DM11A

DATA TEST

MD-11-DZDMB-B

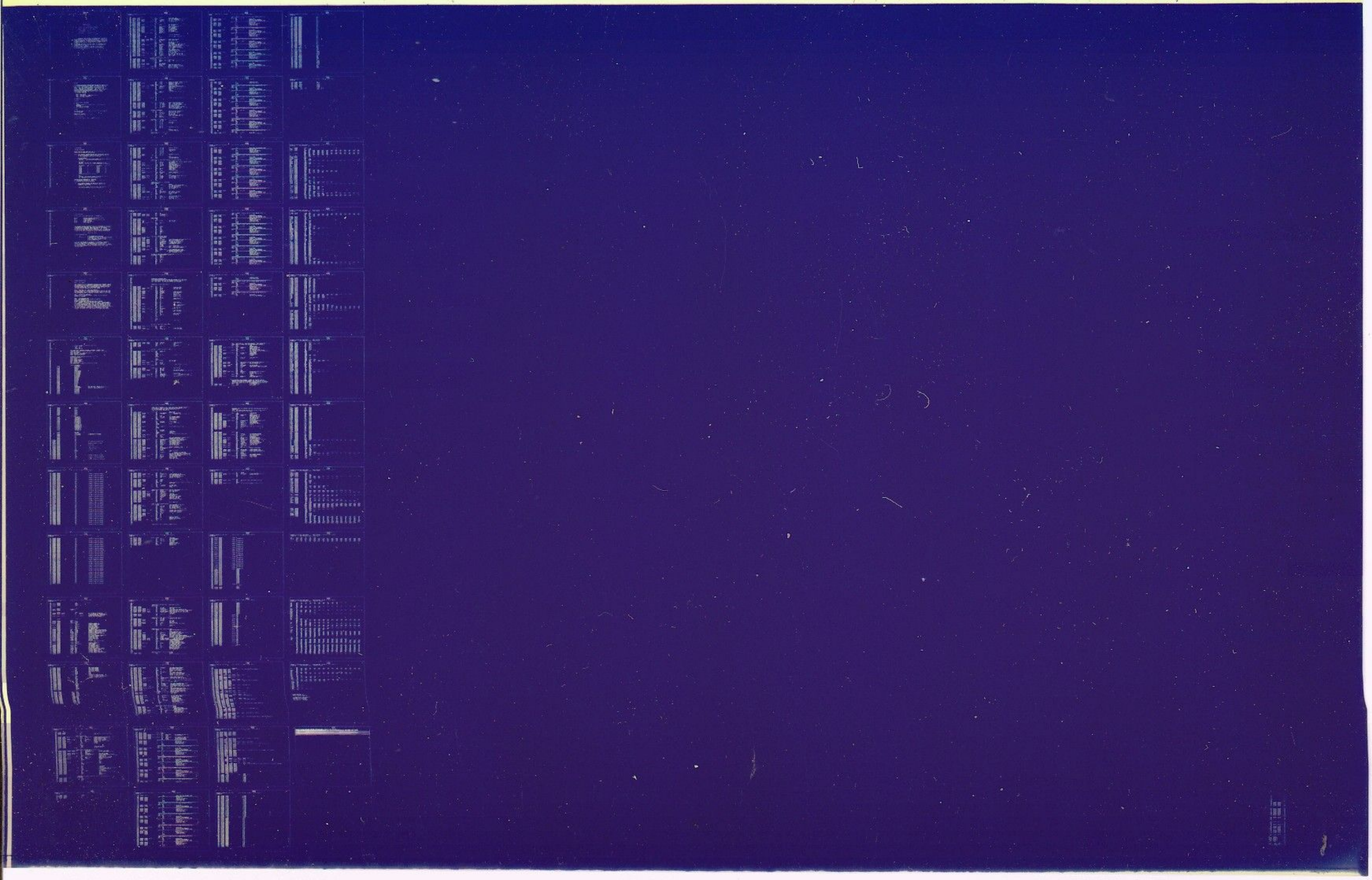
EP-DZDMB-B-DL-A

NOV 1976

digital

MADE IN USA

FICHE 1 OF 1



```
.REM %
```

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDMB-B-D

PRODUCT NAME: DM11 DATA TESTS

DATE RELEASED: MAY,1976

MAINTAINER: DIAGNOSTIC GROUP

1. ABSTRACT

THIS SEPARATE DIAGNOSTIC PROGRAMS ARE PROVIDED FOR TESTING THE DM11 (ASYNCHRONOUS DATA MULTIPLEXER), MAINDEC-11-DZDMA (DM11 LOGIC TESTS), AND MAINDEC-11-DZDMB (DM11 MULTIPLE LINE DATA TESTS). THE LOGIC TESTS INDIVIDUALLY TEST EACH OF THE 16 DM11 LINES AND ALL COMMON LOGIC. THE MULTIPLE LINE DATA TESTS RUN SEVERAL LINES CONCURRENTLY AND ARE USED TO TEST LINE INTERACTION AND DATA TRANSMISSION/RECEPTION RELIABILITY.
THIS DOCUMENT DESCRIBES THE MULTIPLE LINE DATA TESTS.
THE AVAILABLE TESTS ARE:

PRG0 - DATA TESTS
PRG1 - DATA TEST(ALL LINES SIMULTANEOUSLY)
PRG2 - TRANSMIT TO TERMINALS
PRG3 - ECHO RECEIVED DATA

2. REQUIREMENTS

2.1 EQUIPMENT

A. PDP 11 FAMILY PROCESSER
B. DM11
C. JUMPERS CONNECTING 16 TRANSMITTERS TO THEIR RESPECTIVE RECEIVERS.
D. TERMINALS (IF AVAILABLE)
E. DM11 DISTRIBUTION PANEL

2.2 STORAGE

THIS PROGRAM USES ALL OF CORE (4K) EXCEPT THAT AREA RESERVED FOR THE LOADERS.

2.3 PREREQUISITE PROGRAMS
MAINDEC-11-DZDMA (DM11 LOGIC TESTS)

3. LOADING PROCEDURE

THE ABSOLUTE LOADER IS USED TO LOAD THE PROGRAM.

4. USE PROCEDURE

4.1 STARTING PROCEDURE

BEFORE STARTING MAKE SURE THAT THE TTY IS IN REMOTE MODE.
THREE STARTING ADDRESSES ARE PROVIDED.

0200 - THIS STARTING ADDRESS REQUESTS DM11 PARAMETERS, AND MUST BE USED TO INITIALLY START THE PROGRAM, AND WHENEVER ANY OF THE PARAMETERS LISTED BELOW IS CHANGED.

A. VECTOR ADDRESS ?
RESPONSE: TYPE IN THE VECTOR ADDRESS OF THE DM11 RECEIVER UNDER TEST. CARRIAGE RETURN SELECTS 0300

B. UNIT #(0)?
RESPONSE: THE DM11 UNIT NUMBER CORRESPONDS TO THE ADDRESS TO WHICH THE CLOCK STATUS REGISTER (CSR) RESPONDS.

CSR ADDRESS	DM11 UNIT #	CSR ADDRESS	DM11 UNIT #
175000	0	175100	10
175010	1	175110	11
175020	2	175120	12
175030	3	175130	13
175040	4	175140	14
175050	5	175150	15
175060	6	175160	16
175070	7	175170	17

CARRIAGE RETURN SELECTS UNIT # 0.

C. PRG #
RESPONSE: TYPE PROGRAM NUMBER OF PROGRAM YOU WISH TO RUN. CARRIAGE RETURN SELECTS PROGRAM # 0.

CARRIAGE RETURN TERMINATES ALL RESPONSES.
ANY UNACCEPTABLE RESPONSE WILL RESULT IN A ? TYPEOUT AND THE PARAMETER WILL AGAIN BE REQUESTED.

0204 - THIS STARTING ADDRESS USES PREVIOUSLY DEFINED DM11 PARAMETERS AND REQUESTS THE PROGRAM NUMBER OF THE PROGRAM YOU WISH TO RUN.

0210 - THIS STARTING ADDRESS STARTS THE PREVIOUSLY SELECTED PROGRAM USING PREVIOUSLY SELECTED PARAMETERS.

4.2 SWITCH SETTINGS

THE FOLLOWING SWITCH SETTINGS APPLY TO PROGRAM #0.

SR 0-6	ROUTINE TO BE RUN (IF ENABLED BY SR-9)
SR 9	LOOP SELECTED ROUTINE
SR 11	INHIBIT ITERATION (DO EACH ROUTINE ONCE)
SR 13	INHIBIT PRINTOUT
SR 14	SCOPE (LOOP ROUTINE)
SR 15	HALT ON ERROR

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. 176) 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.

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.

5.0 PROGRAM DESCRIPTIONS

5.1 PRG0 - LOGIC TESTS

PRG0 CONSISTS OF 34(8) INDEPENDENT ROUTINES WHICH TRANSMIT VARIOUS DATA PATTERNS ON ALL LINES WITH A DECREASING DELAY BEFORE STARTING SUCCESSIVE LINES. THE DATA IS CHECKED WHEN ALL TRANSMITTERS HAVE COMPLETED TRANSMITTING. IF A DATA ERROR OCCURS THE ERROR TYPEOUT WILL SHOW THE DATA FAILURE AND THE LINE NUMBER.

5.2 PRG1 - DATA TEST (ALL LINES SIMULTANEOUSLY)
PROGRAM 1 TRANSMITS ' A QUICK BROWN FOX JUMPED OVER THE LAZY DOGS BACK 1234567890' ON ALL LINES SIMULTANEOUSLY. WHEN ALL LINES HAVE FINISHED RECEIVED DATA IS VERIFIED. AN ERROR TYPEOUT IS AS IN PRG0.

5.3 PRG2 - TRANSMIT TO TERMINALS
PROGRAM 2 IS THE SAME AS PROGRAM 1 EXCEPT THAT THE RECEIVED DATA IS NOT CHECKED.

5.4 PRG3 - ECHO RECEIVED DATA
NOTE: THIS PROGRAM MAY ONLY BE RUN IF USING AN ASR 33 NOT MODIFIED BY DEC.
PROGRAM 3 ECHOES BACK DATA RECEIVED FROM A TERMINAL.
NOTE: PROGRAM 3 SHOULD BE RUN AND DATA TYPED AT ALL AVAILABLE DM11 TERMINALS. IT IS THE ONLY TEST THAT INSURES CORRECT OPERATION OF THE DM11 DISTRIBUTION PANEL LOGIC. IF THE TERMINALS ARE ASR-33 WITH A PAPER TAPE READER/PUNCH I SUGGEST THAT INDIVIDUAL TAPES BE MADE UP FOR EACH LINE. THIS CAN BE DONE BY RUNNING PROGRAM 2 WITH THE PUNCH TURNED ON. PROGRAM 2 WILL THEN PUNCH A TAPE ON EACH TERMINAL WITH THE LINE NUMBER IDENTIFIER AT THE BEGINNING OF EACH TAPE. PROGRAM 3 CAN BE RUN WITH THESE TAPES IN THE PAPER TAPE READERS.

```
                                %

                                        .TITLE  MAINDEC-11-DZDMB DM11 DATA TESTS
                                        .NLIST  MC,MD
                                        .LIST   ME
                                        .ENABLE ABS,AMA

                                ;DM11 DATA TESTS DIAGNOSTIC MAINDEC-11-DZDMB-B (FORMERLY D9GB)
                                ;COPYRIGHT 1971, 1976 DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
                                ;PRG0- DATA TESTS
                                ;PRG1- DATA TEST (ALL LINES SIMULTANEOUSLY)
                                ;PRG2- TRANSMIT TO TERMINALS
                                ;PRG3-  ECHO RECEIVED DATA

                                ;STANDARD SR SWITCH OPTIONS (SWITCH SET TO A 1 (UP))
                                ;SR15- HALT ON ERROR
                                ;SR14- SCOPE.
                                ;SR13- INHIBIT PRINTOUT
                                ;SR12- INHIBIT TRACE
                                ;SR11- INHIBIT ITERATION.
                                ;SR9- LOOP ROUTINE.
                                ;SR6 THROUGH SR0 - NUMBER OF ROUTINE TO BE LOOPED.

                                ;EQUATE STATEMENTS
177776                                  CC=177776
177776                                  PSW=177776
000004                                  ERRVEC=4
000240                                  NOP=240
000000                                  OPEN=0
100000                                  MANUAL=BIT15
100000                                  BIT15=100000
040000                                  BIT14=40000
020000                                  BIT13=20000
010000                                  BIT12=10000
004000                                  BIT11=4000
002000                                  BIT10=2000
001000                                  BIT9=1000
000400                                  BIT8=400
000200                                  BIT7=200
000100                                  BIT6=100
000040                                  BIT5=40
000020                                  BIT4=20
000010                                  BIT3=10
000004                                  BIT2=4
000002                                  BIT1=2
000001                                  BIT0=1
005726                                  POPSP=5726              ;POP THE STACK. SAME AS TST (SP)+
022626                                  POPSP2=022626           ;POP STACK TWICE. SAME AS CMP (SP)+, (SP)+
000340                                  PRTY7=340               ;PRIORITY LEVEL DEFINITIONS
000300                                  PRTY6=300
000240                                  PRTY5=240
000200                                  PRTY4=200
000140                                  PRTY3=140
000100                                  PRTY2=100
000040                                  PRTY1=40
```

```
                                        PRTY0=0
                                        R0=%0
                                        R1=%1
                                        R2=%2
                                        R3=%3
                                        R4=%4
                                        R5=%5
                                        SP=%6
                                        PC=%7
                                ;EMT CALLS
                                        TYPE=EMT+0
                                        TYPES=EMT+1
                                        STALL=EMT+2
                                        ERROR=EMT+3
                                        DATCHK=EMT+4
                                        STRXV=EMT+6
                                        STTXV=EMT+7
                                        EHALT=EMT+10
                                        SCOPE=EMT+12
                                        SAVREG=EMT+13
                                        RSTREG=EMT+14
                                        ERROR1=EMT+15
                                        SUSWR=EMT+16
                                        KBDIN=EMT+17
                                        CNTLU=EMT+20

                                        BELL=007
                                        ATLAST=-1
                                        ALT0=125252             ;ALTERNATING 0'S PATTERN
                                        ALT1=052525             ;ALTERNATING 1'S PATTERN
                                        Y=0
                                        X=-1
                                        A=0
                                        .=0
000000  000000                          HALT
000002  000000                          HALT
000004  000006                          .+2                     ;SP OVERFLOW, BUS ERROR TRAP
000006  000000                          HALT
000010  000012                          .+2                     ;RESERVED INSTRUCTION TRAP
000012  000000                          HALT
000014  000016                          .+2                     ;TRACE TRAP
000016  000000                          HALT
000020  000022                          .+2                     ;TRAP TO CALL IOX
000022  000000                          HALT
000024  000026                          .+2                     ;POWER FAIL TRAP
000026  000000                          HALT
000030  002514                          EMTINT                  ;EMT TRAP
000032  000340                          PRTY7
000034  000036                          .+2
000036  000000                          HALT
000040  000042                          .+2
000042  000000                          HALT                    ;TRAPPED TO PREVIOUS ADDRESS.
000044  000046                          .+2
000046  000000                          HALT                    ;TRAPPED TO PREVIOUS ADDRESS.
000050  000052                          .+2
000052  000000                          HALT                    ;TRAPPED TO PREVIOUS ADDRESS.
```

```
  315  000054  000056                .+2
  316  000056  000000                HALT                ;TPAPPED TO PREVIOUS ADDRESS.
  317  000060  000062                .+2
  318  000062  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  319  000064  000056                .+2
  320  000066  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  321  000070  000072                .+2
  322  000072  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  323  000074  000076                .+2
  324  000076  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  325  000100  000102                .+2
  326  000102  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  327  000104  000106                .+2
  328  000106  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  329  000110  000112                .+2
  330  000112  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  331  000114  000116                .+2
  332  000116  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  333  000120  000122                .+2
  334  000122  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  335  000124  000126                .+2
  336  000126  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  337  000130  000132                .+2
  338  000132  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  339  000134  000136                .+2
  340  000136  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  341  000140  000142                .+2
  342  000142  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  343  000144  000146                .+2
  344  000146  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  345  000150  000152                .+2
  346  000152  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  347  000154  000156                .+2
  348  000156  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  349  000160  000162                .+2
  350  000162  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  351  000164  000166                .+2
  352  000166  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  353  000170  000172                .+2
  354  000172  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  355  000174  000176                .+2
  356  000176  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  357  000200  000202                .+2
  358  000202  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  359  000204  000206                .+2
  360  000206  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  361  000210  000212                .+2
  362  000212  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  363  000214  000216                .+2
  364  000216  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  365  000220  000222                .+2
  366  000222  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  367  000224  000226                .+2
  368  000226  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
  369  000230  000232                .+2
  370  000232  000000                HALT                ;TRAPPED TO PREVIOUS ADDRESS.
```

```
371  000234  000236                  .+2
372  000236  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
373  000240  000242                  .+2
374  000242  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
375  000244  000246                  .+2
376  000246  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
377  000250  000252                  .+2
378  000252  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
379  000254  000256                  .+2
380  000256  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
381  000260  000262                  .+2
382  000262  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
383  000264  000266                  .+2
384  000266  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
385  000270  000272                  .+2
386  000272  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
387  000274  000276                  .+2
388  000276  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
389  000300  000302                  .+2
390  000302  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
391  000304  000306                  .+2
392  000306  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
393  000310  000312                  .+2
394  000312  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
395  000314  000316                  .+2
396  000316  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
397  000320  000322                  .+2
398  000322  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
399  000324  000326                  .+2
400  000326  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
401  000330  000332                  .+2
402  000332  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
403  000334  000336                  .+2
404  000336  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
405  000340  000342                  .+2
406  000342  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
407  000344  000346                  .+2
408  000346  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
409  000350  000352                  .+2
410  000352  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
411  000354  000356                  .+2
412  000356  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
413  000360  000362                  .+2
414  000362  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
415  000364  000366                  .+2
416  000366  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
417  000370  000372                  .+2
418  000372  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
419  000374  000376                  .+2
420  000376  000000                  HALT                ;TRAPPED TO PREVIOUS ADDRESS.
421
```

```
                000046                          .=46                    ;ACT11 HOOKS
000046  002452                                 $ENDAD
                000052                          .=52
000052  020000                                 020000


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

                000200                          .=200
000200  000137  002076                         JMP     @#START         ;GO TO START OF DIAGNOSTIC.
000204  000137  002114                         JMP     @#RSTAT1        ;GO GET PROGRAM # & RESTART PROGRAM
                                                                        ;USING PREVIOUS DM11 PARAMETERS
000210  000137  002154                         JMP     @#RSTAT2        ;RESTART PREVIOUS PROGRAM USING
                                                                        ;PREVIOUS DM11 PARAMETERS

                001100                          .=1100

001100  000000                  SPBOT:  0
001102  177570                  SWR:    177570
001104  177570                  DISPLAY:177570
001106  000000                  CAT:    OPEN                            ;STARTING ADDRESS OF
                001146                          .=CAT+32.               ;CURRENT ADDRESS TABLE
001146  000000                  WCT:    OPEN                            ;STARTING ADDRES OF
                001206                          .=WCT+32.               ;WORD COUNT TABLE
001206  000000                  BAT:    OPEN                            ;STARTING ADDRESS OF
                001246                          .=BAT+32.               ;BIT ASSEMBLY TABLE
001246  000000                  VAC:    OPEN                            ;32. SPARE WORDS
001250  175000                  CSR:    175000                          ;ADDRESS OF CLOCK STATUS REGISTER
001252  175002                  BAR:    175002                          ;ADDRESS OF BUFFER ACTIVE REGISTER
001254  175004                  BKCSR:  175004                          ;ADDRESS OF BREAK STATUS REGISTER
001256  175006                  BASREG: 175006                          ;ADDRESS OF BASE REGISTER
001260  000000                  CLKINT: OPEN                            ;DM11 VECTOR ADDRESS (RECEIVER)
001262  000240                  CLKLVL: PRTY5                           ;PRIORITY LEVEL
001264  000000                  XMTINT: OPEN                            ;DM11 VECTOR ADDRESS (TRANSMITTER)
001266  000240                  XMTLVL: PRTY5                           ;TRANSMITTER PRIORITY LEVEL
001270  000000                  BARIM:  OPEN                            ;PROGRAM BAR IMAGE
001272  000000                  TTDAT:  OPEN                            ;TUMBLE TABLE DATA
001274  000000                  LINBIT: OPEN                            ;LINE BIT (FOR BAR)
001276  000000                  BARDAT: OPEN                            ;BAR DATA
001300  000000                  TTPTR:  OPEN                            ;PROGRAM TUMBLE TABLE POINTER
                001306                          .=VAC+32.
001306  000000                  TUMTAB: OPEN                            ;STARTING ADDRESS OF
                001506                          .=TUMTAB+128.           ;TUMBLE TABLE
001506  000060                  TKVTR:  60                              ;LSR INTERRUPT VECTOR
001510  000200                  TKLVL:  PRTY4                           ;LSR PRIORITY LEVEL
001512  000064                  TPVTR:  64                              ;LSP INTERRUPT VECTOR
001514  000200                  TPLVL:  PRTY4                           ;LSP PRIORITY LEVEL
001516  000000                  KSTART: OPEN                            ;CURRENT PROGRAM START ADDRESS.
001520  000000                  CURTST: OPEN                            ;CONTAINS ADDR OF CURRENT TEST.
001522  000000                  RTNNO:  OPEN                            ;CONTAINS CURRENT TEST #.
001524  000000                  NXTST:  OPEN                            ;CONTAINS ADDR OF NEXT TEST.
001526  000000                  ICTR:   OPEN                            ;CONTAINS CURRENT ITERATION COUNT
001530  000000                  SCOPTR: OPEN                            ;CONTAINS CURRENT SCOPE POINTER.
001532  177774                  PRGLIM: -4
```

```
   478  001534  005120  PRGTAB: PRG0                    ;PRG0 START ADDRESS
   479  001536  006100          PRG1                    ;PRG1 START ADDRESS
   480  001540  006264          PRG2                    ;PRG2 START ADDRESS
   481  001542  006274          PRG3                    ;PRG3 START ADDRESS
   482  001544  005142  RSTART: PRG0R                   ;PRG0 RESTART ADDRESS
   483  001546  006116          PRG1R                   ;PRG1    "       "
   484  001550  006270          PRG2R                   ;PRG2    "       "
   485  001552  006312          PRG3R                   ;PRG3    "       "
   486  001554  002650  EMTTAB: TYP                     ;POINTER TO TYPEOUT ROUTINE
   487  001556  000000          OPEN                    ;POINTER TO CHAINED MESSAGES ROUTINE
   488  001560  000000          OPEN                    ;POINTER TO RANDOM STALL ROUTINE
   489  001562  001726          ERR                     ;POINTER TO ERROR ROUTINE
   490  001564  001660          DTCHK
   491  001566  000000          OPEN
   492  001570  000000          OPEN
   493  001572  000000          OPEN
   494  001574  000000          OPEN
   495  001576  000000          OPEN
   496  001600  002322          ESCOPE
   497  001602  002546          SAVRG
   498  001604  002606          RSTRG
   499  001606  001744          ERR1
   500  001610  003652          SUSWRR
   501  001612  003526          KBDINTT
   502  001614  003572          CNTLUU
   503
   504  001616  000000  SRT:    OPEN
   505  001620  177560  TKCSR:  177560
   506  001622  177562  TKDBR:  177562
   507  001624  177564  TPCSR:  177564
   508  001626  177566  TPDBR:  177566
   509  001630  000000  RCVDAT: OPEN
   510  001632  000000  XMTDAT: OPEN
   511  001634  000000  CARMSK: OPEN
   512  001636  000000  TEMP:   OPEN
   513  001640  000000  PCADD:  OPEN
   514  001642  000000  APCADD: OPEN
   515  001644  000000  PRVCNT: OPEN
   516  001646  000000  LINE:   OPEN
   517  001650  000000  LINBUF: OPEN
```

```
                                INCRTN: TYPE
001652  104000                          M1                      ;TYPE INCORRECT ROUTINE SELECTED.
001654  012540                          RTS     %7              ;EXIT.
001656  000207

                                ;DATA CHECK ROUTINE.
001660  123737  001630  001632  DTCHK:  CMPB    RCVDAT,XMTDAT   ;COMPARE EXPECTED AND RECEIVED
001666  001416                          BEQ     1$              ;CHARS. BRANCH IF SAME.
001670  004737  002050                  JSR     7,CNVDAT        ;CONVERT RCVDAT & XMTDAT TO ASCII
001674  032777  020000  177200          BIT     #BIT13,@SWR     ;ERROR TYPEOUT DESIRED?
001702  001010                          BNE     1$              ;BRANCH IF NO TYPEOUT DESIRED
001704  004537  004352                  JSR     5,@#OACNV       ;CONVERT LINE
001710  001646                          LINE                    ;NUMBER
001712  012514                          ALINE                   ;TO ASCII
001714  000002                          2
001716  104015                          ERROR1
001720  104000                          TYPE                    ;TYPE LINE # AS PART
001722  012505                          LINEM                   ;OF ERROR MESSAGE
001724  000002                  1$:     RTI                     ; EXIT.

                                ;ERROR SERVICE ROUTINE CALLED BY TRAP (HLT)
001726  012737  000402  002026  ERR:    MOV     #402,ERRB               ;MOV BR .+6 TO ERRB
001734  013737  001640  001642          MOV     @#PCADD,@#APCADD        ;GET PC WHERE ERROR OCCURRED
001742  000410                          BR      ERRA
001744  012737  000240  002026  ERR1:   MOV     #240,ERRB               ;MOVE NOP TO ERRB
001752  013737  001640  001642          MOV     @#PCADD,@#APCADD        ;GET PC WHERE ERROR OCCURRED
001760  004737  002050                  JSR     7,@#CNVDAT              ;CONVERT RCVDAT & XMIT DAT TO ASCII
001764  104017                  ERRA:   KBDIN                           ;CHECK FOR ^G
001766  032777  020000  177106          BIT     #BIT13,@SWR             ;ERROR PRINTOUT DESIRED
001774  001017                          BNE     ERRC                    ;BRANCH IF NO PRINTOUT
001776  004537  004352                  JSR     5,@#OACNV               ;CONVERT
002002  001642                          APCADD                          ;DATA
002004  013174                          APC                             ;TO
002006  000006                          6                               ;ASCII
002010  004537  004352                  JSR     5,@#OACNV               ;FOR
002014  001522                          RTNNO                           ;PRINTOUT
002016  013164                          ATNUMB
002020  000003                          3
002022  104000                          TYPE                            ;TYPE ERROR
002024  013161                          EMO                             ;MESSAGE
002026  000000                  ERRB:   OPEN                            ;NOP IF ERROR1, BR .+6 IF ERROR
002030  104000                          TYPE                            ;TYPE ANOTHER MESSAGE
002032  012440                          ERDAT                           ;IF ERROR 1
002034  005777  177042          ERRC:   TST     @SWR                    ;HALT ON ERROR
002040  100001                          BPL     1$                      ;GO TO EXIT IF NO HALT ON ERROR
002042  000000                          HALT                            ;HALT
002044  104017                  1$:     KBDIN                           ;CHECK FOR ^G
002046  000002                          RTI                             ;RETURN

                                ;SUBROUTINE TO CONVERT RCVDAT AND XMTDAT TO ASCII AND PLACE
                                ;IN MESSAGE.
002050  004537  004352          CNVDAT: JSR     5,OACNV
002054  001632                          XMTDAT
002056  012460                          AASB
002060  000006                          6
002062  004537  004352                  JSR     5,OACNV
```

```
574  002066  001630                  RCVDAT
575  002070  012475                  AWAS
576  002072  000006                  6
577  002074  000207                  RTS     7               ;EXIT
578
579
```

```
                                                ;INITIALIZE STACK
002002  012706  001100          START:  MOV     #SPBOT,%6
002006  104016                  SUSWR           ;CHECK FOR HARDWARE SWITCH REGISTER
002010  004737  003764          JSR     2,@#DMPAR       ;GET DM11 PARAMETERS
002014  004737  003600          JSR     2,@#OVRLAY      ;PUT HALT,.+2 IN VECTOR AREA
002020  012706  001100  RSTAT1: MOV     #SPBOT,%6       ;INITIALIZE STACK
002024  104000                  TYPE
002026  012620                  MO
002030  004537  003346          JSR     5,RECD          ;GET THE PRGNUM &
002034  000000  PRGNUM: 0                               ;PUT IT HERE
002036  043737  001532  002130  BIC     PRGLIM,PRGNUM   ;MASK OFF UNUSED BITS
002044  006337  002130          ASL     PRGNUM          ;SHIFT PROGRAM #
002050  013700  002130          MOV     PRGNUM,%0       ;GET PROGRAM #
002054  000170  001534          JMP     @PRGTAB(0),     ;GO START PROGRAM
002060  012706  001100  RSTAT2: MOV     #SPBOT,%6       ;INITIALIZE STACK
002064  013700  002130          MOV     PRGNUM,%0       ;GET PROGRAM #
002070  000170  001544          JMP     @RSTART(0)      ;GO RESTART PROGRAM
002074  022737  000176  001102  SRSET:  CMP     #SWREG,SWR
002102  001404                  BEQ     1$
002104  104000                  TYPE                    ;TYPE OPTIONS MESSAGE
002106  012500                  M3
002110  000000                  HALT                    ;WAIT FOR USER TO SET OPTIONS
002112  000401                  BR      GETRDY
002114  104020          1$:     CNTLU
002116  013737  001516  001524  GETRDY: MOV     KSTART,NXTST    ;ADDR OF 1ST ROUTINE TO NXTST
002124  012737  000006  000004  GTRDYX: MOV     #6,@#ERRVEC     ;RESET ERROR TRAP.
002132  005037  177776          CLR     PSW
002136  012706  001100          MOV     #SPBOT,%6       ;SET BOTTOM OF STACK.
002142  000005                  RESET                   ;ISSUE RESET.
002144  004737  002462  GTRDYA: JSR     %7,FORWD        ;ROLL FORWARD TO "NEXT" ROUTINE.
002150  032777  001000  176630  BIT     #BIT9,@SWR      ;CHECK SELECT ROUTINE SWITCH
002156  001003                  BNE     GTRDYC          ;BRANCH IF SELECT ROUTINE SWITCH IS SET.
002160  000177  177240          JMP     @CURTST         ;GO RUN CURRENT ROUTINE.
002164  000452                  BR      SCOPED          ;NO GO. MANUAL RTN BYPASSED.
002166  017700  176614  GTRDYC: MOV     @SWR,%0         ;(SR) TO R0
002172  042700  177600          BIC     #177600,%0      ;MASK UNDESIRED BITS
002176  123700  001522          CMPB    RTNNO,%0        ;COMPARE RTNNO TO (R0)
002202  001002                  BNE     GTRDYD          ;BRANCH IF ROUTINE NOT FOUND YET.
002204  000177  177214          JMP     @CURTST         ;GO RUN ROUTINE.
002210  022737  177777  001524  GTRDYD: CMP     #-1,NXTST       ;NO. CHECK FOR LAST ROUTINE.
002216  001352                  BNE     GTRDYA          ;BRANCH IF NOT LAST ROUTINE.
002220  004737  001652          JSR     %7,INCRTN       ;YES. INCORRECT ROUTINE SELECTED.
002224  000734                  BR      GETRDY          ;START OVER.

                        ;SCOPE ROUTINE (CALLED BY EMT INST.)
002226  000240          ESCOPE: NOP
002230  005077  176720          CLR     @CSR            ;INITIALIZE
002234  005077  176720          CLR     @BKCSR          ;THE
002240  005077  176712          CLR     @BAR            ;DM11
002244  104017                  KBDIN
002246  012777  001106  176706  MOV     #CAT,@BASREG
002254  032777  040000  176524  BIT     #BIT14,@SWR     ;CHECK FOR SCOPE OPTION.
002262  001403                  BEQ     SCOPEB          ;BRANCH IF SCOPE SW NOT SET.
002264  013716  001530  SCOPEA: MOV     SCOPTR,@%6      ;SET UP TO RETURN TO ROUTINE.
002270  000002                  RTI                     ;RETURN TO ROUTINE.
002272  032777  004000  176506  SCOPEB: BIT     #BIT11,@SWR     ;TEST INHIBIT ITERATION SWITCH
```

```
002374  001003                  BNE     SCOPEC          ;BRANCH IF INHIBIT ITERATION SW SET.
002376  005337  001526           DEC     ICTR            ;DECREMENT ITERATION COUNT.
002402  001366                  BNE     SCOPEA          ;BRANCH IF COUNT NOT 0.
002404  022626          SCOPEC: POPSP2                  ;POP STACK TWICE
002406  032777  001000  176466  SCOPED: BIT     #BIT9,@SWR      ;CHECK SELECT ROUTINE SWITCH
002414  001276                  BNE     GETRDY          ;BRANCH IF SELECT RTN SW SET
002416  022737  177777  001524  CMP     #-1,NXTST       ;LAST TEST?
002424  001275                  BNE     GTRDYX          ;BRANCH IF NOT LAST TEST.
002426  104000                  TYPE                    ;TYPE
002430  012543                  M2                      ;'PRGEND'
002432  013702  000042           MOV     @#42,R2         ;CHECK DDP ACT11 MONITOR HOOK
002436  001665                  BEQ     GETRDY
002440  000005                  RESET
002442  000240                  NOP
002444  000240                  NOP
002446  000240                  NOP
002450  000240                  NOP
002452  004712          $ENDAD: JSR     PC,(R2)         ;RETURN TO DDP/ACT11 MONITOR
002454  000240                  NOP
002456  000240                  NOP
002460  000240                  NOP

002462  013705  001524  FORWD:   MOV     NXTST,R5        ;ADDR OF NEXT ROUTINE TO RS.
002466  012537  001532          MOV     (R5)+,RTNNO     ;GET NEXT ROUTINE NUMBER.
002472  012537  001534          MOV     (R5)+,NXTST     ;GET ADDR OF NEXT "NEXT" ROUTINE.
002476  012537  001526          MOV     (R5)+,ICTR      ;GET ITERATION COUNT.
002502  012537  001530          MOV     (R5)+,SCOPTR    ;GET SCOPE LOOP ENTRY POINTER.
002506  010537  001520          MOV     R5,CURTST       ;ADDR OF NOW CURRENT TEST TO CURTST.
002512  000207                  RTS     PC              ;EXIT FORWD SUBROUTINE.


                        ;EMT TRAP INTERPRETER
002514  011646          EMTINT: MOV     (SP),-(SP)      ;GET PC OF NEXT INSTRUCTION
002516  162716  000002          SUB     #2,(SP)         ;POINT SP TO PC OF EMT
002522  011637  001640          MOV     (SP),PCADD      ;GET PC OF EMT CALL
002526  017616  000000          MOV     @(SP),(SP)      ;GET EMT CALL
002532  105066  000001          CLRB    1(SP)           ;STRIP EMT $ SAVE IDENTIFIER
002536  006316                  ASL     (SP)            ;SHIFT IDENTIFIER LEFT
002540  062716  001554          ADD     #EMTTAB,(SP)
002544  013607                  MOV     @(SP)+,PC       ;GO TO PROPER EMT


                        ;SAVE REGS 0 TO 4 SUBROUTINE.
002546  012637  002602  SAVRG:  MOV     (SP)+,1$        ;SAVE PC AND PSW.
002552  012637  002604          MOV     (SP)+,2$
002556  010446                  MOV     R4,-(SP)        ;SAVE REGS 0 - 4
002560  010346                  MOV     R3,-(SP)        ;IN STACK.
002562  010246                  MOV     R2,-(SP)
002564  010146                  MOV     R1,-(SP)
002566  010046                  MOV     R0,-(SP)
002570  013746  002604          MOV     2$,-(SP)        ;RESTORE PC AND PSW.
002574  013746  002602          MOV     1$,-(SP)
002600  000002                  RTI                     ;EXIT.
002602  000000          1$:     OPEN                    ;CONTAINS SAVED PC
002604  000000          2$:     OPEN                    ;CONTAINS SAVED PSW


                        ;RESTORE REGS 0 TO 4 SUBROUTINE.
002606  000240          RSTRG:  NOP
```

```
  692  002610  012637  002644          MOV     (6)+,1$         ;SAVE PC AND PSW.
  693  002614  012637  002646          MOV     (6)+,2$
  694  002620  012600                  MOV     (6)+,%0         ;RESTORE REGS 0 - 4
  695  002622  012601                  MOV     (6)+,%1         ;FROM STACK.
  696  002624  012602                  MOV     (6)+,%2
  697  002626  012603                  MOV     (6)+,%3
  698  002630  012604                  MOV     (6)+,%4
  699  002632  013746  002646          MOV     2$,-(6)         ;RESTORE PC AND PSW.
  700  002636  013746  002644          MOV     1$,-(6)
  701  002642  000002                  RTI                     ;EXIT
  702  002644  000000          1$:     OPEN                    ;CONTAINS SAVED PC
  703  002646  000000          2$:     OPEN                    ;CONTAINS SAVED PSW
  704
  705                           ;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER.
  706  002650  000240          TYP:    NOP
  707  002652  011600                  MOV     (SP),%0         ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS.
  708  002654  062716  000002          ADD     #2,(SP)         ;SET UP EXIT.
  709  002660  011000                  MOV     @%0,%0          ;ADDRESS OF MESSAGE TO R0.
  710  002662  112037  002762  1$:     MOVB    (0)+,5$         ;GET CHARACTER
  711  002666  122737  000100  002762  CMPB    #100,5$         ;CHECK FOR"@"CHARACTER
  712  002674  001001                  BNE     2$              ;BRANCH IF NOT"@".
  713  002676  000002                  RTI                     ;TERMINATOR CHAR. DONE. EXIT.
  714  002700  122737  000045  002762  2$:     CMPB    #45,5$          ;CHECK FOR"%".
  715  002706  001412                  BEQ     4$              ;BRANCH IF"%"
  716  002710  004737  002716          JSR     %7,3$           ;TYPE CHAR IN 5$
  717  002714  000762                  BR      1$
  718  002716  113777  002762  176702  3$:     MOVB    5$,@TPDBR       ;OUTPUT CHARACTER TO PRINTER
  719  002724  105777  176674          TSTB    @TPCSR          ;WAIT FOR DONE FLAG.
  720  002730  100375                  BPL     .-4
  721  002732  000207                  RTS     %7              ;EXIT
  722  002734  112737  000015  002762  4$:     MOVB    #15,5$          ;MOVE CARRIAGE RETURN CODE TO 5$
  723  002742  004737  002716          JSR     %7,@#3$         ;GO TYPE CHAR.
  724  002746  112737  000012  002762          MOVB    #12,5$          ;MOVE LF CODE TO 5$.
  725  002754  004737  002716          JSR     %7,3$           ;GO TYPE CHAR.
  726  002760  000740                  BR      1$
  727  002762  000000          5$:     OPEN
  728
  729                           ;SUBROUTINE TO GET DM11 PARAMETERS
  730                           ;VECTOR ADDRESS
  731  002764  000240          DMPAR:  NOP                     ;BEGIN
  732  002766  004737  003312          JSR     7,CVRLAY        ;PUT HALT..+2 IN VECTOR AREA
  733  002772  104000                  TYPE                    ;ASK USER FOR RECEIVER INT. VECTOR
  734  002774  012310                  WHERE                   ;OF UNIT UNDER TEST
  735  002776  004537  003346          JSR     5,RECD          ;GET THE VECTOR $
  736  003002  000000          VECTOR: 0                       ;PUT IT HERE
  737  003004  005737  003002          TST     VECTOR
  738  003010  001003                  BNE     1$
  739  003012  012737  000300  003002          MOV     #300,VECTOR     ;SET VECTOR = TO 0300
  740  003020  023727  003002  000300  1$:     CMP     VECTOR,#300     ;IS VECTOR HIGHER OR
  741  003026  103003                  BHIS    3$              ;EQUAL TO 0300
  742  003030  104000                  2$:     TYPE                    ;TYPE "?"
  743  003032  012540                  MI
  744  003034  000753                  BR      DMPAR           ;ASK FOR ANOTHER VECTOR
  745  003036  023727  003002  000770  3$:     CMP     VECTOR,#770     ;IS VECTOR = TO OR
  746  003044  101371                  BHI     2$              ;LESS THAN 770
  747  003046  032737  000007  003002          BIT     #7,VECTOR       ;LSB OF VECTOR MUST BE ALL 0'S
```

```
                                        BNE     2$
003054  001365
003056  013737  003002  001260        MOV     VECTOR,@#CLKINT
003064  062737  000004  003002        ADD     #4,VECTOR
003072  013737  003002  001254        MOV     VECTOR,@#XMTINT


                                ;UNIT NUMBER
003100  104000                  DMPARB: TYPE
003102  012403                          WHICH
003104  004537  003346                  JSR     5,RECD          ;GET THE UNIT #
003110  000000                  UNIT:   0                       ;PUT IT HERE
003112  023727  003110  000017        CMP     UNIT,#17
003120  101403                        BLOS    1$
003122  104000                        TYPE
003124  012540                        M1
003126  000764                        BR      DMPARB
003130  006337  003110          1$:   ASL     UNIT
003134  006337  003110                ASL     UNIT
003140  006337  003110                ASL     UNIT
003144  012702  000004                MOV     #4,%2
003150  012701  001250                MOV     #CSR,%1
003154  042711  000370          2$:   BIC     #370,(1)
003160  063721  003110                ADD     UNIT,(1)+
003164  005302                        DEC     %2
003166  001372                        BNE     2$


                                ;CALCULATE CHARARACTER LENGTH
003170  012777  001106  176060        MOV     #CAT,@BASREG
003176  005077  176046                CLR     @CSR
003202  012737  177777  005622        MOV     #-1,OUTBUF      ;LOAD OUTBUF WITH CHAR TO BE TRANSMITTED
003210  012737  177777  001146        MOV     #-1,WCT         ;SET UP TO TRANSMIT 1 CHAR
003216  012737  177777  001634        MOV     #-1,@#CARMSK    ;PRE SET THE CHARACTER MASK
003224  012737  006622  001106        MOV     #OUTBUF,CAT     ;1 CHARACTER ON LINE 0
003232  012777  003270  176020        MOV     #3$,@CLKINT     ;LOAD RECEIVER INTERRUPT
003240  012777  000340  176014        MOV     #340,@CLKLVL    ;AND PRIORITY LEVEL
003246  005037  001306                CLR     TUMTAB
003252  012777  000001  175772        MOV     #1,@BAR         ;START TRANSMITTING
003260  012777  000105  175762        MOV     #BIT6+BIT2+BIT0,@CSR    ;SET IE,MAINT AND GO BITS
003266  000001                        WAIT                    ;WAIT FOR RECEIVER INTERRUPT
003270  005077  175754          3$:   CLR     @CSR
003274  143737  001306  001634        BICB    TUMTAB,CARMSK   ;LOAD CHARACTER LENGTH MASK
003302  005037  177776                CLR     PSW             ;RESTORE PROCEESSER TO PRIORITY 0
003306  022626                        POPSP2                  ;RESTORE THE STACK POINTER
003310  000207                        RTS     7               ;EXIT PARAMETERS ROUTINE


                                ;ROUTINE TO LOAD TRAP/INTERRUPT VECTOR AREA WITH HALT..+2. HALTS PROGRAM
                                ;AT ADDRESS OF TRAP/INTERRUPT VECTOR +2.
003312  012701  000300          OVRLAY: MOV     #300,%1
003316  012702  000302                  MOV     #302,%2
003322  010221                  1$:     MOV     %2,(1)+
003324  005021                          CLR     (1)+
003326  020227  000776                  CMP     %2,#776
003332  001403                          BEQ     2$
003334  062702  000004                  ADD     #4,%2
003340  000770                          BR      1$
003342  000240                  2$:     NOP
003344  000207                          RTS     7               ;EXIT
```

```
                                        ;SUBROUTINE TO RECEIVE DATA
                                        ;THIS SUBROUTINE RECEIVES DATA FROM THE KEYBOARD (UP TO SIX OCTAL
                                        ;DIGITS AND PLACES THEM INTO THE ADDRESS FOLLOWING THE SUBROUTINE
                                        ;CALL (JSR 5,RECD).  NO REGISTER CONTENTS ARE DISTURBED.


                                        ;SUBROUTINE TO INPUT DATA FROM TTY

003346  010046                  RECD:   MOV     R0,-(SP)
003350  005015                  1$:     CLR     (5)                     ;CLEAR OLD DATA
003352  012737  000007  003524          MOV     #7,CNT                  ;SET CHAR COUNT
003360  105777  176234          2$:     TSTB    @TKCSR                  ;WAIT FOR CHAR
003364  100375                          BPL     2$
003366  117700  176230                  MOVB    @TKDBR,R0
003372  142700  000200                  BICB    #200,R0                 ;STRIP OFF PARITY
003376  110077  176224                  MOVB    R0,@TPDBR               ;ECHO CHARACTER
003402  122700  000025                  CMPB    #25,R0                  ;IS IT A ^U
003406  001443                          BEQ     5$                      ;BRANCH IF YES
003410  122700  000015                  CMPB    #15,R0                  ;IS IT A <CR>
003414  001415                          BEQ     6$                      ;BRANCH IF YESS
003416  142700  000060                  BICB    #60,R0
003422  132700  000110                  BITB    #110,R0                 ;CHECK FOR 0-7 (8)
003426  001031                          BNE     7$                      ;BRANCH IF NOT
003430  006315                          ASL     (5)
003432  006315                          ASL     (5)
003434  006315                          ASL     (5)                     ;SHIFT DATA
003436  150015                          BISB    R0,(5)                  ;INSET NEW CHAR
003440  005337  003524                  DEC     CNT
003444  001422                          BEQ     7$                      ;O_Y 6 CHAR'S PLEASE
003446  000744                          BR      2$                      ;NEXT CHARACTER
003450  105777  176150          6$:     TSTB    @TPCSR
003454  100375                          BPL     6$                      ;WAIT FOR READY
003456  012777  000012  176142          MOV     #12,@TPDBR              ;TYPE <LF>
003464  105777  176134          8$:     TSTB    @TPCSR
003470  100375                          BPL     8$                      ;WAIT FOR READY
003472  005077  176130                  CLR     @TPDBR                  ;NEXT CHARACTER
003476  105777  176122          9$:     TSTB    @TPCSR
003502  100375                          BPL     9$                      ;WAIT FOR READY
003504  005725                          TST     (R5)+                   ;ADJUST R5
003506  012600                          MOV     (SP)+,R0                ;RESTORE R0
003510  000205                          RTS     R5
003512  104000                  7$:     TYPE
003514  012540                          M!
003516  104000                  5$:     TYPE
003520  012400                          $CTLU
003522  000712                          BR      1$                      ;START OVER
003524  000000                  CNT:    0


                                        ;ROUTINE TO CHECK FOR ^G BEING TYPED

003526  022737  000176  001102  KBDINTT:CMP     #SWREG,SWR
003534  001015                          BNE     1$
003536  005037  003630                  CLR     TMP1                    ;CLEAR TEMP AREA
003542  117737  176054  003630          MOVB    @TKDBR,TMP1             ;FETCH THE BUFFER
```

```
   860  003550  142737  000200  003630          BICB    #200,TMP1               ;STRIP OFF PARITY
   861  003556  122737  000007  003630          CMPB    #7,TMP1                 ;WAS IT ^G
   862  003564  001001                          BNE     1$                      ;NOP
   863  003566  104020                          CNTLU                           ;GO CHANGE IT
   864  003570  000002                  1$:     RTI                             ;EXIT
   865
   866
   867                                          ;ROUTINE TO CHANGE CONTENTS OF SWREG(LOC 176)
   868
   869  003572  022737  000176  001102  CNTLUU: CMP     #SWREG,SWR
   870  003600  001023                          BNE     FAJAG
   871  003602  104000                          TYPE
   872  003604  012351                          $SWREG
   873  003606  004537  004352                  JSR     R5,OACNV                ;CONVERT TO ASCII
   874  003612  000176                          SWREG
   875  003614  012360                          $VALUE
   876  003616  000006                          6
   877  003620  104000                          TYPE
   878  003622  012360                          $VALUE
   879  003624  004537  003346                  JSR     5,RECD          ;GET THE TMP1 &
   880  003630  000000                  TMP1:   0                       ;PUT IT HERE
   881  003632  022737  000007  003524          CMP     #7,CNT
   882  003640  001403                          BEQ     FAJAG
   883  003642  013777  003630  175232          MOV     TMP1,@SWR               ;CHANGE CONTENTS OF SWREG
   884  003650  000002                  FAJAG:  RTI
   885
   886
   887  003652  013746  000006          SUSWRR: MOV     @#6,-(SP)               ;SAVE VECTORS
   888  003656  013746  000004                  MOV     @#4,-(SP)
   889  003662  012737  003702  000004          MOV     #1$,@#4                 ;SET UP FOR TIMEOUT
   890  003670  022777  177777  175204          CMP     #-1,@SWR                ;REFERENCE HARDWARE SWITCH REGISTER
   891  003676  001402                          BEQ     2$
   892  003700  000407                          BR      3$
   893  003702  022626                  1$:     CMP     (SP)+,(SP)+             ;ADJUST STACK
   894  003704  012737  000176  001102  2$:     MOV     #SWREG,SWR              ;POINT TO SOFTWARE SWITCH REG
   895  003712  012737  000174  001104          MOV     #DISPREG,DISPLAY                ;POINT TO SOFT DISPLAY REG
   896  003720  012637  000004          3$:     MOV     (SP)+,@#4               ;RESTORE VECTORS
   897  003724  012637  000006                  MOV     (SP)+,@#6
   898  003730  000002                          RTI
   899
```

```
                                        ;SUBROUTINE TO TRANSMIT ON ALL LINES WITH A DELAY BETWEEN TRANSMITTING
                                        ;ON SUCCESSIVE LINES. THE DELAY FOR THE TEST IS SUPPLIED BY THE
                                        ;CALLING JSR INSTRUCTION. DATA IS CHECKED AFTER ALL
                                        ;LINES HAVE FINISHED TRANSMITTING.

003732  000240                  DLYXMT: NOP                     ;BEGIN TEST
003734  012777  001106  175314          MOV     #CAT,@BASREG    ;SET UP BASE REGISTER
003742  004737  004264                  JSR     7,@#IDENT               ;TRANSMIT LINE # ON EACH LINE
003746  000240                          NOP                     ;NOP
003750  005077  175274                  CLR     @CSR
003754  012537  003764                  MOV     (5)+,10$        ;GET MESSAGE ADDRESS
003760  004537  004440                  JSR     5,@#BMOVE       ;LOAD OUTPUT BUFFER
003764  000000                  10$:    OPEN                    ;WITH DATA TO
003766  006622                          OUTBUF                  ;BE TRANSMITTED
003770  000100                          64.
003772  005037  001306                  CLR     @#TUMTAB        ;CLEAR TUMBLE
003776  004537  004440                  JSR     5,@#BMOVE               ;TABLE (200
004002  001306                          TUMTAB                  ;BYTES)
004004  001307                          TUMTAB+1
004006  000177                          177
004010  004537  004440                  JSR     5,@#BMOVE       ;CLEAR CHARACTER COUNT TABLE
004014  001306                          CNTTAB
004016  012126                          16.
004020  000020
004022  005037  006766                  CLR     @#LNOBUF
004026  004537  004440                  JSR     5,@#BMOVE       ;CLEAR ALL
004032  006766                          LNOBUF                  ;LINE'S INPUT
004034  006767                          LNOBUF+1                ;BUFFERS
004036  003077                          1599.                   ;(16. BUFFERS OF 100. CHARS. EACH)
004040  022737  000006  002130          CMP     #6,PRGNUM
004046  001002                          BNE     .+6
004050  000137  006322                  JMP     PRG3A
004054  012504                          MOV     (5)+,%4         ;GET # OF CHARACTERS TO TRANSMIT BEFORE
                                                                ;TRANSMITTING ON NEXT LINE
004056  012737  001306  001300          MOV     #TUMTAB,@#TTPTR ;INITIALIZE TUMBLE TABLE POINTER
004064  013701  001260                  MOV     @#CLKINT,%1     ;GET RECEIVER VECTOR ADDRESS
004070  012721  004666                  MOV     #RINT,(1)+      ;LOAD RECEIVER VECTOR
004074  013721  001262                  MOV     @#CLKLVL,(1)+   ;AND PRIORITY LEVEL
004100  012721  005060                  MOV     #TINT,(1)+      ;LOAD TRANSMITTER VECTOR
004104  013721  001266                  MOV     @#XMTLVL,(1)+   ;AND PRIORITY LEVEL
004110  005737  002130                  TST     PRGNUM          ;RUNNING PROGRAM 0?
004114  001402                          BEQ     .+6
004116  000137  006126                  JMP     PRG1A           ;RETURN TO PROGRAM 1 CODE
004122  012777  010101  175120          MOV     #BIT12+BIT6+BIT0,@CSR   ;SET IE & GO BITS
004130  012737  000001  001274          MOV     #1,@#LINBIT
004136  005037  001646                  CLR     @#LINE
004142  013700  001646          1$:     MOV     LINE,%0         ;LINE # X2 TO R0
004146  000240                          NOP                     ;NOP
004150  004537  004462                  JSR     5,@#XMITD               ;TRANSMIT 64 CHARACTERS
004154  177700                          -64.                    ;ON LINE # AS SPECIFIED IN ADDRESS LINE
004156  020460  001146          2$:     CMP     %4,WCT(0)       ;WAIT FOR THE WORD COUNT TO DEC TO THE
004162  001375                          BNE     2$              ;CORRECT VALUE BEFORE STARTING NEXT LINE
004164  062737  000002  001646          ADD     #2,LINE         ;FORM NEXT LINE NUMBER
004172  006337  001274                  ASL     LINBIT          ;SHIFT LINE BIT
004176  103361                          BCC     1$              ;START NEXT LINE
004200  005760  001146          3$:     TST     WCT(0)          ;WAIT FOR LAST LINE TO FINISH
```

```
                                                BNE     3$
004206  042777  177400  175034                  BIC     #177400,@CSR    ;CLEAR ODD BYTE OF CSR
004214  062700  000001          31$:            ADD     #1,R0           ;WAIT FOR RECEIVER TO RECEIVE
004220  001375                                  BNE     31$             ;ALL TRANSMITTED DATA
004222  017737  175024  001630                  MOV     @BAR,RCVDAT     ;GET AND TEST BAR CONTENTS
004230  001410                                  BEQ     4$              ;BRANCH IF IS CLEAR

004232  005037  001632                          CLR     XMTDAT
004236  005077  175006                          CLR     @CSR
004242  005077  175004                          CLR     @BAR
004246  104015                                  ERROR1                  ;ERROR! BAR DID NOT CLEAR IN SUFICIENT TIME
004250  000403                                  BR      5$              ;EXIT TEST
004252  000240                  4$:             NOP
004254  004737  004560                          JSR     7,@#CHKDAT      ;GO TEST DATA
004260  022626                  5$:             CMP     (6)+,(6)+       ;RESET THE STACK
004262  104012                                  SCOPE                   ;SCOPE


                                                ;SUBROUTINE TO TRANSMIT ON EACH LINE ITS LINE NUMBER (CRLF YY CRLF).
004264  005037  001646          IDENT:          CLR     @#LINE          ;GET LINE NUMBER 0
004270  012737  000001  001274                  MOV     #1,@#LINBIT     ;GET LINE BIT
004276  013702  001646          1$:             MOV     LINE,%2
004302  016262  012146  001106                  MOV     ID(2),CAT(2)    ;LOAD CAT
004310  012762  177772  001146                  MOV     #-6,WCT(2)      ;LOAD WORD COUNT
004316  053777  001274  174726                  BIS     LINBIT,@BAR     ;SET BAR BIT
004324  062737  000002  001646                  ADD     #2,LINE         ;FORM NEXT LINE NUMBER
004332  006337  001274                          ASL     LINBIT          ;FORM NEXT LINE BIT
004336  103357                                  BCC     1$              ;BRANCH IF NOT DONE
004340  005777  174706          2$:             TST     @BAR            ;WAIT FOR BAR TO CLEAR
004344  001375                                  BNE     2$
004346  000240                                  NOP
004350  000207                                  RTS     7               ;EXIT SUBROUTINE


                                                ;OCTAL TO ASCII CONVERT ROUTINE
004352  104013                  DACNV:          SAVREG                  ;SAVE REGISTERS ON THE STACK
004354  013537  004436                          MOV     @(5)+,2$        ;GET OCTAL VALUE.
004360  012501                                  MOV     (5)+,%1         ;GET DESTINATION ADDR.
004362  012502                                  MOV     (5)+,%2         ;GET CONVERT COUNT.
004364  060201                                  ADD     %2,%1           ;DEVELOP ADDR TO STORE 1ST CHAR.
004366  013703  004436          1$:             MOV     2$,%3
004372  042703  177770                          BIC     #177770,%3      ;ISOLATE LEAST SIGNIFICANT DIGIT.
004376  062703  000060                          ADD     #60,%3          ;CONVERT DIGIT TO ASCII.
004402  110341                                  MOVB    %3,-(1)         ;STORE ASCII CHARACTER.
004404  042737  000007  004436                  BIC     #7,2$
004412  006037  004436                          ROR     2$
004416  006037  004436                          ROR     2$
004422  006037  004436                          ROR     2$
004426  005302                                  DEC     %2              ;DONE ALL DIGITS?
004430  001356                                  BNE     1$              ;BRANCH IF NOT DONE.
004432  104014                                  RSTREG                  ;RESTORE THE REGISTERS
004434  000205                                  RTS     %5              ;DONE. EXIT.
004436  000000          2$:             OPEN


                                                ;SUBROUTINE TO MOVE A VARIABLE NUMBER OF BYTES.
```

```
 1012  004440  104013  BMOVE:  SAVREG                   ;SAVE REGS.
 1013  004442  012501          MOV     (5)+,%1         ;GET"FROM"ADDRESS
 1014  004444  012502          MOV     (5)+,%2         ;GET"TO"ADDRESS
 1015  004446  012503          MOV     (5)+,%3         ;GET COUNT
 1016  004450  112122  1$:     MOVB    (1)+,(2)+       ;MOVE BYTE
 1017  004452  005303          DEC     %3              ;DECREMENT COUNT
 1018  004454  001375          BNE     1$              ;BRANCH IF NOT DONE.
 1019  004456  104014          RSTREG                  ;RESTORE REGS.
 1020  004460  000205          RTS     %5              ;DONE EXIT
```

```
1021
1022
1023                                    ;SUBROUTINE TO TRANSMIT DATA.  SUBROUTINE CALLED BY
1024                                    ;JSR 5,XMITD
1025  004462  000240                    XMITD:  NOP
1026  004464  010046                            MOV     %0,-(SP)        ;SAVE R0 ON THE STACK
1027  004466  013700  001646                    MOV     @#LINE,%0       ;GET LINE
1028  004472  012760  006622  001106            MOV     #OUTBUF,CAT(0)  ;LOAD FIRST CHAR ADDRESS IN CAT
1029  004500  012560  001146                    MOV     (5)+,WCT(0)     ;LOAD WORD COUNT INTO LINE'S TABLE ADDRESS
1030  004504  053737  001274  001270            BIS     @#LINBIT,@#BARIM;LOAD LINE POSITION INTO BAR IMAGE
1031  004512  053777  001274  174532            BIS     LINBIT,@BAR     ;START TRANSMITTING ON LINE SPECIFIED
1032                                                                    ;IN LINBIT
1033  004520  012600                            MOV     (SP)+,%0        ;RESTORE R0
1034  004522  000240                            NOP
1035  004524  000205                            RTS     5               ;EXIT
1036
1037
1038  004526  010046                    ;SUBROUTINE TO FORM LINE BIT POSITION WITH THE LINE # IN LINE
1039  004530  005037  001274            GTLINB: MOV     %0,-(SP)        ;SAVE R0 ON THE STACK
1040  004534  013700  001646                    CLR     @#LINBIT
1041  004540  000261                            MOV     @#LINE,%0       ;GET LINE
1042  004542  006137  001274                    SEC                     ;SET CARRY
1043  004546  162700  000002            1$:     ROL     LINBIT          ;SHIFT LINE BIT
1044  004552  100373                            SUB     #2,%0           ;SUBTRACT 2 FROM LINE NUMBER
1045  004554  012600                            BPL     1$
1046  004556  000207                            MOV     (SP)+,%0        ;RESTORE R0
1047                                            RTS     7               ;EXIT
1048
1049                                    ;SUBROUTINE TO CHECK TRANSMITTED DATA
1050  004560  104013                    CHKDAT: SAVREG                  ;SAVE THE REGISTERS ON THE STACK
1051  004562  000240                            NOP                     ;NOP
1052  004564  005001                            CLR     %1              ;CLEAR CHARACTER COUNT
1053  004566  012702  012066                    MOV     #INTAB,%2       ;GET ADDRESS OF LINE'S INPUT BUFFER
1054  004572  005003                            CLR     %3              ;ADDRESS ;GET LINE COUNT
1055  004574  010337  001646            1$:     MOV     %3,@#LINE       ;MOVE LINE # TO LINE
1056  004600  012237  001650                    MOV     (2)+,@#LINBUF   ;GET LINE'S INPUT BUFFER ADDRESS
1057  004604  005337  001650                    DEC     LINBUF          ;SUBTRACT 1 FROM LINE'S INPUT BUFFER ADDRESS
1058  004610  005237  001650            2$:     INC     LINBUF          ;INCREMENT LINE'S INPUTT BUFFER ADDRESS
1059  004614  117737  175030  001630            MOVB    @LINBUF,@#RCVDAT        ;GET RECEIVED CHARACTER
1060  004622  116137  006622  001632            MOVB    OUTBUF(1),XMTDAT;GET TRANSMITTED CHARACTER
1061  004630  043737  001634  001632            BIC     @#CARMSK,XMTDAT ;CLEAR UNTRANSMITTED BITS
1062  004636  104004                            DATCHK                  ;COMPARE CHARACTERS
1063  004640  005201                            INC     %1              ;INCREMENT CHARACTER COUNT
1064  004642  020127  000100                    CMP     %1,#64.         ;ALL CHARACTERS BEEN COMPARED
1065  004646  001360                            BNE     2$              ;GO CHECK NEXT CHAR. IF NOT
1066  004650  005001                            CLR     %1              ;CLEAR CHARACTER COUNT
1067  004652  005203                            INC     %3              ;INCREMENT LINE COUNT
1068  004654  020327  000020                    CMP     %3,#16.         ;ALL LINES CHECKED?
1069  004660  002745                            BLT     1$              ;BRANCH IF ALL LINES NOT CHECKED
1070  004662  104014                            RSTREG                  ;RESTORE REGISTERS
1071  004664  000207                            RTS     7               ;EXIT SUBROUTINE
1072
1073
1074                                    ;RECEIVER INTERRUPT SERVICE ROUTINE
1075  004666  000240                    RINT:   NOP                     ;BEGIN
1076  004670  104013                            SAVREG                  ;SAVE THE REGISTERS ON THE STACK
```

```
1077  004672  013701  001300          MOV     @#TTPTR,%1      ;GET TUMBLE TABLE POINTER
1078  004676  011137  001272          MOV     (1),TTDAT       ;GET TUMBLE TABLE ENTRY
1079  004702  100410                  BMI     2$              ;BRANCH IF VALID DATA ENTRY
1080  004704  10-003                  ERROR                   ;ERROR! FALSE INTERRUPT
1081  004706  000454                  BR      6$              ;EXIT
1082  004710  011137  001272  1$:     MOV     (1),@#TTDAT     ;GET TUMBLE TABLE ENTRY
1083  004714  001451                  BEQ     6$              ;GO TO EXIT IF NO DATA ENTRY
1084  004716  100402                  BMI     2$              ;BRANCH IF VALID DATA ENTRY
1085  004720  104003                  ERROR                   ;ERROR! NO VALID DATA ENTRY INDICATOR
1086  004722  000425                  BR      3$
1087  004724  005011          2$:     CLR     (1)             ;CLEAR TUMBLE TABLE ENTRY
1088  004726  042737  160400  001272  BIC     #160400,@#TTDAT ;CLEAR ALL BUT CHAR. & LINE #
1089  004734  113702  001273          MOVB    TTDAT+1,%2      ;PUT LINE # IN R2 (LINE WILL BE IN LSH)
1090  004740  010204                  MOV     %2,%4
1091  004742  016237  012066  001650  MOV     INTAB(2),@#LINBUF       ;GET LINE'S INPUT BUFFER ADDRESS
1092  004750  006202                  ASR     %2              ;SHIFT LINE #
1093  004752  005003                  CLR     %3
1094  004754  116203  012126          MOVB    CNTTAB(2),%3    ;GET LINE'S RECEIVED CHAR. COUNT
1095  004760  105262  012126          INCB    CNTTAB(2,       ;INCREMENT CHARACTER COUNT
1096  004764  060337  001650          ADD     %3,LINBUF       ;FORM ADDRESS WHERE CHAR. IS TO BE STORED
1097  004770  113777  001272  174652  MOVB    TTDAT,@LINBUF   ;STORE CHAR. IN LINE'S INPUT BUFFER
1098  004776  000240          3$:     NOP                     ;NOP
1099  005000  016437  001146  001630  MOV     WCT(4),RCVDAT   ;GET TRANSMITTERS WORD COUNT
1100  005006  003405                  BLE     4$              ;BRANCH IF WORD COUNT IS 0 OR NEGATIVE
1101  005010  010437  001632          MOV     %4,XMTDAT       ;GET LINE # OF FAILING LINE
1102  005014  104015                  ERROR1                  ;ERROR! INCORRECT WORD COUNT IN
1103                          ;TYPE OUT SHOWS FAILING LINE #. AND FAILING LINE'S WORD COUNT
1104  005016  000005                  RESET                   ;STOP THE DM11
1105  005020  104012                  SCOPE                   ;EXIT TEST
1106
1107  005022  022701  001504  4$:     CMP     #TUMTAB+176,%1  ;IS THE TUMBLE TABLE POINTER AT THE
1108  005026  001002                  BNE     5$              ;THE END OF THE TABLE
1109  005030  012701  001304          MOV     #TUMTAB-2,%1    ;RESET POINTER
1110  005034  005721          5$:     TST     (1)+            ;INCREMENT POINTER
1111  005036  000724                  BR      1$              ;GO CHECK NEXT ENTRY
1112  005040  042777  000200  174202  6$:     BIC     #BIT7,@CSR      ;CLEAR RECEIVER DONE FLAG
1113  005046  010137  001300          MOV     %1,TTPTR        ;SAVE POINTER
1114  005052  104014                  RSTREG                  ;RESTORE THE REGISTERS
1115  005054  000240                  NOP
1116  005056  000002                  RTI                     ;EXIT SERVICE ROUTINE
1117
1118                          ;TRANSMITTER INTERRUPT SERVICE ROUTINE
1119  005060  000240          TINT:   NOP                     ;BEGIN
1120  005062  032777  060000  174160  BIT     #BIT14+BIT13,@CSR;TEST ERROR FLAGS
1121  005070  001404                  BEQ     1$              ;BRANCH IF NO ERROR FLAGS
1122  005072  104003                  ERROR                   ;ERROR! ERROR FLAG IS SET
1123  005074  042777  060000  174146  BIC     #BIT14+BIT13,@CSR;CLEAR ERROR FLAGS
1124  005102  005777  174142  1$:     TST     @CSR            ;TEST READY FLAG
1125  005106  100003                  BPL     2$              ;BRANCH IF READY IS CLEAR
1126  005110  042777  100000  174132  BIC     #BIT15,@CSR     ;CLEAR READY FLAG
1127  005116  000002          2$:     RTI
1128
1129
```

```
1130
1131  005120  104000                  PRGO:   TYPE
1132  005122  012661                          PRGOM
1133  005124  012737  005160  001516  PRGOA:  MOV     #RTO,KSTART     ;GET ADDRESS OF FIRST TEST
1134  005132  005037  001522                  CLR     RTNNO           ;CLEAR ROUTINE #
1135  005136  000137  002170                  JMP     SRSET
1136  005142  012737  005160  001516  PRGOR:  MOV     #RTO,KSTART     ;GET ADDRESS OF FIRST TEST
1137  005150  005037  001522                  CLR     RTNNO           ;CLEAR ROUTINE NUMBER
1138  005154  000137  002212                  JMP     GETRDY          ;GO AND START PROGRAM
1139                                  ;*****************************************************************
1140  005160  000000                  RTO:    0                       ;ROUTINE # 0                    *
1141  005162  005200                          RT1                     ;ADDR OF NEXT ROUTINE.          *
1142  005164  000002                          2                       ;ITERATION COUNT                *
1143  005166  005170                          RTOA                    ;SCOPE ENTRY POINT.             *
1144           000000                         X=X+1
1145                                  ;*****************************************************************
1146                                  ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1147                                  ;NEXT LINE.
1148  005170  004537  003732          RTOA:   JSR     5,DLYXMT        ;GO DO TEST.
1149  005174  013203                          MSG1                    ;TRANSMIT THIS MESSAGE &
1150  005176  000000                          0                       ;DELAY THIS MUCH BETWEEN LINES
1151                                  ;*****************************************************************
1152  005200  000001                  RT1:    1                       ;ROUTINE # 1                    *
1153  005202  005220                          RT2                     ;ADDR OF NEXT ROUTINE.          *
1154  005204  000002                          2                       ;ITERATION COUNT                *
1155  005206  005210                          RT1A                    ;SCOPE ENTRY POINT.             *
1156           000001                         X=X+1
1157                                  ;*****************************************************************
1158                                  ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1159                                  ;NEXT LINE.
1160  005210  004537  003732          RT1A:   JSR     5,DLYXMT        ;GO DO TEST.
1161  005214  013203                          MSG1                    ;TRANSMIT THIS MESSAGE &
1162  005216  177740                          -32.                    ;DELAY THIS MUCH BETWEEN LINES
1163                                  ;*****************************************************************
1164  005220  000002                  RT2.    2                       ;ROUTINE # 2                    *
1165  005222  005240                          RT3                     ;ADDR OF NEXT ROUTINE.          *
1166  005224  000002                          2                       ;ITERATION COUNT                *
1167  005226  005230                          RT2A                    ;SCOPE ENTRY POINT.             *
1168           000002                         X=X+1
1169                                  ;*****************************************************************
1170                                  ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1171                                  ;NEXT LINE.
1172  005230  004537  003732          RT2A:   JSR     5,DLYXMT        ;GO DO TEST.
1173  005234  013203                          MSG1                    ;TRANSMIT THIS MESSAGE &
1174  005236  177720                          -48.                    ;DELAY THIS MUCH BETWEEN LINES
1175                                  ;*****************************************************************
1176  005240  000003                  RT3:    3                       ;ROUTINE # 3                    *
1177  005242  005260                          RT4                     ;ADDR OF NEXT ROUTINE.          *
1178  005244  000002                          2                       ;ITERATION COUNT                *
1179  005246  005250                          RT3A                    ;SCOPE ENTRY POINT.             *
1180           000003                         X=X+1
1181                                  ;*****************************************************************
1182                                  ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1183                                  ;NEXT LINE.
1184  005250  004537  003732          RT3A:   JSR     5,DLYXMT        ;GO DO TEST.
1185  005254  013203                          MSG1                    ;TRANSMIT THIS MESSAGE &
```

```
1186  005256  177710                        -56.                    ;DELAY THIS MUCH BETWEEN LINES
1187                                ;*************************************************************************
1188  005260  000004                RT4:    4                       ;ROUTINE # 4                            *
1189  005262  005300                        RT5                     ;ADDR OF NEXT ROUTINE.                  *
1190  005264  000002                        2                       ;ITERATION COUNT                        *
1191  005266  005270                        RT4A                    ;SCOPE ENTRY POINT.                     *
1192           000004                       X=X+1
1193                                ;*************************************************************************
1194                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1195                                ;NEXT LINE.
1196  005270  004537  003732        RT4A:   JSR     5,DLYXMT        ;GO DO TEST.
1197  005274  013203                        MSG1                    ;TRANSMIT THIS MESSAGE &
1198  005276  177704                        -60.                    ;DELAY THIS MUCH BETWEEN LINES
1199                                ;*************************************************************************
1200  005300  000005                RT5:    5                       ;ROUTINE # 5                            *
1201  005302  005320                        RT6                     ;ADDR OF NEXT ROUTINE.                  *
1202  005304  000002                        2                       ;ITERATION COUNT                        *
1203  005306  005310                        RT5A                    ;SCOPE ENTRY POINT.                     *
1204           000005                       X=X+1
1205                                ;*************************************************************************
1206                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1207                                ;NEXT LINE.
1208  005310  004537  003732        RT5A:   JSR     5,DLYXMT        ;GO DO TEST.
1209  005314  013203                        MSG1                    ;TRANSMIT THIS MESSAGE &
1210  005316  177702                        -62.                    ;DELAY THIS MUCH BETWEEN LINES
1211                                ;*************************************************************************
1212  005320  000006                RT6:    6                       ;ROUTINE # 6                            *
1213  005322  005340                        RT7                     ;ADDR OF NEXT ROUTINE.                  *
1214  005324  000002                        2                       ;ITERATION COUNT                        *
1215  005326  005330                        RT6A                    ;SCOPE ENTRY POINT.                     *
1216           000006                       X=X+1
1217                                ;*************************************************************************
1218                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1219                                ;NEXT LINE.
1220  005330  004537  003732        RT6A:   JSR     5,DLYXMT        ;GO DO TEST.
1221  005334  013203                        MSG1                    ;TRANSMIT THIS MESSAGE &
1222  005336  177701                        -63.                    ;DELAY THIS MUCH BETWEEN LINES
1223                                ;*************************************************************************
1224  005340  000007                RT7:    7                       ;ROUTINE # 7                            *
1225  005342  005360                        RT10                    ;ADDR OF NEXT ROUTINE.                  *
1226  005344  000002                        2                       ;ITERATION COUNT                        *
1227  005346  005350                        RT7A                    ;SCOPE ENTRY POINT.                     *
1228           000007                       X=X+1
1229                                ;*************************************************************************
1230                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1231                                ;NEXT LINE.
1232  005350  004537  003732        RT7A:   JSR     5,DLYXMT        ;GO DO TEST.
1233  005354  013203                        MSG1                    ;TRANSMIT THIS MESSAGE &
1234  005356  177700                        -64.                    ;DELAY THIS MUCH BETWEEN LINES
1235                                ;*************************************************************************
1236  005360  000010                RT10:   10                      ;ROUTINE # 10                           *
1237  005362  005370                        R*11                    ;ADDR OF NEXT ROUTINE.                  *
1238  005364  000002                        2                       ;ITERATION COUNT                        *
1239  005366  005400                        RT10A                   ;SCOPE ENTRY POINT.                     *
1240           000010                       X=X+1
1241                                ;*************************************************************************
```

```
                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
                                ;NEXT LINE.
005370  004537  003732          RT10A:  JSR     5,DLYXMT        ;GO DO TEST.
005374  013304                          MSG2                    ;TRANSMIT THIS MESSAGE $
005376  177740                          -32.                    ;DELAY THIS MUCH BETWEEN LINES
                                ;*******************************************************************
005400  000011                  RT11:   11                      ;ROUTINE # 11                    *
005402  005420                          RT12                    ;ADDR OF NEXT ROUTINE.           *
005404  000002                          2                       ;ITERATION COUNT                 *
005406  005410                          RT11A                   ;SCOPE ENTRY POINT.              *
        000011                          X=X+1
                                ;*******************************************************************
                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
                                ;NEXT LINE.
005410  004537  003732          RT11A:  JSR     5,DLYXMT        ;GO DO TEST.
005414  013304                          MSG2                    ;TRANSMIT THIS MESSAGE $
005416  177720                          -48.                    ;DELAY THIS MUCH BETWEEN LINES
                                ;*******************************************************************
005420  000012                  RT12:   12                      ;ROUTINE # 12                    *
005422  005440                          RT13                    ;ADDR OF NEXT ROUTINE.           *
005424  000002                          2                       ;ITERATION COUNT                 *
005426  005430                          RT12A                   ;SCOPE ENTRY POINT.              *
        000012                          X=X+1
                                ;*******************************************************************
                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
                                ;NEXT LINE.
005430  004537  003732          RT12A:  JSR     5,DLYXMT        ;GO DO TEST.
005434  013304                          MSG2                    ;TRANSMIT THIS MESSAGE $
005436  177710                          -56.                    ;DELAY THIS MUCH BETWEEN LINES
                                ;*******************************************************************
005440  000013                  RT13:   13                      ;ROUTINE # 13                    *
005442  005460                          RT14                    ;ADDR OF NEXT ROUTINE.           *
005444  000002                          2                       ;ITERATION COUNT                 *
005446  005450                          RT13A                   ;SCOPE ENTRY POINT.              *
        000013                          X=X+1
                                ;*******************************************************************
                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
                                ;NEXT LINE.
005450  004537  003732          RT13A:  JSR     5,DLYXMT        ;GO DO TEST.
005454  013304                          MSG2                    ;TRANSMIT THIS MESSAGE $
005456  177704                          -60.                    ;DELAY THIS MUCH BETWEEN LINES
                                ;*******************************************************************
005460  000014                  RT14:   14                      ;ROUTINE # 14                    *
005462  005500                          RT15                    ;ADDR OF NEXT ROUTINE.           *
005464  000002                          2                       ;ITERATION COUNT                 *
005466  005470                          RT14A                   ;SCOPE ENTRY POINT.              *
        000014                          X=X+1
                                ;*******************************************************************
                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
                                ;NEXT LINE.
005470  004537  003732          RT14A:  JSR     5,DLYXMT        ;GO DO TEST.
005474  013304                          MSG2                    ;TRANSMIT THIS MESSAGE $
005476  177702                          -62.                    ;DELAY THIS MUCH BETWEEN LINES
                                ;*******************************************************************
005500  000015                  RT15:   15                      ;ROUTINE # 15                    *
005502  005520                          RT16                    ;ADDR OF NEXT ROUTINE.           *
```

```
:298  005504  000002                  2                       ;ITERATION COUNT                *
:299  005506  005510                  RT15A                   ;SCOPE ENTRY POINT.             *
:300          000015                  X=X+1
:301
:302
:303
:304  005510  004537  003732  RT15A:  JSR     5,DLYXMT        ;GO DO TEST.
:305  005514  013304                  MSG2                    ;TRANSMIT THIS MESSAGE &
:306  005516  177701                  -63.                    ;DELAY THIS MUCH BETWEEN LINES
:307
:308  005520  000016          RT16:   16                      ;ROUTINE # 16                   *
:309  005522  005540                  RT17                    ;ADDR OF NEXT ROUTINE.          *
:310  005524  000002                  2                       ;ITERATION COUNT                *
:311  005526  005530                  RT16A                   ;SCOPE ENTRY POINT.             *
:312          000016                  X=X+1
:313
:314
:315
:316  005530  004537  003732  RT16A:  JSR     5,DLYXMT        ;GO DO TEST.
:317  005534  013304                  MSG2                    ;TRANSMIT THIS MESSAGE &
:318  005536  177700                  -64.                    ;DELAY THIS MUCH BETWEEN LINES
:319
:320  005540  000017          RT17:   17                      ;ROUTINE # 17                   *
:321  005542  005560                  RT20                    ;ADDR OF NEXT ROUTINE.          *
:322  005544  000002                  2                       ;ITERATION COUNT                *
:323  005546  005550                  RT17A                   ;SCOPE ENTRY POINT.             *
:324          000017                  X=X+1
:325
:326
:327
:328  005550  004537  003732  RT17A:  JSR     5,DLYXMT        ;GO DO TEST.
:329  005554  013404                  MSG3                    ;TRANSMIT THIS MESSAGE &
:330  005556  177720                  -48.                    ;DELAY THIS MUCH BETWEEN LINES
:331
:332  005560  000020          RT20:   20                      ;ROUTINE # 20                   *
:333  005562  005600                  RT21                    ;ADDR OF NEXT ROUTINE.          *
:334  005564  000002                  2                       ;ITERATION COUNT                *
:335  005566  005570                  RT20A                   ;SCOPE ENTRY POINT.             *
:336          000020                  X=X+1
:337
:338
:339
:340  005570  004537  003732  RT20A:  JSR     5,DLYXMT        ;GO DO TEST.
:341  005574  013404                  MSG3                    ;TRANSMIT THIS MESSAGE &
:342  005576  177704                  -60.                    ;DELAY THIS MUCH BETWEEN LINES
:343
:344  005600  000021          RT21:   21                      ;ROUTINE # 21                   *
:345  005602  005620                  RT22                    ;ADDR OF NEXT ROUTINE.          *
:346  005604  000002                  2                       ;ITERATION COUNT                *
:347  005606  005610                  RT21A                   ;SCOPE ENTRY POINT.             *
:348          000021                  X=X+1
:349
:350
:351
:352  005610  004537  003732  RT21A:  JSR     5,DLYXMT        ;GO DO TEST.
:353  005614  013404                  MSG3                    ;TRANSMIT THIS MESSAGE &
```

```
1354  005616  177701                  -63.                    ;DELAY THIS MUCH BETWEEN LINES
1355                  ;*****************************************************************
1356  005620  000022  RT22:   22                      ;ROUTINE # 22                    *
1357  005622  005640         RT23                    ;ADDR OF NEXT ROUTINE.           *
1358  005624  000002         2                       ;ITERATION COUNT                 *
1359  005626  005630         RT22A                   ;SCOPE ENTRY POINT.              *
1360           000022         X=X+1
1361                  ;*****************************************************************
1362                  ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1363                  ;NEXT LINE.
1364  005630  004537  003732 RT22A:  JSR     5,DLYXMT        ;GO DO TEST.
1365  005634  013404         MSG3                    ;TRANSMIT THIS MESSAGE &
1366  005636  177700         -64.                    ;DELAY THIS MUCH BETWEEN LINES
1367                  ;*****************************************************************
1368  005640  000023 RT23:   23                      ;ROUTINE # 23                    *
1369  005642  005660         RT24                    ;ADDR OF NEXT ROUTINE.           *
1370  005644  000002         2                       ;ITERATION COUNT                 *
1371  005646  005650         RT23A                   ;SCOPE ENTRY POINT.              *
1372           000023         X=X+1
1373                  ;*****************************************************************
1374                  ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1375                  ;NEXT LINE.
1376  005650  004537  003732 RT23A:  JSR     5,DLYXMT        ;GO DO TEST.
1377  005654  013504         MSG4                    ;TRANSMIT THIS MESSAGE &
1378  005656  177740         -32.                    ;DELAY THIS MUCH BETWEEN LINES
1379                  ;*****************************************************************
1380  005660  000024 RT24:   24                      ;ROUTINE # 24                    *
1381  005662  005700         RT25                    ;ADDR OF NEXT ROUTINE.           *
1382  005664  000002         2                       ;ITERATION COUNT                 *
1383  005666  005670         RT24A                   ;SCOPE ENTRY POINT.              *
1384           000024         X=X+1
1385                  ;*****************************************************************
1386                  ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1387                  ;NEXT LINE.
1388  005670  004537  003732 RT24A:  JSR     5,DLYXMT        ;GO DO TEST.
1389  005674  013504         MSG4                    ;TRANSMIT THIS MESSAGE &
1390  005676  177710         -56.                    ;DELAY THIS MUCH BETWEEN LINES
1391                  ;*****************************************************************
1392  005700  000025 RT25:   25                      ;ROUTINE # 25                    *
1393  005702  005720         RT26                    ;ADDR OF NEXT ROUTINE.           *
1394  005704  000002         2                       ;ITERATION COUNT                 *
1395  005706  005710         RT25A                   ;SCOPE ENTRY POINT.              *
1396           000025         X=X+1
1397                  ;*****************************************************************
1398                  ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1399                  ;NEXT LINE.
1400  005710  004537  003732 RT25A:  JSR     5,DLYXMT        ;GO DO TEST.
1401  005714  013504         MSG4                    ;TRANSMIT THIS MESSAGE &
1402  005716  177702         -62.                    ;DELAY THIS MUCH BETWEEN LINES
1403                  ;*****************************************************************
1404  005720  000026 RT26:   26                      ;ROUTINE # 26                    *
1405  005722  005740         RT27                    ;ADDR OF NEXT ROUTINE.           *
1406  005724  000002         2                       ;ITERATION COUNT                 *
1407  005726  005730         RT26A                   ;SCOPE ENTRY POINT.              *
1408           000026         X=X+1
1409                  ;*****************************************************************
```

```
1410
1411
1412  005730  004537  003732        RT26A:  JSR     5,DLYXMT        ;GO DO TEST.
1413  005734  013504                        MSG4                    ;TRANSMIT THIS MESSAGE &
1414  005736  177700                        -64.                    ;DELAY THIS MUCH BETWEEN LINES
1415                                ;*****************************************************************
1416  005740  000027                RT27:   27                      ;ROUTINE # 27                  *
1417  005742  005760                        RT30                    ;ADDR OF NEXT ROUTINE.         *
1418  005744  000002                        2                       ;ITERATION COUNT               *
1419  005746  005750                        RT27A                   ;SCOPE ENTRY POINT.            *
1420           000027                       X=X+1
1421                                ;*****************************************************************
1422                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1423                                ;NEXT LINE.
1424  005750  004537  003732        RT27A:  JSR     5,DLYXMT        ;GO DO TEST.
1425  005754  013604                        MSG5                    ;TRANSMIT THIS MESSAGE &
1426  005756  177720                        -48.                    ;DELAY THIS MUCH BETWEEN LINES
1427                                ;*****************************************************************
1428  005760  000030                RT30:   30                      ;ROUTINE # 30                  *
1429  005762  006000                        RT31                    ;ADDR OF NEXT ROUTINE.         *
1430  005764  000002                        2                       ;ITERATION COUNT               *
1431  005766  005770                        RT30A                   ;SCOPE ENTRY POINT.            *
1432           000030                       X=X+1
1433                                ;*****************************************************************
1434                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1435                                ;NEXT LINE.
1436  005770  004537  003732        RT30A:  JSR     5,DLYXMT        ;GO DO TEST.
1437  005774  013604                        MSG5                    ;TRANSMIT THIS MESSAGE &
1438  005776  177710                        -56.                    ;DELAY THIS MUCH BETWEEN LINES
1439                                ;*****************************************************************
1440  006000  000031                RT31:   31                      ;ROUTINE # 31                  *
1441  006002  006020                        RT32                    ;ADDR OF NEXT ROUTINE.         *
1442  006004  000002                        2                       ;ITERATION COUNT               *
1443  006006  006010                        RT31A                   ;SCOPE ENTRY POINT.            *
1444           000031                       X=X+1
1445                                ;*****************************************************************
1446                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1447                                ;NEXT LINE.
1448  006010  004537  003732        RT31A:  JSR     5,DLYXMT        ;GO DO TEST.
1449  006014  013604                        MSG5                    ;TRANSMIT THIS MESSAGE &
1450  006016  177704                        -60.                    ;DELAY THIS MUCH BETWEEN LINES
1451                                ;*****************************************************************
1452  006020  000032                RT32:   32                      ;ROUTINE # 32                  *
1453  006022  006040                        RT33                    ;ADDR OF NEXT ROUTINE.         *
1454  006024  000002                        2                       ;ITERATION COUNT               *
1455  006026  006030                        RT32A                   ;SCOPE ENTRY POINT.            *
1456           000032                       X=X+1
1457                                ;*****************************************************************
1458                                ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1459                                ;NEXT LINE.
1460  006030  004537  003732        RT32A:  JSR     5,DLYXMT        ;GO DO TEST.
1461  006034  013604                        MSG5                    ;TRANSMIT THIS MESSAGE &
1462  006036  177702                        -62.                    ;DELAY THIS MUCH BETWEEN LINES
1463                                ;*****************************************************************
1464  006040  000033                RT33:   33                      ;ROUTINE # 33                  *
1465  006042  006060                        RT34                    ;ADDR OF NEXT ROUTINE.         *
```

```
1466  006044  000002                  2                       ;ITERATION COUNT                        *
1467  006046  006050                  RT33A                   ;SCOPE ENTRY POINT.                     *
1468          000033                  X=X+1
1469                                  ;*************************************************************
1470                                  ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1471                                  ;NEXT LINE.
1472  006050  004537  003732  RT33A:  JSR     5,DLYXMT        ;GO DO TEST.
1473  006054  013604                  MSG5                    ;TRANSMIT THIS MESSAGE &
1474  006056  177701                  -63.                    ;DELAY THIS MUCH BETWEEN LINES
1475                                  ;*************************************************************
1476  006060  000034  RT34:   34                      ;ROUTINE # 34                           *
1477  006062  177777                  RT35                    ;ADDR OF NEXT ROUTINE.                  *
1478  006064  000002                  2                       ;ITERATION COUNT                        *
1479  006066  006070                  RT34A                   ;SCOPE ENTRY POINT.                     *
1480          000034                  X=X+1
1481                                  ;*************************************************************
1482                                  ;TEST TO TRANSMIT ON EACH LINE WITH A DELAY BEFORE STATING THE
1483                                  ;NEXT LINE.
1484  006070  004537  003732  RT34A:  JSR     5,DLYXMT        ;GO DO TEST.
1485  006074  013604                  MSG5                    ;TRANSMIT THIS MESSAGE &
1486  006076  177700                  -64.                    ;DELAY THIS MUCH BETWEEN LINES
1487          177777  RT35=-1
```

```
1498
1499                                    :PRG1-  DATA TESTS ALL LINES SIMULTANEOUSLY. DATA TRANSMITTED IS 'THE
1490                                    ;QUICK BROWN FOX JUMPED OVER THE LAZY DOGS BACK 1234567890'
1491  006100  104000                    PRG1:   TYPE                            ;TYPE
1492  006102  012707                            PRG1M                           ;PROGRAM TITLE
1493  006104  022737  000176  001102    PRGX:   CMP     #SWREG.SWR              ;SEE IF SWITCH-LESS
1494  006112  001001                            BNE     PRG1R                   ;BRANCH IF NOT
1495  006114  104020                            CNTLU                           ;GET SWREG SETTINGS
1496  006116  004537  003732            PRG1R:  JSR     5.DLYXMT                ;GO TO DLYXMT TO SET UP DM11
1497  006122  013203                            MSG1                            ;MSG1 WILL BE THE DATA TRANSMITTED
1498  006124  177700                            -64.                            ;DO NOT DELAY
1499  006126  012737  006622  001106    PRG1A:  MOV     #OUTBUF,CAT             ;LOAD CURRENT
1500  006134  004537  004440                    JSR     5.BMOVE                 ;ADDRESS TABLE
1501  006140  001106                            CAT                             ;TO POINT TO
1502  006142  001110                            CAT+2                           ;OUTBUF
1503  006144  000040                            32.
1504  006146  012737  177700  001146            MOV     #-64.,WCT               ;LOAD WORD COUNT
1505  006154  004537  004440                    JSR     5.BMOVE                 ;TO -64.
1506  006160  001146                            WCT
1507  006162  001150                            WCT+2
1508  006164  000040                            32.
1509  006166  012777  010100  173054            MOV     #BIT12+BIT6,@CSR        ;SET TRANSMITTER & RECIEVER ** BITS
1510  006174  023727  002130  000004            CMP     PRGNUM,#4               ;RUNNING PROGRAM #2?
1511  006202  001403                            BEQ     .+10
1512  006204  052777  000001  173036            BIS     #BIT0,@CSR              ;SET THE GO BIT
1513  006212  012777  177777  173032            MOV     #-1,@BAR                ;START TRANSMITTING ON ALL LINES
1514  006220  005777  173026                    TST     @BAR                    ;WAIT FOR ALL LINES TO COMPLETE
1515  006224  001375                            BNE     .-4
1516  006226  005205                            INC     %5
1517  006230  001376                            BNE     .-2
1518  006232  005077  173012                    CLR     @CSR
1519  006236  023727  002130  000004    PRG1C:  CMP     PRGNUM,#4               ;DO NOT CHECK DATA IF RUNNING
1520  006244  001402                            BEQ     PRG1D                   ;PROGRAM # 2
1521  006246  004737  004560                    JSR     7.CHKDAT                ;GO CHECK RECEIVED DATA
1522  006252  104000                    PRG1D:  TYPE                            ;TYPE
1523  006254  012543                            M2                              ;'PRGEND'
1524  006256  012706  001100            PRG1EX: MOV     #SPBOT,SP               ;RESET THE STACK POINTER
1525  006262  000715                            BR      PRG1R                   ;GO RESTART TEST
1526
1527
1528                                    :PRG2-PROGRAM 2 RUNS PROGRAM 1 EXCEPT FOR THE DATA CHECKING
1529                                    ;WHEN ALL LINES ARE FINISHED TRANSMITTING. THIS ALLOWS THE DATA
1530                                    ;TRANSMITTED TO BE SENT TO TERMINALS. BEFORE STARTING THIS PROGRAM
1531                                    ;REMOVE THE JUMPERS CONNECTING THE TRANSMITTERS TO THE RECEIVERS.
1532  006264  104000                    PRG2:   TYPE                            ;TYPE PROGRAM TITLE
1533  006266  012767                            PRG2M                           ;AND INSTRUCTIONS
1534  006270  000137  006104            PRG2R:  JMP     PRGX                    ;GO RUN PRG1
```

```
1535
1536                                    ;PRG3-ECHO TEST THIS PROGRAM ECHOS BACK DATA RECEIVED FROM ANY DM11
1537                                    ;TERMINAL(S)
1538                                    ;NOTE:  THIS TEST IS THE ONLY TEST THAT INSURES PROPER OPERATION
1539                                    ;OF THE DM11 DISTRIBUTION PANEL LOGIC.
1540
1541  006274  104000                    PRG3:   TYPE                    ;TYPE PROGRAM
1542  006276  013016                            PRG3M                   ;TITLE
1543  006300  022737  000176  001102            CMP     #SWREG,SWR      ;SEE IF SWITCH-LESS
1544  006306  001001                            BNE     PRG3R           ;BRANCH IF NOT
1545  006310  104020                            CNTLU                   ;GET SWREG SETTINGS
1546  006312  004537  003732           PRG3R:  JSR     5,DLYXMT        ;USE PART OF THE
1547  006316  013203                            MSG1                    ;DLYXMT ROUTINE TO
1548  006320  000240                            NOP                     ;SET UP DM11
1549  006322  012737  001306  001300   PRG3A:  MOV     #TUMTAB,TTPTR   ;INITIALIZE SOFTWARE POINTER
1550  006330  013701  001260                    MOV     CLKINT,%1       ;LOAD RECEIVER
1551  006334  012721  006372                    MOV     #RINT3,(1)+     ;AND TRANSMITTER
1552  006340  013721  001262                    MOV     CLKLVL,(1)+     ;VECTORS AND PRIORITY
1553  006344  012721  006576                    MOV     #TINT3,(1)+     ;LEVELS
1554  006350  013721  001266                    MOV     XMTLVL,(1)+
1555  006354  012777  010101  172666            MOV     #BIT12+BIT6+BIT0,@CSR ;SET IE AND GO BITS
1556  006362  012700  000001                    MOV     #1,%0
1557  006366  005200                            INC     %0
1558  006370  000776                            BR      .-2
1559
1560
1561  006372  000240           RINT3:  NOP
1562  006374  000240                   NOP
1563  006376  013701  001300           MOV     TTPTR,%1        ;GET SOFTWARE POINTER
1564  006402  011137  001272   RINT3A: MOV     (1),TTDAT       ;GET TUMBLE TABLE ENTRY
1565  006406  001463                   BEQ     RINT3X          ;EXIT IF NO ENTRY
1566  006410  005011                   CLR     (1)             ;CLEAR ENTRY
1567  006412  032737  040000  001272   BIT     #BIT14,TTDAT    ;WAS BREAK RECEIVED
1568  006420  001047                   BNE     RINT3B          ;DO NOTHING ABOUT IT
1569  006422  042737  160400  001272   BIC     #160400,TTDAT   ;CLEAR ALL BUT LINE # AND DATA
1570  006430  113702  001273           MOVB    TTDAT+1,%2      ;GET LINE NUMBER
1571  006434  010237  001646           MOV     %2,LINE         ;FETCH LINE NUMBER
1572  006440  004737  004526           JSR     7,GTLINB        ;FORM LINE BIT FOR BAR
1573  006444  033777  001274  172600   BIT     LINBIT,@BAR     ;IS THIS LINE ACTIVE
1574  006452  001414                   BEQ     NONACT          ;LINE NOT ACTIVE
1575  006454  033777  001274  172570   BIT     LINBIT,@BAR     ;WAIT FOR LINE
1576  006462  001374                   BNE     .-6
1577  006464  032777  060000  172556   BIT     #BIT14+BIT13,@CSR
1578  006472  001401                   BEQ     .+4             ;BRANCH IF NO ERRORS
1579  006474  104003                   ERROR
1580  006476  042777  100000  172544   BIC     #BIT15,@CSR     ;CLEAR TRANSMIT DONE
1581  006504  113762  001272  006622   NONACT: MOVB    TTDAT,OUTBUF(2) ;STORE RECEIVED CHARACTER
1582  006512  012762  177777  001146           MOV     #-1,WCT(2)      ;LOAD LINE'S WORD COUNT
1583  006520  010203                           MOV     %2,%3
1584  006522  062703  006622                   ADD     #OUTBUF,%3
1585  006526  010362  001106                   MOV     %3,CAT(2)       ;AND CURRENT ADDRESS
1586  006532  053777  001274  172512           BIS     LINBIT,@BAR     ;ECHO RECEIVED CHARACTER
1587  006540  022701  001504   RINT3B: CMP     #TUMTAB+176,%1  ;CHECK TUMBLE
1588  006544  001002                   BNE     .+6             ;TABLE POINTER
1589  006546  012701  001304           MOV     #TUMTAB-2,%1
1590  006552  005721                   TST     (1)+
```

```
1591   006554  000712                          BR      RINT3A
1592   006556  042777  000200  172464  RINT3X: BIC     #BIT7,@CSR      ;CLEAR CHARACTER DONE FLAG
1593   006564  010137  001300                  MOV     %1,TTPTR        ;RESTORE POINTER
1594   006570  000240                          NOP
1595   006572  000240                          NOP
1596   006574  000002                          RTI                     ;EXIT
1597
1598   006576  000240                  TINT3:  NOP
1599   006600  032777  060000  172442          BIT     #BIT14+BIT13,@CSR ;ANY ERROR FLAGS SE
1600   006606  001401                          BEQ     .+4
1601   006610  104003                          ERROR
1602   006612  042777  160000  172430          BIC     #BIT15+BIT14+BIT13,@CSR ;CLEAR ALL FLAGS
1603   006620  000002                          RTI                     ;EXIT
```

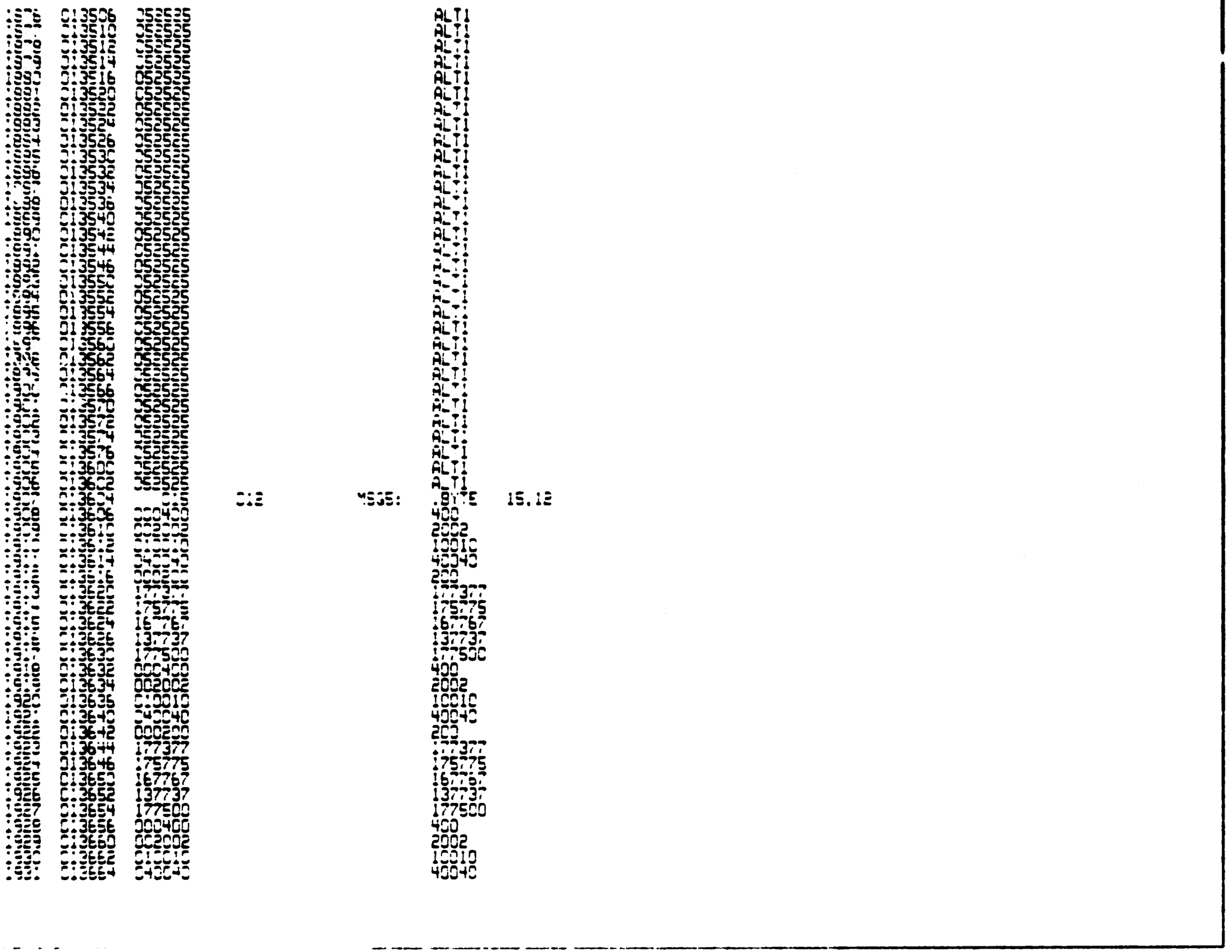
```
1604  006622  000000          OUTBUF: 0
1605                                  .=OUTBUF+100.
1606  006766  000000          LN0BUF: 0
1607                                  .=LN0BUF+100.
1608  007132  000000          LN1BUF: 0
1609                                  .=LN1BUF+100.
1610  007276  000000          LN2BUF: 0
1611                                  .=LN2BUF+100.
1612  007442  000000          LN3BUF: 0
1613                                  .=LN3BUF+100.
1614  007606  000000          LN4BUF: 0
1615                                  .=LN4BUF+100.
1616  007752  000000          LN5BUF: 0
1617                                  .=LN5BUF+100.
1618  010116  000000          LN6BUF: 0
1619                                  .=LN6BUF+100.
1620  010262  000000          LN7BUF: 0
1621                                  .=LN7BUF+100.
1622  010426  000000          LN10BF: 0
1623                                  .=LN10BF+100.
1624  010572  000000          LN11BF: 0
1625                                  .=LN11BF+100.
1626  010736  000000          LN12BF: 0
1627                                  .=LN12BF+100.
1628  011102  000000          LN13BF: 0
1629                                  .=LN13BF+100.
1630  011246  000000          LN14BF: 0
1631                                  .=LN14BF+100.
1632  011412  000000          LN15BF: 0
1633                                  .=LN15BF+100.
1634  011556  000000          LN16BF: 0
1635                                  .=LN16BF+100.
1636  011722  000000          LN17BF: 0
1637                                  .=LN17BF+100.
1638  012066  006766          INTAB:  LN0BUF
1639  012070  007132                  LN1BUF
1640  012072  007276                  LN2BUF
1641  012074  007442                  LN3BUF
1642  012076  007606                  LN4BUF
1643  012100  007752                  LN5BUF
1644  012102  010116                  LN6BUF
1645  012104  010262                  LN7BUF
1646  012106  010426                  LN10BF
1647  012110  010572                  LN11BF
1648  012112  010736                  LN12BF
1649  012114  011102                  LN13BF
1650  012116  011246                  LN14BF
1651  012120  011412                  LN15BF
1652  012122  011556                  LN16BF
1653  012124  011722                  LN17BF
1654  012126  000000          CNTTAB: 0
1655                                  .=CNTTAB+16.
1656
1657  012146  012206          ID:     IDENT0
1658  012150  012212                  IDENT1
1659  012152  012216                  IDENT2
```

```
 1660  012154  012222                    IDENT3
 1661  012156  012226                    IDENT4
 1662  012160  012232                    IDENT5
 1663  012162  012236                    IDENT6
 1664  012164  012242                    IDENT7
 1665  012166  012246                    IDNT10
 1666  012170  012252                    IDNT11
 1667  012172  012256                    IDNT12
 1668  012174  012262                    IDNT13
 1669  012176  012266                    IDNT14
 1670  012200  012272                    IDNT15
 1671  012202  012276                    IDNT16
 1672  012204  012302                    IDNT17
 1673  012206  105215          IDENT0: CRLF
 1674  012210  030060                    "00
 1675  012212  105215          IDENT1: CRLF
 1676  012214  030460                    "01
 1677  012216  105215          IDENT2: CRLF
 1678  012220  031060                    "02
 1679  012222  105215          IDENT3: CRLF
 1680  012224  031460                    "03
 1681  012226  105215          IDENT4: CRLF
 1682  012230  032060                    "04
 1683  012232  105215          IDENT5: CRLF
 1684  012234  032460                    "05
 1685  012236  105215          IDENT6: CRLF
 1686  012240  033060                    "06
 1687  012242  105215          IDENT7:CRLF
 1688  012244  033460                    "07
 1689  012246  105215          IDNT10: CRLF
 1690  012250  030061                    "10
 1691  012252  105215          IDNT11: CRLF
 1692  012254  030461                    "11
 1693  012256  105215          IDNT12: CRLF
 1694  012260  031061                    "12
 1695  012262  105215          IDNT13: CRLF
 1696  012264  031461                    "13
 1697  012266  105215          IDNT14: CRLF
 1698  012270  032061                    "14
 1699  012272  105215          IDNT15: CRLF
 1700  012274  032461                    "15
 1701  012276  105215          IDNT16: CRLF
 1702  012300  033061                    "16
 1703  012302  105215          IDNT17: CRLF
 1704  012304  033461                    "17
 1705  012306  105215                    CRLF
 1706          105215          CRLF=105215
 1707
```

```
1708
1709                                        ;MESSAGES
1710  012310  042045  030515  020061  WHERE:  .ASCII  '%DM11 RECEIVER VECTOR ADDRESS = @'
1711  012316  042522  042503  053111
1712  012324  051105  053040  041505
1713  012332  047524  020122  042101
1714  012340  051104  051505  020123
1715  012346  020075     100
1716  012351     045  053523  036522  $SWREG: .ASCII  '%SWR= @'
1717  012356  040040
1718  012360  020040  020040  020040  $VALUE: .ASCII  '          NEW= @'
1719  012366  020040  020040  042516
1720  012374  036527  040040
1721  012400  036445     040          $CTLU:  .ASCII  '%= '
1722  012403     045  044127  041511  WHICH:  .ASCII  '%WHICH DM11 ARE YOU TESTING @'
1723  012410  020110  046504  030461
1724  012416  040440  042522  054440
1725  012424  052517  052040  051505
1726  012432  044524  043516  040040
1727  012440  042045  052101  020101  ERDAT:  .ASCII  '%DATA ERR  S/B: '
1728  012446  051105  020122  051440
1729  012454  041057  020072
1730  012460  020040  020040  020040  AASB:   .ASCII  '        WAS: '
1731  012466  020040  040527  035123
1732  012474     040
1733  012475     040  020040  020040  AWAS:   .ASCII  '       @'
1734  012502  020040     100
1735  012505     114  047111  020105  LINEM:  .ASCII  'LINE # '
1736  012512  020043
1737  012514  020040  040040          ALINE:  .ASCII  '  @'
1738  012520  052045  050131  020105  M0:     .ASCII  '%TYPE PROGRAM #@'
1739  012526  051120  043517  040522
1740  012534  020115  040043
1741  012540  037445     100          M1:     .ASCII  '%?@'
1742  012543     045  042524  052123  M2:     .ASCII  '%TEST DZDMB COMPLETE@'
1743  012550  042040  042132  041115
1744  012556  041440  046517  046120
1745  012564  052105  040105
1746  012570  051445  052105  051440  M3:     .ASCII  '%SET SR OPTIONS. NORMAL OPERATION'
1747  012576  020122  050117  044524
1748  012604  047117  027123  047040
1749  012612  051117  040515  020114
1750  012620  050117  051105  052101
1751  012626  047511     116
1752  012631     123  020122  020075          .ASCII  'SR = 000000 PRESS CONT.@'
1753  012636  030060  030060  030060
1754  012644  050040  042522  051523
1755  012652  041440  047117  027124
1756  012660     100
1757  012661     045  040504  040524  PRGOM:  .ASCII  '%DATA TEST ALL LINES @'
1758  012666  052040  051505  020124
1759  012674  046101  020114  044514
1760  012702  042516  020123     100
1761  012707     045  040504  040524  PRG1M:  .ASCII  '%DATA TEST TRANSMIT ON ALL LINES SIMULTANEOUSLY@'
1762  012714  052040  051505  020124
1763  012722  051124  047101  046523
```

```
 1764  012730  052111  047440  020116
 1765  012736  046101  020114  044514
 1766  012744  042516  020123  044523
 1767  012752  052515  052114  047101
 1768  012760  047505  051525  054514
 1769  012766     100
 1770
 1771  012767     045  051124  047101  PRG2M:  .ASCII  '%TRANSMIT TO TERMINALS@'
 1772  012774  046523  052111  052040
 1773  013002  020117  042524  046522
 1774  013010  047111  046101  040123
 1775  013016  042445  044103  020117  PRG3M:  .ASCII  '%ECHO TEST@'
 1776  013024  042524  052123     100
 1777  013031     045  052520  020124  PRGI:   .ASCII  '%PUT CHAR IN SR(0-7),DELAY IN SR(8-15)@'
 1778  013036  044103  051101  044440
 1779  013044  020116  051123  030050
 1780  013052  033455  026051  042504
 1781  013060  040514  020131  047111
 1782  013066  051440  024122  026470
 1783  013074  032461  040051
 1784  013100  052045  050131  020105  POPPAR: .ASCII  '%TYPE PARITY OPTION (N=NOT DESIRED O=ODD, E=EVEN@'
 1785  013106  040520  044522  054524
 1786  013114  047440  052120  047511
 1787  013122  020116  047050  047075
 1788  013130  052117  042040  051505
 1789  013136  051111  042105  047440
 1790  013144  047475  042104  020054
 1791  013152  036505  053105  047105
 1792  013160     100
 1793  013161     045  020122          EMO:    .ASCII  '%R '
 1794  013164  020040  020040  041520  ATNUMB: .ASCII  '    PC= '
 1795  013172  020075
 1796  013174  020040  020040  020040  APC:    .ASCII  '      @'
 1797  013202     100
 1798  013203     015     012          MSG1:   .BYTE   15,12
 1799  013205     040  044124  020105          .ASCII  ' THE QUICK BROWN FOX JUMPED OVER THE LAZY DOGS BACK 1234567890'
 1800  013212  052521  041511  020113
 1801  013220  051102  053517  020116
 1802  013226  047506  020130  052512
 1803  013234  050115  042105  047440
 1804  013242  042526  020122  044124
 1805  013250  020105  040514  054532
 1806  013256  042040  043517  020123
 1807  013264  040502  045503  030440
 1808  013272  031462  032464  033466
 1809  013300  034470     060
 1810          013304                  .EVEN
 1811  013304     015     012          MSG2:   .BYTE   15,12
 1812  013306  177400                          177400
 1813  013310  177400                          177400
 1814  013312  177400                          177400
 1815  013314  177400                          177400
 1816  013316  177400                          177400
 1817  013320  177400                          177400
 1818  013322  177400                          177400
 1819  013324  177400                          177400
```

```
 1820  013326  177400                          177400
 1821  013330  177400                          177400
 1822  013332  177400                          177400
 1823  013334  177400                          177400
 1824  013336  177400                          177400
 1825  013340  177400                          177400
 1826  013342  177400                          177400
 1827  013344  177400                          177400
 1828  013346  177400                          177400
 1829  013350  177400                          177400
 1830  013352  177400                          177400
 1831  013354  177400                          177400
 1832  013356  177400                          177400
 1833  013360  177400                          177400
 1834  013362  177400                          177400
 1835  013364  177400                          177400
 1836  013366  177400                          177400
 1837  013370  177400                          177400
 1838  013372  177400                          177400
 1839  013374  177400                          177400
 1840  013376  177400                          177400
 1841  013400  177400                          177400
 1842  013402  177400                          177400
 1843  013404     015      012          MSG3:   .BYTE   15,12
 1844  013406  125252                          ALTO
 1845  013410  125252                          ALTO
 1846  013412  125252                          ALTO
 1847  013414  125252                          ALTO
 1848  013416  125252                          ALTO
 1849  013420  125252                          ALTO
 1850  013422  125252                          ALTO
 1851  013424  125252                          ALTO
 1852  013426  125252                          ALTO
 1853  013430  125252                          ALTO
 1854  013432  125252                          ALTO
 1855  013434  125252                          ALTO
 1856  013436  125252                          ALTO
 1857  013440  125252                          ALTO
 1858  013442  125252                          ALTO
 1859  013444  125252                          ALTO
 1860  013446  125252                          ALTO
 1861  013450  125252                          ALTO
 1862  013452  125252                          ALTO
 1863  013454  125252                          ALTO
 1864  013456  125252                          ALTO
 1865  013460  125252                          ALTO
 1866  013462  125252                          ALTO
 1867  013464  125252                          ALTO
 1868  013466  125252                          ALTO
 1869  013470  125252                          ALTO
 1870  013472  125252                          ALTO
 1871  013474  125252                          ALTO
 1872  013476  125252                          ALTO
 1873  013500  125252                          ALTO
 1874  013502     015      012          MSG4:   .BYTE   15,12
```



```
                                                    200
                                                    177777
                                                    175775
                                                    16776
                                                    137737
                                                    177500
                                    012             .BYTE   15.12
                                    012             .BYTE   15.12
                                                    .END
```

```
.      = 000000          291#
ABSB     012460          571    1730#
ABTIME   013514          531    1737#
ACTO   = 125252          289#   1844    1845    1846    1847    1848    1849    1850    1851    1852    1853    1854    1855
                        1856    1857    1858    1859    1860    1861    1862    1863    1864    1865    1866    1867    1868
                        1869    1870    1871    1872    1873    1874
ACT1   = 052525          288#   1876    1877    1878    1879    1880    1881    1882    1883    1884    1885    1886    1887
                        1888    1889    1890    1891    1892    1893    1894    1895    1896    1897    1898    1899    1900
                        1901    1902    1903    1904    1905    1906
ADD      013174          550    1796#
ADDCADD  001642          514#    540*    543*    549
ADDLAST= 177776          286#
ADDMB    013164          554    1794#
AWAS     012475          575    1733#
BAR      001252          452#    628*    783*    960     965*    980*    984    1031*   1513*   1514    1573    1575    1585*
BARCAT   001276          462#
BBCMB    001270          459#   1030*
BBSREG   001256          454#    630*    774*    906*
BBT      001206          448#    449
BELL   = 000007          295#
BITO   = 000001          249#    784     943     1512    1555
BIT1   = 000002          248#
BIT10  = 002000          239#
BIT11  = 004000          238#    635
BIT12  = 010000          237#    943     1509    1555
BIT13  = 020000          236#    527     546     1120    1123    1577    1593    1602
BIT14  = 040000          235#    631     1120    1123    1567    1577    1599    1602
BIT15  = 100000          233#    234#    1126    1580    1602
BIT2   = 000004          247#    784
BIT3   = 000010          246#
BIT4   = 000020          245#
BIT5   = 000040          244#
BIT6   = 000100          243#    784     943     1509    1555
BIT7   = 000200          242#   1112    1592
BIT8   = 000400          241#
BIT9   = 001000          240#    610     640
BKCSR    001254          452#    627*
CMCSR    004440          911     916     920     925     1012#   1500    1505
CRAMSK   001634          511#    778*    787*   1061
CRT      001106          444#    445     630     774     779*    906     975*   1013*   1499*   1501    1502    1535*
CSR    = 177560          228#
CSRCAT   004560          969    1050#   1521
CSRINT   001260          455#    749*    780*    935    1550
CSRLVL   001262          456#    781*    937    1552
CSRT     003524          815*    832*    851#    881
CSRTBO = 104020          283#    603     863    1495    1545
CSRTBR   003572          502     869#
CSRTBR   012126          922    1094    1095*   1654#   1655
CSRTMP   002050          526     544     569#
CSR_F  = 105215          1673    1675    1677    1679    1681    1683    1685    1687    1689    1691    1693    1695    1697
                        1699    1701    1703    1705    1706#
DSR      001250          451#    626*    767     775*    784*    786*    909*    943*    957*    964*   1112*   1120    1123*
                        1124    1126*   1509*   1512*   1518*   1555*   1577    1580*   1592*   1599    1602*
DSRCAT   001520          472#    612     618     663*
DSRINT=  177774          273#   1062
DSRLVL   001264          443#    895*
```

```
$$SPRE  000174   429#   895
$CYNMT  003732   955#  1148  1160  1172  1184  1196  1208  1220  1232  1244  1256  1268  1280
                1292  1304  1316  1328  1340  1352  1364  1376  1388  1400  1412  1424  1436
                1448  1460  1472  1484  1496  1546
$MPAR   002764   583   731#   744
$MPARB  003100   754#   762
$TCHK   001660   490   524#
EHALT = 104010   276#
EMT     013161   557  1793#
EMTINT  002614   305   667#
EMTTAB  001554   486#  513
ERCAT   012440   560  1727#
ERR     001726   489   539#
ERRA    001764   541   545#
ERRB    002026   539*  542*   558#
ERRC    002034   547   561#
ERROR = 104003   272#  1080  1085  1122  1579  1631
ERROR1= 104015   280#   533   966  1102
ERRVEC= 000004   230#   605*
ERR1    001745   499   542#
ESCOPE  002322   496   625#
FAIAG   003650   870   882   884#
FORWD   002462   609   658#
GETRDY  002212   602   604#   622   641   647  1138
GTLINB  004526  1038#  1572
GTRDYA  002240   609#   620
GTRDYC  002262   611   614#
GTRDYD  002304   617   619#
GTRDYX  002220   605#   643
IOTR    001526   475#   637*   661*
ID      012146   978  1657#
IDENT   004264   907   975#
IDENT0  012206  1657  1673#
IDENT1  012212  1658  1675#
IDENT2  012216  1659  1677#
IDENT3  012222  1660  1679#
IDENT4  012226  1661  1681#
IDENT5  012232  1662  1683#
IDENT6  012236  1663  1685#
IDENT7  012242  1664  1687#
IDNT10  012246  1665  1689#
IDNT11  012252  1666  1691#
IDNT12  012256  1667  1693#
IDNT13  012262  1668  1695#
IDNT14  012266  1669  1697#
IDNT15  012272  1670  1699#
IDNT16  012276  1671  1701#
IDNT17  012302  1672  1703#
INCRTN  001652   519#   621
INTAB   012066  1053  1091  1638#
KBCIN = 104017   282#   545   564   629
KBCINT  003526   501   956#
KSTART  001516   471#   604  1133*  1136*
LINBIT  001274   461#   944*   953*   976*   980   988*  1030  1031  1039*  1042*  1573  1575  1596
LINBUF  001650   517#  1056*  1057*  1058*  1059*  1060*  1061*  1062*  1063*
LINE    001646   516#   530   945*   946   952*   954*   957*   981*  1027  1040  1055*  1571*
```

```
LINEM   012505   535  1735#
LNOBUF  006766   924*  926  927  1636#  1607  1638
LN1BUF  007132  1608#  1609  1639
LN1OBF  010426  1622#  1623  1646
LN11BF  010572  1624#  1625  1647
LN12BF  010736  1626#  1627  1648
LN13BF  011102  1628#  1629  1649
LN14BF  011246  1630#  1631  1650
LN15BF  011412  1632#  1633  1651
LN16BF  011556  1634#  1635  1652
LN17BF  011722  1636#  1637  1653
LN2BUF  007276  1610#  1611  1640
LN3BUF  007442  1612#  1613  1641
LN4BUF  007606  1614#  1615  1642
LN5BUF  007752  1616#  1617  1643
LN6BUF  010116  1618#  1619  1644
LN7BUF  010262  1620#  1621  1645
MANUAL= 100000   233#
MSG1    013203  1149  1161  1173  1185  1197  1209  1221  1233  1497  1547  1792#
MSG2    013304  1245  1257  1269  1281  1293  1305  1317  1811#
MSG3    013404  1329  1341  1353  1365  1843#
MSG4    013504  1377  1389  1401  1413  1875#
MSG5    013604  1425  1437  1449  1461  1473  1485  1907#
M0      012520   587  1738#
M1      012540   520   743   761   847  1741#
M2      012543   645  1523  1742#
M3      012570   62?  1746#
NONACT  006504  157?  1581#
NOP   = 000240   231#
NXTST   001524   474#  604*  619  642  658  660*
OACNV   004352   529  548  552  569  573  873  990#
OPEN  = 000000   232#  444  446  448  450  455  457  459  460  461  462  463  465
                 471  472  473  474  475  476  487  488  491  492  493  494  495
                 504  509  510  511  512  513  514  515  516  517  558  687  688
                 702  703  727  912  1007
OUTBUF  006622   776*  779  913  1028  1060  1499  1581*  1584  1604#  1605
OVRLAY  003312   584  732  794#
PC    =%000007   267#
PCADD   001640   513#  540  543  669*
POPPAR  013100  1784#
POPSP =  005726   250#
POPSP2=  022626   251#  639  789
PRGI    013031  1777#
PRGLIM  001532   477#  590
PRGNUM  002130   589#  590*  591*  592  595  929  940  1510  1519
PRGTAB  001534   478#  593
PRGX    006104  1493#  1534
PRGO    005120   478  1131#
PRGOA   005124  1133#
PRGOM   012661  1132  1757#
PRGOR   005142   482  1136#
PRGI    006100   479  1491#
PRGIA   006126   942  1499#
PRGIC   006236  1519#
PRGID   006252  1520  1522#
PRGIEX  006256  1524#
```

```
PRG1M   012707          1492    1761#
PRG1R   006116           483    1494    1496#   1525
PRG2    006264           480    1532#
PRG2M   012767          1533    1771#
PRG2R   006270           484    1534#
PRG3    006274           481    1541#
PRG3A   006322           931    1543#
PRG3M   013016          1542    1775#
PRG3R   006312           485    1544    1546#
PRTY0 = 000000           259#
PRTY1 = 000040           256#
PRTY2 = 000100           257#
PRTY3 = 000140           256#
PRTY4 = 000200           255#    468     470
PRTY5 = 000240           254#    456     458
PRTY6 = 000300           253#
PRTY7 = 000340           252#    306
PRVCNT  001644           515#
PSW   = 177776           229#    606*    788*
RCVDAT  001630           509#    524     574     960*    1055*   1099*
RECD    003346           588     735     756     813#     879
RINT    004666           936    1075#
RINT3   006372          1551    1561#
RINT3A  006402          1564#   1591
RINT3B  006540          1569    1587#
RINT3X  006556          1565    1592#
RSTART  001544           482#    596
RSTAT1  002114           434     585#
RSTAT2  002154           436     594#
RSTREG= 104014           279#   1005    1019    1070    1114
RSTRG   002606           498     691#
RTNO    001522           473#    553     616     659*    1134*   1137*
RT10    005160          1133    1136    1140#
RT10B   005180          1143    1148#
RT11    005200          1141    1152#
RT11A   005210          1155    1160#
RT12    005360          1225    1236#
RT12A   005370          1239    1244#
RT13    005440          1237    1248#
RT13A   005450          1255    1256#
RT13B   005460          1243    1260#
RT14    005470          1263    1268#
RT14A   005500          1261    1272#
RT15    005510          1275    1280#
RT15A   005520          1273    1284#
RT16    005530          1287    1292#
RT16A   005540          1285    1296#
RT17    005550          1299    1304#
RT17A   005560          1297    1308#
RT20    005630          1311    1316#
RT20A   005640          1309    1320#
RT21    005650          1323    1328#
RT21A   005660          1153    1164#
RT22    005230          1167    1172#
RT22A   005240          1321    1332#
RT23    005250          1335    1340#
```

```
RT21    005600          1333    1344#
RT21A   005610          1347    1352#
RT22    005620          1345    1356#
RT22A   005630          1359    1364#
RT23    005640          1357    1368#
RT23A   005650          1371    1376#
RT24    005660          1369    1380#
RT24A   005670          1383    1388#
RT25    005700          1381    1392#
RT25A   005710          1395    1400#
RT26    005720          1393    1404#
RT26A   005730          1407    1412#
RT27    005740          1405    1416#
RT27A   005750          1419    1424#
RT3     005240          1165    1176#
RT3A    005250          1179    1184#
RT30    005760          1417    1428#
RT30A   005770          1431    1436#
RT31    006000          1429    1440#
RT31A   006010          1443    1448#
RT32    006020          1441    1452#
RT32A   006030          1455    1460#
RT33    006040          1453    1464#
RT33A   006050          1467    1472#
RT34    006060          1465    1476#
RT34A   006070          1479    1484#
RT35  = 177777          1477    1487#
RT4     005260          1177    1189#
RT4A    005270          1191    1196#
RT5     005300          1189    1200#
RT5A    005310          1203    1208#
RT6     005320          1201    1212#
RT6A    005330          1215    1220#
RT7     005340          1213    1224#
RT7A    005350          1227    1232#
R0    =%000000           260#     813     818*    819*    820     821     823     825*    826     831     944*    959*
R1    =%000001           261#
R2    =%000002           262#
R3    =%000003           263#
R4    =%000004           264#
R5    =%000005           265#     843     845*    873*
SAVREG= 104013           278#     990    1012    1050    1076
SAVRG   002546           497      677#
SCOPE = 104012           277#     971    1105
SCOPEA  002360           633#     638
SCOPEB  002366           632      635#
SCOPEC  002404           636      639#
SCOPED  002406           613      640#
SCOPTR  001530           476#     633     662#
SP    =%000006           266#     707     708*    813*    844     887*    888*    893     896     997    1026*   1033    1039*
                        1045    1524#
SPBOT   001100           441#     581     585     594     607    1524
SRSET   002170           597#    1135
SRT     001616           504#
STALL = 104002           271#
START   002076           433      581#
```

```
STRXV = 104006   274#
STTXV = 104007   275#
SUSWR = 104016   281#  582
SUSWRR  003652   500   887#
SWR     001102   442#  527   546   561   597   610   614   631   635   640   856   869   883*
                 890   894*  1493  1543
SWREG   000176   430#  597   856   869   874   894   1493  1543
TEMP    001636   512#
TINT    005060   938   1119#
TINT3   006576  1553   1598#
TKCSR   001620   505#  816
TKDBR   001622   506#  818   859
TKLVL   001510   468#
TKVTR   001506   467#
TMP1    003630   858*  859*  860*  861   880#  883
TPCSR   001624   507#  719   835   838   841
TPDBR   001626   508#  718*  820*  837*  840*
TPLVL   001514   470#
TPVTR   001512   469#
TTDAT   001272   460#  1078*  1082*  1088*  1089  1097  1564*  1567  1569*  1570  1591
TTPTR   001300   463#  934*  1077  1113*  1549*  1563  1593*
TUMTAB  001306   465#  466   782*  787   915*  917   918   921   934   1107  1109  1549  1587
                1589
TYP     002650   486   706#
TYPE  = 104000   269#  519   534   556   559   586   599   644   733   742   754   760   846
                 848   871   877   1131  1491  1522  1532  1541
TYPES = 104001   270#
UNIT    003110   757#  758   763*  764*  765*  769
VAC     001246   450#  464
VECTOR  003002   736#  737   739*  740   745   747   749   750*  751
WCT     001146   446#  447   777*  950   955   979*  1029*  1099  1504*  1506  1507  1582*
WHERE   012310   734   1710#
WHICH   012403   755   1722#
X     = 000034   290#  1139  1144#  1151  1156#  1163  1168#  1175  1180#  1187  1192#  1199  1204#
                1211  1216#  1223  1228#  1235  1240#  1247  1252#  1259  1264#  1271  1276#  1293
                1288#  1295  1300#  1307  1312#  1319  1324#  1331  1336#  1343  1348#  1355  1360#
                1367  1372#  1379  1384#  1391  1396#  1403  1408#  1415  1420#  1427  1432#  1439
                1444#  1451  1456#  1463  1468#  1475  1480#
XMITD   004462   948   1025#
XMTDAT  001632   510#  524   570   963*  1060*  1061*  1101*
XMTINT  001264   457#  751*
XMTLVL  001266   458#  939   1554
Y     = 000000   289#
$CTLU   012400   849   1721#
$ENDAD  002452   423   653#
$SWREG  012351   872   1716#
$VALUE  012360   875   878   1718#
.     = 013706   292#  295   297   299   301   303   307   309   311   313   315   317   319
                 321   323   325   327   329   331   333   335   337   339   341   343   345
                 347   349   351   353   355   357   359   361   363   365   367   369   371
                 373   375   377   379   381   383   385   387   389   391   393   395   397
                 399   401   403   405   407   409   411   413   415   417   419   422#  424#
                 428#  432#  439#  445#  447#  449#  464#  466#  720   930   941   1511  1515
                1517  1558  1576  1578  1588  1600  1605#  1607#  1609#  1611#  1613#  1615#  1617#
                1619#  1621#  1623#  1625#  1627#  1629#  1631#  1633#  1635#  1637#  1655#  1913#
```

GET	211#	598	735	756	879										
HEADER	211#	1139	1151	1163	1175	1187	1199	1211	1223	1235	1247	1259	1271	1283	1295
	1307	1319	1331	1343	1355	1367	1379	1391	1403	1415	1427	1439	1451	1463	1475
XMTDLY	211#	1139	1151	1163	1175	1187	1199	1211	1223	1235	1247	1259	1271	1283	1295
	1307	1319	1331	1343	1355	1367	1379	1391	1403	1415	1427	1439	1451	1463	1475

```
ADD      673   708   750   769   800   952   958   981   994   997  1096  1584
ASL      591   672   763   764   765   828   929   830   953   982
ASR     1092
BCC      954   983
BEQ      525   598   632   647   715   799   822   924   833   882   891   941   961  1083  1121
        1511  1520  1565  1574  1578  1600
BHI      746
BHIS     741
BIC      590   615   768   957   996   999  1061  1088  1112  1123  1126  1569  1580  1592  1602
BICB     787   819   825   860
BIS      980  1030  1031  1512  1586
BISB     831
BIT      527   546   610   631   635   640   747  1120  1567  1573  1575  1577  1599
BITB     826
BLE     1100
BLOS     759
BLT     1069
BMI     1079  1094
BNE      528   547   611   617   620   636   638   641   643   712   738   748   771   827   857
         862   870   930   951   956   959   985  1004  1018  1065  1108  1494  1515  1517  1544
        1568  1576  1588
BPL      562   720   817   836   839   842  1044  1125
BR       541   602   613   622   717   726   744   762   801   834   850   892   967  1081  1086
        1111  1525  1558  1591
CLR      506   626   627   628   775   782   786   788   797   814   840   858   909   915   924
         945   963   964   965   975  1039  1052  1054  1066  1087  1093  1134  1137  1518  1566
CLRB     671
CMP      597   619   642   740   745   758   798   856   869   881   890   893   929   950   970
        1064  1068  1107  1493  1510  1519  1543  1587
CMPB     524   616   711   714   821   823   861
DEC      637   770   832  1003  1017  1057
EMT      269   270   271   272   273   274   275   276   277   278   279   280   281   282   283
HALT     293   294   296   298   300   302   304   308   310   312   314   316   318   320   322
         324   326   328   330   332   334   336   338   340   342   344   346   348   350   352
         354   356   358   360   362   364   366   368   370   372   374   376   378   380   382
         384   386   388   390   392   394   396   398   400   402   404   406   409   410   412
         414   416   418   420   563   601
INC     1058  1063  1067  1516  1557
INCB    1095
JMP      433   434   436   593   596   612   618   931   942  1135  1138  1534
JSR      526   529   544   548   552   569   573   583   584   588   609   621   653   716   723
         725   732   735   756   873   879   907   911   916   920   925   948   969  1149  1160
        1172  1184  1196  1208  1220  1232  1244  1256  1268  1280  1292  1304  1316  1328  1340
        1352  1364  1376  1388  1400  1412  1424  1436  1448  1460  1472  1484  1496  1500  1505
        1521  1546  1572
MOV      539   540   542   543   581   585   592   594   595   604   605   607   614   630   633
         646   658   659   660   661   662   663   667   669   670   674   677   678   679   680
         681   682   683   684   685   692   693   694   695   696   697   698   699   700   707
         709   739   749   751   766   767   774   776   777   778   779   780   781   783   784
         794   795   796   813   815   837   844   883   887   888   889   894   895   896   897
         906   910   932   934   935   936   937   938   939   943   944   946   960   976   977
         978   979   991   992   993   995  1013  1014  1015  1026  1027  1028  1029  1033  1038
        1040  1045  1053  1055  1056  1077  1078  1082  1090  1091  1099  1101  1109  1113  1133
        1136  1499  1504  1509  1513  1524  1549  1550  1551  1552  1553  1554  1555  1556  1563
        1564  1571  1582  1583  1585  1589  1593
MOVB     710   718   722   724   818   820   859   998  1016  1059  1060  1089  1094  1097  1570
```

DZDMBB.SRC CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
        1591
NOP      625   649   650   651   652   654   655   656   691   706   731   802   905   908   947
         968   986  1025  1034  1051  1075  1098  1115  1119  1548  1561  1562  1594  1595  1598
RESET    608   648  1104
ROL     1042
RCR     1000  1001  1002
RTI      536   565   634   686   701   713   864   884   898  1116  1127  1596  1603
RTS      521   577   664   721   790   803   845   987  1006  1020  1035  1046  1071
SEC     1041
SUB      668  1043
TST      561   737   843   940   955   984  1110  1124  1514  1590
TSTB     719   816   835   838   841
WAIT     785
.ASCII  1710  1716  1718  1721  1722  1727  1730  1733  1735  1737  1738  1741  1742  1746  1752
        1757  1761  1771  1775  1777  1784  1793  1794  1796  1799
.BYTE   1798  1811  1843  1875  1907  1938  1939
.ENABL   209
.END    1940
.EVEN   1810
.LIST    208   211
.MACR    211
.NLIST   207   211
.REM       3
.REPT   1812  1844  1876  1908
.TITLE   206
```

```
ERRORS DETECTED:  0
DEFAULT GLOBALS GENERATED:  0

*,DZDMBB.SEQ/SOL/CRF=DZDMBB.SRC
RUN-TIME:  5 12 2 SECONDS
RUN-TIME RATIO:  120/20=5.7
CORE USED:  8K  (15 PAGES)
```

Spooler runtime 7 Seconds, 35 KCS, 187 disk reads, 4 disk writes, 50 pages