010600                                                                MOV     SP,R0
     044344  104417                                                                TRAP    C$PNTF
     044346  062706  000004                                                        ADD     #4,SP
7346 044352  017737  136514  045372  435$:  MOV     @FREE,T38WR             ;SET UP WRITE BUFFER ADDRESS
7347 044360  012704  045370           MOV     #T38PK4,R4              ;GET PACKET ADDRESS
7348 044364  010465  177776           MOV     R4,TSDB(R5)             ;SET THE PACKET ADDRESS
7349 044370  004737  017120           JSR     PC,WAITF                ;WAIT FOR SSR TO SET
7350 044374  016501  000000           MOV     TSSR(R5),R1             ;GET TSSR
7351 044400  012702  100206           MOV     #SC!SSR!BIT1!BIT2,R2    ;SET UP EXPECTED
7352 044404  020102                   CMP     R1,R2                   ;ARE THEY EQUAL (CORRECT)
7353 044406  001404                   BEQ     440$                    ;BR, IF CORRECT STATUS
7357 044410                           ERRHRD  ERRNO,T38WRT,PKTSSR     ;"TSSR INCORRECT AFTER WRITE COMMAND
     044410  104456                                                                TRAP    C$ERHRD
     044412  000774                                                                .WORD   508
     044414  045455                                                                .WORD   T38WRT
     044416  011700                                                                .WORD   PKTSSR
7358 044420                    440$:  CKLOOP                          ;LOOP ON ERROR, IF FLAG SET
     044420  104406                                                                TRAP    C$CLP1
7359 044422  013701  044644           MOV     T38BFR+6,R1             ;READ XSTO CONTENTS
7360 044426  010102                   MOV     R1,R2                   ;SET UPR EXPECTED
7361 044430  052702  004000           BIS     #BIT11,R2               ;SET THE WRITE LOCK ERROR BIT (XSTO)
7362 044434  020102                   CMP     R1,R2                   ;WAS THE BIT SET
7363 044436  001404                   BEQ     450$                    ;BR, IF IT WAS (GOOD)
7367 044440                           ERRHRD  ERRNO,T38WLE,EXPREC     ;"WRITE LOCK ERROR BIT NOT SET"
     044440  104456                                                                TRAP    C$ERHRD
     044442  000775                                                                .WORD   509
     044444  045602                                                                .WORD   T38WLE
     044446  016344                                                                .WORD   EXPREC
```

```
7368 044450                        450$:   CKLOOP                          ;LOOP IF SELECTED
     044450  104406                                                                  TRAP    C$CLP1
7369 044452  000137  043510                 JMP     2$                      ;GO BACK TO MENU
7370
7371 044456                        500$:   PRINTF  #T38MS6                 ;TELL'EM WHAT TO DO
     044456  012746  046162                                                          MOV     #T38MS6,-(SP)
     044462  012746  000001                                                          MOV     #1,-(SP)
     044466  010600                                                                  MOV     SP,R0
     044470  104417                                                                  TRAP    C$PNTF
     044472  062706  000004                                                          ADD     #4,SP
7372 044476  004737  016644        510$:   JSR     PC,SOFINIT              ;DO SOFT INIT OF CONTROLLER
7373 044502  103405                        BCS     520$                    ;BR IF SOFT INIT = OK
7377 044504  010001                        MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
7378 044506                                ERRDF   ERRNO,SFIERR,SFIMSG     ;DEVICE FATAL ERROR DURING INIT
     044506  104455                                                                  TRAP    C$ERDF
     044510  000776                                                                  .WORD   510
     044512  003550                                                                  .WORD   SFIERR
     044514  011666                                                                  .WORD   SFIMSG
7379 044516                        520$:
7380 044516  016503  000000                 MOV     TSSR(R5),R3             ;IS DRIVE OFF-LINE
7381 044522  030327  000100                 BIT     R3,#OFL                 ;LOOK AT THE OFF-LINE BIT
7382 044526  001412                        BEQ     550$                    ;BR, IF DRIVE IS ON-LINE
7383 044530                                PRINTF  #T380FL                 ;"DRIVE IS NOW OFF-LINE"
     044530  012746  045772                                                          MOV     #T380FL,-(SP)
     044534  012746  000001                                                          MOV     #1,-(SP)
     044540  010600                                                                  MOV     SP,R0
     044542  104417                                                                  TRAP    C$PNTF
     044544  062706  000004                                                          ADD     #4,SP
7384 044550  000137  043510                 JMP     2$                      ;RETURN TO MENU
7385
7386
7387 044554                        550$:   PRINTF  #T380NM                 ;"DRIVE IS NOW ON-LINE"
     044554  012746  045740                                                          MOV     #T380NM,-(SP)
     044560  012746  000001                                                          MOV     #1,-(SP)
     044564  010600                                                                  MOV     SP,R0
     044566  104417                                                                  TRAP    C$PNTF
     044570  062706  000004                                                          ADD     #4,SP
7388 044574  000137  043510                 JMP     2$                      ;RETURN TO MENU
7389
7390 044600                        63$:
7391 044600                        64$:    EXIT    TST                     ;LEAVE TEST
     044600  104432                                                                  TRAP    C$EXIT
     044602  002752                                                                  .WORD   L10064-.
7392
7393                               ;+
7394                               ;LOCAL TEXT MESSAGES FOR TEST
7395                               ;-
7396
7397                               ;LOCAL STORAGE FOR THIS TEST
7398                               ;
7399                               ;+
7400                               ;LOCAL STORAGE FOR THIS TEST
7401                               ;-
7402
7403 044604  000000                TTION2:         .WORD   0               ;WORD SET IF SUPERVISOR TTI INTER OFF
7404 044606  000000                TVSAV2: .WORD   0               ;SAVE TTI VECTOR
7405 044610  000000                TPSAV2: .WORD   0               ;SAVE TTI PRIORITY
```

```
7406
7407 044612  000000                T38DLY: .WORD   0                       ;DELAY COUNTER FOR TEST
7409 044614                              .BLKB   10-<.-TUV2A67>
7411 044620                      T38PACKET:                              ;COMMAND PACKET FOR TEST
7412 044620  140006                      .WORD   140006                  ;WRITE SUBSYSTEM MEM. CMD.  ACK,CVC=1
7413 044622  044630                      .WORD   T38TAD                  ;ADDRESS OF CHARACTERISTICS BLOCK
7414 044624  000000                      .WORD   0
7415 044626  000012                      .WORD   10.                     ;STARTING VALUE OF BLOCK SIZE
7416 044630                      T38TAD:                         ;CHARACTERISTICS DATA BLOCK
7417 044630     000              T38BS0: .BYTE   0                       ;BSEL0 BYTE
7418 044631     000              T38BS1: .BYTE   0                       ;BSEL1 BYTE
7419 044632  000000              T38BS2: .WORD   0                       ;BSEL1 WORD
7420 044634  000000                      .WORD   0                       ;DATA
7421 044636                      T38BFR: .BLKW   150.                    ;MESSAGE BUFFER
7422 045312  000000              T38EB:  .WORD                           ;END OF BUFFER ADDRESS
7423
7424
7426 045314                              .BLKB   10-<.-TUV2A67>
7428 045320                      T38PK2:                                 ;COMMAND PACKET FOR TEST
7429 045320  140004                      .WORD   140004                  ;WRITE CHARA. MEM. CMND.,  ACK,CVC=1
7430 045322  045330                      .WORD   T38DTA                  ;ADDRESS OF SELECT DATA BLOCK
7431 045324  000000                      .WORD   0
7432 045326  000012                      .WORD   10.                     ;STARTING VALUE OF BLOCK SIZE
7433
7434
7435 045330                      T38DTA:                                 ;SELECT DATA BLOCK
7436 045330  044636                      .WORD   T38BFR                  ;ADDRESS OF MESSAGE BUFFER
7437 045332  000000                      .WORD   0
7438 045334  000400                      .WORD   256.                    ;LENGTH OF MESSAGE BUFFER
7439 045336  000000              T38EAI: .WORD   0                       ;EAI BIT WORD
7441 045340                              .BLKB   10-<.-TUV2A67>
7443 045350  140412              T38PK3: .WORD   140412                  ;REWIND AND UNLOAD COMMAND
7444 045352  000000                      .WORD   0                       ;NOT USED
7445 045354  000000                      .WORD   0                       ;NOT USED
7446 045356  000000                      .WORD   0                       ;NOT USED
7447 045360  000000                      .WORD   0                       ;NOT USED
7448                             ;
7449                             ;WRITE TAPE PACKET
7450                             ;
7452 045362                              .BLKB   10-<.-TUV2A67>
7454 045370  140005              T38PK4: .WORD   140005                  ;WRITE, ACK, CVC=1 COMMAND
7455 045372  000000              T38WR:  .WORD   0                       ;ADDRESS OF WRITE BUFFER
7456 045374  000000                      .WORD   0                       ;MORE ADDRESS OF WRITE BUFFER
7457 045376  000400              T38SIZ: .WORD   256.                    ;SIZE OF RECORD
7458
7459
7460
7461
7462                             ;+
7463                             ;LOCAL TEXT MESSAGES FOR TEST
7464                             ;-
7465
7466
7467
7468
7469
7470 045400     123     164     141  T38NE:  .ASCIZ  'Stand-alone Manual Intervention Not Executed'
```

```
7471 045455  124  123  123  T38WRT: .ASCIZ  'TSSR Not Correct After WRITE, With WRITE PROTECT On'
7472 045541  127  122  111  T38WRL: .ASCIZ  'WRITE LOCKED Bit Not Set In XSTO'
7473 045602  127  122  111  T38WLE: .ASCIZ  'WRITE LOCK ERROR Bit Not Set In XSTO, After Attempted WRITE'
7474 045676  045  116  045  T38ONL: .ASCIZ  '<N>A ERROR Drive Is Still ON-LINE'
7475 045740  045  116  045  T38ONM: .ASCIZ  '<N>A Drive Is Now ON-LINE'
7476 045772  045  116  045  T38OFL: .ASCIZ  '<N>A Drive Is Now  OFF-LINE'
7477 046026  045  116  045  T38MS4: .ASCIZ  '<N>A Set Drive To On-line and At Or Beyond BOT'
7478 046105  045  116  045  T38MS5: .ASCIZ  '<N>A Drive Should Now Rewind and Go Off-line'
7479
7480 046162  045  116  045  T38MS6: .ASCIZ  '<N>A Front Panel On-line/Off-line Switch Test'
7481 046240  103  157  156  T38SST: .ASCIZ  'Contents Of TSSR Incorrect After REWIND And RELEASE'
7482 046324  045  116  045  T38MS2: .ASCIZ  '<N>AType RETURN To Return To Menu<N>'
7483 046370  111  163  040  T38MSG: .ASCIZ  'Is Write-Protected Tape Mounted'
7484 046430  045  116  045  T38WOK: .ASCIZ  '<N>A Drive Is Write Protected'
7485 046466  115  141  156  T38ID:  .ASCIZ  'Manual Intervention'
7486                          .EVEN
7487 046512 046532 046604 046632  MIMENU: .WORD  1$,2$,5$,6$,7$
7488 046524 046776 047041 000000          .WORD  9$,10$,0
7489
7490 046532  012  123  105  1$:   .ASCIZ  <12>'SELECT OPERATION FROM FOLLOWING OPTIONS:'
7491 046604  012  011  060  2$:   .ASCIZ  <12>'    0       Display This Menu'
7492 046632  011  061  011  5$:   .ASCIZ  '       1       Rewind and Unload Command Test'
7493 046674  011  062  011  6$:   .ASCIZ  '       2       Write Protect Test'
7494 046722  011  063  011  7$:   .ASCIZ  '       3       Front Panel On-line/Off-line Switch Test'
7495 046776  011  064  011  9$:   .ASCIZ  '       4       Return to Diagnostic Supervisor'
7496 047041  000            10$:  .ASCIZ  ''
7497                          .EVEN
7498
7499                    ;+
7500                    ;LOCAL STORAGE FOR THIS TEST
7501                    ;-
7502
7503 047042  000000    T38DAT: .WORD   0               ;LOGICAL RESPONSE TO QUESTION
7504 047044            T38REST:
7505 047044                    SAVREG                  ;SAVE THE REGISTERS
7506 047050  012701  044620      MOV     #T38PACKET,R1   ;START OF THE PACKET
7507 047054  012721  140206      MOV     #140206,(R1)+   ;WRITE SUBSYSTEM MEM. WITH ACK,CVC=1
7508 047060  012721  044630      MOV     #T38TAD,(R1)+   ;ADDRESS OF DATA BLOCK
7509 047064  005021              CLR     (R1)+           ;EXTENDED ADDRESS
7510 047066  012721  000006      MOV     #6.,(R1)+       ;SIZE OF DATA BLOCK IN BYTES
7511 047072  005021              CLR     (R1)+           ;CLEAR BSEL0 AND BSEL1
7512 047074  005021              CLR     (R1)+           ;CLEAR SEL2
7513 047076  005011              CLR     (R1)            ;CLEAR DATA AREA
7514 047100  000207              RTS     PC              ;RETURN
7515
7516
7517                    ;+
7518                    ;
7519                    ;THIS ROUTINE PRINTS THE CONTENTS OF
7520                    ;THE 256 BYTE MESSAGE BUFFER RETURNED BY THE
7521                    ;TUV-05.
7522                    ;
7523                    ;INPUT:
7524                    ;
7525                    ;       R0      LOW ORDER ADDRESS OF MESSAGE BUFFER
7526                    ;       R1      HIGH ORDER ADDRESS OF MESSAGE BUFFER
7527                    ;       NOTE: R1 IS IGNORED IF KTENABLE FLAG IS CLEAR
```

```
7528                                    ;
7529                                    ;
7530                                    ;-
7531
7532 047102                             T38MBP:
7533 047102                                     SAVREG                  ;SAVE THE REGISTERS
7534 047106  010005                             MOV     R0,R5           ;SAVE LOW ORDER ADDRESS
7535 047 10  005737  003102                     TST     KTENABLE        ;ADDRESS ABOVE 28K?
7536 047.14  001001                             BNE     910$            ;BR IF YES
7537 047116  005001                             CLR     R1              ;SET HIGH ORDER ADDRESS TO 0
7538 047120  010103                     910$:   MOV     R1,R3           ;SAVE HIGH ORDER ADDRESS
7539 047122  006100                             ROL     R0              ;SHIFT BIT15 TO C BIT
7540 047124  006101                             ROL     R1              ;SHIFT TO HIGH ORDER FOR PRINTOUT
7541 047126                                     PRINTX  #T38AS0,R1,R5   ;PRINT MESSAGE BUFFER ADDRESS
     047126  010546                                                             MOV     R5,-(SP)
     047130  010146                                                             MOV     R1,-(SP)
     047132  012746  047404                                                     MOV     #T38AS0,-(SP)
     047136  012746  000003                                                     MOV     #3,-(SP)
     047142  010600                                                             MOV     SP,R0
     047144  104415                                                             TRAP    C$PNTX
     047146  062706  000010                                                     ADD      10,SP
7542 047152                                     PRINTX  #T38AS1         ;PRINT HEADER FOR CONTENTS
     047152  012746  047451                                                     MOV     #T38AS1,-(SP)
     047156  012746  000001                                                     MOV     #1,-(SP)
     047162  010600                                                             MOV     SP,R0
     047164  104415                                                             TRAP    C$PNTX
     047166  062706  000004                                                     ADD     #4,SP
7543 047172  010501                             MOV     R5,R1           ;COPY LOW ORDER ADDRESS
7544 047174  010300                             MOV     R3,R0           ;COPY HIGH ORDER ADDRESS
7545 047176  001403                             BEQ     913$            ;BR IF NOT ABOVE 28K
7546 047200  004737  020270                     JSR     PC,SETMAP       ;SETUP PAR ADDRESS IN R0
7547 047204  010005                             MOV     R0,R5           ;GET PAR FORMAT ADDRESS ABOVE 28K
7548 047206  010537  047552             913$:   MOV     R5,T38CNT       ;HOLD ADDRESS
7549 047212  011504                     911$:   MOV     (R5),R4         ;GET BUFFER ENTRY
7550 047214  022704  125252                     CMP     #125252,R4      ;CHECK FOR NO LOAD CONDITION
7551 047220  001417                             BEQ     912$            ;BR, IF BUFFER WASN'T LOADED
7552 047222  010403                             MOV     R4,R3           ;MAKE COPY
7553 047224  042704  170377                     BIC     #170377,R4      ;ONLY BITS 11,10,9 AND 8 ARE SAVED
7554 047230  000241                             CLC                     ;CLEAR CARRY
7555 047232  006003                             ROR     R4              ;11 TO 10 BIT POSITION
7556 047234  006004                             ROR     R4              ;10 TO 9 BIT POSITION
7557 047236  006004                             ROR     R4              ;9 TO 8 BIT POSITON
7558 047240  006004                             ROR     R4              ;8 TO 7 BIT POSITION
7559 047242  042703  177760                     BIC     #177760,R3      ;ONLY BITS 3,2,1 AND 0 ARE SAVED
7560 047246  060403                             ADD     R4,R3           ;"OR'EM TOGETHER
7561 047250  010325                             MOV     R3,(R5)+        ;PUT BACK IN BUFFER
7562 047252  020527  045312                     CMP     R5,#T38EB       ;END OF BUFFER YET
7563 047256  001355                             BNE     911$            ;BR, IF NOT AT END YET
7564 047260  013705  047552             912$:   MOV     T38CNT,R5       ;PUT ADDRESS BACK
7565 047264  012704  000001                     MOV     #1,R4           ;START BYTE NUMBER AT ONE
7566 047270                             915$:   PRINTX  #T38ASN,R4,(R5)+        ;PRT MEM BUFFER W/NEWLINE
     047270  012546                                                             MOV     (R5)+,-(SP)
     047272  010446                                                             MOV     R4,-(SP)
     047274  012746  047526                                                     MOV     #T38ASN,-(SP)
     047300  012746  000003                                                     MOV     #3,-(SP)
     047304  010600                                                             MOV     SP,R0
     047305  104415                                                             TRAP    C$PNTX
```

```
     047310  062706  000010                                                          ADD     #10,SP
7567 047314  005037  047552          CLR     T38CNT                  ;CLEAR COUNTER
7568 047320  000412                  BR      921$                    ;SKIP OTHER PRINT
7569 047322                  920$:   PRINTX  #T38ASC,R4,(R5)+        ;PRINT THE CONTENTS OF MEMORY BUFFER
     047322  012546                                                          MOV     (R5)+,-(SP)
     047324  010446                                                          MOV     R4,-(SP)
     047326  012746  047507                                                  MOV     #T38ASC,-(SP)
     047332  012746  000003                                                  MOV     #3,-(SP)
     047336  010600                                                          MOV     SP,R0
     047340  104415                                                          TRAP    C$PNTX
     047342  062706  000010                                                  ADD     #10,SP
7570 047346  005237  047552          921$:   INC     T38CNT                  ;BUMP COUNTER
7571 047352  005204                  INC     R4              ;NUMBER OF THE NEXT
7572 047354  020427  000200          CMP     R4,#128.                ;DONE ALL YET ?
7573 047360  003010                  BGT     50$             ;BRANCH IF ALL DONE
7574 047362  023727  047552  000004  CMP     T38CNT,#4               ;DONE FOUR YET
7575 047370  001401                  BEQ     925$                    ;BR. IF THREE DONE
7576 047372  000753                  BR      920$            ;KEEP GOING
7577 047374  005037  047552          925$:   CLR     T38CNT                  ;CLEAR COUNTER
7578 047400  000733                  BR      915$                    ;PRINT WITH NEW LINE
7579 047402  000207                  50$:    RTS     PC              ;RETURN
7580
7581 047404     045     116     045  T38AS0: .ASCIZ  '%N%A Message Buffer Address - %01%05'
7582 047451     045     116     045  T38AS1: .ASCIZ  '%N%A Message Buffer Contents:'
7583 047507     045     101     040  T38ASC: .ASCIZ  '%A  %D4%A: %03'
7584 047526     045     116     045  T38ASN: .ASCIZ  '%N%A Byte%D4%A: %03'
7585                                          .EVEN
7586 047552  000000          T38CNT: .WORD           ;COUNTER FOR PRINT
7587 047554                          ENDTST
     047554                                                  L10064:
     047554  104401                                                  TRAP    C$ETST
```

```
                        .SBTTL  TEST 6: CONFIGURATION TYPEOUT

                ;THIS TEST MUST BE STARTED AS FOLLOWS:
                ;
                ;AT THE DIAGNOSTIC SUPERVISOR PROMPT "DR>"
                ;TYPE "START/FLAG:PNT/TEST:6/PASS:1"
                ;       -------- --------------------
                ;
                ;THIS IS A STANDALONE ROUTINE THAT PRINTS OUT ON THE CONSOLE TERMINAL
                ;THE CONFIGURATION OF THE CONTROLLER MODULE AND TK25 SUBSYSTEM. SPECIFICALLY,
                ;THE FOLLOWING INFORMATION IS PRESENTED:
                ;
                ;
                ; 1.0 MICROCODE REVSION LEVEL OF THE CONTROLLER,
                ;
                ; 2.0 NUMBER OF TAPE TRANSPORTS CONNECTED TO THE CONTROLLER,
                ;
                ; 3.0 UNIT SELECT CODE AND STATE (ONLINE/OFFLINE, WRITE ENABLED/PROTECTED)
                ;     OF EACH CONNECTED TRANSPORT.
                ;
                ;THE OPERATOR IS EXPECTED TO READ THE PRINTOUT AND VERIFY THAT IT MATCHES
                ;THE ACTUAL CONFIGURATION AT HAND.  IF, FOR EXAMPLE, THE PROGRAM INDICATES
                ;THAT IT "SEES" TWO TRANSPORTS CONNECTED WHEN IN FACT ONLY ONE IS PRESENT,
                ;THE OPERATOR MUST INTERPRET THIS AS AN ERROR AND ATTEMPT TO FIND THE
                ;CAUSE (BAD CABLE, FAULTY UNIT-SELECT DECODING IN THE TRANSPORT, ETC.).
                ;(SINCE THE CONTROLLER CAN ONLY ACCESS UNIT 0 IF IT IS IN "STANDARD"
                ;MODE, THE PROGRAM WILL FORCE THE MODULE INTO EXTENDED MODE VIA THE
                ;WRITE SUBSYSTEM MEMORY COMMAND IN ORDER TO SCAN FOR CONNECTED TRANSPORTS.)
                ;
                ;
                ;THIS ROUTINE, WHEN ITS ACTIONS ARE COMPLETED, WILL EXIT BACK TO THE
                ;DIAGNOSTIC SUPERVISOR SO THAT IF ADDITIONAL UNITS (CONTROLLERS) ARE
                ;SELECTED (E.G., FROM THE INITIAL STARTUP DIALOG), THE ROUTINE WILL BE
                ;REENTERED SO THAT THEIR CONFIGURATIONS CAN BE PRINTED.
                ;
                        BGNTST
                                                                                T6::
                        CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
                        CLR     KTFLG           ;HOLD OF KT11
                        TST     TSTCNT                  ;IS THIS FIRST TEST IN SEQUENCE ?
                        BEQ     10$                     ;BR, IF FIRST TEST
                        MOV     #T39NE,R0               ;"TEST NOT EXECUTED"
                        BR      11$                     ;JUMP OUT OF TEST IF NOT
                10$:    MOV     #TST39ID,R0             ;TEST ID MESSAGE
                11$:    JSR     PC,TSTSETUP             ;DO THE COMMON SETUP
                        JSR     PC,CHKMAN               ;IS MANUAL INTERVENTION ALLOWED?
                        BCS     15$                     ;BR, IF MANUAL INTERVENTION ALLOWED
                        JMP     64$                     ;JUMP TO OUT IF NOT
                15$:    CMP     #1,TSTCNT               ;IS THIS THE FIRST TEST IN SEQ
                        BEQ     20$                     ;BR, IF FIRST TEST
                        JMP     64$                     ;JMP IF IT WASN'T
                20$:
                        JSR     PC,SOFINIT              ;DO SOFT INIT OF CONTROLLER
                        BCS     25$                     ;BR IF SOFT INIT = OK
```

```
7652 047646  010001                    MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
7653 047650                             ERRDF   ERRNO,SFIERR,SFIMSG     ;DEVICE FATAL ERROR DURING INIT
     047650  104455                                                             TRAP    C$ERDF
     047652  001131                                                             .WORD   601
     047654  003550                                                             .WORD   SFIERR
     047656  011666                                                             .WORD   SFIMSG
7654 047660                     25$:    CKLOOP                          ;LOOP IF SELECTED
     047660  104406                                                             TRAP    C$CLP1
7655 047662  012704  050730            MOV     #T39PK2,R4              ;SUBROUTINE NEEDS PACKET ADDRESS
7656 047666  004737  010332            JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
7657 047672  103405                    BCS     50$                     ;BR, IF COMMAND ISSUED OK
7661 047674  010001                    MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
7662 047676                             ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     047676  104456                                                             TRAP    C$ERHRD
     047700  001132                                                             .WORD   602
     047702  004754                                                             .WORD   WRTMSG
     047704  011666                                                             .WORD   SFIMSG
7663 047706                     50$:    CKLOOP                          ;LOOP IF SELECTED
     047706  104406                                                             TRAP    C$CLP1
7664 047710  013701  050252    130$:   MOV     T39BFR+4,R1             ;PICK UP THE RES. BYTE CNTR AREA
7665 047714  042701  177700            BIC     #177700,R1              ;ONLY LEAVE MICROCODE REV LEVEL
7666 047720  010137  051406            MOV     R1,T39RL                ;LOAD UP REV LEVEL
7667 047724                             PRINTX  #T39MCL,T39RL           ;"MICROCODE REVISION LEVEL =000XXX"
     047724  013746  051406                                                     MOV     T39RL,-(SP)
     047730  012746  051323                                                     MOV     #T39MCL,-(SP)
     047734  012746  000002                                                     MOV     #2,-(SP)
     047740  010600                                                             MOV     SP,R0
     047742  104415                                                             TRAP    C$PNTX
     047744  062706  000006                                                     ADD     #6,SP
7668 047750  004737  016644            JSR     PC,SOFINIT              ;DO SOFT INIT OF CONTROLLER
7669 047754  103405                    BCS     140$                    ;BR IF SOFT INIT = OK
7673 047756  010001                    MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
7674 047760                             ERRDF   ERRNO,SFIERR,SFIMSG     ;DEVICE FATAL ERROR DURING INIT
     047760  104455                                                             TRAP    C$ERDF
     047762  001133                                                             .WORD   603
     047764  003550                                                             .WORD   SFIERR
     047766  011666                                                             .WORD   SFIMSG
7675 047770                     140$:   CKLOOP                          ;LOOP IF SELECTED
     047770  104406                                                             TRAP    C$CLP1
7676 047772  012704  050730            MOV     #T39PK2,R4              ;SUBROUTINE NEEDS PACKET ADDRESS
7677 047776  004737  010332            JSR     PC,WRTCHR               ;ISSUE WRITE CHARACTERISTICS
7678 050002  103405                    BCS     150$                    ;BR, IF COMMAND ISSUED OK
7682 050004  010001                    MOV     R0,R1                   ;SAVE CONTENTS OF TSSR
7683 050006                             ERRHRD  ERRNO,WRTMSG,SFIMSG     ;WRITE CHARACTERISTISC FAILED
     050006  104456                                                             TRAP    C$ERHRD
     050010  001134                                                             .WORD   604
     050012  004754                                                             .WORD   WRTMSG
     050014  011666                                                             .WORD   SFIMSG
7684 050016                     150$:   CKLOOP                          ;LOOP IF SELECTED
     050016  104406                                                             TRAP    C$CLP1
7685 050020  005037  002150            CLR     UNITN                   ;SET TO DRIVE 0
7686
7687 050024  016501  000000    190$:   MOV     TSSR(R5),R1             ;GET TSSR STATUS
7688 050030  032701  000100            BIT     #OFL,R1                 ;CHECK FOR OFF-LINE
7689 050034  001414                    BEQ     200$                    ;BR, IF DRIVE IS ON-LINE
7690 050036                             PRINTX  #T390F2,UNITN           ;"DRIVE NUMBER XX IS OFF LINE"
     050036  013746  002150                                                     MOV     UNITN,-(SP)
```

```
        050042  012746  051062                                                  MOV     #T390F2,-(SP)
        050046  012746  000002                                                  MOV     #2,-(SP)
        050052  010600                                                          MOV     SP,R0
        050054  104415                                                          TRAP    C$PNTX
        050056  062706  000006                                                  ADD     #6,SP
7691    050062  000137  050200          JMP     63$                    ;DO NOT TRY TO GET ANYMORE INFO.
7692    050066                  200$:   PRINTX  #T390N2,UNITN          ;"DRIVE NUMBER XX IS ON-LINE"
        050066  013746  002150                                                  MOV     UNITN,-(SP)
        050072  012746  051126                                                  MOV     #T390N2,-(SP)
        050076  012746  000002                                                  MOV     #2,-(SP)
        050102  010600                                                          MOV     SP,R0
        050104  104415                                                          TRAP    C$PNTX
        050106  062706  000006                                                  ADD     #6,SP
7693    050112  013701  050254          MOV     T39BFR+6,R1            ;READ EXTENDED STATUS (XSTO)
7694    050116  032701  000004          BIT     #BIT2,R1               ;IS DRIVE WTITE PROTECTED
7695    050122  001013                  BNE     210$                   ;BR, IF WRITE PROTECTED
7696    050124                          PRINTX  #T39WPN,UNITN          ;"DRIVE NUMBER IS NOT WRT PRO"
        050124  013746  002150                                                  MOV     UNITN,-(SP)
        050130  012746  051244                                                  MOV     #T39WPN,-(SP)
        050134  012746  000002                                                  MOV     #2,-(SP)
        050140  010600                                                          MOV     SP,R0
        050142  104415                                                          TRAP    C$PNTX
        050144  062706  000006                                                  ADD     #6,SP
7697    050150  000413                  BR      63$                    ;SKIP OVER
7698    050152                  210$:   PRINTX  #T39WRT,UNITN          ;"DRIVE NUMBER XX IS WRT PRO"
        050152  013746  002150                                                  MOV     UNITN,-(SP)
        050156  012746  051171                                                  MOV     #T39WRT,-(SP)
        050162  012746  000002                                                  MOV     #2,-(SP)
        050166  010600                                                          MOV     SP,R0
        050170  104415                                                          TRAP    C$PNTX
        050172  062706  000006                                                  ADD     #6,SP
7699    050176  000400                  BR      63$                    ;BR, IF NO MORE DRIVES
7700    050200                  63$:    PRINTX  #T39NFL                        ;NEW LINE
        050200  012746  051000                                                  MOV     #T39NFL,-(SP)
        050204  012746  000001                                                  MOV     #1,-(SP)
        050210  010600                                                          MOV     SP,R0
        050212  104415                                                          TRAP    C$PNTX
        050214  062706  000004                                                  ADD     #4,SP
7701    050220                  64$:    EXIT    TST                    ;EXIT THIS SECTION
        050220  104432                                                          TRAP    C$EXIT
        050222  001254                                                          .WORD   L10065-.
7702                                    ;+
7703                                    ;LOCAL TEXT MESSAGES FOR TEST
7704                                    ;-
7705
7706                                    ;LOCAL STORAGE FOR THIS TEST
7707                                    ;-
7708
7709    050224  000000          T39DLY: .WORD   0                              ;DELAY COUNTER FOR TEST
7711    050226                          .BLKB   10.-<.-TUV2A67>
7713    050230                  T39PACKET:                             ;COMMAND PACKET FOR TEST
7714    050230  140006                  .WORD   140006                 ;WRITE SUBSYSTEM MEM. CMD.  ACK,CVC+1
7715    050232  050240                  .WORD   T39TAD                 ;ADDRESS OF CHARACTERISTICS BLOCK
7716    050234  000000                  .WORD   0
7717    050236  000012                  .WORD   10.                    ;STARTING VALUE OF BLOCK SIZE
7718    050240                  T39TAD:                        ;CHARACTERISTICS DATA BLOCK
7719    050240     000          T39BSO: .BYTE   0                      ;BSELO BYTE
```

```
7720 050241      000                T39BS1: .BYTE   0                       ;BSEL1 BYTE
7721 050242  000000                T39BS2: .WORD   0                       ;BSEL1 WORD
7722 050244  000000                        .WORD   0                       ;DATA
7723 050246                        T39BFR: .BLKW   150.                    ;MESSAGE BUFFER
7724
7725
7727 050722                                .BLKB   10-<.-TUV2A&7>
7729 050730                        T39PK2:                                 ;COMMAND PACKET FOR TEST
7730 050730  140004                        .WORD   140004                  ;WRITE CHARA. MEM. CMND.,  ACK,CVC=1
7731 050732  050740                        .WORD   T39DTA                  ;ADDRESS OF SELECT DATA BLOCK
7732 050734  000000                        .WORD   0
7733 050736  000012                        .WORD   10.                     ;STARTING VALUE OF BLOCK SIZE
7734
7735
7736 050740                        T39DTA:                                 ;SELECT DATA BLOCK
7737 050740  050246                        .WORD   T39BFR                  ;ADDRESS OF MESSAGE BUFFER
7738 050742  000000                        .WORD   0
7739 050744  000400                        .WORD   256.                    ;LENGTH OF MESSAGE BUFFER
7740 050746  000000                T39EAI: .WORD   0                       ;EAI BIT WORD
7742 050750                                .BLKB   10-<.-TUV2A&7>
7744 050760  140012                T39PK3: .WORD   140012                  ;MESSAGE BUFFER RELEASE COMMAND
7745 050762  000000                        .WORD   0                       ;NOT USED
7746                                   ;
7747                                   ;WRITE TAPE PACKET
7748                                   ;
7750 050764                                .BLKB   10-<.-TUV2A&7>
7752 050770  140005                T39PK4: .WORD   140005                  ;WRITE, ACK, CVC=1 COMMAND
7753 050772  000000                T39WR:  .WORD   0                       ;ADDRESS OF WRITE BUFFER
7754 050774  000000                        .WORD   0                       ;MORE ADDRESS OF WRITE BUFFER
7755 050776  000400                T39SIZ: .WORD   256.                    ;SIZE OF RECORD
7756
7757
7758
7759
7760                                   ;+
7761                                   ;LOCAL TEXT MESSAGES FOR TEST
7762                                   ;-
7763
7764
7765
7766
7767 051000      045      116      000  T39NFL: .ASCIZ  '%N'
7768 051003      123      164      141  T39NE:  .ASCIZ  'Stand-alone Configuration Typeout Not Executed'
7769 051062      045      116      045  T39OF2: .ASCIZ  '%N%A Drive Number %D2%A Is Off-Line'
7770 051126      045      116      045  T39ON2: .ASCIZ  '%N%A Drive Number %D2%A Is On-Line'
7771 051171      045      116      045  T39WRT: .ASCIZ  '%N%A Drive Number %D2%A Is Write Protected'
7772 051244      045      116      045  T39WPN: .ASCIZ  '%N%A Drive Number %D2%A Is NOT Write Protected'
7773 051323      045      116      045  T39MCL: .ASCIZ  '%N%A Controller Microcode Revision Level     =%02'
7774                                            .EVEN
7775 051406  000000                T39RL:  .WORD   0
7776                                            .EVEN
7777                                            .EVEN
7778
7779                                   ;+
7780                                   ;LOCAL STORAGE FOR THIS TEST
7781                                   ;-
7782
```

```
7783 051410  000000                T39DAT: .WORD   0               ;LOGICAL RESPONSE TO QUESTION
7784 051412                        T39REST:
7785 051412                                SAVREG                  ;SAVE THE REGISTERS
7786 051416  012701  050230                MOV     #T39PACKET,R1   ;START OF THE PACKET
7787 051422  012721  140006                MOV     #140006,(R1)+   ;WRITE SUBSYSTEM MEM. WITH ACK,CVC=1
7788 051426  012721  050240                MOV     #T39TAD,(R1)+   ;ADDRESS OF DATA BLOCK
7789 051432  005021                        CLR     (R1)+           ;EXTENDED ADDRESS
7790 051434  012721  000006                MOV     #6.,(R1)+       ;SIZE OF DATA BLOCK IN BYTES
7791 051440  005021                        CLR     (R1)+           ;CLEAR BSEL0 AND BSEL1
7792 051442  005021                        CLR     (R1)+           ;CLEAR SEL2
7793 051444  005011                        CLR     (R1)            ;CLEAR DATA AREA
7794 051446  000207                        RTS     PC              ;RETURN
7795
7796                               ;+
7797                               ;LOCAL TEXT MESSAGES FOR TEST
7798                               ;-
7799
7800 051450     103     157     156  TST39ID:        .ASCIZ  'Configuration Typeout'
7801                                        .EVEN
7802 051476                                ENDTST
     051476                                                          L10065:
     051476  104401                                                        TRAP    C$ETST
7803
```

```
                        .SBTTL  TEST 7: SCOPE LOOPS

;+
;THIS TEST MUST BE STARTED AS FOLLOWS:
;
;AT THE DIAGNOSTIC SUPERVISOR PROMPT "DR>"
;TYPE "START/FLAG:PNT/TEST:7/PASS:1"
;       ----------------------------
;
;
;
;         THIS IS A STANDALONE ROUTINE PROVIDING A NUMBER OF TIGHT "SCOPE
;         LOOPS" USEFUL FOR DEBUGGING BASIC REGISTER ACCESS PROBLEMS WITH
;         THE CONTROLLER MODULE.  THESE SCOPE LOOPS CAN BE USED WHEN THE NORMAL
;         "LOOP ON ERROR" OR "LOOP ON TEST (SUBTEST)" FACILITIES DON'T
;         SEEM TO ALLOW THE OPERATOR TO ZERO IN A PROBLEM IN THE EARLY
;         TESTS (I.E. THE HARDWARE MAY NOT BE RESPONDING TO A REGISTER
;         ACCESS, CAUSING A BUS ERROR TRAP, EVEN THOUGH THE DEVICE ADDRESS
;         SELECTED BY THE PROGRAM MATCHES THE CONFIGURATION SET UP IN THE
;         HARDWARE DIP SWITCHES).  THE FOLLOWING MENU OF SCOPE LOOPS ARE
;         AVAILABLE:
;
;                 CODE    SCOPE LOOP
;                 0       HELP. DISPLAY THIS MENU.
;                 1       TSBA READ ACCESS
;                 2       TSSR READ ACCESS
;                 3       INITIALIZE (TSSR WRITE ACCESS)
;                 4       TSDB HIGH BYTE WRITE ACCESS
;                 5       TSDB LOW BYTE WRITE ACCESS
;                 6       TSSR BYTE WRITE (SELF-TEST)
;                 7       RETURN TO DIAGNOSTIC SUPERVISOR
;
;        FOR SCOPE LOOPS THAT WRITE INTO REGISTERS, THE PROGRAM PROMPTS
;        THE OPERATOR FOR THE DATA TO BE WRITTEN.  TYPING <RETURN> CAUSES
;        AN EXIT FROM THE SCOPE LOOP BACK TO MENU LEVEL.
;
;-


051500                                  BGNTST
051500                                                          T7::
051500  005037  002170                  CLR     FATFLG                  ;CLEAR FATAL ERROR FLAG
051504  005037  003100                  CLR     KTFLG           ;HOLD OF KT11
051510  005737  002162                  TST     TSTCNT                  ;IS THIS FIRST TEST IN RUN ?
051514  001403                          BEQ     1$                      ;CONTINUE TEST IF FIRST IN RUN
051516  012700  052711                  MOV     #T4ONE,R0               ;"TEST NOT EXECUTED"
051522  000402                          BR      100$                    ;JUST EXIT IF NOT
051524  012700  052756          1$:     MOV     #TST4OID,R0             ;TEST ID MESSAGE
051530  004737  017410          100$:   JSR     PC,TSTSETUP             ;DO THE COMMON SETUP
051534  004737  021404                  JSR     PC,CHKMAN               ;SEE IF MANUAL INTERVENTION ALLOWED
051540  103402                          BCS     102$                    ;CARRY SET IF INTERVENTION ALLOWED
051542  000137  052174                  JMP     64$                     ;EXIT IF NO MANUAL INTERVENTION
051546  022737  000001  002162  102$:   CMP     #1,TSTCNT               ;WAS THIS THE FIRST TEST IN SEQ
051554  001402                          BEQ     2$                      ;BR, IF IT WAS
051556  000137  052174                  JMP     64$                     ;JMP TO END OF TEST
```

```
7865 051562  004737  016644          2$:     JSR     PC,SOFINIT              ;DO A SOFT INIT
7866 051566  103405                          BCS     5$                      ;BRANCH IF OK
7867 051570  010001                          MOV     R0,R1                   ;CONTENTS OF TSSR REGISTER
7871 051572                                  ERRDF   ERRNO,SFIERR,SFIMSG     ;REPORT FATAL ERROR
     051572  104455                                                          TRAP    C$ERDF
     051574  001275                                                          .WORD   701
     051576  003550                                                          .WORD   SFIERR
     051600  011666                                                          .WORD   SFIMSG
7872 051602  012700  052206          5$:     MOV     #SCMENU,R0              ;MENU OF SCOPE LOOP SELECTIONS
7873 051606  012701  000010                  MOV     #8.,R1                  ;MAXIMUM ALLOWED SELECTION
7874 051612  004737  021162                  JSR     PC,GETSEL               ;GO GET THE OPERATORS SELECTION
7875 051616  005700                          TST     R0                      ;WAS ZERO SPECIFIED ?
7876 051620  001760                          BEQ     2$                      ;REPEAT MENU IF YES.
7877 051622  010004          3$:     MOV     R0,R4                   ;SAVE THE MENU SELECTION
7878 051624                                  SETPRI  #PRI07                  ;RAISE THE PRIORITY
     051624  012700  000340                                                  MOV     #PRI07,R0
     051630  104441                                                          TRAP    C$SPRI
7879 051632  005037  052200                  CLR     TTION                   ;ASSUME INTERRUPTS ARE ENABLED
7880 051636  032737  000100  177560          BIT     #100,@#TTICSR           ;ARE TTI INTERRUPTS ON ?
7881 051644  001005                          BNE     4$                      ;BRANCH IF YES
7882 051646  005237  052200                  INC     TTION                   ;FLAG SET IF INTERRUPTS OFF
7883 051652  052737  000100  177560          BIS     #100,@#TTICSR           ;ENABLE INTERRUPTS
7884 051660  012701  000060          4$:     MOV     #TTIVEC,R1              ;START OF TTI VECTORS
7885 051664  011137  052202                  MOV     (R1),TVECSAV            ;SAVE THE CURRENT TTI VECTOR
7886 051670  012721  052106                  MOV     #60$,(R1)+              ;SET NEW INTERRUPT ROUTINE
7887 051674  011137  052204                  MOV     (R1),TPRISAV            ;SAVE THE VECTOR PRIORITY
7888 051700  012711  000340                  MOV     #PRI07,(R1)             ;USE PRIORITY SEVEN
7889 051704                                  SETPRI  #PRI00                  ;LOWER INTERRUPT BR LEVEL
     051704  012700  000000                                                  MOV     #PRI00,R0
     051710  104441                                                          TRAP    C$SPRI
7890 051712  006304                          ASL     R4                      ;CONVERT TO WORD OFFSET
7891 051714  000174  051720                  JMP     @6$(R4)                 ;JUMP TO PROPER LOOP
7892 051720  051562          6$:     .WORD   2$                      ;RETYPE THE MENU
7893 051722  051740                  .WORD   10$                     ;TSBA READ ACCESS
7894 051724  051746                  .WORD   15$                     ;TSSR READ ACCESS
7895 051726  051754                  .WORD   20$                     ;TSSR WRITE ACCESS
7896 051730  051770                  .WORD   25$                     ;TSDB HIGH BYTE WRITE ACCESS
7897 051732  052004                  .WORD   30$                     ;TSDB LOW BYTE WRITE ACCESS
7898 051734  052020                  .WORD   35$                     ;TSSR BYTE WRITE (SELF-TEST)
7899 051736  052174                  .WORD   65$                     ;LEAVE THE TEST
7900
7901
7902 051740                  10$:
7903 051740  016500  177776  12$:    MOV     TSBA(R5),R0             ;READ TSBA REGISTER
7904 051744  000775                  BR      12$                     ;LOOP UNTIL HALTED
7905
7906 051746                  15$:
7907 051746  016500  000000  18$:    MOV     TSSR(R5),R0             ;READ TSSR REGISTER
7908 051752  000775                  BR      18$                     ;LOOP UNTIL STOPPED
7909
7910
7911 051754  004737  021100  20$:    JSR     PC,GETPAT               ;READ THE DATA PATTERN
7912 051760  010001                  MOV     R0,R1                   ;DATA PATTERN FOR LOOP
7913 051762  010165  000000  22$:    MOV     R1,TSSR(R5)             ;WRITE DATA TO TSSR
7914 051766  000775                  BR      22$                     ;LOOP
7915
7916
```

```
7917 051770                          25$:
7918 051770  004737  021100                  JSR     PC,GETPAT           ;READ THE DATA PATTERN
7919 051774  010001                          MOV     R0,R1               ;DATA PATTERN FOR LOOP
7920 051776  110165  177777          27$:    MOVB    R1,TSDBH(R5)        ;WRITE THE DATA TO TSDB, HIGH BYTE
7921 052002  000775                          BR      27$                 ;LOOP UNTIL STOPPED
7922
7923
7924 052004                          30$:
7925 052004  004737  021100                  JSR     PC,GETPAT           ;READ THE DATA PATTERN
7926 052010  010001                          MOV     R0,R1               ;DATA PATTERN FOR LOOP
7927 052012  110165  177776          32$:    MOVB    R1,TSDB(R5)         ;WRITE DATA TO TSSR, LOW BYTE
7928 052016  000775                          BR      32$                 ;LOOP UNTIL HALTED BY OPERATOR
7929
7930 052020  004737  021100          35$:    JSR     PC,GETPAT           ;PICK UP THE PATTERN FROM OPR
7931 052024  010001                          MOV     R0,R1               ;STORE IN R1
7932 052026  110165  000000          37$:    MOVB    R1,TSSR(R5)         ;WRITE BYTE TO TSSR, THIS STARTS MDIAG
7933 052032                                  DELAY   250                 ;WAIT 2500US
     052032  012727  000250                                                      MOV     #250,(PC)+
     052036  000000                                                              .WORD   0
     052040  013727  002116                                                      MOV     L$DLY,(PC)+
     052044  000000                                                              .WORD   0
     052046  005367  177772                                                      DEC     -6(PC)
     052052  001375                                                              BNE     .-4
     052054  005367  177756                                                      DEC     -22(PC)
     052060  001367                                                              BNE     .-20
7934 052062  036527  000000  000200          BIT     TSSR(R5),#SSR       ;CHECK FOR READY SET, IF A TRAP OCCURS
7935                                                                     ;THE TSSR WAS "NOT BACK TO THE BUS"
7936                                                                     ;IN TIME.
7937 052070  001356                          BNE     37$                 ;BR, IF SSR WAS SET (GOOD) KEEP LOOPING
7941 052072                                  ERRDF   ERRNO,T40NSR,SFIMSG ;PRINT DEVICE FATAL ERROR MESSAGE
     052072  104455                                                              TRAP    C$ERDF
     052074  001276                                                              .WORD   702
     052076  052772                                                              .WORD   T40NSR
     052100  011666                                                              .WORD   SFIMSG
7942 052102  000137  000200                  JMP     200                 ;GO TO SUPERVISOR ETC.
7943
7944
7945                                 ;+
7946                                 ;PROCESS CONSOLE INTERRUPTS
7947                                 ;-
7948
7949 052106  010046                  60$:    MOV     R0,-(SP)            ;SAVE WORK REGISTER
7950 052110  113700  177562                  MOVB    @#TTIBFR,R0         ;GET THE OPERATOR INPUT
7951 052114  042700  000200                  BIC     #200,R0             ;STRIP OFF PARITY BIT
7952 052120  122700  000015                  CMPB    #15,R0              ;IS IT A CARRIAGE RETURN ?
7953 052124  001021                          BNE     61$                 ;JUST EXIT IF NOT
7954 052126  012766  051562  000002          MOV     #2$,2(SP)           ;RETURN TO MASTER MENU
7955 052134  005066  000004                  CLR     4(SP)               ;FORCE PRIORITY ZERO
7956 052140  013737  052202  000060          MOV     IVECSAV,@#TTIVEC    ;RESTORE SUPERVISOR VECTOR
7957 052146  013737  052204  000062          MOV     IPRISAV,@#TTIVEC+2  ;RESTORE SUPERVISOR PRIORITY
7958 052154  005737  052200                  TST     ITION               ;ARE SUPERVISOR INTERRUPTS ENABLED ?
7959 052160  001403                          BEQ     61$                 ;BRANCH IF YES
7960 052162  042737  000100  177560          BIC     #100,@#TTICSR       ;TURN OFF TTI INTERRUPTS
7961 052170  012600                  61$:    MOV     (SP)+,R0            ;RESTORE REGISTER
7962 052172  000002                          RTI                         ;RETURN FROM INTERUPT
7963
7964 05217A                          64$:
```

```
                                63$:
                                65$:    EXIT    TST                     ;EXIT THE TEST
052174  104432                                                          TRAP    C$EXIT
052176  000664                                                          .WORD   L10066-.

                                ;+
                                ;LOCAL STORAGE FOR THIS TEST
                                ;-

052200  000000                  TTION:          .WORD   0               ;WORD SET IF SUPERVISOR TTI INTER OFF
052202  000000                  TVECSAV:        .WORD   0               ;SAVE TTI VECTOR
052204  000000                  TPRISAV:        .WORD   0               ;SAVE TTI PRIORITY


                                ;+
                                ;MENU FOR OPERATOR INPUT FOR SCOPE LOOPS
                                ;-

                                        .EVEN
052206  052236  052311  052337  SCMENU: .WORD   1$,2$,3$,4$,5$,6$
052222  052510  052546  052605          .WORD   7$,8$,10$,11$,12$,0

052236      012     123     105 1$:     .ASCIZ  <12>'SELECT SCOPE LOOP FROM FOLLOWING OPTIONS:'
052311      012     011     060 2$:     .ASCIZ  <12>'    0      Display This Menu'
052337      011     061     011 3$:     .ASCIZ  '       1       TSBA Read Access'
052363      011     062     011 4$:     .ASCIZ  '       2       TSSR Read Access'
052407      011     063     011 5$:     .ASCIZ  '       3       Initialize (TSSR Write Access)'
052451      011     064     011 6$:     .ASCIZ  '       4       TSDB High Byte Write Access'
052510      011     065     011 7$:     .ASCIZ  '       5       TSDB Low Byte Write Access'
052546      011     066     011 8$:     .ASCIZ  '       6       TSSR Write Byte (Self Test)'
052605      011     067     011 10$:    .ASCIZ  '       7       Return to Diagnostic Supervisor'
052650      000             011 11$:    .ASCIZ  ''
052651      124     171     160 12$:    .ASCIZ  'Type RETURN To Stop Scope Loops'
052711      123     164     141 T40NE:  .ASCIZ  'Stand-alone Scope Loops Not Executed'
052756      123     143     157 TST40ID:        .ASCIZ  'Scope Loops'
052772      123     123     122 T40NSR: .ASCIZ  'SSR Failed To Set After TSSR Write Byte And 10ms Delay'
                                .EVEN
                                ENDTST
053062                                                                  L10066:
053062  104401                                                                  TRAP    C$ETST


                                .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.
                                ;--
                                        BGNHRD
```

```
      053064  000015                          .WORD L10067-L$HARD/2
      053066                          ' $HARD::
 8032
 8033 053066                                  GPRMA   HPM1,0,0,160000,177776,YES      ;GET TSBA/TSDB REGISTER ADDRESS.
      053066  000031                          .WORD   T$CODE
      053070  053120                          .WORD   HPM1
      053072  160000                          .WORD   T$LOLIM
      053074  177776                          .WORD   T$HILIM
 8034 053076                                  GPRMA   HPM2,2,0,0,776,YES             ;GET VECTOR ADDRESS.
      053076  001031                          .WORD   T$CODE
      053100  053147                          .WORD   HPM2
      053102  000000                          .WORD   T$LOLIM
      053104  000776                          .WORD   T$HILIM
 8035 053106                                  GPRMD   HPM3,4,0,340,0,7,YES           ;GET INTERRUPT PRIORITY.
      053106  002032                          .WORD   T$CODE
      053110  053173                          .WORD   HPM3
      053112  000340                          .WORD   340
      053114  000000                          .WORD   T$LOLIM
      053116  000007                          .WORD   T$HILIM
 8036 053120                                  ENDHRD
                                              .EVEN
      053120                          L10067:
 8037 053120      104     105     126 HPM1:   .ASCIZ  'DEVICE ADDRESS (TSSR) '
 8038 053147      111     116     124 HPM2:   .ASCIZ  'INTERRUPT VECTOR   '
 8039 053173      111     116     124 HPM3:   .ASCIZ  'INTERRUPT PRIORITY     '
 8040                                         .EVEN
 8041
```

```
                                        .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.
                                ;--
053224                                  BGNSFT
053224  000006                          .WORD L10070-L$SOFT/2
053226                          L$SOFT::
053226                                  GPRML   SPM1,0,-1,YES                   ;GET RAM DUMP FLAG
053226  000130                          .WORD   T$CODE
053230  053242                          .WORD   SPM1
053232  177777                          .WORD   -1
053234                                  GPRML   SPM4,2,-1,YES                   ; GET ITERATION CONTROL.
053234  001130                          .WORD   T$CODE
053236  053306                          .WORD   SPM4
053240  177777                          .WORD   -1
                                ;       GPRMD   SPM6,4,D,7777,0,7777,YES        ; GET LOCAL ERROR LIMIT
                                ;       GPRMD   SPM7,6,D,7777,0,7777,YES        ; GET GLOBAL ERROR LIMIT
053242                                  ENDSFT
                                        .EVEN
053242                          L10070:

053242  105  116  101  SPM1:    .ASCIZ  'ENABLE CONTROLLER RAM DUMP ON ERROR'
053306  111  116  110  SPM4:    .ASCIZ  'INHIBIT ITERATIONS     '
053336  120  105  122  SPM6:    .ASCIZ  'PER TEST ERROR LIMIT   '
053366  120  105  122  SPM7:    .ASCIZ  'PER UNIT ERROR LIMIT   '
                                        .EVEN
                                        .SBTTL  PATCH AREA
                                ;+
                                ;DISPATCH TABLE
                                ;
                                ; *** MOVE TO FRONT OF PROGRAM FOR RELEASE ***
                                ;-

053416                                  DISPATCH        TESTNO
053416  000007                          .WORD   7
053420                          L$DISPATCH::
053420  023634                          .WORD   T1
053422  024674                          .WORD   T2
053424  026226                          .WORD   T3
053426  032160                          .WORD   T4
053430  043426                          .WORD   T5
053432  047556                          .WORD   T6
053434  051500                          .WORD   T7

                                ;
                                ; FINALLY A GENEROUS PATCH AREA.
                                ;
                                ; AND AN ADJUSTMENT TO ACCOUNT FOR THE "LASTAD BIT7" HACK
                                ; DESCRIBED IN "SUPPRG.MEM" (FOR REV C).
                                ;
```

PATCH AREA

```
8081
8082 053436                          PATCH::
8083                                         .EVEN
8084                                 ;       .IF     NZ,.&377
8085                                 ;       .=.!377+1
3086                                 ;       .ENDC
8087 053436                                  LASTAD          ;SET LAST USED ADDRESS.
                                             .EVEN
     053436  053454                          .WORD T$FREE
     053440  000005                          .WORD T$SIZE
     053442                          L$LAST::
8088                                         .SBTTL  HARD CODED P-TABLE
8089                                 ;++
8090                                 ;       DIAGNOSTIC IS PRE-PARAMETERIZED PER THIS TABLE
8091                                 ;--
8092 053442                                  BGNSETUP        1
8093 053442                                  BGNPTAB
     053442  000000                          .WORD   0
     053444  000003                          .WORD   L10073-./2-1
     053446                          L10071:
8094 053446  172522                          .WORD           172522
8095 053450  000224                          .WORD           224
8096 053452  000240                          .WORD           PRI05
8097 053454                                  ENDPTAB
     053454                          L10073:
8098 053454                                  ENDSETUP
8099
8100         000001                          .END
```

SYMBOL TABLE

```
ADDSSR  011772 G      C$AU  = 000052      DEBUGM  011464        FATFLG  002170 G      HIMEM = 007776
ADR   = 000020 G      C$AUTO= 000061      DEVCNT  002166 G      FERCM   011554        HOE   = 100000 G
AMBTSS  006337        C$BRK = 000022      DEVDRO  023564        FIFEXP  012022 G      HPM1    053120
ASSEMB= 000010        C$BSEG= 000004      DEVNRD  023503        FIF1MS  012074        HPM2    053147
A1716 = 000003        C$BSUB= 000002      DEVNXR  023421        FIF2MS  012143        HPM3    053173
BADDAT  003110 G      C$CEFG= 000045      DEVONL  023351        FILLME  020374        IBE   = 010000 G
BADSSR  016550 G      C$CLCK= 000062      DEVSUM  023314        FNOINT  004113        IDU   = 000040 G
BAR   = 174402        C$CLEA= 000012      UFPTBL  002124 G      FORCER  002144 G      IER   = 020000 G
BENBSW  002174 G      C$CLOS= 000035      DIAGMC= 000000        FREE    003072 G      IFAULT  004154
BIE   = 040000        C$CLP1= 000006      DLCYL = 000177        FREEHI  003076        INCERR  017744
BIT0  = 000001 G      C$CVEC= 000036      DLDNER= 100200        FRESIZ  003074 G      INTCPC  017020
BIT00 = 000001 G      C$DCLN= 000044      DLERR = 177730        FUSI    004015        INTFLA  017015
BIT01 = 000002 G      C$DODU= 000051      DLGETS= 000004        F$AU  = 000015        INTMAS  017014
BIT02 = 000004 G      C$DRPT= 000024      DLRDMD= 000010        F$AUTO= 000020        INTR    017066 G
BIT03 = 000010 G      C$DU  = 000053      DLRDNM= 000016        F$BGN = 000040        INTREC  002172 G
BIT04 = 000020 G      C$EDIT= 000003      DLSR  = 000013        F$CLEA= 000007        INTVEC  017016
BIT05 = 000040 G      C$ERDF= 000055      DLUN  = 000006        F$DU  = 000016        INTX    004176
BIT06 = 000100 G      C$ERHR= 000056      DSBINT  017054        F$END = 000041        IOKCKI= 000200
BIT07 = 000200 G      C$ERRO= 000060      DUAD12  004541        F$HARD= 000004        IOKSTP= 000001
BIT08 = 000400 G      C$ERSF= 000054      DUFLG   003060 G      F$HW  = 000013        IPRI    002160 G
BIT09 = 001000 G      C$ERSO= 000057      DUMMY   003030        F$INIT= 000006        ISR   = 000100 G
BIT1  = 000002 G      C$ESCA= 000010      EF.CON= 000036 G      F$JMP = 000050        IVEC    002156 G
BIT10 = 002000 G      C$ESEG= 000005      EF.NEW= 000035 G      F$MOD = 000000        IXE   = 004000 G
BIT11 = 004000 G      C$ESUB= 000003      EF.PWR= 000034 G      F$MSG = 000011        I$AU  = 000041
BIT12 = 010000 G      C$ETST= 000001      EF.RES= 000037 G      F$PROT= 000021        I$AUTO= 000041
BIT13 = 020000 G      C$EXIT= 000032      EF.STA= 000040 G      F$PWR = 000017        I$CLN = 000041
BIT14 = 040000 G      C$GETB= 000026      EMAXDU  017677        F$RPT = 000012        I$DU  = 000041
BIT15 = 100000 G      C$GETW= 000027      EN    = 000000        F$SEG = 000003        I$HRD = 000041
BIT2  = 000004 G      C$GMAN= 000043      ENAINT  017022        F$SOFT= 000005        I$INIT= 000041
BIT3  = 000010 G      C$GPHR= 000042      ENVIRN  021534        F$SRV = 000010        I$MOD = 000040
BIT4  = 000020 G      C$GPLO= 000030      EPRTSW  002146 G      F$SUB = 000002        I$MSG = 000041
BIT5  = 000040 G      C$GPRI= 000040      EPRT1   005762        F$SW  = 000014        I$PROT= 000040
BIT6  = 000100 G      C$INIT= 000011      EPRT2   005672        F$TEST= 000001        I$PTAB= 000041
BIT7  = 000200 G      C$INLP= 000020      EPRT3   006026        GDDAT   003112 G      I$PWR = 000041
BIT8  = 000400 G      C$MAN1= 000050      ERCM    011565        GERRMA  002142 G      I$RPT = 000041
BIT9  = 001000 G      C$MEM = 000031      ERRHI   002202 G      GETPAT  021100 G      I$SEG = 000041
BOE   = 000400 G      C$MSG = 000023      ERRK    017656        GETSEL  021162 G      I$SETU= 000041
BRINIT  004355        C$OPEN= 000034      ERRLO   002204 G      G$CNTO= 000200        I$SFT = 000041
BSEL0 = 000000        C$PNTP= 000014      ERRNO = 001276        G$DELM= 000372        I$SRV = 000041
BSEL1 = 000001        C$PNTF= 000017      ERRVEC= 000004 G      G$DISP  000003        I$SUB = 000041
CHKAMB  016714        C$PNTS= 000016      ERTABE  003330        G$EXCP  000400        I$TST = 000041
CHKMAN  021404 G      C$PNTX= 000015      ERTABL  003130        G$HILI= 000002        JUMP  = 000167
CHKTSS  017236        C$QIO = 000377      ESUM    017660        G$LOLI= 000001        KIPAR0= 172340
CKDROP  020154        C$RDBU= 000007      EVL   = 000004 G      G$NO  = 000000        KIPAR1= 172342
CKEMAX  020002        C$REFG= 000047      EXBCNT= 000010        G$OFFS= 000400        KIPAR2= 172344
CKMSG   011212 G      C$RESE= 000033      EXPBRE  016352 G      G$OFSI= 000376        KIPAR3= 172346
CKMSG2  011332 G      C$REVI= 000003      EXPD    002176 G      G$PRMA= 000001        KIPAR4= 172350
CKRAM   010534 G      C$RFLA= 000021      EXPGOT  004431        G$PRMD= 000002        KIPAR5= 172352
CKRAM2  011110 G      C$RPT = 000025      EXPGT2  004465        G$PRML= 000000        KIPAR6= 172354
CMPMEM  020560        C$SEFG= 000046      EXPMSG  002266 G      G$RADA= 000140        KIPAR7= 172356
CONFIG  020222        C$SPRI= 000041      EXPREC  016344 G      G$RADB= 000000        KIPDR0= 172300
COUNT   002254 G      C$SVEC= 000037      EXTA    005232        G$RADD= 000040        KIPDR1= 172302
CSR   = 174400        C$TPRI= 000013      EXTEND  005230        G$RADL= 000120        KIPDR2= 172304
CSRADD  002154 G      DAR   = 174404      E$END = 002100        G$RADO= 000020        KIPDR3= 172306
CTAB    003116 G      DATA    002256 G    E$LOAD= 000035        G$XFER= 000004        KIPDR4= 172310
CTABE   003130 G      DATAFL  015064      FATCHK  020102        G$YES = 000010        KIPDR5= 172312
CTABM   003116 G      DATASC  021136      FATERR= 000060        HIADDR= 001400        KIPDR6= 172314
```

SYMBOL TABLE

```
KIPDR7= 172316          L$SOFT  053226 G        L10061  037544          O$BGNR= 000001          PROASC  014714
KTENAB  003102 G        L$SPC   002056 G        L10062  040034          O$BGNS= 000001          PR1ASC  014761
KTFLG   003100 G        L$SPCP  002020 G        L10063  040510          O$DU  = 000001          PST32W  003104 G
KTINIT  021622          L$SPTP  002024 G        L10064  047554          O$ERRT= 000000          PUNIT   022476
KTOFF   020246          L$STA   002030 G        L10065  051476          O$GNSW= 000001          PW.D11= 000021
KTON    020230          L$SW    002134 G        L10066  053062          O$POIN= 000001          PW.D13= 000022
LERRMA  002140 G        L$TEST  002114 G        L10067  053120          O$SETU= 000001          PW.D22= 000020
LISTAL= 000001          L$TIML  002014 G        L10070  053242          PASRPT  022242          PW.NOP= 000000
LOE   = 040000 G        L$UNIT  002012 G        L10071  053446          PATCH   053436 G        PW.NO1= 000023
LOOPCN  002164 G        L10000  002132          L10073  053454          PATDAT  021134          PW.RDE= 000024
LOOPCO  012760          L10001  002144          MEMADD  013606 G        PC.ERA= 002400          PW.RDR= 000001
LOOPFL  003114 G        L10002  005226          MENASC  021353          PC.IER= 002000          PW.RDS= 000005
LOT   = 000010 G        L10003  011676          MENERR  021300          PC.NOO= 001000          PW.RFI= 000003
L$ACP   002110 G        L10004  011726          MENRES  021402          PC.REL= 000000          PW.WCT= 000006
L$APT   002036 G        L10005  011744          MESBFA  002716 G        PC.REW= 000400          PW.WFI= 000004
L$AU    022544 G        L10006  011752          MESBFN  014634          PKBCNT= 000006          PW.WFM= 000007
L$AUT   002070 G        L10007  011770          MESHEA  015017          PKHI  = 000004          PW.WMI= 000010
L$AUTO  002750 G        L10010  012006          MIMEMJ  046512          PKLOW = 000002          PW.WNP= 000011
L$CCP   002106 G        L10011  012020          MMVEC = 000250          PKTADD  007276          PW.WTR= 000002
L$CLEA  023024 G        L10012  012072          MPR   = 174406          PKTFRM  007240          P.ACK = 100000
L$CO    002032 G        L10013  012242          MSA.FR= 000006          PKTGET  011730 G        P.CMD = 000037
L$DEPO  002011 G        L10014  012756          MSA.NO= 000000          PKTMES  011754 G        P.CONT= 000012
L$DESC  003342 G        L10015  013604          MSA.NR= 000004          PKTNEW  007333          P.CVC = 040000
L$DESP  002076 G        L10016  013626          MSA.VO= 000002          PKTRAM  004643 G        P.FMT = 000140
L$DEVP  002060 G        L10017  016350          MSGEXP  012010 G        PKTSSR  011700 G        P.FORM= 000011
L$DISP  053420 G        L10020  016356          MSGLOO  012716 G        PNT   = 001000 G        P.GETS= 000017
L$DLY   002116 G        L10021  016364          MSGSTA  012202 G        PRAMPK  013630          P.IE  = 000200
L$DTP   002040 G        L10022  016376          MSGSUB  013574 G        PRBEXP  016340          P.INIT= 000013
L$DTYP  002034 G        L10023  016420          MS.ATT= 000006          PRBMSG  016206          P.MODE= 007400
L$DU    022642 G        L10024  016446          MS.EXT= 000200          PRBREC  016342          P.OPP = 020000
L$DUT   002072 G        L10025  016606          MS.RSD= 000001          PRBTOT  016273          P.POSI= 000010
L$DVTY  003334 G        L10026  017116          MS.RSF= 000020          PRBYTE  015772 G        P.READ= 000001
L$EF    002052 G        L10030  022474          MS.RST= 000010          PRI   = 002000 G        P.SWB = 010000
L$ENVI  002044 G        L10031  022640          NBA   = 002000          PRIADD  007712          P.WRIT= 000005
L$ETP   002102 G        L10032  022746          NEWPAS  022176          PRIAO   007762          P.WRTC= 000004
L$EXP1  002046 G        L10033  023022          NODEV   003062 G        PRIBXO  007344 G        P.WRTS= 000006
L$EXP4  002064 G        L10034  023050          NOINIT  004233          PRIEQU  007612          QVP     002152 G
L$EXP5  002066 G        L10035  023312          NOINTR  004117          PRIPKT  007072 G        RAMASC  013776
L$HARD  053066 G        L10036  024672          NOITS   002136 G        PRIRAM  007620          RAMDAT  002206 G
L$HIME  002120 G        L10037  026224          NOMAN   021440          PRITAD  010026          RAMER   010636 G
L$HPCP  002016 G        L10040  025134          NP.IR = 000200          PRITSS  005264          RAMERR  016360 G
L$HPTP  002022 G        L10041  032156          NP.LOO= 000040          PRITO   010076          RAMEXP  016400 G
L$HW    002124 G        L10042  026624          NP.OUT= 000100          PRIXOR  007474 G        RAMFHR  014536
L$ICP   002104 G        L10043  027302          NP.WRP= 000020          PRI00 = 000000 G        RAMFOR  007650
L$INIT  021764 G        L10044  027510          NSI     004050          PRI01 = 000040 G        RAMHLD  011020
L$LADP  002026 G        L10045  027776          NSINIT  004305          PRI02 = 000100 G        RAMIOP  011024
L$LAST  053442 G        L10046  030320          NUL     004425          PRI03 = 000140 G        RAMPD   011075
L$LOAD  002100 G        L10047  043424          NULCR   004426          PRI04 = 000200 G        RAMRSH  011022
L$LUN   002074 G        L10050  032776          NXM   = 004000          PRI05 = 000240 G        RAMSIZ  002246 G
L$MREV  002050 G        L10051  033544          NXR     003636          PRI06 = 000300 G        RAMTAD  016366 G
L$NAME  002000 G        L10052  034240          NXRERR  005176 G        PRI07 = 000340 G        RBPCRA  015131
L$PRIO  002042 G        L10053  034676          NXRX    003675          PRMESS  014062          RCVHIA  002250 G
L$PROT  021754 G        L10054  035324          NXTU    022210          PRMNO   002264 G        RCVLOA  002252 G
L$PRT   002112 G        L10055  035752          O%L   = 000100          PRMSGE  015422 G        RDERR   005104
L$REPP  002062 G        L10056  036334          ONEFIL= 000000          PRMSG0  015602          READ  = 000014
L$REV   002010 G        L10057  037010          O$APTS= 000000          PRMSG1  015647          READY = 000001
L$RPT   023052 G        L10060  037252          O$AU  = 000001          PRMSG2  015705          RECMSG  002432 G
```

SYMBOL TABLE

```
RECV    002200 G    S1.ICE= 002000      TST21I  024516      T$$PTA= 010072      T23ET   030573
REGSAV  021040      S1.IEO= 010000      TST22I  026032      T$$RPT= 010035      T23LOO  026276
REWIND  010434 G    S1.IFM= 001000      TST23I  031761      T$$SOF= 010070      T23OFL  031301
RMCHBE= 000167      S1.IHE= 000400      TST24I  043172      T$$SRV= 010026      T23PAC  030360
RMCHEN= 000200      S1.IID= 004000      TST39I  051450      T$$SUB= 010063      T23PK2  030470
RMMSGB= 000104      S1.I1R= 020000      TST40I  052756      T$$SW = 010001      T23PK3  030510
RMMSGE= 000117      S1.I2R= 040000      TTIBFR= 177562 G    T$$TES= 010066      T23RES  031776
RMPKTB= 000020      S1.PAR= 100000      TTICSR= 177560 G    T1      023634 G    T23RNC  031160
RMPKTE= 000027      S2.ATI= 000010      TTION   052200      T2      024674 G    T23RSZ  030520
RMR   = 010000      S2.BTI= 000004      TTION2  044604      T2.1    024734      T23RT2  032070
RWPACK  010530      S2.DIM= 000200      TTIVEC= 000060 G    T21AM3  024375      T23RT3  032132
SC    = 100000      S2.ILW= 000100      TTOBFR= 177566      T21BFR  024170      T23RWN  031111
SCE   = 020000      S2.INR= 000020      TTOCSR= 177564      T21BF2  024270      T23SSR  030540
SCME    004711      S2.OUT= 000040      TUV2A   002000 G    T21BS0  024270      T23SZ   030516
SCMENU  052206      S2.UND= 000003      TVECSA  052202      T21BS1  024271      T23S2   030524
SDELAY  010330      TBLEND= 003030 G    TVSAV2  044606      T21DAT  024160      T23S3   030526
SEEK  = 000006      TCOASC  006200      T$ARGC= 000001      T21DLY  024276      T23TM   030736
SELASC  021346      TCOCOD  006400      T$CODE= 001130      T21LOO  023674      T23TMA  031025
SELDAT= 000004      TEMP1   003064 G    T$ERRN= 001276      T21OFL  024475      T23VCK  031545
SEL2  = 000002      TEMP2   003066 G    T$EXCP= 000000      T21PAC  024150      T23WB   030512
SETMAP  020270      TERCLS= 000016      T$FLAG= 000040      T21PK2  024260      T23WD   030530
SETU    022274      TESTNO= 000007      T$FREE= 053454      T21RES  024540      T23WDC  031443
SFFMSG  011746 G    TEXASC  006137      T$GMAN= 000000      T21RT2  024630      T23WDD  031354
SFHERR  003603      TFCASC  006241      T$HILI= 000007      T21SSR  024300      T23WDR  030532
SFIERR  003550      TIMEXP  016422 G    T$LAST= 000001      T21S2   024272      T23WSS  031672
SFIMSG  011666 G    TIMSGO  016450      T$LOLI= 000000      T21S3   024274      T24AM3  042160
SFPTBL  002134 G    TINERR  011653      T$LSYM= 010000      T22AM3  025435      T24BA   042512
SIFLAG  003106 G    TKB   = 177562      T$LTNO= 000007      T22BFR  025222      T24BFR  040570
SIMSG   011620      TKS   = 177560      T$NEST= 000000      T22BF2  025320      T24BF2  040710
SKIP    030320      TMPBFR  002576 G    T$NSC = 000000      T22BS0  025320      T24BOT  041553
SKIPT   003332      TNAM    017604      T$NS1 = 000005      T22BS1  025321      T24BS0  040710
SOFINI  016644 G    TPB   = 177566      T$NS2 = 000002      T22DAT  025210      T24BS1  040711
SPACE   010140 G    TPRISA  052204      T$PCNT= 000000      T22FOR  025334      T24CON  040722
SPM1    053242      TPS   = 177564      T$PTAB= 010072      T22LOO  024734      T24DAT  040560
SPM4    053306      TPSAV2  044610      T$PTHV= 000001      T22OFL  025535      T24DLY  040726
SPM6    053336      TRANST  002134 G    T$PTNU= 000001      T22PAC  025200      T24DTA  041620
SPM7    053366      TSBA  = 177776 G    T$SAVL= 177777      T22PK2  025310      T24EOT  041706
SR0   = 177572      TSBAH = 177777 G    T$SEGL= 177777      T22POS  025332      T24ILA  041302
SR1   = 177574      TSBAL = 177776 G    T$SIZE= 000005      T22RD   025326      T24LON  042652
SR2   = 177576      TSDB  = 177776 G    T$SUBN= 000000      T22RES  026066      T24LOO  032224
SR3   = 172516      TSDBH = 177777 G    T$TAGL= 177777      T22R12  026100      T24LOP  042734
SSR   = 000200      TSDBL = 177776 G    T$TAGN= 010074      T22RWJ  025704      T24LOQ  041366
STARTCO 012244      TSFCOD  006740      T$TEMP= 000010      T22SSR  025340      T24LOR  041002
SVCGBL= 000000      TSREJ = 000006      T$TEST= 000007      T22S2   025322      T24NEF  040730
SVCINS= 000000      TSSDEF  006310      T$TSTM= 177777      T22S3   025324      T24NXM  041141
SVCSUB= 000001      TSSR  = 000000 G    T$TSTS= 000001      T22TM   025610      T24OFL  042225
SVCTAG= 000000      TSSRBI  003400 G    T$$AU = 010031      T22VCK  025757      T24PAC  040550
SVCTST= 000001      TSSRFO  006117      T$$AUT= 010033      T22WRT  025330      T24PBP  043016
S$LSYM= 010000      TSSRH = 000001 G    T$$CLE= 010034      T23AM3  031233      T24PK2  040660
SO.IDB= 000010      TSSX    003716      T$$DAT= 010073      T23BA   031620      T24PK3  040700
SO.IFB= 000002      TSTBLK  002720 G    T$$DU = 010032      T23BFR  030400      T24RB   040702
SO.IFP= 000001      TSTCNT  002162 G    T$$HAR= 010067      T23BF2  030522      T24RES  043240
SO.ILD= 000020      TSTEND  017620      T$$HW = 010000      T23BS0  030522      T24RN   040716
SO.ION  000010      TSTFLA  002260 G    T$$INI= 010030      T23BS1  030523      T24RNC  042105
SO.IRD= 000040      TSTLOO  017356 G    T$$MSG= 010025      T23CON  030534      T24RT2  043332
SO.IRW= 000004      TSTPTR  002262 G    T$$PC = 000001      T23DAT  030370      T24RT3  043374
SO.ISP= 000200      TSTSET  017410 G    T$$PRO= 010027      T23EOT  030660      T24RWN  042036
```

SYMBOL TABLE

```
T24SSR  041447          T38MS2  046324          T39PK4  050770          WC.I1T= 000040          XSOWLE= 004000
T24SZ   040706          T38MS4  046026          T39RES  051412          WC.I5R= 000020          XSOWLK= 000004
T24S2   040712          T38MS5  046105          T39RL   051406          WF.IED= 000010          XS1CON  015243
T24S3   040714          T38MS6  046162          T39SIZ  050776          WF.IER= 000004          XS2CON  015310
T24TM   041763          T38NE   045400          T39TAD  050240          WF.IHI= 000200          XS3CON  015355
T24TRL  043104          T38OFL  045772          T39WPN  051244          WF.IRE= 000040          XXCOMM  003070 G
T24VCK  042437          T38ONL  045676          T39WR   050772          WF.IWF= 000020          X$ALWA= 000000
T24WB   040702          T38ONM  045740          T39WRT  051171          WF.IWR= 000100          X$FALS= 000040
T24WDC  042366          T38PAC  044620          T4      032160 G        WF.I3R= 000002          X$OFFS= 000400
T24WDD  042300          T38PK2  045320          T4.1    032224          WF.I4R= 000001          X$TRUE= 000020
T24WDE  041501          T38PK3  045350          T4.10   037270          WRTCHR  010332 G        X1.COR= 020000
T24WDF  041225          T38PK4  045370          T4.11   037562          WRTERR  005011          X1.DLT= 100000
T24WDG  041052          T38RES  047044          T4.12   040052          WRTMSG  004754          X1.MBZ= 017375
T24WDR  040720          T38SIZ  045376          T4.2    033014          XFERAS  016610          X1.RBP= 000400
T24WSS  042563          T38SST  046240          T4.3    033562          XNXM    017276          X1.SPA= 040000
T3      026226 G        T38TAD  044630          T4.4    034256          XORBFO  007426          X1.UNC= 000002
T3.1    026276          T38WLE  045602          T4.5    034714          XORFOR  007544          X2.BUF= 000100
T3.2    026642          T38WOK  046430          T4.6    035342          XST0  = 000006 G        X2.EXT= 000200
T3.3    027320          T38WR   045372          T4.7    035770          XST1  = 000010 G        X2.OPM= 100000
T3.4    027526          T38WRL  045541          T4.8    036352          XST2  = 000012 G        X2.RCE= 040000
T3.5    030014          T38WRT  045455          T4.9    037026          XST3  = 000014 G        X2.REV= 000077
T38ASC  047507          T39BFR  050246          T40NE   052711          XST4  = 000016 G        X2.SPA= 035400
T38ASN  047526          T39BS0  050240          T40NSR  052772          XS0BOT= 000002          X2.UNI= 000007
T38AS0  047404          T39BS1  050241          T5      043426 G        XS0CON  015176          X2.WCF= 002000
T38AS1  047451          T39BS2  050242          T6      047556 G        XS0EOT= 000001          X3.DCK= 000010
T38BFR  044636          T39DAT  051410          T7      051500 G        XS0IE = 000040          X3.MBZ= 000006
T38BS0  044630          T39DLY  050224          UAM   = 000200 G        XS0ILA= 000400          X3.MDE= 177400
T38BS1  044631          T39DTA  050740          UNITN   002150 G        XS0ILC= 001000          X3.OPI= 000100
T38BS2  044632          T39EAI  050746          UNREC = 000006          XS0LET= 020000          X3.REV= 000040
T38CNT  047552          T39MCL  051323          USI     004021          XS0MOT= 000200          X3.RIB= 000001
T38DAT  047042          T39NE   051003          WAITF   017120 G        XS0NEF= 002000          X3.SPA= 000200
T38DLY  044612          T39NFL  051000          WC.IFA= 000200          XS0ONL= 000100          X3.TRF= 000020
T38DTA  045330          T39OF2  051062          WC.IFE= 000002          XS0PED= 000010          X4.HSP= 100000
T38EAI  045336          T39ON2  051126          WC.IGO= 000001          XS0RLL= 010000          X4.MBZ= 017400
T38EB   045312          T39PAC  050230          WC.IRE= 000010          XS0RLS= 040000          X4.RCE= 040000
T38ID   046466          T39PK2  050730          WC.IRW= 000004          XS0TMK= 100000          X4.TSM= 020000
T38MBP  047102          T39PK3  050760          WC.IOT= 000100          XS0VCK= 000020          X4.WRC= 000377
T38MSG  046370

. ABS.  053454      000
        000000      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  29424 WORDS  ( 115 PAGES)
DYNAMIC MEMORY:  20060 WORDS  ( 77 PAGES)
ELAPSED TIME:  00:30:09
CZTKFA,BIC,CZTKFA/-SP=SVC/ML,CZTKFA

.
```

SYMBOL TABLE B1
SYMBOL TABLE C1
SYMBOL TABLE D1
SYMBOL TABLE E1