TMA-11

INSTRUCTION TEST

MD-11-DZTMF-D

EP-DZTMF-D-DL-A

FICHE 1 OF 1

NOV 1976

digital

MADE IN USA

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZTMF-D-D

PRODUCT NAME: TMA,B-11/TU10,N,W SUPPLEMENTAL INST. TEST

PROGRAM DATE: AUGUST 1976

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: R. B. BARNES

REVISED BY: RON PLATUKIS/R. SOLER

TABLE OF CONTENTS

1. ABSTRACT

THIS PROGRAM IS INTENDED TO BE USED IN ADDITION TO
THE TMA,B-11 INSTRUCTION TEST (MAINDEC-11-DZTMA)
TO COMPLETE TESTING OF THE MAG TAPE CONTROLLER.
THE PROGRAM CONSISTS OF ONLY FOUR (4) TESTS WHICH
CHECK ONLY THE TMA,B-11 FEATURES OF DATA TRANSFER AT ODD
BYTE STARTING ADDRESS AND OPERATION INCOMPLETE TIME OUT.

2. REQUIREMENTS

A. ANY PDP-11 PROCESSOR
B. 4K OF CORE
C. CONSOLE TTY
D. TMA-11 OR TMB-11 TAPE CONTROLLER (ONLY)
E. 1-8 TAPE TRANSPORTS (TU10,N,W)

3. LOADING PROCEDURE

A. USE STANDARD PROCEDURE FOR LOADING BINARY PAPER TAPE
B. THIS PROGRAM IS LOADABLE AND CHAINABLE PER XXDP, ACT11,
AND SLIDE, IN 8K OF MEMORY. (SEE 7.1)

4. STARTING PROCEDURE

THERE ARE TWO (2) STARTING ADDRESSES THAT MAY BE USED:
200(8) AND 210(8).

A. 200(8): STARTING AT THIS ADDRESS WILL CAUSE A PROGRAM
IDENTIFICATION HEADER TO BE PRINTED AND ALSO A
REQUEST FOR ENTRY OF THE UNIT NUMBER (TAPE TRANSPORT SELECT).
THE DEFAULT SELECTION OF UNIT ZERO (0) IS DISPLAYED, AND
MAY BE CHANGED TO ANY NUMBER (0-7) OR UNCHANGED BY
TYPING THE DESIRED NUMBER OR A CARRIAGE RETURN.
IF THE SELECTED UNIT IS NOT AVAILABLE, A MESSAGE WILL
BE PRINTED SO STATING, AND THE UNIT SELECT REQUEST REPEATED.

B. 210(8): STARTING AT THIS ADDRESS WILL NOT PRINT THE HEADER OR THE
UNIT SELECT REQUEST AND IS INTENDED AS A RESTART ADDRESS ONLY.

5. CONSOLE SWITCH SETTING

ALL SWITCHES EXCEPT 3-9 ARE USED AND THE NORMAL, OR DEFAULT, RUN
IS DONE WITH ALL SWITCHES SET TO ZERO (0).
ALL SWITCHES ARE DYNAMIC AND MAY BE CHANGED AT ANY TIME.

SW15: 1=HALT ON ERROR
0=CONTINUE
SW14: 1=LOOP ON ERROR (SCOPE)
0=CONTINUE
SW13: 1=INHIBIT ERROR TYPE OUT
0=PRINT ALL ERRORS
SW12: 1=INHIBIT ITERATION**(FIRST PASS IS SINGLE ITERATION)**
0=ITERATE EACH TEST ITS ASSIGNED AMOUNT
SW11: 1=CONTINUOUS CYCLE
0=HALT AT END OF PASS
SW10: 1=HALT AT END OF CURRENT TEST
0=CONTINUE
SW9-3: NOT USED
SW2-0: SELECT INDIVIDUAL TEST (1-4)** 00 = DO ALL TESTS

5.1 THIS PROGRAM HAS BEEN MODIFIED TO RUN ON A PROCESSOR WITH OR WITHOUT
A HARDWARE SWITCH REGISTER. WHEN FIRST EXECUTED THE PROGRAM TESTS
THE EXISTENCE OF A HARDWARE SWITCH REGISTER. IF NOT FOUND A
SOFTWARE SWITCH REGISTER LOCATION (SWREG=LOC. 17[illegible]) IS DEFAULTED TO.
IF THIS IS THE CASE, UPON EXECUTION THE CONTENTS OF THE SWREG ARE
DUMPED IN OCTAL ON THE CONSOLE TTY AND ANY CHANGES ARE REQUESTED

(IE) SWR=XXXXXX NEW=

POSSIBLE RESPONSES ARE:

1.	<CR>	IF NO CHANGES ARE TO BE MADE
2.	6 DIGITS 0-7	TO REPRESENT IN OCTAL THE NEW SWITCH REGISTER VALUE ;LAST DIGIT FOLLOWED BY <CR>.
3.	↑U	TO ALLOW REENTERING VALUE IF ERROR IS COMMITTED KEYING IN SWREG VALUE.
4.	<LF>	ONLY VALID FOR ACT-11 SYSTEMS-DO NOT USE

BUILT INTO THE PROGRAM IS THE ABILITY TO DYNAMICALLY CHANGE THE
CONTENTS OF SWREG DURING PROGRAM EXECUTION. BY STRIKING ↑G
(CNTL G) ON CONSOLE TTY THE OPERATOR SETS A REQUEST FLAG TO CHANGE
THE CONTENTS OF SWREG, WHICH IS PROCESSED IN KEY AREAS OF THE PROGRAM
CODE (IE) ERROR ROUTINES, AFTER HALTS END OF PASS, AND OTHER
APPLICABLE AREAS.

6. ERROR PRINTOUTS

THERE ARE THREE (3) TYPES OF ERROR PRINTOUTS WHICH MAY APPEAR: STATUS ERROR, DATA ERROR, POSITION ERROR.

A. STATUS ERROR: ANY READ, WRITE, OR SPACE OPERATION WHICH RESULTS IN SOME BAD STATUS (BIT 15 OF MTC), OR UNEXPECTED BUS ADDRESS, OR INCORRECT BYTE COUNT, WILL BE PRINTED.

B. DATA ERROR: ANY READ OPERATION WHICH RESULTS IN UNEXPECTED DATA WILL BE PRINTED.

C. POSITION ERROR: ANY SPACE OR REWIND OPERATION RESULTING IN UNEXPECTED STATUS WILL BE PRINTED.

EXAMPLES***

1. THE FOLLOWING EXAMPLE SHOWS A TYPICAL STATUS ERROR.

```
TEST1: WRITE FROM ODD BYTE
WRITE ERROR                 THIS PRINT SHOWS THAT WHILE EXECUTING
MTS: 10101                  TEST 1 ON UNIT 2 AT 800 BPI, A WRITE
MTC: 161204                 PARITY ERROR OCCURED.  THE BYTE COUNT
MTBC: 0                     IS ZERO AS IT SHOULD BE AND THE CURRENT
MTCA: 6003 6003             ADDRESS IS AS EXPECTED.
```

2. THE FOLLOWING EXAMPLE SHOWS A TYPICAL DATA ERROR.

```
TEST 2: READ TO ODD BYTE
DATA ERROR                  THIS PRINT SHOWS THAT A SINGLE BIT WAS
CN: 0                       PICKED UP IN BOTH CHARACTER NUMBER ZERO
G: 00000000                 (0) AND CHARACTER NUMBER THREE (3).
B: 01000000
CN: 3
G: 00000011
B: 01000011
```

3. THE FOLLOWING EXAMPLE SHOWS AN ERROR DURING A REWIND OPERATION.

```
TEST4: OPI TOO LONG
REWIND ERROR: NO BOT
```

7. OPERATION

THE PROCEDURES FOR OPERATING THIS PROGRAM ARE QUITE SIMPLE AND REQUIRE ONLY A FEW STEPS:

1. LOAD ADDRESS 200 OR 210
2. SET SWICHES FOR DESIRED TEST SEQUENCE
3. PRESS START

ALL CONSOLE SWITCHES ARE DYNAMIC AND MAY BE CHANGED AT ANY TIME. THE NORMAL OPERATING SEQUENCE IS ALL SWITCHES DOWN (0). THE PROGRAM WILL TAKE APPROXIMATELY 1.25 MINUTES TO RUN; HOWEVER, IF ITERATIONS ARE INHIBITED (SW11=1), THE PROGRAM WILL RUN IN ABOUT .75 MINUTES. THE END OF PASS IS NOTED BY A PRINTOUT STATING END OF PASS AND THE NUMBER OF THAT PASS.

SINGLE TEST SELECTION: (SW0-SW3)

WHEN SW0-3 ARE SET TO ZERO (0), THE SCHEDULAR WILL EXECUTE ALL TESTS (1-4) IN SEQUENCE AS A SINGLE PASS. IF SW0-3 ARE SET TO SOME NUMBER BETWEEN 1 AND 4, THEN THAT PARTICULAR TEST WILL BE EXECUTED CONTINUOUSLY. THE PROGRAM MAY BE STOPPED AT THE END OF THE CURRENT TEST (EITHER IN SEQUENCE OR SINGLE TEST MODE) BY SETTING SWITCH TEN (SW10) TO A ONE (1). YOU MAY SELECT TEST NUMBERS IN ANY ORDER (UP OR DOWN) BECAUSE EACH TEST IS SELF CONTAINED.

7.1 CHAIN MODE RUNS A SINGLE PASS ON DRIVE 0 WITH 7 OR 9 TRACK AT THE STANDARD UNIBUS ADDRESS.

8. TEST DESCRIPTION

TEST1: WRITE FROM ODD BYTE

THE PURPOSE OF THIS TEST IS TO ASSURE THAT DATA MAY BE TRANSFERRED FROM MEMORY TO TAPE STARTING FROM AN ODD BYTE ADDRESS. THE TEST WILL WRITE A SIX (6) BYTE RECORD FROM AN ODD ADDRESS (WDATA+1) AND READ THAT RECORD BACK INTO AN EVEN ADDRESS (RDATA). NO STATUS ERROR SHOULD OCCUR, AND THE READ DATA SHOULD BE POSITIONED PROPERLY. THE RECORD IS SIX BYES LONG, EACH BYTE IS ITS NUMBER (0,1,2,3,4,5)

TEST2: READ TO ODD BYTE

THE PURPOSE OF THIS TEST IS TO ASSURE THAT DATA MAY BE TRANSFERRED FROM TAPE TO MEMORY STARTING AT AN ODD BYTE ADDRESS. THE PROCEDURE IS THE SAME AS IN TEST ONE (1), EXCEPT THAT THE WRITE IS FROM AN EVEN ADDRESS (WDATA) AND THE READ IS TO AN ODD ADDRESS (RDATA+1).

TEST3: OPI TOO LONG (OPI = BIT 8 OF MTS)

THE PURPOSE OF THIS TEST IS TO ASSURE THAT THE OPI TIMER WILL SHUTDOWN THE DRIVE BEFORE THIRTY FIVE FEET OF BLANK TAPE IS PASSED. THE PROCEDURE IS TO PERFORM A WRITE WITH IRG, BACKSPACE, WRITE WITH IRG 105(10) TIMES IN ORDER TO ERASE 35 FEET OF TAPE. AFTER REWIND, ISSUE A READ COMMAND AND OPI SHOULD TIME OUT BEFORE THE FIRST RECORD (35 FEET DOWN TAPE) IS FOUND. THE NOMINAL VALUE FOR OPI IS SEVEN SECONDS (7SEC) OR ABOUT TWENTY-SIX FEET (26 FT) OF TAPE. THIRTY-FIVE FEET OF TAPE REFLECTS THE MAXIMUM TOLERANCE FOR OPI.

TEST4: OPI TOO SHORT (OPI = BIT 8 OF MTS)

THE PURPOSE OF THIS TEST IS TO ASSURE THAT THE OPI TIMER WILL NOT SHUTDOWN THE DRIVE BEFORE SIXTEEN FEET (16 FT) OF BLANK TAPE IS PASSED. THE PROCEDURE IS THE SAME AS IN TEST THREE (3), HOWEVER OPI IS NOT EXPECTED BEFORE THE FIRST RECORD IS FOUND (16 FEET DOWN TAPE). THE SIXTEEN FEET OF TAPE RELECTS THE MINIMUM TOLERANCE FOR OPI.

9. LISTING

```
                                        .TITLE TMA,B-11 SUPPLEMENTAL INST TEST
                                        ;MAINDEC-11-DZTMF-D-D
                                        ;JUNE 76
                                        ;R. BARNES
                                        .ENABLE ABS,AMA
                                        .MCALL  .$ACT11



                                        ;REGISTER EQUIVS**********

000000                                  R0=%0
000001                                  R1=%1
000002                                  R2=%2
000003                                  R3=%3
000004                                  R4=%4
000005                                  R5=%5
000006                                  SP=%6
000007                                  PC=%7

                                        ;TRAP CATCHERS**********


                                .SBTTL  ACT11 HOOKS

                                ;;**********************************************************************
                                ;HOOKS REQUIRED BY ACT11
001000                                  $SVPC=.                 ;SAVE PC
000046                                  .=46
000046  001660                          $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
000052                                  .=52
000052  000000                          .WORD   0               ;;2)SET LOC.52 TO ZERO
001000                                  .=$SVPC                 ;; RESTORE PC


                                        ;**********************************************************
                                        ;NOTE: PROGRAM HAS BEEN MODIFIED TO RUN WITH OR WITHOUT
                                        ;      A HARDWARE SWITCH REGISTER-REFER TO DOCUMENT
                                        ;**********************************************************
```

```
                                        ;TTY INTERRUPT VECTOR**********

                000060                  .=60
000060  004410                          TTINT                   ;TTY INTERRUPT HANDLER
000062  000000                          0


                                        ;SOFTWARE SWITCH REGISTER LOCATIONS************

                000174                  .=174
000174  000000                          DISPREG:0
000176  000000                          SWREG:  0


                                        ;STARTING ADDRESS***************
                000200                  .=200
000200  005000                          CLR     R0
000202  000137  001000                  JMP     START           ;PROGRAM START

                000210                  .=210
000210  012700  000001                  MOV     #1,R0
000214  000137  001006                  JMP     RSTART          ;NO HEADER START

                                        ;TMA,B-11 INTERRUPT VECTOR**********

                000224                  .=224
000224  004370                          MTINT                   ;TAPE INTERRUPT HANDLER
000226  000340                          340
```

```
                000600                          .=600
                                                ;CONSTANTS**********

000600  172520                  MTS:    172520  ;TAPE STATUS REGISTER
000602  172522                  MTC:    172522  ;TAPE COMMAND REGISTER
000604  172524                  MTBC:   172524  ;TAPE BYTE COUNTER
000606  172526                  MTBA:   172526  ;TAPE BUS ADDRESS
000610  000000                  UDES:   0       ;UNIT DESCRIPTION(PRESET FOR UNIT 0)
000612  000020                  RCNT:   20      ;RECORD COUNT
000614  177760                  CCNT:   -20     ;CHARACTER COUNT
000616  177776                  PSW:    177776  ;PROCESSOR STATUS
000620  177570                  SWR:    177570  ;CONSOLE SWITCH REGISTER
000622  177570                  DISPLAY:177570  ;CONSOLE DISPLAY REGISTER
000624  177560                  TKS:    177560  ;TTY READ STATUS
000626  177562                  TKB:    177562  ;TTY READ BUFFER
000630  177564                  TPS:    177564  ;TTY PUNCH STATUS
000632  177566                  TPB:    177566  ;TTY OUTPUT BUFFER
000634  000010                  ITAMT:  10      ;NUMBER OF ITERATIONS
000636  000004                  STALL:  4       ;READY DELAY MULTIPLIER
```

```
                                        ;FLAGS AND COUNTERS**********

000640  000000              TINF:   0
000642  000000              TOB:    0
000644  000000              TIB:    0
000646  000000              TEMP1:  0
000650  000000              TEMP2:  0
000652  000000              TEMP3:  0
000654  000000              EMADDR: 0
000656  000000              ERRAD:  0
000660  000000              LTADD:  0
000662  000000              ITRLP:  0
000664  000000              SPFLG:  0
000666  000000              STFLG:  0
000670  000000              PCNTR:  0
000672  000000              BADR:   0
000674  000000              BYTES:  0
000676  000000              SCNT:   0
000700  000000              FUN:    0
000702  000000              ITCNT:  0
000704  000000              CRCNT:  0
000706  000000              DERFL:  0
000710  000000              HDRFL:  0
000712  000000              PFLG:   0
000714  000000              UNP:    0
000716  000000              BCNT:   0
000720  000000              COUNT:  0
000722  000000              TEMPST: 0
000724  000000              RDSW:   0

                                        ;TEST ENTRY TABLE**********

000726  000000              TSTTBL: 0
000730  000000                      0
000732  001720              T1AD:   LT1
000734  001720              T1IAD:  LT1
000736  002146              T2AD:   LT2
000740  002146              T2IAD:  LT2
000742  002370              T3AD:   LT3
000744  002466              T3IAD:  LT3IT
000746  002620              T4AD:   LT4
000750  002716              T4IAD:  LT4IT
000752  000000                      0
```

```
                001000                          .=1000
                                                ;*****************************
                                                ;PROGRAM START AND HOUSEKEEPING
                                                ;*****************************
001000  012737  177570  000620  START:  MOV     #177570,SWR     ;PRESET TO CONSOLE SWITCHES
001006  012777  000340  177602  RSTART: MOV     #340,@PSW       ;SET PRIORITY
001014  012706  000500                  MOV     #500,SP         ;SET STACK POINTER
001020  013746  000006          SUSWR:  MOV     @#6,-(SP)               ;SAVE VECTORS
001024  013746  000004                  MOV     @#4,-(SP)
001030  012737  001050  000004          MOV     #1$,@#4                 ;SET UP FOR TIMEOUT
001036  022777  177777  177554          CMP     #-1,@SWR                ;REFERENCE HARDWARE SWITCH REGISTER
001044  001402                          BEQ     2$
001046  000407                          BR      3$
001050  022626                  1$:     CMP     (SP)+,(SP)+             ;ADJUST STACK
001052  012737  000176  000620  2$:     MOV     #SWREG,SWR              ;POINT TO SOFTWARE SWITCH REG
001060  012737  000174  000622          MOV     #DISPREG,DISPLAY                ;POINT TO SOFT DISPLAY REG
001066  012637  000004          3$:     MOV     (SP)+,@#4               ;RESTORE VECTORS
001072  012637  000006                  MOV     (SP)+,@#6
001076  122737  000004  000041          CMPB    #4,@#41         ;TM-11 MAG TAPE?
001104  001006                          BNE     4$              ;IF NO , BR
001106  012704  006377                  MOV     #MSG22,R4
001112  004737  004656                  JSR     PC,TTOUT        ;LOADER TM-11 MAG TAPE
001116  000137  001630                  JMP     TEND            ;END OP
001122  012777  010000  177452  4$:     MOV     #10000,@MTC     ;POWER CLEAN
001130  005700                          TST     R0              ;SEE IF SKIP HEADER
001132  001116                          BNE     ST4             ;IF SO: BR
001134  012704  005724                  MOV     #MSG1,R4
001140  004737  004656                  JSR     PC,TTOUT        ;PRINT HEADER
001144  012704  006012          ST0:    MOV     #MSG2,R4
001150  004737  004656                  JSR     PC,TTOUT        ;REQUEST DRIVE NUMBER
001154  005037  000610                  CLR     UDES            ;PRESET UNIT 0
001160  013703  000610                  MOV     UDES,R3         ;GET UNIT NUMBER
001164  000303                          SWAB    R3              ;POSITION
001166  042703  177770                  BIC     #177770,R3      ;MASK UNIT NUMBER
001172  004737  005054                  JSR     PC,OCTP         ;PRINT CURRENT VALUE
001176  005737  000042                  TST     @#42            ;CHAIN MODE?
001202  001404                          BEQ     1$              ;IF YES, BR
001204  012737  000176  000620          MOV     #176,SWR        ;STORE SWR
001212  000434                          BR      ST1
001214  012705  000652          1$:     MOV     #TEMP3,R5       ;SET SAVE LOCATION
001220  012701  000001                  MOV     #1,R1           ;SET SIZE OF ENTRY
001224  012702  000007                  MOV     #7,R2           ;SET UPPER LIMIT
001230  012703  000000                  MOV     #0,R3           ;SET LOWER LIMIT
001234  004737  004420                  JSR     PC,TTR          ;GO GET UNIT NUMBER
001240  000337  000652                  SWAB    TEMP3           ;POSITION UNIT NUMBER
001244  042737  003400  000610          BIC     #3400,UDES      ;CLEAR OLD NUMBER
001252  053737  000652  000610          BIS     TEMP3,UDES      ;LOAD NEW NUMBER
001260  013777  000610  177314          MOV     UDES,@MTC       ;SELECT UNIT
001266  005000                          CLR     R0
001270  022737  000176  000620          CMP     #SWREG,SWR
001276  001002                          BNE     ST1
001300  004737  005462                  JSR     PC,CNTLU
001304  032777  000100  177266  ST1:    BIT     #100,@MTS       ;SEE IF SELECT REMOTE
001312  001013                          BNE     ST2             ;IF SO: BR
001314  005300                          DEC     R0
001316  001372                          BNE     ST1             ;DELAY FOR SELECT REMOTE
```

```
   479  001320  012704  006356                    MOV     #MSG21,R4
   480  001324  004737  004656                    JSR     PC,TTOUT        ;PRINT NOT AVAILABLE
   481  001330  005737  000042                    TST     @#42            ;SEE IF CHAIN MODE
   482  001334  001703                           BEQ     ST0             ;IF YES, BR
   483  001336  000137  001630                    JMP     TEND            ;END OP
   484  001342  032777  000020  177230  ST2:     BIT     #20,@MTS        ;SEE IF 7 CHANNEL
   485  001350  001404                           BEQ     ST3             ;IF NOT: BR
   486  001352  052737  040000  000610            BIS     #40000,UDES     ;SET TO 800 BPI 7 CHAN
   487  001360  000403                           BR      ST4
   488  001362  052737  060000  000610  ST3:     BIS     #60000,UDES     ;SET TO 800 BPI 9 CHAN
   489  001370  000240                   ST4:    NOP
   490  001372  012702  000642                    MOV     #TOB,R2         ;GET START OF TABLE
   491  001376  012700  000027                    MOV     #27,R0          ;SET SIZE OF TABLE
   492  001402  005022                   ST5:    CLR     (R2)+           ;CLEAR TABLE
   493  001404  005300                           DEC     R0
   494  001406  001375                           BNE     ST5             ;DO ALL
   495  001410  005037  000670                    CLR     PCNTR           ;CLEAR PASS COUNTER
   496
```

```
                                                        ;TEST SCHEDULAR**********
                                                
001414  000240                          TSCD:   NOP
001416  005037  002666                          CLR     STFLG           ;CLEAR SINGLE TEST FLAG
001422  017700  177172                          MOV     @SWR,R0         ;GET SWITCH REGISTER
001426  042700  177700                          BIC     #177700,R0      ;MASK TEST SELECT
001432  005700                                  TST     R0              ;SEE IF SINGLE TEST SELECT
001434  001050                                  BNE     STSCD           ;IF SO: BR
001436  012737  000726  000660                  MOV     #TSTTBL,LTADD   ;GET TABLE START
001444  062737  000004  000660          TSCD0:  ADD     #4,LTADD
001452  013737  000660  000662                  MOV     LTADD,ITRLP     ;SET ITERATION ADDRESS
001460  062737  000002  000662                  ADD     #2,ITRLP
001466  005777  177165                          TST     @LTADD          ;SEE IF END OF CYCLE
001472  001002                                  BNE     TSCD1           ;IF NOT: BR
001474  000137  001630                          JMP     TEND            ;GO TO END ROUTINE
001500  005037  000710                  TSCD1:  CLR     HDRFL           ;CLEAR HEADER FLAG
001504  017700  177150                          MOV     @LTADD,R0       ;GET TEST ADDRESS
001510  000110                                  JMP     (R0)            ;GO TO TEST
001512  004737  005410                  TSCD2:  JSR     PC,CKSWR
001516  032777  002000  177074                  BIT     #2000,@SWR      ;SEE IF HALT ON TEST
001524  001401                                  BEQ     TSCD3           ;IF NOT: BR
001526  000000                                  HALT
001530  004737  005410                  TSCD3:  JSR     PC,CKSWR        ;GO TEST FOR ^G
001534  005737  000666                          TST     STFLG           ;SEE IF SINGLE TEST
001540  001741                                  BEQ     TSCD0           ;IF NOT: BR
001542  017700  177052                          MOV     @SWR,R0
001546  042700  177760                          BIC     #177760,R0      ;GET TEST NUMBER
001552  005700                                  TST     R0              ;SEE IF ALL TESTS
001554  001717                                  BEQ     TSCD            ;IF SO: BR
001556  012737  000001  000666  STSCD:  MOV     #1,STFLG        ;SET SINGLE TEST FLAG
001564  022700  000005                          CMP     #5,R0           ;SEE IF EXCEEDED TEST NUMBER
001570  003417                                  BLE     TEND            ;IF SO: BR
001572  000241                                  CLC
001574  006100                                  ROL     R0
001576  006100                                  ROL     R0              ;POSITION NUMBER
001600  012737  000726  000660                  MOV     #TSTTBL,LTADD   ;GET START OF TABLE
001606  060037  000660                          ADD     R0,LTADD        ;SET POINTER
001612  013737  000660  000662                  MOV     LTADD,ITRLP
001620  062737  000002  000662                  ADD     #2,ITRLP        ;SET ITERATION ADDRESS
001626  000724                                  BR      TSCD1           ;GO DO TEST
001630  012704  006031          TEND:   MOV     #MSG3,R4
001634  004737  004656                          JSR     PC,TTOUT        ;PRINT END OF PASS
001640  013703  000670                          MOV     PCNTR,R3
001644  004737  005054                          JSR     PC,OCTP         ;PRINT PASS NUMBER
001650  013703  000042                          MOV     @#42,R3
001654  001405                                  BEQ     HERE
001656  000005                                  RESET
001660  004713                  SENDAD: JSR     PC,(R3)
001662  000240                                  NOP
001664  000240                                  NOP
001666  000240                                  NOP
001670  000240                  HERE:   NOP
001672  032777  004000  176720                  BIT     #4000,@SWR      ;SEE IF HALT ON PASS
001700  001001                                  BNE     TENDX           ;IF NOT: BR
001702  000000                                  HALT
001704  004737  005410          TENDX:  JSR     PC,CKSWR        ;GO TEST FOR ^G
```

```
   553  001710  005237  000670                  INC     PCNTR           ;BUMP PASS COUNTER
   554  001714  000137  001414                  JMP     TSCD            ;RESTART
   555
   556
   557                                          ;*************************************
   558                                          ;TEST 1: WRITE FROM ODD BYTE
   559                                          ;
   560                                          ;THIS TEST WILL WRITE A SIX (6) BYTE RECORD
   561                                          ;FROM AN ODD BYTE STARTING ADDRESS. THE RECORD
   562                                          ;WILL BE READ BACK INTO AN EVEN STARTING ADDRESS
   563                                          ;TO TEST FOR PROPER TRANSFER.
   564                                          ;*************************************
   565
   566  001720  000240                  LT1:    NOP
   567  001722  012737  006466  000654          MOV     #LT1MSG,EMADDR  ;SET HEADER
   568  001730  012702  006636                  MOV     #WDATA,R2       ;GET BUFFER START
   569  001734  112722  000377                  MOVB    #377,(R2)+      ;INSERT BACKGROUND DATA
   570  001740  005000                          CLR     R0
   571  001742  110022                  LT1B:   MOVB    R0,(R2)+
   572  001744  005200                          INC     R0              ;LOAD WRITE BUFFER (0,1,2,3,4,5)
   573  001746  022700  000006                  CMP     #6,R0
   574  001752  001373                          BNE     LT1B
   575  001754  004737  003026                  JSR     PC,RWND         ;GO REWIND
   576  001760  012737  000004  000700          MOV     #4,FUN          ;SET WRITE FUNCTION CODE
   577  001766  012737  006637  000672          MOV     #WDATA+1,BADR   ;SET DATA POINTER
   578  001774  012737  177772  000674          MOV     #-6,BYTES       ;SET SIZE OF RECORD
   579  002002  012737  006256  000656          MOV     #MSG17,ERRAD    ;SET WRITE ERROR CODE
   580  002010  004737  003304                  JSR     PC,EXEC         ;GO EXECUTE COMMAND
   581  002014  000240                  LT1C:   NOP
   582  002016  004737  003534                  JSR     PC,ERCHK        ;GO CHECK FOR STATUS ERROR
   583  002022  012737  177777  000676          MOV     #-1,SCNT
   584  002030  004737  003212                  JSR     PC,BKSP         ;GO BACKSPACE ONE RECORD
   585  002034  012702  006740                  MOV     #RDATA,R2
   586  002040  012700  000010                  MOV     #10,R0
   587  002044  012722  177777          LT1D:   MOV     #-1,(R2)+       ;BACKGROUND READ BUFFER
   588  002050  005300                          DEC     R0
   589  002052  001374                          BNE     LT1D            ;DO ALL
   590  002054  012737  000002  000700          MOV     #2,FUN          ;SET READ FUNCTION CODE
   591  002062  012737  006740  000672          MOV     #RDATA,BADR     ;SET READ POINTER
   592  002070  012737  177772  000674          MOV     #-6,BYTES       ;SET SIZE OF RECORD
   593  002076  012737  006273  000656          MOV     #MSG18,ERRAD    ;SET READ ERROR CODE
   594  002104  004737  003304                  JSR     PC,EXEC         ;GO DO READ
   595  002110  000240                  LT1E:   NOP
   596  002112  004737  003534                  JSR     PC,ERCHK        ;GO CHECK ERRORS
   597  002116  012701  006637                  MOV     #WDATA+1,R1     ;SET EXPT DATA POINTER
   598  002122  012702  006740                  MOV     #RDATA,R2       ;SET RCVD DATA POINTER
   599  002126  012700  000006                  MOV     #6,R0           ;SET SIZE OF RECORD
   600  002132  004737  004022                  JSR     PC,DCHK         ;GO CHECK DATA
   601  002136  004737  004312                  JSR     PC,ITER         ;GO SEE IF ITERATIONS
   602  002142  000137  001512                  JMP     TSCD2           ;RETURN TO SCHEDULAR
```

```
                                                ;*************************
                                                ;TEST 2: READ INTO ODD BYTE
                                                ;
                                                ;THIS TEST WILL WRITE A SIX (6) BYTE RECORD
                                                ;FROM AN EVEN BYTE STARTING ADDRESS. THE RECORD
                                                ;WILL BE READ BACK INTO AN ODD STARTING ADDRESS
                                                ;TO TEST FOR PROPER TRANSFER.
                                                ;*************************

002146  000240                         LT2:    NOP
002150  012737  006524  000654                  MOV     #LT2MSG,EMADDR  ;SET HEADER POINTER
002156  012702  006636                          MOV     #WDATA,R2       ;POINT TO START OF WRITE BUFFER
002162  005000                                  CLR     R0
002164  110022                         LT2B:   MOVB    R0,(R2)+        ;LOAD DATA PATTERN
002166  005200                                  INC     R0              ;BUMP PATTERN
002170  022700  000006                          CMP     #6,R0           ;SEE IF DONE
002174  001373                                  BNE     LT2B            ;IF NOT: BR
002176  004737  003026                          JSR     PC,RWND         ;GO REWIND TO BOT
002202  012737  000004  000700                  MOV     #4,FUN          ;SET WRITE OP-CODE
002210  012737  006636  000672                  MOV     #WDATA,BADR     ;SET STARTING ADDRESS
002216  012737  177772  000674                  MOV     #-6,BYTES       ;SET SIZE OF RECORD
002224  004737  003304                          JSR     PC,EXEC         ;GO EXECUTE COMMAND
002230  000240                         LT2C:   NOP
002232  012737  006256  000656                  MOV     #MSG17,ERRAD    ;SET ERROR CODE
002240  004737  003534                          JSR     PC,ERCHK        ;GO CHECK FOR STATUS ERROR
002244  012737  177777  000676                  MOV     #-1,SCNT
002252  004737  003212                          JSR     PC,BKSP         ;GO BACKSPACE ONE RECORD
002256  012702  006740                          MOV     #RDATA,R2       ;GET READ BUFFER POINTER
002262  012700  000010                          MOV     #10,R0          ;SET SIZE
002266  012722  177777                 LT2D:   MOV     #-1,(R2)+       ;BACKGROUND POINTER
002272  005300                                  DEC     R0              ;SEE IF DONE
002274  001374                                  BNE     LT2D            ;IF NOT: BR
002276  012737  000002  000700                  MOV     #2,FUN          ;SET READ FUNCTION CODE
002304  012737  006741  000672                  MOV     #RDATA+1,BADR   ;SET START OF READ BUFFER
002312  012737  177772  000674                  MOV     #-6,BYTES       ;SET SIZE OF RECORD
002320  004737  003304                          JSR     PC,EXEC         ;GO EXECUTE COMMAND
002324  000240                         LT2E:   NOP
002326  012737  006273  000656                  MOV     #MSG18,ERRAD    ;SET ERROR CODE
002334  004737  003534                          JSR     PC,ERCHK        ;GO CHECK FOR STATUS ERROR
002340  012701  006636                          MOV     #WDATA,R1       ;POINT TO EXPT DATA
002344  012702  006741                          MOV     #RDATA+1,R2     ;POINT TO RCVD DATA
002350  012700  000006                          MOV     #6,R0           ;SET SIZE OF RECORD
002354  004737  004022                          JSR     PC,DCHK         ;GO CHECK DATA
002360  004737  004312                          JSR     PC,ITER         ;GO SEE IF ITERATION
002364  000137  001512                          JMP     TSCD2           ;RETURN TO SCHEDULAR
```

```
                                                ;*************************
                                                ;TEST 3: OPI TOO LONG
                                                ;
                                                ;THIS TEST WILL ERASE APPROXIMATELY THIRTYFIVE (35)
                                                ;FEET OF TAPE BY WRITING WITH IRG, BACKSPACING
                                                ;AND REPEATING THE SEQUENCE 105(10) TIMES. TAPE
                                                ;WILL REWIND AND A READ FORWARD ISSUED.  THE
                                                ;OPI TIMER SHOULD SHUTDOWN THE UNIT BEFORE
                                                ;REACHING THE FIRST RECORD ON TAPE.
                                                ;*************************

002370  000240                        LT3:    NOP
002372  012737  006557  000654                MOV     #LT3MSG,EMADDR  ;SET TEST HEADER
002400  012700  000151                        MOV     #151,R0         ;SET NUMBER OF WRITE IRG/BACKSPACE
002404  004737  003026                        JSR     PC,RWND         ;GO REWIND UNIT
002410  012737  000014  000700        LT3A:   MOV     #14,FUN         ;SET WRITE IRG FUNCTION CODE
002416  012737  006636  000672                MOV     #WDATA,BADR     ;SET BUS ADDRESS
002424  012737  177760  000674                MOV     #-20,BYTES      ;SET SIZE OF RECORD
002432  004737  003304                        JSR     PC,EXEC         ;GO EXECUTE COMMAND
002436  012737  006256  000656        LT3B:   MOV     #MSG17,ERRAD    ;SET ERROR CODE
002444  004737  003534                        JSR     PC,ERCHK        ;GO CHECK FOR STATUS ERROR
002450  012737  177777  000676                MOV     #-1,SCNT
002456  004737  003212                        JSR     PC,BKSP         ;GO BACKSPACE ONE RECORD
002462  005300                                DEC     R0              ;SEE IF DONE ALL
002464  001351                                BNE     LT3A            ;IF NOT: BR
002466  000240                        LT3IT:  NOP
002470  004737  003026                        JSR     PC,RWND         ;GO REWIND
002474  012737  000500  000636                MOV     #500,STALL      ;SET OPI STALL
002502  012737  006740  000672                MOV     #RDATA,BADR     ;SET START OF READ BUFFER
002510  012737  177760  000674                MOV     #-20,BYTES      ;SET SIZE OF RECORD
002516  012737  000002  000700                MOV     #2,FUN          ;SET READ FUNCTION CODE
002524  012737  006307  000656                MOV     #MSG19,ERRAD    ;SET ERROR CODE
002532  004737  003304                        JSR     PC,EXEC         ;GO EXECUTE COMMAND
002536  000240                        LT3C:   NOP
002540  012737  000004  000636                MOV     #4,STALL        ;RESET NORMAL STALL
002546  032777  000400  176024                BIT     #400,@MTS       ;SEE IF BTE IS SET
002554  001007                                BNE     LT3X            ;IF SO: BR
002556  012737  000001  000664                MOV     #1,SPFLG        ;SET NO BA PRINT FLAG
002564  004737  003612                        JSR     PC,ERPT         ;GO PRINT ERROR
002570  005037  000664                        CLR     SPFLG           ;RESET FLAG
002574  012737  000002  000634        LT3X:   MOV     #2,ITAMT        ;SET TO TWO (2) ITERATIONS
002602  004737  004312                        JSR     PC,ITER         ;GO SEE IF ITERATION
002606  012737  000010  000634                MOV     #10,ITAMT       ;RESET ITERATIONS
002614  000137  001512                        JMP     TSCD2           ;RETURN TO SCHEDULAR
```

```
                                                ;*************************
                                                ;TEST 4: OPI TOO SHORT
                                                ;
                                                ;THIS TEST WILL ERASE APPROXIMATELY SIXTEEN (16) FEET
                                                ;OF TAPE BY WRITING WITH IRG, BACKSPACING
                                                ;ONE (1) RECORD AND REPEATING THIS SEQUENCE
                                                ;50(10) TIMES.  TAPE WILL REWIND AND BE READ
                                                ;FORWARD.  THE FIRST RECORD ON TAPE SHOULD BE
                                                ;REACHED BEFORE OPI TIMES OUT.
                                                ;*************************

002620  000240                          LT4:    NOP
002622  012737  006606  000654                  MOV     #LT4MSG,EMADDR  ;SET HEADER
002630  004737  003026                          JSR     PC,RWND         ;GO REWIND
002634  012700  000062                          MOV     #62,R0          ;SET NUMBER OF WRITE IRG/BACKSPACES
002640  012737  000014  000700          LT4A:   MOV     #14,FUN         ;SET WRITE IRG FUNCTION CODE
002646  012737  006636  000672                  MOV     #WDATA,BADR     ;SET START OF WRITE BUFFER
002654  012737  177760  000674                  MOV     #-20,BYTES      ;SET SIZE OF RECORD
002662  012737  006256  000656                  MOV     #MSG17,ERRAD    ;SET ERROR CODE
002670  004737  003304                          JSR     PC,EXEC         ;GO EXECUTE COMMAND
002674  004737  003534                  LT4B:   JSR     PC,ERCHK        ;GO CHECK FOR STATUS ERROR
002700  012737  177777  000676                  MOV     #-1,SCNT
002706  004737  003212                          JSR     PC,BKSP         ;GO BACKSPACE ONE RECORD
002712  005300                                  DEC     R0              ;SEE IF DONE ALL SEQUENCES
002714  001351                                  BNE     LT4A            ;IF NOT: BR
002716  000240                          LT4IT:  NOP
002720  004737  003026                          JSR     PC,RWND         ;REWIND
002724  012737  000500  000636                  MOV     #500,STALL      ;SET OPI STALL
002732  012737  006740  000672                  MOV     #RDATA,BADR     ;SET START OF READ BUFFER
002740  012737  177760  000674                  MOV     #-20,BYTES      ;SET SIZE OF RECORD
002746  012737  000002  000700                  MOV     #2,FUN          ;SET READ FUNCTION CODE
002754  012737  006332  000656                  MOV     #MSG20,ERRAD    ;SET ERROR CODE
002762  004737  003304                          JSR     PC,EXEC         ;GO EXECUTE COMMAND
002766  004737  003534                  LT4C:   JSR     PC,ERCHK        ;GO CHECK FOR STATUS ERRORS
002772  000240                                  NOP
002774  012737  000004  000636                  MOV     #4,STALL        ;RESET NORMAL STALL
003002  012737  000002  000634                  MOV     #2,ITAMT        ;SET TO TWO (2) ITERATIONS
003010  004737  004312                          JSR     PC,ITER         ;GO SEE IF ITERATIONS
003014  012737  000010  000634                  MOV     #10,ITAMT       ;RESET ITERATIONS
003022  000137  001512                          JMP     TSCD2           ;RETURN TO SCHEDULAR
```

```
                                                        ;REWIND SUBROUTINE**********

003026  000240                          RWND:   NOP
003030  013777  000610  175544                  MOV     UDES,@MTC       ;SELECT UNIT
003036  032777  000040  175534                  BIT     #40,@MTS        ;SEE IF AT BOT
003044  001056                                  BNE     RWNDXX          ;IF SO: BR
003046  052777  000017  175526                  BIS     #17,@MTC        ;START REWIND
003054  105777  175522          1$:     TSTB    @MTC
003060  100375                                  BPL     1$              ;AWAIT CUR
003062  032777  000001  175510  RWND1:  BIT     #1,@MTS         ;AWAIT TUR
003070  001774                                  BEQ     RWND1
003072  032777  000040  175500                  BIT     #40,@MTS        ;SEE IF BOT SET
003100  001040                                  BNE     RWNDXX          ;IF SO: BR
003102  032777  020000  175510                  BIT     #20000,@SWR     ;SEE IF PRINT ERROR
003110  001030                                  BNE     RWNDX           ;IF NOT: BR
003112  013704  000654                          MOV     EMADDR,R4
003116  004737  004656                          JSR     PC,TTOUT        ;PRINT HEADER
003122  012704  006052                          MOV     #MSG4,R4
003126  004737  004656                          JSR     PC,TTOUT        ;PRINT REWIND ERROR
003132  012704  006102                          MOV     #MSG5,R4
003136  004737  004656                          JSR     PC,TTOUT        ;PRINT MTS TAG
003142  017703  175432                          MOV     @MTS,R3
003146  004737  005044                          JSR     PC,OCTPE        ;PRINT MTS
003152  012704  006111                          MOV     #MSG6,R4
003156  004737  004656                          JSR     PC,TTOUT        ;PRINT MTC TAG
003162  017703  175414                          MOV     @MTC,R3
003166  004737  005044                          JSR     PC,OCTPE        ;PRINT MTC
003172  005777  175422          RWNDX:  TST     @SWR            ;SEE IF HALT ON ERROR
003176  100001                                  BPL     RWNDXX          ;IF NOT: BR
003200  000000                                  HALT
003202  004737  005410          RWNDXX: JSR     PC,CKSWR        ;GO TEST FOR ↑G
003206  000240                                  NOP
003210  000207                                  RTS     PC              ;RETURN

                                                        ;BACKSPACE SUBROUTINE**********

003212  000240                          BKSP:   NOP
003214  013777  000610  175360                  MOV     UDES,@MTC       ;SELECT UNIT
003222  013777  000676  175354                  MOV     SCNT,@MTBC      ;SET NUMBER OF RECORDS TO SPACE
003230  052777  000013  175344                  BIS     #13,@MTC        ;START SPACE REVERSE
003236  105777  175340          1$:     TSTB    @MTC
003242  100375                                  BPL     1$              ;AWAIT CUR
003244  032777  000001  175326  BKSP1:  BIT     #1,@MTS
003252  001774                                  BEQ     BKSP1           ;AWAIT TUR
003254  012737  000001  000664                  MOV     #1,SPFLG        ;SET SPACE FLAG
003262  012737  006140  000656                  MOV     #MSG9,ERRAD
003270  004737  003534                          JSR     PC,ERCHK        ;GO CHECK FOR ERROR
003274  005037  000664                          CLR     SPFLG           ;CLEAR SPACE FLAG
003300  000240                                  NOP
003302  000207                                  RTS     PC              ;RETURN
```

```
                                                ;COMMAND EXECUTE SUBROUTINE**********
                                        EXEC:   NOP
003306  005005                                  CLR     R5
003310  032777  000200  175264          EXEC0:  BIT     #200,@MTC       ;SEE IF CUR
003316  001021                                  BNE     EXEC2           ;IF SO: BR
003320  005305                                  DEC     R5              ;SEE IF TIMED OUT
003322  001372                                  BNE     EXEC0           ;IF NOT: BR
003324  005737  000710                          TST     HDRFL           ;SEE IF DONE HEADER
003330  001004                                  BNE     EXEC1           ;IF SO: BR
003332  013704  000654                          MOV     EMADDR,R4
003336  004737  004656                          JSR     PC,TTOUT        ;ELSE PRINT HEADER
003342  012704  006163                  EXEC1:  MOV     #MSG10,R4
003346  004737  004656                          JSR     PC,TTOUT        ;PRINT NOT READY ERROR
003352  005777  175242                          TST     @SWR            ;SEE IF HALT ON ERROR
003356  100001                                  BPL     EXEC2           ;IF NOT: BR
003360  000000                                  HALT
003362  004737  005410                  EXEC2:  JSR     PC,CKSWR        ;GO TEST FOR ↑G
003366  000240                                  NOP
003370  013777  000610  175204                  MOV     UDES,@MTC       ;SELECT UNIT
003376  013777  000672  175202                  MOV     BADR,@MTBA      ;SET BUS MEMORY ADDRESS
003404  013777  000674  175172                  MOV     BYTES,@MTBC     ;SET BYTE COUNT
003412  013701  000700                          MOV     FUN,R1          ;GET FUNCTION
003416  052701  000101                          BIS     #101,R1         ;SET IN GO BIT AND INTERRUPT ENABLE
003422  050177  175154                          BIS     R1,@MTC         ;LOAD COMMAND+GO+IE
003426  000240                                  NOP
003430  005077  175162                          CLR     @PSW            ;ALLOW INTERRUPTS
003434  013737  000636  000646                  MOV     STALL,TEMP1     ;SET READY STALL
003442  005001                                  CLR     R1
003444  005301                          EXEC3:  DEC     R1
003446  001376                                  BNE     EXEC3           ;AWAIT INTERRUPT
003450  005337  000646                          DEC     TEMP1
003454  001373                                  BNE     EXEC3
003456  032777  020000  175134                  BIT     #20000,@SWR     ;SEE IF PRINT ERROR
003464  001013                                  BNE     EXECX           ;IF NOT: BR
003466  005737  000710                          TST     HDRFL           ;SEE IF DONE HEADER
003472  001004                                  BNE     EXEC4           ;IF SO: BR
003474  013704  000654                          MOV     EMADDR,R4
003500  004737  004656                          JSR     PC,TTOUT        ;PRINT HEADER
003504  012704  006200                  EXEC4:  MOV     #MSG11,R4
003510  004737  004656                          JSR     PC,TTOUT        ;PRINT NO INTERRUPT MESSAGE
003514  005777  175100                  EXECX:  TST     @SWR            ;SEE IF HALT ON ERROR
003520  100001                                  BPL     EXECXX          ;IF NOT: BR
003522  000000                                  HALT
003524  004737  005410                  EXECXX: JSR     PC,CKSWR        ;GO TEST FOR ↑G
003530  000240                                  NOP
003532  000207                                  RTS     PC              ;RETURN TO CALLER
```

```
                                                ;STATUS ERROR CHECK SUBROUTINE**********

003534  005777  175042          ERCHK:  TST     @MTC            ;SEE IF ANY ERROR BITS
003540  100002                          BPL     ERCHK1          ;IF NOT: BR
003542  000137  003612                  JMP     ERPT            ;ELSE PRINT ERROR
003546  005777  175032          ERCHK1: TST     @MTBC           ;SEE IF BYTE COUNT IS ZERO
003552  001402                          BEQ     ERCHK2          ;IF SO: BR
003554  000137  003612                  JMP     ERPT            ;ELSE PRINT ERROR
003560  013703  000674          ERCHK2: MOV     BYTES,R3
003564  005403                          NEG     R3
003566  063703  000672                  ADD     BADR,R3         ;SET EXPT BUS ADDRESS
003572  005737  000664                  TST     SPFLG           ;SEE IF SPACE OPERATION
003576  001401                          BEQ     ERCHK3          ;IF NOT: BR
003600  000207                          RTS     PC
003602  020377  175000          ERCHK3: CMP     R3,@MTBA        ;SEE IF EXPT=RCVD
003606  001001                          BNE     ERPT            ;IF NOT: BR
003610  000207                          RTS     PC              ;ELSE EXIT
003612  000240          ERPT:   NOP
003514  032777  020000  174776          BIT     #20000,@SWR     ;SEE IF SHOULD PRINT
003622  001067                          BNE     ERPTX           ;IF NOT: BR
003624  005737  000710                  TST     HDRFL           ;SEE IF DONE HEADER
003630  001006                          BNE     ERPT1           ;IF SO: BR
003632  013704  000654                  MOV     EMADDR,R4
003636  004737  004656                  JSR     PC,TTOUT        ;ELSE PRINT HEADER
003642  005237  000710                  INC     HDRFL           ;SET FLAG
003646  013704  000656          ERPT1:  MOV     ERRAD,R4
003652  004737  004656                  JSR     PC,TTOUT        ;PRINT ERROR CODE
003656  012704  006102                  MOV     #MSG5,R4
003662  004737  004656                  JSR     PC,TTOUT        ;PRINT MTS TAG
003666  017703  174706                  MOV     @MTS,R3
003672  004737  005044                  JSR     PC,OCTPE        ;PRINT MTS
003676  012704  006111                  MOV     #MSG6,R4
003702  004737  004656                  JSR     PC,TTOUT        ;PRINT MTC TAG
003706  017703  174670                  MOV     @MTC,R3
003712  004737  005044                  JSR     PC,OCTPE        ;PRINT MTC
003716  012704  006120                  MOV     #MSG7,R4
003722  004737  004656                  JSR     PC,TTOUT        ;PRINT BYTE COUNT TAG
003726  017703  174652                  MOV     @MTBC,R3
003732  004737  005054                  JSR     PC,OCTP         ;PRINT BYTE COUNT
003736  005737  000664                  TST     SPFLG           ;SEE IF PRINT BA
003742  001017                          BNE     ERPTX           ;IF NOT: BR
003744  012704  006130                  MOV     #MSG8,R4
003750  004737  004656                  JSR     PC,TTOUT        ;PRINT BUS ADDRESS TAG
003754  017703  174626                  MOV     @MTBA,R3
003760  004737  005054                  JSR     PC,OCTP         ;PRINT CURRENT ADDRESS
003764  013703  000674                  MOV     BYTES,R3
003770  005403                          NEG     R3
003772  063703  000672                  ADD     BADR,R3
003776  004737  005054                  JSR     PC,OCTP         ;PRINT EXPT ADDRESS
004002  005777  174612          ERPTX:  TST     @SWR            ;SEE IF HALT ON ERROR
004006  100001                          BPL     ERPTXX          ;IF NOT: BR
004010  000000                          HALT
004012  004737  005410          ERPTXX: JSR     PC,CKSWR        ;GO TEST FOR ↑G
004016  000137  004262                  JMP     SCOPE           ;GO SEE IF SCOPE ON ERROR
```

```
                                                ;DATA CHECK SUBROUTINE**********

004022  000240                          DCHK:   NOP
004024  005037  000704                          CLR     CRCNT           ;CLEAR COUNTER
004030  121112                          DCHK0:  CMPB    (R1),(R2)       ;SEE IF EXPT DATA=RCVD DATA
004032  001007                                  BNE     DCHKE           ;IF NOT: BR
004034  005237  000704                  DCHK1:  INC     CRCNT           ;BUMP CHARACTER COUNTER
004040  122122                                  CMPB    (R1)+,(R2)+
004042  005300                                  DEC     R0              ;SEE IF DONE
004044  001371                                  BNE     DCHK0           ;IF NOT: BR
004046  000137  004222                          JMP     DCHKX           ;ELSE GO TO EXIT ROUTINE
004052  000240                          DCHKE:  NOP
004054  012737  000001  000706                  MOV     #1,DERFL        ;SET ERROR FLAG
004062  032777  020000  174530                  BIT     #20000,@SWR     ;SEE IF PRINT ERROR
004070  001054                                  BNE     DCHKX           ;IF NOT: BR
004072  005737  000710                          TST     HDRFL           ;SEE IF DONE HEADER
004076  001007                                  BNE     DCHKE1          ;IF SO: BR
004100  013704  000654                          MOV     EMADDR,R4
004104  004737  004656                          JSR     PC,TTOUT        ;PRINT HEADER
004110  012737  000001  000710                  MOV     #1,HDRFL        ;SET HEADER FLAG
004116  012704  006220                  DCHKE1: MOV     #MSG12,R4
004122  005737  000712                          TST     PFLG            ;SEE IF PRINTED DATA ERROR TAG
004126  001004                                  BNE     DCHKE2          ;IF SO: BR
004130  005237  000712                          INC     PFLG
004134  004737  004656                          JSR     PC,TTOUT        ;ELSE PRINT DATA ERROR TAG
004140  012704  006234                  DCHKE2: MOV     #MSG13,R4
004144  004737  004656                          JSR     PC,TTOUT        ;PRINT CHAR NUMBER TAG
004150  013703  000704                          MOV     CRCNT,R3
004154  004737  005054                          JSR     PC,OCTP         ;PRINT CHAR NUMBER
004160  012704  006242                          MOV     #MSG14,R4
004164  004737  004656                          JSR     PC,TTOUT        ;PRINT GOOD TAG
004170  111103                                  MOVB    (R1),R3
004172  004737  005302                          JSR     PC,DOUT         ;PRINT GOOD CHARACTER
004176  012704  006247                          MOV     #MSG15,R4
004202  004737  004656                          JSR     PC,TTOUT        ;PRINT BAD TAG
004206  111203                                  MOVB    (R2),R3
004210  004737  005302                          JSR     PC,DOUT         ;PRINT BAD CHARACTER
004214  000240                                  NOP
004216  000137  004034                          JMP     DCHK1           ;CONTINUE FOR ALL BYTES
004222  000240                          DCHKX:  NOP
004224  005737  000706                          TST     DERFL           ;SEE IF ANY ERROR
004230  001404                                  BEQ     DCHKXX          ;IF NOT: BR
004232  005777  174362                          TST     @SWR            ;SEE IF HALT ON ERROR
004236  100001                                  BPL     DCHKXX          ;IF NOT: BR
004240  000000                                  HALT
004242  004737  005410                  DCHKXX: JSR     PC,CKSWR        ;GO TEST FOR ↑G
004246  000240                                  NOP
004250  005037  000712                          CLR     PFLG            ;CLEAR PRINT FLAG
004254  005037  000706                          CLR     DERFL           ;CLEAR DATA ERROR FLAG
004260  000207                                  RTS     PC              ;RETURN
```

```
                                                ;SCOPE LOOP ON ERROR SUBROUTINE**********
                                                
004262  000240                  SCOPE:  NOP
004264  032777  040000  174326          BIT     #40000,@SWR     ;SEE IF LOOP ON ERROR
004272  001001                          BNE     SCOPE1          ;IF SO: BR
004274  000207                          RTS     PC              ;ELSE EXIT
004276  000240                  SCOPE1: NOP
004300  005726                          TST     (SP)+           ;RESET STACK
004302  000240                          NOP
004304  017703  174350                  MOV     @LTADD,R3
004310  000113                          JMP     (R3)            ;LOOP ON ERROR
                                                
                                                ;TEST ITERATION SUBROUTINE**********
                                                
004312  000240                  ITER:   NOP
004314  004737  005410                  JSR     PC,CKSWR
004320  032777  010000  174272          BIT     #10000,@SWR     ;SEE IF ITERATIONS
004326  001403                          BEQ     ITER1           ;IF SO: BR
004330  005037  000702          ITER0:  CLR     ITCNT           ;CLEAR ITERATION COUNTER
004334  000207                          RTS     PC              ;ELSE EXIT
004336  005737  000670          ITER1:  TST     PCNTR           ;SEE IF FIRST PASS
004342  001772                          BEQ     ITER0           ;IF SO: BR
004344  005237  000702                  INC     ITCNT           ;BUMP COUNTER
004350  023737  000702  000634          CMP     ITCNT,ITAMT     ;SEE IF DONE ALL
004356  001764                          BEQ     ITER0           ;IF SO: BR
004360  005726                          TST     (SP)+           ;RESET STACK
004362  017700  174274                  MOV     @ITRLP,R0       ;SET ITERATION POINTER
004366  000110                          JMP     (R0)            ;GO ITERATE
                                                
                                                ;MAG TAPE INTERRUPT HANDLER**********
                                                
004370  000240                  MTINT:  NOP
004372  022626                          CMP     (SP)+,(SP)+     ;RESET STACK POINTER
004374  042777  000100  174200          BIC     #100,@MTC       ;CLEAR INTERRUPT ENABLE
004402  000240                          NOP
004404  000240                          NOP
004406  000207                          RTS     PC              ;RETURN
                                                
                                                ;TTY INTERRUPT HANDLER**********
                                                
004410  000240                  TTINT:  NOP
004412  000240                          NOP
004414  000240                          NOP
004416  000002                          RTI
```

```
                                        ;*************************************************************
                                        ;TTY ENTRY SUBROUTINE:
                                        ;
                                        ;THIS SUBROUTINE IS USED BY THE TEST CONDITION
                                        ;ENTRY ROUTINE TO READ THE RESPONSE ENTERED
                                        ;AT THE TTY AND CHECK THEM FOR LEGALITY AND
                                        ;LIMITS.  ALL RESPONSE MUST BE TYPED IN OCTAL
                                        ;(0-7) AND MUST FALL WITHIN THE LIMITS SET BY
                                        ;THE CALLING ROUTINE.
                                        ;IF AN ENTRY IS ILLEGAL OR OUTSIDE THE LIMITS,
                                        ;A QUESTION MARK IS TYPED (?) AND THE RESPONSE
                                        ;MAY BE REENTERED.
                                        ;ENTRIES MAY NOT EXCEED SIX (6) CHARACTERS AND
                                        ;MAY BE TERMINATED AT LESS THAN SIZ BY TYPING A
                                        ;CARRIAGE RETURN
                                        ;*************************************************************

004420  005037  000646          TTR:    CLR     TEMP1           ;CLEAR FIRST CHARACTER FLAG
004424  005000                          CLR     R0
004426  004737  004604          TTR0:   JSR     PC,TTIN         ;GO READ CHARACTER
004432  122737  000215  000644          CMPB    #215,TIB        ;SEE IF CR
004440  001005                          BNE     TTR1            ;IF NOT: BR
004442  005737  000646                  TST     TEMP1           ;SEE IF FIRST CHARACTER
004446  001446                          BEQ     TTR5            ;IF SO: BR
004450  000137  004542                  JMP     TTR2            ;ELSE GO LOAD VALUE
004454  122737  000260  000644  TTR1:   CMPB    #260,TIB        ;SEE IF CHAR IS LESS THAN 0
004462  101402                          BLOS    TTR1A           ;IF NOT: BR
004464  000137  004566                  JMP     TINER           ;ELSE GO TO ERROR
004470  122737  000270  000644  TTR1A:  CMPB    #270,TIB        ;SEE IF CHAR IS GREATER THAN 7
004476  101002                          BHI     TTR1B           ;IF NOT: BR
004500  000137  004566                  JMP     TINER           ;ELSE GO TO ERROR
004504  005237  000646          TTR1B:  INC     TEMP1           ;SET FIRST CHARACTER FLAG
004510  000241                          CLC
004512  006100                          ROL     R0
004514  000241                          CLC
004516  006100                          ROL     R0              ;SHIFT 3 LEFT
004520  000241                          CLC
004522  006100                          ROL     R0
004524  042737  177770  000644          BIC     #177770,TIB     ;STRIP ASCII
004532  053700  000644                  BIS     TIB,R0          ;LOAD CHARACTER
004536  005301                          DEC     R1              ;SEE IF DONE
004540  001332                          BNE     TTR0            ;IF NOT: BR
004542  020002                  TTR2:   CMP     R0,R2           ;SEE IF EXCEEDED MAXIMUM LIMIT
004544  101402                          BLOS    TTR3            ;IF OT: BR
004546  000137  004566                  JMP     TINER           ;ELSE GO TO ERROR
004552  020300                  TTR3:   CMP     R3,R0           ;SEE IF BELOW MINIMUM LIMIT
004554  101402                          BLOS    TTR4            ;IF NOT: BR
004556  000137  004566                  JMP     TINER           ;ELSE GO TO ERROR
004562  010015                  TTR4:   MOV     R0,(R5)         ;LOAD VALUE
004564  000207                  TTR5:   RTS     PC              ;EXIT
```

```
                                                ;TTY ENTRY ERROR SUBROUTINE**********
                                                
004566  012704  006254          TINER:  MOV     #MSG16,R4
004572  004737  004656                  JSR     PC,TTOUT        ;PRINT?
004576  162716  000020                  SUB     #20,(SP)        ;RESET SP TO START OF VALUE ROUTINE
004602  000207                          RTS     PC              ;REDO VALUE ENTRY
                                                
                                                ;TTY READ SUBROUTINE**********
                                                
004604  005077  174014          TTIN:   CLR     @TKS
004610  005077  174012                  CLR     @TKB
004614  005037  000644                  CLR     TIB
004620  005277  174000                  INC     @TKS
004624  105777  173774          TTIN1:  TSTB    @TKS
004630  100375                          BPL     TTIN1
004632  017737  173770  000644          MOV     @TKB,TIB
004640  105777  173764          TTIN2:  TSTB    @TPS
004644  100375                          BPL     TTIN2
004646  113777  000644  173756          MOVB    TIB,@TPB
004654  000207                          RTS     PC
                                                
                                                ;TTY OUTPUT SUBROUTINE**********
                                                
004656  112437  000642          TTOUT:  MOVB    (R4)+,TOB
004662  122737  000043  000642          CMPB    #43,TOB
004670  001452                          BEQ     TEX
004672  122737  000045  000642          CMPB    #45,TOB
004700  001407                          BEQ     TCRLF
004702  122737  000041  000642          CMPB    #41,TOB
004710  001443                          BEQ     TBELL
004712  004737  005002                  JSR     PC,TOG
004716  000757                          BR      TTOUT
004720  112737  000015  000642  TCRLF:  MOVB    #15,TOB
004726  004737  005002                  JSR     PC,TOG
004732  012703  000004                  MOV     #4,R3
004736  005037  000642          TCRLFA: CLR     TOB
004742  004737  005002                  JSR     PC,TOG
004746  005303                          DEC     R3
004750  001372                          BNE     TCRLFA          ;DO FILLERS
004752  112737  000012  000642          MOVB    #12,TOB
004760  004737  005002                  JSR     PC,TOG
004764  105737  000724                  TSTB    RDSW
004770  100401                          BMI     1$
004772  000731                          BR      TTOUT
004774  005037  000724          1$:     CLR     RDSW
005000  000406                          BR      TEX
005002  105777  173622          TOG:    TSTB    @TPS
005006  100375                          BPL     TOG
005010  113777  000642  173614          MOVB    TOB,@TPB
005016  000207                  TEX:    RTS     PC
005020  012703  000002          TBELL:  MOV     #2,R3
005024  012737  000007  000642  TBELA:  MOV     #7,TOB
005032  004737  005002                  JSR     PC,TOG
005036  005303                          DEC     R3
005040  001371                          BNE     TBELA
005042  000705                          BR      TTOUT
```

```
1089
1090                                            ;OCTAL OUTPUT SUBROUTINE**********
1091
1092  005044  012737  000001  005300  OCTPE:  MOV     #1,OFL
1093  005052  000402                  BR      OCTPE1
1094  005054  005037  005300          OCTP:   CLR     OFL             ;CLEAR FLAG FOR LEADING ZERO
1095  005060  010304                  OCTPE1: MOV     R3,R4           ;SEE IF NUMBER IS ZERO
1096  005062  001007                  BNE     OCTPO           ;IF NOT ZERO: BR
1097  005064  005737  005300          TST     OFL             ;SEE IF PRINT ALL 0
1098  005070  001004                  BNE     OCTPO           ;IF SO: BR
1099  005072  004737  005260          JSR     PC,OCTPG1       ;ELSE PRINT ZERO
1100  005076  000137  005222          JMP     OCTP3           ;SPACE AND EXIT
1101  005102  032704  100000          OCTPO:  BIT     #100000,R4      ;SEE IF MSD = 1
1102  005106  001406                  BEQ     OCTP1           ;IF NOT: BR
1103  005110  012704  000001          MOV     #1,R4
1104  005114  004737  005236          JSR     PC,OCTPG        ;PRINT 1
1105  005120  000137  005132          JMP     OCTP2
1106  005124  005004                  OCTP1:  CLR     R4
1107  005126  004737  005236          JSR     PC,OCTPG        ;PRINT 0
1108  005132  010304                  OCTP2:  MOV     R3,R4
1109  005134  006004                  ROR     R4
1110  005136  006004                  ROR     R4
1111  005140  006004                  ROR     R4              ;POSITION DIGIT
1112  005142  006004                  ROR     R4
1113  005144  000304                  SWAB    R4
1114  005146  004737  005236          JSR     PC,OCTPG        ;PRINT DIGIT 2
1115  005152  010304                  MOV     R3,R4
1116  005154  006004                  ROR     R4
1117  005156  000304                  SWAB    R4
1118  005160  004737  005236          JSR     PC,OCTPG        ;PRINT DIGIT 3
1119  005164  010304                  MOV     R3,R4
1120  005166  006104                  ROL     R4
1121  005170  006104                  ROL     R4
1122  005172  000304                  SWAB    R4
1123  005174  004737  005236          JSR     PC,OCTPG        ;PRINT DIGIT 4
1124  005200  010304                  MOV     R3,R4
1125  005202  006004                  ROR     R4
1126  005204  006004                  ROR     R4
1127  005206  006004                  ROR     R4
1128  005210  004737  005236          JSR     PC,OCTPG
1129  005214  010304                  MOV     R3,R4
1130  005216  004737  005236          JSR     PC,OCTPG        ;PRINT DIGIT 5
1131  005222  012737  000240  000642  OCTP3:  MOV     #240,TOB
1132  005230  004737  005002          JSR     PC,TOG          ;PRINT SPACE
1133  005234  000207                  RTS     PC              ;EXIT
```

```
1134
1135                                        ;OCTAL PRINT SUBROUTINE**********
1136
1137  005236  042704  177770        OCTPG:  BIC     #177770,R4
1138  005242  001004                        BNE     OCTPGO
1139  005244  005737  005300                TST     OFL
1140  005250  001001                        BNE     OCTPGO
1141  005252  000207                        RTS     PC
1142  005254  005237  005300        OCTPGO: INC     OFL
1143  005260  052704  000260        OCTPG1: BIS     #260,R4
1144  005264  010437  000642                MOV     R4,TOB
1145  005270  004737  005002                JSR     PC,TOG
1146  005274  010304                        MOV     R3,R4
1147  005276  000207                        RTS     PC
1148  005300  000000                OFL:    0                       ;FIRST CHAR FLAG
1149
1150                                        ;DATA CHARACTER OUTPUT SUBROUTINE**********
1151
1152  005302  005037  000642        DOUT:   CLR     TOB
1153  005306  012704  000010                MOV     #10,R4          ;SET NUMBER TO PRINT
1154  005312  110337  000642                MOVB    R3,TOB
1155  005316  105777  173306        DOUT1:  TSTB    @TPS
1156  005322  100375                        BPL     DOUT1
1157  005324  132737  000200  000642        BITB    #200,TOB
1158  005332  001404                        BEQ     DOUT2
1159  005334  012777  000061  173270        MOV     #061,@TPB
1160  005342  000403                        BR      DOUT3
1161  005344  012777  000060  173260  DOUT2:MOV     #060,@TPB
1162  005352  006137  000642        DOUT3:  ROL     TOB
1163  005356  005304                        DEC     R4
1164  005360  001356                        BNE     DOUT1
1165  005362  000207                        RTS     PC
1166  005364  013703  000652        DOUTD:  MOV     TEMP3,R3
1167  005370  000303                        SWAB    R3
1168  005372  004737  005302                JSR     PC,DOUT
1169  005376  013703  000652                MOV     TEMP3,R3
1170  005402  004737  005302                JSR     PC,DOUT
1171  005406  000207                        RTS     PC
1172
1173
1174                                        ;SUBROUTINE TO CHANGE CONTENTS OF SOFTWARE SWITCH REGISTER
1175  005410  022737  000176  000620  CKSWR: CMP     #SWREG,SWR              ;SOFTWARE SWITCH REG PRESENT
1176  005416  001041                        BNE     OUT                     ;NO, GET OUT
1177  005420  105777  173200                TSTB    @TKS                    ;YES,WAIT FOR
1178  005424  100036                        BPL     OUT                     ;READY, GET CHARACTER
1179  005426  017737  173174  000644        MOV     @TKB,TIB                ;AND STRIP OFF
1180  005434  042737  177600  000644        BIC     #177600,TIB             ;THE GARBAGE
1181  005442  022737  000007  000644        CMP     #7,TIB                  ;IS IT A <^G>
1182  005450  001024                        BNE     OUT
1183  005452  012704  006433                MOV     #$CNTG,R4
1184  005456  004737  004656                JSR     PC,TTOUT
1185  005462  012704  006437        CNTLU:  MOV     #$MSWR,R4
1186  005466  004737  004656                JSR     PC,TTOUT
1187  005472  017703  173122                MOV     @SWR,R3
1188  005476  004737  005044                JSR     PC,OCTPE
1189  005502  012704  006451                MOV     #$MNEW,R4
```

```
1190  005506  004737  004656                      JSR     PC,TTOUT
1191  005512  005037  000722                      CLR     @#TEMPST
1192  005516  004737  005524                      JSR     PC,$READ                ;GO READ A LINE
1193  005522  000207                      OUT:    RTS     PC                      ;RETURN TO MAIN BODY OF PROGRAM
1194
1195  005524  005037  000722              $READ:  CLR     TEMPST
1196  005530  012737  000007  000720              MOV     #7,COUNT
1197  005536  004737  004604              1$:     JSR     PC,TTIN         ;GO  READ A CHARACTER
1198  005542  042737  177600  000644              BIC     #177600,TIB     ;STRIP OFF GARBAGE
1199  005550  122737  000025  000644              CMPB    #25,TIB         ;IS IT A ^U?
1200  005556  001002                              BNE     2$              ;BRANCH IF NOT
1201  005560  005726                      3$:     TST     (SP)+           ;POP THE STACK
1202  005562  000737                              BR CNTLU                ;START OVER
1203  005564  122737  000015  000644      2$:     CMPB    #15,TIB         ;IS IT A <CR>?
1204  005572  001013                              BNE     4$              ;BRANCH IF NOT
1205  005574  012737  000200  000724              MOV     #200,RDSW
1206  005602  004737  004720                      JSR     PC,TCRLF        ;ECHO IT WITH <LF>
1207  005606  022737  000007  000720              CMP     #7,COUNT        ;WAS IT FIRST CHARACTER
1208  005614  001037                              BNE     7$              ;CHANGE SWR IF NOT FIRST ONE
1209  005616  005726                      8$:     TST     (SP)+           ;POP THE STACK
1210  005620  000740                              BR      OUT             ;GET OUT
1211  005622  122737  000060  000644      4$:     CMPB    #60,TIB
1212  005630  003004                              BGT     5$
1213  005632  122737  000067  000644              CMPB    #67,TIB
1214  005640  002005                              BGE     6$
1215  005642  012704  006461              5$:     MOV     #$QUEST,R4
1216  005646  004737  004656                      JSR     PC,TTOUT
1217  005652  000742                              BR      2$                      ;START OVER IF NOT LEGAL CHARACTER
1218  005654  006337  000722              6$:     ASL     TEMPST
1219  005660  006337  000722                      ASL     TEMPST
1220  005664  006337  000722                      ASL     TEMPST
1221  005670  142737  000060  000644              BICB    #60,TIB         ;GET NITTY-GRITTY
1222  005676  153737  000644  000722              BISB    TIB,TEMPST
1223  005704  005337  000720                      DEC     COUNT           ;ONLY WANT 6   DIGITS
1224  005710  001754                              BEQ     5$
1225  005712  000711                              BR      1$
1226  005714  013777  000722  172676      7$:     MOV     TEMPST,@SWR     ;CHANGE SWITCH REGISTER CONTENTS
1227  005722  000735                              BR      8$
1228
```

```
                                                        ;MESSAGE TABLE**********
                                                        
005724  022445  046524  026101  MSG1:   .ASCII  /%%TMA,B-11 :TU10,N,W SUPPLEMENTAL INST TEST (DZTMF-D)#/
005732  026502  030461  035040
005740  052524  030061  047054
005746  053454  051440  050125
005754  046120  046505  047105
005762  040524  020114  047111
005770  052123  052040  051505
005776  020124  042050  052132
006004  043115  042055  021451
006012  052445  044516  020124  MSG2:   .ASCII  /%UNIT NUMBER: #/
006020  052516  041115  051105
006026  020072     043
006031     045  020445  047105  MSG3:   .ASCII  /%%!END OF PASS: #/
006036  020104  043117  050040
006044  051501  035123  021440
006052  020445  051041  053505  MSG4:   .ASCII  /%!!REWIND ERROR: NO POT#/
006060  047111  020104  051105
006066  047522  035122  047040
006074  020117  047502  021524
006102  046445  051524  020072  MSG5:   .ASCII  /%MTS: #/
006110     043
006111     045  052115  035103  MSG6:   .ASCII  /%MTC: #/
006116  021440
006120  046445  041124  035103  MSG7:   .ASCII  /%MTBC: #/
006126  021440
006130  046445  041524  035101  MSG8:   .ASCII  /%MTCA: #/
006136  021440
006140  020445  041041  041501  MSG9:   .ASCII  /%!!BACKSPACE ERROR#/
006146  051513  040520  042503
006154  042440  051122  051117
006162     043
006163     045  020441  047516  MSG10:  .ASCII  /%!!NOT READY#/
006170  020124  042522  042101
006176  021531
006200  020445  047041  020117  MSG11:  .ASCII  /%!!NO INTERRUPT#/
006206  047111  042524  051122
006214  050125  021524
006220  042045  052101  020101  MSG12:  .ASCII  /%DATA ERROR#/
006226  051105  047522  021522
006234  041445  035116  021440  MSG13:  .ASCII  /%CN: #/
006242  043445  020072     043  MSG14:  .ASCII  /%G: #/
006247     045  035102  021440  MSG15:  .ASCII  /%B: #/
006254  021477                  MSG16:  .ASCII  /?#/
006256  053445  044522  042524  MSG17:  .ASCII  /%WRITE ERROR#/
006264  042440  051122  051117
006272     043
006273     045  042522  042101  MSG18:  .ASCII  /%READ ERROR#/
006300  042440  051122  051117
006306     043
006307     045  047516  047440  MSG19:  .ASCII  /%NO OPI IN 35 FEET#/
006314  044520  044440  020116
006322  032463  043040  042505
006330  021524
006332  047445  044520  053440  MSG20:  .ASCII  /%OPI WITHIN 16 FEET#/
```

```
1285  006340  052111  044510  020116
1286  006346  033061  043040  042505
1287  006354  021524
1288  006356  020440  047041  052117  MSG21:  .ASCII  / !!NOT AVAILABLE#/
1289  006364  040440  040526  046111
1290  006372  041101  042514     043
1291  006377     045  040503  047116  MSG22:  .ASCII  /%CANNOT TEST LOAD MEDIUM!!%#/
1292  006404  052117  052040  051505
1293  006412  020124  047514  042101
1294  006420  046440  042105  052511
1295  006426  020515  022441     043
1296  006433     045  043536     043  $CNTG:  .ASCII  /%↑G#/
1297  006437     045  020445  051441  $MSWR:  .ASCII  /%%!!SWR= #/
1298  006444  051127  020075     043
1299  006451     040  047040  053505  $MNEW:  .ASCII  /  NEW= #/
1300  006456  020075     043
1301  006461     045  022477  021445  $QUEST: .ASCII  /%?%%#/
1302
1303                                          ;TEST HEADER**********
1304
1305  006466  022445  042524  052123  LT1MSG: .ASCII  /%%TEST 1: WRITE FROM ODD BYTE#/
1306  006474  030440  020072  051127
1307  006502  052111  020105  051106
1308  006510  046517  047440  042104
1309  006516  041040  052131  021505
1310  006524  022445  042524  052123  LT2MSG: .ASCII  /%%TEST 2: READ TO ODD BYTE#/
1311  006532  031040  020072  042522
1312  006540  042101  052040  020117
1313  006546  042117  020104  054502
1314  006554  042524     043
1315  006557     045  052045  051505  LT3MSG: .ASCII  /%%TEST 3: OPI TOO LONG#/
1316  006564  020124  035063  047440
1317  006572  044520  052040  047517
1318  006600  046040  047117  021507
1319  006606  022445  042524  052123  LT4MSG: .ASCII  /%%TEST 4: OPI TOO SHORT#/
1320  006614  032040  020072  050117
1321  006622  020111  047524  020117
1322  006630  044123  051117  021524
1323                                          .EVEN
1324
1325  006636  177777                  WDATA:  -1
1326          006740                          .=.+100
1327  006740  000000                  RDATA:  0
1328          000001                          .END
```

```
BADR    000672          394#    577*    591*    622*    636*    665*    677*    710*    722*    905     843     880
BCNT    000716          404#
BKSP    003212          584     629     671     716     770#
BKSP1   003244          776#    777
BYTES   000E74          395#    579*    592*    623*    637*    666*    678*    711*    723*    906     841     878
CCNT    000614          368#
CKSWR   005410          515     519     552     764     802     829     985     932     952     1175#
CNTLU   005462          474     1185#   1202
COUNT   000720          405#    1196*   1207    1223*
CRCNT   000704          399#    890*    893*    914
DCHK    004022          600     645     889#
DCHKE   004052          892     898#
DCHKE1  004116          903     907#
DCHKE2  004140          909     912#
DCHKX   004222          897     901     926#
DCHKXX  004242          928     930     932#
DCHK0   004030          891#    896
DCHK1   004034          893#    925
DERFL   000706          400#    899*    927     935*
DISPLA  000622          371#    438*
DISPRE  000174          340#    438
DOUT    005302          919     923     1152#   1168    1170
DOUTD   005364          1166#
DOUT1   005316          1155#   1156    1164
DOUT2   005344          1158    1161#
DOUT3   005352          1160    1162#
EMADDR  000654          387#    567*    613*    661*    706*    749     795     822     855     904
ERCHK   003534          582     596     627     641     669     714     727     780     835#
ERCHK1  003546          836     838#
ERCHK2  003560          839     841#
ERCHK3  003602          845     847#
ERPT    003612          687     837     840     848     850#
ERPTX   004002          852     873     882#
ERPTXX  004012          883     885#
ERPT1   003646          854     858#
ERRAD   000656          388#    579*    593*    626*    640*    668*    680*    712*    725*    779*    858
EXEC    003304          580     594     624     638     667     681     713     726     787#
EXECX   003514          819     826#
EXECXX  003524          827     829#
EXEC0   003310          789#    792
EXEC1   003342          794     797#
EXEC2   003362          790     800     802#
EXEC3   003444          814#    815     817
EXEC4   003504          821     824#
FUN     000700          397#    576*    590*    621*    635*    664*    679*    709*    724*    807
HORFL   000710          401#    512*    793     820     853     857*    902     906*
HERE    001670          542     548#
ITAMT   000634          376#    689*    691*    730*    732*    960
ITCNT   000702          398#    955*    959*    960
ITER    004312          601     646     690     731     951#
ITER0   004330          955#    958     961
ITER1   004336          954     957#
ITRLP   000662          390#    507*    508*    534*    535*    963
L^ADD   000660          389#    505*    506*    507     509     513     532*    533*    534     946
LT1     001720          413     414     566#
LT1B    001742          571#    574
```

```
LT1C    002014            581#
LT1D    002044            587#    589
LT1E    002110            595#
LT1MSG  006466            567    1305#
LT2     002146            415     416     612#
LT2B    002164            616#    619
LT2C    002230            625#
LT2D    002266            632#    634
LT2E    002324            639#
LT2MSG  006524            613    1310#
LT3     002370            417     660#
LT3A    002410            664#    673
LT3B    002436            668#
LT3C    002536            682#
LT3IT   002466            418     674#
LT3MSG  006557            661    1315#
LT3X    002574            685     689#
LT4     002620            419     705#
LT4A    002640            709#    718
LT4B    002674            724#
LT4C    002766            727#
LT4IT   002716            420     719#
LT4MSG  006606            706    1319#
MSG1    005724            449    1231#
MSG10   006163            797    1262#
MSG11   006200            824    1265#
MSG12   006220            907    1268#
MSG13   006234            912    1270#
MSG14   006242            916    1271#
MSG15   006247            920    1272#
MSG16   006254           1035    1273#
MSG17   006256            579     626     668     712    1274#
MSG18   006273            593     640    1277#
MSG19   006307            680    1280#
MSG2    006012            451    1240#
MSG20   006332            725    1284#
MSG21   006356            479    1288#
MSG22   006377            443    1291#
MSG3    006031            537    1243#
MSG4    006052            751    1246#
MSG5    006102            753     860    1250#
MSG6    006111            757     864    1252#
MSG7    006120            868    1254#
MSG8    006130            874    1256#
MSG9    006140            779    1258#
MTBA    000606            365#    805#    847     876
MTBC    000604            364#    772#    806#    838     870
MTC     000602            363#    446#    470#    737#    740#    741     759     771#    773#    774     789     804#    809#
                          835     866     970#
MTINT   004370            356     968#
MTS     000600            362#    475     484     684     738     743     745     755     776     862
OCTP    005054            457     540     871     877     881     915    1094#
OCTPE   005044            756     760     863     867    1092#   1188
OCTPE1  005060           1093    1095#
OCTPG   005236           1104    1107    1114    1118    1123    1128    1130    1137#
OCTPGO  005254           1138    1140    1142#
```

```
OCTPG1  005260          1099    1143#
OCTP0   005102          1096    1098    1101#
OCTP1   005124          1102    1106#
OCTP2   005132          1105    1108#
OCTP3   005222          1100    1131#
OFL     005300          1092*   1094*   1097    1139    1142*   1148#
OUT     005522          1176    1178    1182    1193#   1210
PC    =%000007           308#    444*    450*    452*    457*    466*    474*    480*    515*    519*    538*    540*    544*
                         552*    575*    580*    582*    584*    594*    596*    600*    601*    620*    624*    627*    629*
                         638*    641*    645*    646*    663*    667*    669*    671*    675*    681*    687*    690*    707*
                         713*    714*    716*    720*    726*    727*    731*    750*    752*    754*    756*    758*    760*
                         764*    766*    780*    783*    796*    798*    802*    823*    825*    829*    831*    846*    849*
                         856*    859*    861*    863*    865*    867*    869*    871*    875*    877*    881*    885*    905*
                         911*    913*    915*    917*    919*    921*    923*    932*    936*    942*    952*    956*    973*
                        1001*   1031*   1036*   1038*   1052*   1063*   1066*   1069*   1073*   1082*   1085*   1099*   1104*
                        1107*   1114*   1118*   1123*   1128*   1130*   1132*   1133*   1141*   1145*   1147*   1165*   1168*
                        1170*   1171*   1184*   1186*   1188*   1190*   1192*   1193*   1197*   1206*   1216*
PCNTR   000670           393#    495*    539     553*    957
PFLG    000712           402#    908     910*    934*
PSW     000616           369#    428*    811*
RCNT    000612           367#
RDATA   006740           585     591     598     630     636     643     677     722     1327#
RDSW    000724           407#    1074    1077*   1205*
RSTART  001006           351     428#
RWND    003026           575     620     663     675     707     720     736#
RWNDX   003172           748     761#
RWNDXX  003202           739     746     762     764#
RWND1   003062           743#    744
R0    =%000000           301#    346*    350*    447     471*    477*    491*    493*    501*    502*    503     513*    514
                         522*    523*    524     527     530*    531*    533     570*    571     572*    573     586*    588*
                         599*    615*    616     617*    618     631*    633*    644*    662*    672*    708*    717*    895*
                         963*    964     1000*   1015*   1017*   1019*   1021*   1024    1027    1030
R1    =%000001           302#    463*    597*    642*    807*    808*    809     813*    814*    891     894     918     1022*
R2    =%000002           303#    464*    490*    492*    568*    569*    571*    585*    587*    598*    614*    616*    630*
                         632*    643*    891     894     922     1024
R3    =%000003           304#    454*    455*    456*    465*    539*    541*    544     755*    759*    841*    842*    843*
                         847     862*    866*    870*    876*    878*    879*    880*    914*    918*    922*    946*    947
                        1027    1067*   1070*   1083*   1086*   1095    1108    1115    1119    1124    1129    1146    1154
                        1166*   1167*   1169*   1187*
R4    =%000004           305#    443*    449*    451*    479*    537*    749*    751*    753*    757*    795*    797*    822*
                         824*    855*    858*    860*    864*    868*    874*    904*    907*    912*    916*    920*    1035*
                        1056    1095*   1101    1103*   1106*   1108*   1109*   1110*   1111*   1112*   1113*   1115*   1116*
                        1117*   1119*   1120*   1121*   1122*   1124*   1125*   1126*   1127*   1129*   1137*   1143*   1144
                        1146*   1153*   1163*   1183*   1185*   1189*   1215*
R5    =%000005           306#    462*    788*    791*    1030*
SCNT    000676           396#    583*    628*    670*    715*    772
SCOPE   004262           886     939#
SCOPE1  004276           941     943#
SP    =%000006           307#    429*    430*    431*    436     439     440     944     962     969     1037*   1201    1209
SPFLG   000664           391#    686*    688*    778*    781*    844     872
STALL   000636           377#    676*    683*    721*    729*    812
START   001000           347     427#
STFLG   000656           392#    500*    520     526*
STSCD   001556           504     526#
ST0     001144           451#    482
ST1     001304           461     473     475#    478
```

```
ST2     001342          476     484*
ST3     001362          485     488*
ST4     001370          448     487     489*
ST5     001402          492*    494
SUSWR   001020          430*
SWR     000620          370*    427*    433     437*    460*    472     501     516     522     543     747     761     799
                        818     826     851     882     900     929     940     953     1175    1187    1226*
SWREG   000176          341*    437     472     1175
TBELA   005024          1084*   1087
TBELL   005020          1062    1083*
TCRLF   004720          1060    1065*   1206
TCRLFA  004736          1068*   1071
TEMPST  000722          406*    1191*   1195*   1218*   1219*   1220*   1222*   1226
TEMP1   000646          384*    812*    816*    999*    1004    1013*
TEMP2   000650          385*
TEMP3   000652          386*    462     467*    469     1166    1169
TEND    001630          445     483     511     528     537*
TENDX   001704          550     552*
TEX     005016          1058    1078    1082*
TIB     000644          383*    1002    1007    1010    1020*   1021    1044*   1048*   1051    1179*   1180*   1181    1198*
                        1199    1203    1211    1213    1221*   1222
TINER   004566          1009    1012    1026    1029    1035*
TINF    000640          381*
TKB     000626          373*    1043*   1048    1179
TKS     000624          372*    1042*   1045*   1046    1177
TOB     000642          382*    490     1056*   1057    1059    1061    1065*   1068*   1072*   1081    1084*   1131*   1144*
                        1152*   1154*   1157    1162*
TOG     005002          1063    1066    1069    1073    1079*   1080    1085    1132    1145
TPB     000632          375*    1051*   1081*   1159*   1161*
TPS     000630          374*    1049    1079    1155
TSCD    001414          499*    525     554
TSCD0   001444          506*    521
TSCD1   001500          510     512*    536
TSCD2   001512          515*    602     647     692     733
TSCD3   001530          517     519*
TSTTBL  000726          411*    505     532
TTIN    004604          1001    1042*   1197
TTINT   004410          333     977*
TTIN1   004624          1046*   1047
TTIN2   004640          1049*   1050
TTOUT   004656          444     450     452     480     538     750     752     754     758     796     798     823     825
                        856     859     861     865     869     875     905     911     913     917     921     1036    1056*
                        1064    1076    1088    1184    1186    1190    1216
TTR     004420          466     999*
TTR0    004426          1001*   1023
TTR1    004454          1003    1007*
TTR1A   004470          1008    1010*
TTR1B   004504          1011    1013*
TTR2    004542          1006    1024*
TTR3    004552          1025    1027*
TTR4    004562          1028    1030*
TTR5    004564          1005    1031*
T1AD    000732          413*
T11AD   000734          414*
T2AD    000736          415*
T21AD   000740          416*
```

```
T3AD     000742            417#
T3IAD    000744            418#
T4AD     000746            419#
T4IAD    000750            420#
UDES     000610            366#   453#   454    469#   469#   470    486#   488#   737    771    804
UNP      000714            403#
WDATA    006636            568    577    597    614    622    642    665    710    1325#
$CNTG    006433           1183   1296#
$ENDAD   001660            319    544#
$MNEW    006451           1189   1299#
$MSWR    006437           1185   1297#
$QUEST   006461           1215   1301#
$READ    005524           1192   1195#
$SVPC  = 001000            317#   322
.      = 006742            312#   317    318#   320#   322#   332#   339#   345#   349#   355#   359#   423#   1326#
```

```
COMMEN      1#
ENDCOM      1#
ESCAPE      1#
GETPRI      1#
GETSWR      1#
MULT        1#
NEWTST      1#
POP         1#
PUSH        1#
REPORT      1#
SETPRI      1#
SETUP       1#
SKIP        1#
SLASH       1#
STARS       1#    315
SWRSU       1#
TYPBIN      1#
TYPDEC      1#
TYPNAM      1#
TYPNUM      1#
TYPOCS      1#
TYPOCT      1#
TYPTXT      1#
$$ESCA      1#
$$NEWT      1#
$$SKIP      1#
.EQUAT      1#
.HEADE      1#
.KT11       1#
.SETUP      1#
.SWRHI      1#
.$ACTI      1#    295#    313
.$APTB      1#
.$APTH      1#
.$APTY      1#
.$ASTA      1#
.$CATC      1#
.$CMTA      1#
.$DB2D      1#
.$DB2O      1#
.$DIV       1#
.$EOP       1#
.$ERRO      1#
.$ERRT      1#
.$MULT      1#
.$POWE      1#
.$RAND      1#
.$RDDE      1#
.$RDOC      1#
.$READ      1#
.$R2AZ      1#
.$SAVE      1#
.$SB2D      1#
.$SB2O      1#
.$SCOP      1#
.$SIZE      1#
```

DZTMFD.SRC CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
ADD      506   508   533   535   843   880
ASL     1218  1219  1220
BEQ      434   459   482   485   517   521   525   542   744   777   839   845   928   954   958
         961  1005  1058  1060  1062  1102  1158  1224
BGE     1214
BGT     1212
BHI     1011
BIC      456   468   502   523   970  1020  1137  1180  1198
BICB    1221
BIS      469   486   488   740   773   808   809  1021  1143
BISB    1222
BIT      475   484   516   549   684   738   743   745   747   776   789   818   851   900   940
         953  1101
BITB    1157
BLE      528
BLOS    1008  1025  1028
BMI     1075
BNE      442   448   473   476   478   494   504   510   550   574   589   619   634   673   685
         718   739   746   748   790   792   794   815   817   819   821   848   852   854   873
         892   896   901   903   909   941  1003  1023  1071  1087  1096  1098  1138  1140  1164
        1176  1182  1200  1204  1208
BPL      742   762   775   800   827   836   883   930  1047  1050  1080  1156  1178
BR       435   461   487   536  1064  1076  1078  1088  1093  1160  1202  1210  1217  1225  1227
CLC      529  1014  1016  1018
CLR      346   453   471   492   495   500   512   570   615   688   781   788   811   813   890
         934   935   955   999  1000  1042  1043  1044  1068  1077  1094  1106  1152  1191  1195
CMP      433   436   472   527   573   618   847   960   969  1024  1027  1175  1181  1207
CMPB     441   891   894  1002  1007  1010  1057  1059  1061  1199  1203  1211  1213
DEC      477   493   588   633   672   717   791   814   816   895  1022  1070  1086  1163  1223
HALT     312   518   551   763   801   828   884   931
INC      553   572   617   857   893   910   959  1013  1045  1142
JMP      347   351   445   483   511   514   554   602   647   692   733   837   840   886   897
         925   947   964  1006  1009  1012  1026  1029  1100  1105
JSR      444   450   452   457   466   474   480   515   519   538   540   544   552   575   580
         582   584   594   596   600   601   620   624   627   629   638   641   645   646   663
         657   669   671   675   681   687   690   707   713   714   716   720   726   727   731
         750   752   754   756   758   760   764   780   796   798   802   823   825   829   856
         859   861   863   865   867   869   871   875   877   881   885   905   911   913   915
         917   919   921   923   932   952  1001  1036  1063  1066  1069  1073  1085  1099  1104
        1107  1114  1118  1123  1128  1130  1132  1145  1168  1170  1184  1186  1188  1190  1192
        1197  1206  1216
MOV      350   427   428   429   430   431   432   437   438   439   440   443   446   449   451
         454   460   462   463   464   465   470   479   490   491   501   505   507   513   522
         526   532   534   537   539   541   567   568   576   577   578   579   583   585   586
         587   590   591   592   593   597   598   599   613   614   621   622   623   626   628
         630   631   632   635   636   637   640   642   643   644   661   662   664   665   666
         668   670   676   677   678   679   680   683   686   689   691   706   708   709   710
         711   712   715   721   722   723   724   725   729   730   732   737   749   751   753
         755   757   759   771   772   778   779   795   797   804   805   806   807   812   822
         824   841   855   858   860   862   864   866   868   870   874   876   878   899   904
         906   907   912   914   916   920   946   963  1030  1035  1048  1067  1083  1084  1092
        1095  1103  1108  1115  1119  1124  1129  1131  1144  1146  1153  1159  1161  1166  1169
        1179  1183  1185  1187  1189  1196  1205  1215  1226
MOVB     569   571   616   918   922  1051  1056  1065  1072  1081  1154
NEG      842   879
NOP      489   499   545   546   547   548   566   581   595   612   625   639   660   674   692
```

N03


```
            705    719    728    736    765    770    782    787    803    810    830    850    889    898    924
            926    933    939    943    945    951    968    971    972    977    978    979
RESET       543
ROL         530    531   1015   1017   1019   1120   1121   1162
ROR        1109   1110   1111   1112   1116   1125   1126   1127
RTI         980
RTS         766    783    831    846    849    936    942    956    973   1031   1038   1052   1082   1133   1141
           1147   1165   1171   1193
SUB        1037
SWAB        455    467   1113   1117   1122   1167
TST         447    458    481    503    509    520    524    761    793    799    820    826    835    838    844
            853    872    882    902    908    927    929    944    957    962   1004   1097   1139   1201   1209
TSTB        741    774   1046   1049   1074   1079   1155   1177
.ASCII     1231   1240   1243   1246   1250   1252   1254   1256   1258   1262   1265   1268   1270   1271   1272
           1273   1274   1277   1280   1284   1288   1291   1296   1297   1299   1301   1305   1310   1315   1319
.ENABL        1    294
.END       1328
.ENDC       316    320    322
.EVEN      1323
.IF         315    318    320
.IFF        316    320    322
.LIST         1    288    312
.MACRO        1
.MCALL      295
.NLIST        1    288    312
.REM          4
.REPT       312
.SBTTL      313
.TITLE      290
.WORD       321


ERRORS DETECTED:  0
DEFAULT GLOBALS GENERATED:  0

*,DZTMFD.SEQ/SOL/CRF/PAGNUM/NL:TOC=DZTMFD.SML,DZTMFD.SRC
RUN-TIME:  23 27 2 SECONDS
RUN-TIME RATIO: 117/52=2.2
CORE USED:  32K  (63 PAGES)
```

B04

Spooler prolog 5 Seconds, 26 KCS, 132 disk reads, 3 disk writes, 39 pages