VT20

BOOTSTRAP LOADER

MD-11-DZBMK-B

EP-DZBMK-B-DL

FICHE 1 OF 1

MAY 1978

digital

MADE IN USA

IDENTIFICATION

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

PRODUCT NAME: M792YK VT20 BOOTSTRAP LOADER

DATE CREATED JANUARY 11, 1974

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: ED BADGER

1. ABSTRACT

THE DZBNK DIAGNOSTIC PROGRAM IS WRITTEN TO BE USED AS AN AID
TO HARDWARE DEBUGGING AND MAINTENANCE OF THE M792YK
VT20 BOOTSTRAP LOADER). THIS PROGRAM MAY ALSO BE USED
AS A DATA PELIABILITY TEST.

THE AVAILABLE TESTS ARE
PRG0 = LOGIC TESTS
PRG1 = ROM DATA DUMP
PRG2 = SINGLE ROM ADDRESS READ DATA LOOP

2. REQUIREMENTS

2.1 EQUIPMENT

A. PDP 11 FAMILY CENTRAL PROCESSOR/WITH VT20

B. M792-YK MODULE

2.2 STORAGE

THIS PROGRAM USES CORE 0-4100(8)

3. LOADING AND STARTING PROCEDURE

LOAD PROGRAM INTO MEMORY USING ABS LOADER.
LOAD ADDRESS = 00200
SET SWR = DESIRED STANDARD PDP-11 DIAGNOSTIC OPTIONS (SEE
SECT 4.0)
NOTE: ALL SWITCHES = 0 SELECTS AND STARTS PROGRAM 0
DEPRESS START. THE PROGRAM WILL THEN TYPE OUT INSTRUCTIONS. ALL
USER RESPONSES ARE VIA THE "TUBE 0" KEYBOARD (CARRIAGE RETURN
TERMINATES THE RESPONSE)
TO RESTART THE SELECTED PROGRAM LOAD ADDRESS = 000210 AND
DEPRESS START

4.0 SWITCH SETTINGS

SW15 1 OR UP HALT ON ERROR
SW14 1 OR UP SCOPE LOOP
SW13 1 OR UP INHIBIT PRINTOUT
SW12 1 OR UP INHIBIT TRACE TRAPPING (NOT USED)
SW11 1 OR UP INHIBIT ITERATION

5. PROGRAM DESCRIPTIONS

5.1 PRG0 - LOGIC TESTS

THE LOGIC TESTS CONSIST OF 4 ROUTINES TO TEST THE M792YK LOGIC. PROGRAM 0 LOOPS WITHIN ITSELF UNTIL A NEW PROGRAM NUMBER IS PROVIDED I.E. PRG#= ?

5.1.1 ROUTINE DESCRIPTIONS

ROUTINE	TESTS
T1	ADDRESSABILITY OF M792YK
T2	DATA RELIABILITY
T3	THAT M792YK TIMES OUT WHEN REFERENCED BY A DATIP BUS CYCLE
T4	THAT DATA READ IS CORRECT

5.1.2 ERROR PRINTOUT

IF A ROUTINE FAILS AND THE INHIBIT PRINTOUT SWITCH IS NOT ENABLED (SW13) A PRINTOUT RESULTS. I.E. THE PC AT THE TIME OF FAILURE IS TYPED.

IF AN ERROR OCCURS IN T4, THE ROM DATA, CORRECT DATA, AND THE ADDRESS OF EACH IS TYPED OUT (THE ERROR TYPEOUT CANNOT BE DISABLED). THE FORMAT IS

ROM ADDRESS/ROM DATA
IMAGE ADDRESS*CORRECT DATA

5.2 PRG1 - ROM DATA DUMP

THIS PROGRAM TYPES OUT THE 32 WORDS OF ROM DATA AND THEN TYPES OUT 'PRG#=' REQUESTING WHAT PROGRAM TO PERFORM NEXT.

5.3 PRG2 - SINGLE ROM ADDRESS READ DATA LOOP

THIS PROGRAM CONTINUOUSLY READS DATA FROM A TYPED IN ROM ADDRESS. TO CHANGE THE ADDRESS TYPE IN A NEW ADDRESS.(MUST BE EVEN) FOLLOWED BY A CARRIAGE RETURN.

NOTE: THE ROM WORDS STARTING AT LOCATION 4000 ARE DESIGNATED AS FOLLOWS -

LOCATION	CONTENTS	WORD NO.
4000	012701	0
4002	160000	2
4004	012702	4
4006	000006	6
4010	012703	10

ETC.

```
                        ;LOAD ADDRESS=0200
                        ;DEPRESS START
                        ;RESTART ADDRESS=0210
                        ;STACK POINTER IS AT 500

                                .MCALL  $CPREGS,$CPVECS
                                .MCALL  $CMTAG
                        ;DEFINITIONS AND REGISTER ASSIGNMENTS
                        ;GENERAL REGISTER ASSIGNMENTS
000000                          R0=%0
000001                          R1=%1
000002                          R2=%2
000003                          R3=%3
000004                          R4=%4
000005                          R5=%5
000006                          SP=%6
000007                          PC=%7
000000                          R10=%0
000001                          R11=%1
000002                          R12=%2
000003                          R13=%3
000004                          R14=%4
000005                          R15=%5

                        ;REGISTER ADDRESSES
177776                          PSW=    177776          ;PROCESSER STATUS WORD
177774                          SLR=    177774          ;STACK LIMIT REGISTER (11/40,11/45)
177772                          PIRQ=   177772          ;PROGRAM INTERRUPT REQ. (11/45)
177770                          UBREAK= 177770          ;MICRO-BREAK REGISTER (11/45)
177570                          SWR=    177570          ;SWITCH REGISTER
177570                          DISPLAY=177570          ;DISPLAY REGISTER (11/45)
177560                          TKS=    177560          ;KEYBOARD CSR
177562                          TKB=    177562          ;KEYBOARD DATA BUFFER REGISTER
177564                          TPS=    177564          ;TELEPRINTER CSR
177566                          TPB=    177566          ;TELEPRINTER DATA BUFFER REGISTER

170334                          FKS = 170334            ;TUBE 0 KEYBOARD CSR
170336                          FKB = 170336            ;TUBE 0 KEYBOARD DATA
170304                          VTC = 170304            ;TUBE 0 DISPLAY CSR
170302                          VTS = 170302            ;TUBE 0 DISPLAY STARTING ADR.
                        ;VECTOR ADDRESSES
000004                          ERRVEC=4                ;ADDRESS OF ERROR VECTOR
000010                          RESVEC=10               ;ADDRESS OF RESERVED INST. TRAP VECTOR
000014                          TBITVEC=14              ;ADDRESS OF 'T' BIT TRAP VECTOR
000014                          TRTVEC=14               ;ADDRESS OF 'TRACE' TRAP VECTOR
000014                          BPTVEC=14               ;ADDRESS OF 'BREAKPOINT' TRAP VECTOR
000020                          IOTVEC=20               ;ADDRESS OF IOT TRAP VECTOR
000024                          PFVEC=24                ;ADDRESS OF POWER FAIL TRAP VECTOR
000030                          EMTVEC=30               ;ADDRESS OF EMT VECTOR
000034                          TRAPVEC=34              ;ADDRESS OF TRAP VECTOR
000060                          TKVEC=  60              ;ADDRESS OF TTY KEYBOARD INT. VECTOR
000064                          TPVEC=64                ;ADDRESS OF TTY PRINTER INTERRUPT VECTOR
000114                          PARVEC= 114             ;ADDRESS OF MA/MF PARITY ERROR VECTOR
000240                          PIRVEC=240              ;ADDRESS OF PIRQ VECTOR
```

```
                000244                          FPEVEC=244              ;ADDRESS OF FLOATING POINT INT. VECTOR
                000250                          MMVEC=250               ;ADDRESS OF MEM MGMT ERROR TRAP VECTOR

                000320                          FKVEC = 320
                000020                          .=20
      000020    003234                          .WORD   VTOUT
      000022    000240                          .WORD   240
                000030                          .=30
      000030    002764                          .SCOPE
      000032    000340                          340
      000034    003074                          .HLT
      000036    000000                          0
                                        ;EQUATE STATEMENTS
                000500                          STKPTR= 500
                000004                          TYPE=   IOT
                104400                          HLT=    TRAP
                104000                          SCOPE=  EMT

                000200                          .=200
      000200    000167  001132        START1: JMP     PRMTRS
                000210                          .=210
      000210    000167  001210        START3: JMP     RESTART

                001300                          .=1300

                                        ;WORDS LOADED BY 'SCOPE'
      001300    000000                SCPBLK: .WORD   0               ;CONTAINS PASS COUNT
                001300                          ICNT=SCPBLK
      001302    000002                ITCNT:  .WORD   2               ;CONTAINS SUBTEST ITERATION COUNT
      001304    000004                LASTPC: .WORD   4               ;CONTAINS LAST SCOPE CALL PC

      001306    000000                EPC:    .WORD   0               ;CONTAINS SCOPE RETURN FOR ERRORS
      001310    000000                ERRFLG: .WORD   0               ;CONTAINS ERROR FLAG
      001312    000000                TICKS:  .WORD   0               ;CONTAINS TICK COUNT FOR CLOCKS
      001314    177560                INCSR:  .WORD   177560          ;ADDRESS OF INPUT CSR
      001316    177562                INDAT:  .WORD   177562          ;ADDRESS OF INPUT DEVICE DATA BUFFER REG
      001320    000040                WORDS:  32.
      001322    004000                IMAGE:  4000
      001324    000000                TEMP:   0
      001326    173000                ROMADD: 173000                  ;FIRST ADDRESS OF DATA
      001330    001442                PRGTAB: PRG0
      001332    002202                        PRG1
      001334    002322                        PRG2
      001336    012706  000500        PRMTRS: MOV     #STKPTR,%6      ;SET STACK PTR
      001342    005067  000040                CLR     PRGNUM
      001346    004767  001750                JSR     %7,RESBF                ;OVERLAY PRINTOUT AREA
      001352    010467  166724                MOV     R4,VTS
      001356    012767  000001  166720        MOV     #1,VTC
      001364    005737  177570                TST     @#SWR
      001370    001415                        BEQ     RESTART
      001372    000004                FIXED:  TYPE
      001374    003553                        MHED
      001376    000004                        TYPE
      001400    003365                        M6
```

```
001402  004567  001120                  JSP     5,RECD          ;RECEIVE DATA AND PUT
001406  000000                  PRGNUM: 0                       ;IT HERE
001410  000004                          TYPE
001412  003414                          %8

001414  026727  177766  000002          CMP     PRGNUM,#2       ;SEE IF LEGAL PROGRAM NUMBER
001422  003345                          BGT     PRMIPS          ;IF NOT RETYPE QUESTION


001424  005067  000550          RESTART:CLR     PCNT            ;CLEAR PASS COUNT
001430  016700  177752                  MOV     PRGNUM,%0       ;GET PROGRAM #
001434  006300                          ASL     %0              ;SHIFT PROGRAM #
001436  000170  001330                  JMP     @PRGTAB(0)      ;GO TO PROGRAM
```

```
                                        ;PROGRAM 0 LOGIC TESTS
001442  005067  177632        PRG0:   CLR     ICNT            ;CLEAR PASS COUNT
001446  012767  001462  177630 PRG0A:  MOV     #T1,LASTPC      ;SET RETURN ADDRESS FOR SCOPE
001454  016737  177620  177570        MOV     ICNT,@#DISPLAY  ;DISPLAY PASS COUNT

                                        ;TEST1 TEST ABILITY TO REFERENCE ROM WITHOUT TIMING OUT
001462  012706  000500        T1:     MOV     #STKPTR,SP      ;SET STACK PTR
001466  016700  177634                MOV     ROMADD,%0       ;GET ROM ADDRESS
001472  016701  177622                MOV     WORDS,%1        ;GET ADDRESS COUNTER
001476  012767  001536  176300        MOV     #ERROR1,4       ;SET UP TIME OUT VECTOR
001504  011003                T1A:    MOV     (0),%3          ;REFERENCE
001506  005720                        TST     (0)+            ;ROM
001510  064067  177610                ADD     -(0),TEMP       ;
001514  021010                        CMP     (0),(0)         ;
001516  132020                        BITB    (0)+,(0)+       ;
001520  164067  177600                SUB     -(0),TEMP
001524  062700  000002                ADD     #2,%0           ;INCREMENT POINTER
001530  005301                        DEC     %1              ;DECREMENT ADDRESS COUNTER
001532  001364                        BNE     T1A             ;BRANCH IF NOT FINISHED
001534  000403                        BR      T1B             ;GO TO SCOPE LOOP
001536  022626                ERROR1: CMP     (6)+,(6)+       ;REPOSITION STACK
001540  104400                        HLT                     ;ERROR: ROM TIMED OUT WHEN REFERENCED
                                                                ;ADDRESS IS IN R0
001542  000760                        BR      T1A             ;LOOP ON ERROR
001544  104000                T1B:    SCOPE

                                        ;TEST2  TEST THAT ROM DATA CAN BE READ RELIABLY.
001546  016700  177554        T2:     MOV     ROMADD,%0       ;GET ROM ADDRESS
001552  016701  177542                MOV     WORDS,%1        ;GET ADDRESS COUNTER
001556  012767  000006  176220        MOV     #6,4            ;INITIALIZE TIME OUT VECTOR
001564  005067  177534        T2A:    CLR     TEMP            ;INITIALIZE TEMP
001570  011003                        MOV     (0),%3          ;GET DATA
001572  062067  177526                ADD     (0)+,TEMP       ;ADD DATA TO TEMP
001576  166703  177522                SUB     TEMP,%3         ;SUBTRACT DATA FROM DATA
001602  001402                        BEQ     T2B             ;BRANCH IF EQUAL
001604  104400                ERROR2: HLT                     ;DATA ERROR
001606  000766                        BR      T2A             ;LOOP ON ERROR
001610  044067  177510        T2B:    BIC     -(0),TEMP       ;CLEAR TEMP BITS
001614  001402                        BEQ     T2C             ;BRANCH IF EQUAL TO 0
001616  104400                        HLT                     ;DATA ERROR
001620  000773                        BR      T2B             ;LOOP ON ERROR
001622  021010                T2C:    CMP     (0),(0)         ;COMPARE DATA
001624  001402                        BEQ     T2D             ;BRANCH IF EQUAL
001626  104400                        HLT                     ;DATA ERROR
001630  000774                        BR      T2C             ;LOOP ON ERROR
001632  122040                T2D:    CMPB    (0)+,-(0)       ;COMPARE DATA (BYTE OPERATION)
001634  001402                        BEQ     T2E             ;BRANCH IF EQUAL
001636  104400                        HLT                     ;DATA ERROR
001640  000774                        BR      T2D             ;LOOP ON ERROR
001642  005720                T2E:    TST     (0)+            ;INCREMENT ADDRESS POINTER
001644  005301                        DEC     %1              ;DECREMENT ADDRESS COUNTER
001646  001346                        BNE     T2A             ;RETURN IF NOT DONE
001650  104000                        SCOPE
```

```
                                        ;TEST3 TEST THAT ROM TIMES OUT IF REFERENCED BY OTHER
                                        ;THAN DATI BUS CYCLE
001652  012706  000500          T3:     MOV     #STKPTR,%6      ;SET STACK PTR
001656  016700  177444                  MOV     ROMADD,%0       ;GET ROM ADDRESS
001662  016701  177432                  MOV     WORDS,%1        ;GET ADDRESS COUNTER
001666  012767  001702  176110  T3AA:   MOV     #T3B,4          ;SET UP TIME OUT VECTOR
001674  010010                  T3A:    MOV     %0,(0)          ;ATTEMPT TO ALTER DATA
001676  104400                          HLT                     ;HERE IF DID NOT TIME OUT
001700  000775                          BR      T3A             ;LOOP ON ERROR
001702  012767  001720  176074  T3B:    MOV     #T3D,4          ;SET UP TIME OUT VECTOR
001710  022626                          CMP     (6)+,(6)+       ;REPOSITION STACK
001712  005210                  T3C:    INC     (0)             ;ATTEMPT TO ALTER DATA
001714  104400                          HLT                     ;HERE IF DID NOT TIME OUT
001716  000775                          BR      T3C             ;LOOP ON ERROR
001720  012767  001740  176056  T3D:    MOV     #T3F,4          ;SET UP TIME OUT VECTOR
001726  022626                          CMP     (6)+,(6)+       ;REPOSITION STACK
001730  005077  177372          T3E:    CLR     @ROMADD         ;ATTEMPT TO ALTER DATA
001734  104400                          HLT                     ;HERE IF DID NOT TIME OUT
001736  000774                          BR      T3E             ;LOOP ON ERROR
001740  005720                  T3F:    TST     (0)+            ;INCREMENT ADDRESS POINTER
001742  022626                          CMP     (6)+,(6)+       ;REPOSITION STACK
001744  005301                          DEC     %1              ;DECREMENT ADDRESS COUNTER
001746  001347                          BNE     T3AA            ;RETURN IF NOT DONE
001750  012737  000006  000004          MOV     #6,@#4          ;RESTORE TIME OUT TRAP
001756  104000                          SCOPE                   ;SCOPE LOOP

                                        ;THIS TEST COMPARES ROM AND IMAGE DATA
                                        ;AND TYPES OUT DIFFERENCES
001760  012706  000500          T4:     MOV     #STKPTR,%6      ;SET STACK PTR
001764  016701  177330                  MOV     WORDS,%1        ;GET # OF WORDS
001770  016700  177332                  MOV     ROMADD,%0       ;GET ROM ADDRESS
001774  016703  177322                  MOV     IMAGE,%3        ;GET IMAGE ADDRESS
002000  021013                  T4B:    CMP     (0),(3)         ;COMPARE DATA
002002  001004                          BNE     T4D
002004  005301                  T4C:    DEC     %1              ;ALL DATA BEEN COMPARED
002006  001441                          BEQ     T4E
002010  022023                          CMP     (0)+,(3)+       ;INCREMENT ADDRESS POINTERS
002012  000772                          BR      T4B
002014  000004                  T4D:    TYPE
002016  003414                          M8
002020  010067  001110                  MOV     %0,D2BTYP       ;TYPE
002024  004767  001106                  JSR     7,D2A           ;ROM ADDRESS
002030  000004                          TYPE
002032  003501                          M10                     ;SEPARATOR
002034  011067  001074                  MOV     (0),D2BTYP      ;TYPE
002040  004767  001072                  JSR     7,D2A           ;ROM DATA
002044  000004                          TYPE
002046  003414                          M8                      ;CR/LF
002050  032737  040000  177570  1$:     BIT     #40000,@#SWR
002056  001374                          BNE     1$
002060  010367  001050                  MOV     %3,D2BTYP       ;TYPE
002064  004767  001046                  JSR     7,D2A           ;IMAGE ADDRESS
002070  000004                          TYPE
002072  003505                          M12                     ;SEPARATOR
```

```
002074  011367  001034              MOV     (3),D2BTYP      ;TYPE
002100  004767  001032              JSR     7,D2A           ;IMAGE DATA
002104  000004                      TYPE
002106  003414                      MB                      ;CR/LF
002110  000735                      BR      T4C             ;GO TO T4C
002112  104000              T4E:    SCOPE

002114  005267  177160      END:    INC     ICNT            ;INCREMENT PASS COUNT
002120  026727  177154  000100      CMP     ICNT,#100
002126  001402                      BEQ     DONE
002130  000167  177312              JMP     PRGOR           ;GO RESTART PROGRAM
002134  000004              DONE:   TYPE                    ;RING THE BELL
002136  003507                      MBELL
002140  005267  000034              INC     PCNT
002144  016767  000030  000762      MOV     PCNT,D2BTYP     ;GET PASS COUNT
002152  004767  000760              JSR     %7,D2A
002156  013700  000042              MOV     @#42,%0         ;RETURN TO DECTAPE MONITOR?
002162  001404                      BEQ     DONE1
002164  004710                      JSR     7,(0)           ;RETURN!
002166  000240                      NOP
002170  000240                      NOP
002172  000240                      NOP
002174  000167  177242      DONE1:  JMP     PRGO
002200  000000              PCNT:   0                       ;PASS COUNT
```

```
                                ;THIS PROGRAM TYPES OUT ROM DATA

002202  005067  000234        PRG1:   CLR     PRG2B   ;INITATE A DELAY
002206  005267  000230                INC     PRG2B           ;AND EXECUTE DELAY
002212  001375                        BNE     .-4
002214  004767  001102                JSR     %7,RESBF        ;RESET BUFFER AREA
002220  012706  000500                MOV     #STKPTR,%6      ;INITIALIZE STACK
002224  000004                        TYPE
002226  003377                        M7                      ;'ROM DATA'
002230  016701  177064                MOV     WORDS,%1        ;GET # OF WORDS
002234  016700  177066        PRG1A:  MOV     ROMADD,%0       ;GET STARTING ADDRESS
002240  012702  000011                MOV     #11,%2          ;GET ADDRESS INDICATOR
002244  010067  000664        PRG1B:  MOV     %0,D2BTYP       ;GET ADDRESS
002250  004767  000662                JSR     7,D2A           ;AND TYPE IT
002254  000004                        TYPE
002256  003414                        M6                      ;CR/LF
002260  012067  000650        PRG1C:  MOV     (0)+,D2BTYP     ;TYPE
002264  004767  000646                JSR     7,D2A           ;DATA
002270  000004                        TYPE
002272  003503                        M11
002274  005301                        DEC     %1              ;ALL DATA TYPED.
002276  001407                        BEQ     PRG1D           ;GO TO FINISH
002300  005302                        DEC     %2
002302  001366                        BNE     PRG1C           ;RETURN TO PRG1B
002304  012702  000011                MOV     #11,%2          ;GET ADDRESS INDICATOR
002310  000004                        TYPE
002312  003414                        M8                      ;CR/LF
002314  000753                        BR      PRG1B           ;RETURN TO PRG1B
002316  000167  177050        PRG1D:  JMP     FIXED           ;GO GET NEXT TEST

                                ;THIS PROGRAM CYCLES A SINGLE ADDRESS (ADDRESS MUST BE EVEN) TO CHANGE
                                ;THE ADDRESS TYPE NEW ADDRESS ON THE TTY.
002322  012706  000500        PRG2:   MOV     #STKPTR,%6      ;INITIALIZE STACK POINTER
002326  012737  002510  000004        MOV     #PRG2C,@#4      ;LOAD TRAP ERROR VECTOR
002334  005067  175436                CLR     PSW             ;CLEAR PROCESSOR STATUS
002340  012737  002412  000320        MOV     #PRG2A,@#FKVEC  ;LOAD KEYBOARD INTERRUPT VECTOR
002346  012737  000340  000322        MOV     #340,@#FKVEC+2  ;LOAD KEYBOARD PRIORITY
002354  052737  000101  170334        BIS     #101,@#FKS      ;SET INTERRUPT ENABLE BIT
002362  016700  176740                MOV     ROMADD,%0       ;GET ROM ADDRESS
002366  000004        PRG20:  TYPE
002370  003663                MCYC
002372  010067  000536                MOV     %0,D2BTYP
002376  004767  000534                JSR     7,D2A
002402  000004                        TYPE
002404  003414                        M8
002406  005710                        TST     (0)             ;READ ROM ADDRESS
002410  000776                        BR      .-2             ;LOOP
002412  105767  165716        PRG2A:  TSTB    FKS             ;DID KB INTR?
002416  100407                        BMI     4$              ;YES, GET INFO
002420  052767  000002  165706        BIS     #2,FKS
002426  052767  000101  165700        BIS     #101,FKS
002434  000002                        RTI                     ;NO - EXIT
002436  004567  000064        4$:     JSR     %5,RECD         ;GO GET ADDR &
002442  000000        PRG2B:  0                               ;PUT IT HERE
```

```
002444  016700  177772                  MOV     PRG2B,R0
002450  000004                           TYPE
002452  003414                           .BYTE   8.,13.                  ;CR/LF
002454  026727  177762  173000           CMP     PRG2B,#173000
002462  002415                           BLT     PRER
002464  026727  177752  173100           CMP     PRG2B,#173100
002472  002011                           BGE     PRER
002474  012737  002412  000320  PRERN:   MOV     #PRG2A,@#FKVEC
002502  022626                           CMP     (6)+,(6)+               ;RESET STACK POINTER
002504  000167  177656                  JMP     PRG20
002510  104400                 PRG2C:    HLT                             ;ERROR! DID YOU TYPE AN ODD ADDRESS?
002512  000167  177726                  JMP     PRG2B+2                 ;TRY ADDRESS AGAIN

002516  000004                 PRER:     TYPE
002520  003632                           .BYTE   27.,'A
002522  000167  177746                  JMP     PRERN
```

```
                                        ;ROUTINE TO RECEIVE DATA TYPED IN ON THE KEYBOARD, THE DATA IS PLACED IN
                                        ;THE ADDRESS FOLLOWING THE JSR CALL:
                                        ;               JSP     5,RECD  ;CALL RECEIVE DATA ROUTINE
                                        ;               0               ;DATA IS PLACED HERE
002526  012767  002572  175564  RECD:   MOV     #RECDI,FKVEC
002534  005015                        CLR     (5)
002536  005067  000104                CLR     DONF
002542  052767  000002  165564        BIS     #2,FKS
002550  052767  000101  165556        BIS     #101,FKS
002556  005067  175214                CLR     PSW
002562  005767  000060        RECD1:  TST     DONF
002566  001775                        BEQ     RECD1
002570  000205                        RTS     R5
002572  105737  170334        RECDI:  TSTB    @#FKS           ;TEST KEYBOARD FLAG
002576  100063                        BPL     REX             ;AND WAIT FOR CHARACTER
002600  113746  170336                MOVB    @#FKB,-(SP)     ;GET CHARACTER
002604  042716  177700                BIC     #177700,(SP)    ;STRIP PARITY BIT
002610  122716  000015                CMPB    #15,(SP)        ;CHECK IF CARRIAGE RETURN
002614  001015                        BNE     DONF+2          ;BRANCH IF NOT CARRIAGE RETURN
002616  000004                        TYPE
002620  003414                        M8
002622  022526                        CMP     (R5)+,(SP)+     ;ADJUST R5 AND THE STACK PTR
002624  005267  000016                INC     DONF            ;RETURN TO CALLER
002630  052767  000002  165476        BIS     #2,FKS
002636  052767  000101  165470        BIS     #101,FKS
002644  000002                        RTI
002646  000000        DONF:   0
002650  022716  000003                CMP     #3,(SP)
002654  001002                        BNE     .+6
002656  000137  000200                JMP     @#200
002662  022716  000077                CMP     #77,(SP)        ;DID OPERATOR TYPE RUBOUT?
002666  001012                        BNE     2$              ;NO-CONTINUE
002670  112744  000040                MOVB    #40,-(4)        ;YES-REPLACE LAST CHARACTER
002674  000241                        CLC                     ;WITH A SPACE AND
002676  006015                        ROR     (5)             ;RIGHT JUSTIFY CURRENT NUMBER
002700  000241                        CLC
002702  006015                        ROR     (5)
002704  000241                        CLC
002706  006015                        ROR     (5)
002710  005726                        TST     (6)+
002712  000415                        BR      REX

002714  011667  000214        2$:     MOV     (SP),D2BTYP     ;ECHO CHARACTER
002720  042767  177400  000206        BIC     #177400,D2BTYP  ;STRIP WORD
002726  000004                        TYPE                    ;ECHO CHARACTER
002730  003134                        D2BTYP
002732  042716  177770        3$:     BIC     #177770,(SP)    ;STRIP AWAY ALL BUT 3 LSB
002736  006315                        ASL     (5)             ;ROTATE
002740  006315                        ASL     (5)             ;PREVIOUS
002742  006315                        ASL     (5)             ;DATA
002744  052615                        BIS     (SP)+,(5)       ;AND INSERT CHARACTER
002746  052767  000002  165360  REX:   BIS     #2,FKS          ;GET NEXT CHARACTER
002754  052767  000101  165352        BIS     #101,FKS
```

```
002762  000002                          RTI

                                ;SCOPE ROUTINE, THIS ROUTINE IS ENTERED AT THE END OF EACH SUBTEST.
002764  032737  040000  177570  ,SCOPE: BIT     #40000,@#SWR    ;TEST SR FOR SCOPE
002772  001030                  BNE     A28             ;YES SCOPE
002774  000411                  BR      A38             ;NOP IF FOR XOR
002776  013746  000004          MOV     @#4,-(%6)
003002  012737  003064  000004  MOV     #XOR,@#4
003010  005737  177060          TST     @#177060
003014  012637  000004          MOV     (%6)+,@#4
003020  032737  004000  177570  A38:    BIT     #4000,@#SWR     ;TEST FOR ITERATION
003026  001006                  BNE     A18             ;INHIBIT ITERATION
003030  005267  176246          INC     ITCNT           ;INCREMENT ITERATION COUNT
003034  026767  176242  000020  CMP     ITCNT,ICOUNT    ;ITERATION COMPLETE
003042  003004                  BGT     A28             ;BRANCH IF ITERATIONS NOT COMPLETE
003044  005067  176232          A18:    CLR     ITCNT           ;CLEAR ITERATION COUNT
003050  011667  176230          MOV     (SP),LASTPC     ;GET ADDRESS OF NEXT TEST
003054  016716  176224          A28:    MOV     LASTPC,(SP)
003060  000002                  RTI                     ;EXIT
003062  000005          ICOUNT: 5
003064  012626          XOR:    MOV     (%6)+,(%6)+
003066  012637  000004          MOV     (%6)+,@#4
003072  000770                  BR      A28

                                ;ERROR ROUTINE, THIS ROUTINE IS ENTERED WHEN AN ERROR IS DETECTED.
003074  033727  177570  020000  ,HLT:   BIT     @#SWR,#20000    ;INHIBIT PRINTOUT?
003102  001401                  BEQ     .+4             ;BRANCH IF ERROR PRINT OUT
003104  000002                  RTI                     ;RETURN TO TEST
003106  000004                  TYPE
003110  003356                  ERRORM                  ;'PC=      '
003112  011667  000016          MOV     (6),D2BTYP      ;TYPE PROGRAM COUNTER
003116  004767  000014          JSR     7,D2A
003122  005737  177570          TST     @#SWR           ;HALT ON ERROR?
003126  100001                  BPL     .+4             ;
003130  000000                  HALT                    ;YES HALT
003132  000002                  RTI                     ;RETURN TO TEST
```

```
                                        ;THIS ROUTINE CONVERTS AN OCTAL NUMBER TO ASCII AND TYPES IT ON THE TTY.
003134  000000                          D2BTYP: 0
003136  010246                          O2A:    MOV     %2,-(6)         ;SAVE R2
003140  010146                                  MOV     %1,-(6)         ;SAVE R1
003142  010046                                  MOV     %0,-(6)         ;SAVE R0
003144  016700  177764                          MOV     D2BTYP,%0       ;GET DATA TO BE TYPED
003150  012701  000006                          MOV     #6,%1           ;GET COUNTER
003154  005002                                  CLR     %2              ;CLEAR WORKING REGISTER
003156  006100                                  ROL     %0              ;MOV FIRST BIT (MSB) INTO
003160  006102                                  ROL     %2              ;R2
003162  062702  000260                  1$:     ADD     #260,%2         ;FORM ASCII CODE+STOP CODE (31)
003166  042702  177700                          BIC     #177700,R2
003172  010267  177736                          MOV     R2,D2BTYP
003176  000004                                  TYPE
003200  003134                                  D2BTYP
003202  005002                                  CLR     %2              ;CLEAR WORKING REGISTER
003204  006100                                  ROL     %0              ;ROTATE THE
003206  006102                                  ROL     %2              ;NEXT
003210  006100                                  ROL     %0              ;OCTAL CHARACTER
003212  006102                                  ROL     %2              ;INTO
003214  006100                                  ROL     %0              ;REGISTER
003216  006102                                  ROL     %2              ;TWO
003220  005301                                  DEC     %1              ;DECREMENT COUNTER
003222  001357                                  BNE     1B              ;GO TO O2AA IF NOT 0
003224  012600                                  MOV     (6)+,%0         ;FINISHED.  RESTORE REGISTERS
003226  012601                                  MOV     (6)+,%1         ;
003230  012602                                  MOV     (6)+,%2         ;
003232  000207                                  RTS     7               ;AND EXIT

                                                ;
                                                ;THIS ROUTINE SENDS MESSAGES TO
                                                ;TUBE 0
003234  017667  000000  000056  VTOUT:  MOV     @(6),OUTT
003242  117724  000052          VTOUT1: MOVB    @OUTT,(4)+
003246  001420                          BEQ     VTOUTE
003250  121427  000012                  CMPB    (4),#12
003254  001004                          BNE     1$
003256  005267  000072                  INC     RETCNT
003262  100401                          BMI     1$
003264  000405                          BR      2$
003266  005267  000026          1$:     INC     OUTT
003272  020427  005022                  CMP     %4,#PBE
003276  002761                          BLT     VTOUT1
003300  004767  000016          2$:     JSR     %7,RESBF
003304  000167  177724                  JMP     VTOUT

003310  105744                  VTOUTE: TSTB    -(4)
003312  062716  000002                  ADD     #2,(6)
003316  000002                          RTI
003320  000000                  OUTT:   0
003322  012704  004100          RESBF:  MOV     #PBUFF,R4       ;THIS ROUTINE RESETS PRINTOUT
003326  112724  000031          1$:     MOVB    #31,(4)+        ;AREA TO ALL EOS
003332  020427  005023                  CMP     R4,#PBE+1
003336  001373                          BNE     1$
```

```
003340  012704  004100                  MOV     #PBUFF,R4
003344  012767  177762  000002          MOV     #-14.,,RETCNT
003352  000207                          RTS     PC

003354  177762                  RETCNT: -14.
```

```
                                        ;ASCII MESSAGES
003356  005015  041520  020075  ERRORM: .ASCIZ  <15><12>'PC= '
003364     000
003365     015  173012  051120  M6:     .ASCIZ  <15><12><366>'PRG#='<360>
003372  021507  170075     000
003377     015  051012  046517  M7:     .ASCIZ  <15><12>'ROM DATA'<15><12>
003404  042040  052101  006501
003412  000012
003414  005015     000          M8:     .ASCIZ  <15><12>
003417     015  051012  046517  M9:     .ASCIZ  <15><12>'ROM ADDRESS/IMAGE ADDRESS ROM DATA*IMAGE DATA'<15><12>
003424  040440  042104  042522
003432  051523  044457  040515
003440  042507  040440  042104
003446  042522  051523  051040
003454  046517  042040  052101
003462  025101  046511  043501
003470  020105  040504  040524
003476  005015     000
003501     057     000          M10:    .ASCIZ  '/'
003503     040     000          M11:    .ASCIZ  ' '
003505     052     000          M12:    .ASCIZ  '*'
003507     015  042412  042116  MBELL:  .ASCII  <15><12>/END PASS  /
003514  050040  051501  020123
003522     040
003523     040     040     040          .BYTE   40,40,40,40,40,40,40,40,40,40,40,40
003526     040     040     040
003531     040     040     040
003534     040     040     040
003537     040     040     040          .BYTE   40,40,40,40,40,40,40,40,40,15,12,0
003542     040     040     040
003545     040     040     040
003550     015     012     000
003553     015  005012  046764  MHED:   .ASCII  <15><12><12><364>/MD-11-DZBMK-B /
003560  026504  030461  042055
003566  041132  045515  041055
003574     040
003575     374  052126  030062          .ASCIZ  <374>/VT20 BOOTSTRAP LOADER TEST/<360>
003602  041040  047517  051524
003610  051124  050101  046040
003616  040517  042504  020122
003624  042524  052123  000360
003632  005015  047362  052117  WA:     .ASCII  <15><12><362>/NOT/<40><374>
003640  176040
003642  020101  047522  020115          .ASCIZ  /A ROM ADDRESS/<360><12><15>
003650  042101  051104  051505
003656  170123  006412     000
003663     015  051012  040505  MCYC:   .ASCIZ  <15><12>/READING DATA FROM ADDRESS: /
003670  044504  043516  042040
003676  052101  020101  051106
003704  046517  040440  042104
003712  042522  051523  020072
003720     000

        003776                          .=3776
```

```
003776  000000                          .=OBD
                                ;DATA CUT INTO THE M792-YK
004000  012701  160000  012702          012701,160000,012702,000006
004006  000006
004010  012703  173100  005012          012703,173100,005012,010742
004016  010742
004020  110706  014304  005714          110706,014304,005714,100775
004026  100775
004030  010712  012706  000024          010712,012706,000024,010441
004036  010441
004040  040601  010111  011102          040601,010111,011102,005214
004046  005214
004050  105714  100376  116412          105714,100376,116412,000002
004056  000002
004060  005211  120227  000375          005211,120227,000375,001366
004066  001366
004070  105222  000142  177560          105222,000142,177560,175610
004076  175610
004100  000000                  PBUFF:  0
        005022                          .=.+720
005022     031     031     031  PBE:    .BYTE   31,31,31,31
005025     031

        000001                          .END
```

```
A1S       003044          A2S       003054          A3S       003020          BPTVEC = 000014
DISPLA = 177570           DONE      002134          DONE1     002174          DONE      002646
D2BTYP    003134          EMTVEC = 000030           END       002114          EPC       001306
ERRFLG    001310          ERRORM    003356          ERROR1    001536          ERROR2    001604
ERRVEC = 000004           FIXED     001372          FKB    = 170336           FKS    = 170334
FKVEC  = 000320           FPEVEC = 000244           HLT    = 104400           ICNT   = 001300
ICOUNT    003062          IMAGE     001322          INCSR     001314          INDAT     001316
IOTVEC = 000020           ITCNT     001302          LASTPC    001304          MBELL     003507
MCYC      003663          MHED      003553          MMVEC  = 000250           M10       003501
M11       003503          M12       003505          M6        003365          M7        003377
M8        003414          M9        003417          OUTT      003320          O2A       003136
PARVEC = 000114           PBE       005022          PBUFF     004100          PC     =%000007
PCNT      002200          PFVEC  = 000024           PIRQ   = 177772           PIRVEC = 000240
PPER      002516          PRERR     002474          PRGNUM    001406          PRGTAB    001330
PRG0      001442          PRG0R     001446          PRG1      002202          PRG1A     002234
PRG1B     002244          PRG1C     002260          PRG1D     002316          PRG2      002322
PRG2A     002412          PRG2B     002442          PRG2C     002510          PRG20     002366
PRMTRS    001336          PSW    = 177776           RECD      002526          RECDI     002572
RECD1     002562          RESBF     003322          RESTAR    001424          RESVEC = 000010
RETCNT    003354          REX       002746          ROMADD    001326          R0     =%000000
R1     =%000001           R10    =%000000           R11    =%000001           R12    =%000002
R13    =%000003           R14    =%000004           R15    =%000005           R2     =%000002
R3     =%000003           R4     =%000004           R5     =%000005           SCOPE  = 104000
SCPBLK    001300          SLR    = 177774           SP     =%000006           START1    000200
START3    000210          STKPTR = 000500           SWR    = 177570           TBITVE = 000014
TEMP      001324          TICKS     001312          TKB    = 177562           TKS    = 177560
TKVEC  = 000060           TPB    = 177566           TPS    = 177564           TPVEC  = 000064
TRAPVE = 000034           TRTVEC = 000014           TYPE   = 000004           T1        001462
T1A       001504          T1B       001544          T2        001546          T2A       001564
T2B       001610          T2C       001622          T2D       001632          T2E       001642
T3        001652          T3A       001674          T3AA      001666          T3B       001702
T3C       001712          T3D       001720          T3E       001730          T3F       001740
T4        001760          T4B       002000          T4C       002004          T4D       002014
T4E       002112          UBREAK = 177770           VTC    = 170304           VTOUT     003234
VTOUTE    003310          VTOUT1    003242          VTS    = 170302           WA        003632
WORDS     001320          XOR       003064          .HLT      003074          .SCOPE    002764
.      = 005026
```

ERRORS DETECTED: 0