```
        .REM    _

                         IDENTIFICATION
                         --------------

    PRODUCT CODE:        AC-F066B-MC

    PRODUCT NAME:        CXBMFB0 BM873-YH MODULE

    PRODUCT DATE:        SEPTEMBER 1978

    MAINTAINER:          DEC/X11 SUPPORT GROUP

    THE INFORMATION  IN  THIS  DOCUMENT  IS  SUBJECT  TO  CHANGE
    WITHOUT  NOTICE  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT
    BY  DIGITAL  EQUIPMENT   CORPORATION.   DIGITAL   EQUIPMENT
    CORPORATION  ASSUMES  NO  RESPONSIBILITY FOR ANY ERRORS THAT
    MAY APPEAR IN THIS MANUAL.

    THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO  THE
    PURCHASER  UNDER  A  LICENSE  FOR  USE  ON A SINGLE COMPUTER
    SYSTEM  AND  CAN  BE  COPIED  (WITH  INCLUSION  OF  DIGITALS
    COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY
    OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

    DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY  FOR
    THE  USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS
    NOT SUPPLIED BY DIGITAL.

  COPYRIGHT (C)  1976,1978 DIGITAL EQUIPMENT CORPORATION
```

```
        1.0     ABSTRACT
                --------

                BMF IS A BACKGROUND MODULE THAT EXERCISES A SINGLE BM873-YH
                BOOTSTRAP ROM OPTION. IT COMPARES THE CONTENTS OF EACH OF THE
                256(10) LOCATIONS STORED IN THE ROM WITH THE CONTENTS OF A
                256(10) WORD CORE MEMORY BUFFER TO VERIFY THAT EACH LOCATION
                IN THE ROM CAN BE UNIQUELY ADDRESSED AND CONTAINS THE CORRECT
                DATA. ALL ERRORS ARE REPORTED VIA THE CONSOLE DEVICE.
        2.0     REQUIREMENTS
                ------------

                HARDWARE:       A PDP11 COMPUTER WITH A BM873-YH OPTION

                STORAGE:: BMF REQUIRES:
                                1. DECIMAL WORDS: 368
                                2. OCTAL WORDS: 0560
                                3. OCTAL BYTES: 1340

        3.0     PASS DEFINITION
                ---------------

                THE INITIAL PASS CONSISTS OF EXECUTING THE BASIC TEST SEQUENCE
                ONE TIME BEFORE REPORTING END OF PASS.
                SUBSEQUENT PASSES OF THE BMFB MODULE CONSISTS OF 100(8) ITERATIONS
                OF THE BASIC TEST SEQUENCE DESCRIBED IN PARA. 7 BELOW.
        4.0     EXECUTION TIME
                --------------

                PASS TIME VARIES DEPENDENT UPON CPU TYPE AND THE CONFIGURATION
                BEING EXERCISED. WHEN RUNNING ALONE ON A PDP11/40 THE FIRST PASS
                SHOULD TAKE LESS THAN 10 SECONDS AND SUBSEQUENT PASSES LESS
                THAN ONE MINUTE.
        5.0     CONFIGURATION OPTIONS
                ---------------------

                DEFAULT PARAMETERS:

                DVA:     173000

                REQUIRED PARAMETERS:

                NONE
        6.0     DEVICE OPTION SETUP
                -------------------

                NONE REQUIRED
        7.0     MODULE OPERATION
                ----------------
```

```
                 TEST SEQUENCE:

                 1. R1 IS SET UP TO POINT TO THE FIRST WORD IN THE ROM
                 2. R2 IS SET UP TO POINT TO THE CORRESPONDING WORD IN THE
                    CORE MEMORY BUFFER.

                    THE ADDRESS IN R1 IS CHECKED FOR EQUALITY TO EITHER 173024
                    OR 173224 AND IF FOUND EQUAL GOES TO STEP (5) - IF NOT
                    IT PROCEEDS WITH STEP (3). THESE TWO ADDRESSES ARE NOT
                    CHECKED BECAUSE THEIR CONTENTS AS READ ON THE BUS WILL
                    VARY DEPENDENT UPON WHICH PARTICULAR "LOAD" BUTTON HAD
                    BEEN INITIALLY DEPRESSED TO LOAD THE PROGRAM.
                 3. R1 AND R2 ARE USED TO COMPARE A ROM WORD WITH ITS CORE
                    IMAGE COUNTERPART. IF THE WORDS DON'T COMPARE A SUB-ROUT-
                    INE IS CALLED TO SET UP THE ERROR INFORMATION AND REPORT
                    IT VIA  A "DATER" CALL TO THE MONITOR.
                 4. STEP (3) IS REPEATED.
                 5. R1 AND R2 ARE UPDATED TO POINT TO THE NEXT WORD AND A
                    TEST MADE ON R2 TO SEE IF 256(10) WORDS HAVE BEEN CHECKED.
                    IF YES, GO TO STEP (6) IF NOT REPEAT (3) THRU (5).
                 6. A PASS COUNTER IS DECREMENTED AND TESTED TO SEE IF 100(8)
                    ITERATIONS OF STEPS (1) THRU (5) HAVE OCCURRED - IF YES
                    GO TO STEP (7) IF NOT REPEAT (1) THRU (5).
                 7. REPORT END OF PASS AND REPEAT (1) THRU (6).
        8.0      OPERATOR OPTIONS
                 ----------------

                         (NONE)

        9.0      NON-STANDARD PRINTOUTS
                 ----------------------

                         (NONE)
        -
```

```
        000000'                 BKMOD   <BMFB >,173000,,,,,100,155
        000000'                 MODULE  40020,BMFB ,173000,,,,,100,155
                                .TITLE  BMFB DEC/X11 SYSTEM EXERCISER MODULE
                ;               DDXCOM  VERSION  6         23-MAY-78
                                                           .LIST   BIN
                                                 ;*****************************************************************************
        000000'                                  BEGIN:
        000000' 046502  041106       040         MODNAM: .ASCII  /BMFB / ;MODULE NAME.
        000005'    000                           XFLAG:  .BYTE   OPEN                   ;USED TO KEEP TRACK OF WBUFF USAGE
        000006' 173000                           ADDR:   173000+0                       ;1ST DEVICE ADDR.
        000010' 000000                           VECTOR: +0                             ;1ST DEVICE VECTOR.
        000012'    000                           BR1:    .BYTE   PRTY+0                 ;1ST BR LEVEL.
        000013'    000                           BR2:    .BYTE   PRTY+0                 ;2ND BR LEVEL.
        000014' 000001                           DVID1:  +1                             ;DEVICE INDICATOR 1.
        000016' 000000                           SR1:    OPEN                           ;SWITCH REGISTER 1
        000020' 000000                           SR2:    OPEN                           ;SWITCH REGISTER 2
        000022' 000000                           SR3:    OPEN                           ;SWITCH REGISTER 3
        000024' 000000                           SR4:    OPEN                           ;SWITCH REGISTER 4
                                                 ;****************************************************************************
        000026' 040020                           STAT:   40020                          ;STATUS WORD.
        000030' 000224'                          INIT:   START                          ;MODULE START ADDR.
        000032' 000224'                          SPOINT: MODSP                          ;MODULE STACK POINTER.
        000034' 000000                           PASCNT: 0                              ;PASS COUNTER.
        000036' 000100                           ICONT:  100                            ;# OF ITERATIONS PER PASS=100
        000040' 000000                           ICOUNT: 0                              ;LOC TO COUNT ITERATIONS
        000042' 000000                           SOFCNT: 0                              ;LOC TO SAVE TOTAL SOFT ERRORS
        000044' 000000                           HRDCNT: 0                              ;LOC TO SAVF TOTAL HARD ERRORS
        000046' 000000                           SOFPAS: 0                              ;LOC TO SAVE SOFT ERRORS PER PASS
        000050' 000000                           HRDPAS: 0                              ;LOC TO SAVE HARD ERRORS PER PASS
        000052' 000000                           SYSCNT: 0                              ;# OF SYS ERRORS ACCUMULATED
        000054' 000000                           RANNUM: 0                              ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
        000056'                                  CONFIG:                                ;RESERVED FOR MONITOR USE
        000056' 000000                           RES1:   0                              ;RESERVED FOR MONITOR USE
        000060' 000000                           RES2:   0                              ;RESERVED FOR MONITOR USE
        000062' 000000                           SVR0:   OPEN                           ;LOC TO SAVE R0.
        000064' 000000                           SVR1:   OPEN                           ;LOC TO SAVE R1.
        000066' 000000                           SVR2:   OPEN                           ;LOC TO SAVE R2.
        000070' 000000                           SVR3:   OPEN                           ;LOC TO SAVE R3.
        000072' 000000                           SVR4:   OPEN                           ;LOC TO SAVE R4.
        000074' 000000                           SVR5:   OPEN                           ;LOC TO SAVE R5.
        000076' 000000                           SVR6:   OPEN                           ;LOC TO SAVE R6.
        000100' 000000                           CSRA:   OPEN                           ;ADDR OF CURRENT CSR.
        000102'                                  SBADR:                                 ;ADDR OF GOOD DATA, OR
        000102' 000000                           ACSR:   OPEN                           ;CONTENTS OF CSR.
        000104'                                  WASADR:                                ;ADDR OF BAD DATA, OR
        000104' 000000                           ASTAT:  OPEN                           ;STATUS REG CONTENTS.
        000106'                                  ERRTYP:                                ;TYPE OF ERROR
        000106' 000000                           ASB:    OPEN                           ;EXPECTED DATA.
        000110' 000000                           AWAS:   OPEN                           ;ACTUAL DATA.
        000112' 000230'                          RSTRT:  RESTRT                         ;RESTART ADDRESS AFTER END OF PASS
        000114' 000000                           WDTO:   OPEN                           ;WORDS TO MEMORY PER ITERATION
        000116' 000000                           WDFR:   OPEN                           ;WORDS FROM MEMORY PER ITERATION
        000120' 000000                           INTR:   OPEN                           ;# OF INTERRUPTS PER ITERATION
        000122' 000155                           IDNUM:  155                            ;MODULE IDENTIFICATION NUMBER=155
                000040                                   .REPT   SPSIZ                  ;MODULE STACK STARTS HERE.
                                                         .NLIST
```

```
                                                         .WORD   0
                                                         .LIST
                                                         .ENDR
        000224'                                  MODSP:
                                                 ;****************************************************************************
```

```

000224' 016705  177556          START:  MOV     ADDR,R5         ;GET FIRST ROM ADDRFSS INTO R5

000230'                         RESTRT:
000230' 010501                  AGAIN:  MOV     R5,R1           ;R1 POINTS TO ROM WORD
000232' 012702  000336'                 MOV     #BMTAB,R2       ;R2 POINTS TO ROM IMAGE IN CORE
000236' 022701  173024          1$:     CMP     #173024,R1      ;ROM ADDRESS = 173024 ??
000242' 001413                          BEQ     3$              ;BR IF YES
000244' 022701  173224                  CMP     #173224,R1      ;ROM ADDRESS = 173224 ??
000250' 001410                          BEQ     3$              ;BR IF YES
000252' 021112                          CMP     (R1),(R2)       ;CHECK ONE LOCATION
000254' 001402                          BEQ     2$              ;BR IF [ROM] = [CORE]
000256' 004767  000026                  JSR     PC,BMERR        ;GO SETUP AND REPORT ERROR
000262' 021112                  2$:     CMP     (R1),(R2)       ;CHECK IT AGAIN
000264' 001402                          BEQ     3$              ;BR IF [ROM] = [CORE]
000266' 004767  000016                  JSR     PC,BMERR        ;GO SETUP AND REPORT ERROR
000272' 022122                  3$:     CMP     (R1)+,(R2)+     ;ADD +2 TO BOTH POINTERS
000274' 022702  001336'                 CMP     #TABEND,R2      ;DONE LAST WORD ??
000300' 001356                          BNE     1$              ;BR IF NOT
000302'                         4$:
000302' 104413  000000'                 ENDITS,BEGIN            ;SIGNAL END OF ITERATION.
                                                                ;MONITOR SHALL TEST END OF PASS
000306' 000750                          BR      AGAIN



                                ;THIS ROUTINE SETS UP AND REPORTS ALL DATA COMPARE ERRORS

000310' 010267  177566          BMERR:  MOV     R2,SBADR        ;SAVE THE ADDR. OF GOOD DATA
000314' 010167  177564                  MOV     R1,WASADR       ;SAVE ADDR. OF THE BAD DATA
000320' 011167  177564                  MOV     (R1),AWAS       ;GET WAS DATA
000324' 011267  177556                  MOV     (R2),ASB        ;GET THE S/B DATA
                                ;****************************************************************
000330' 104404  000000'                 DATERS,BEGIN            ;DATA ERROR!!!
                                ;****************************************************************
000334' 000207                          RTS     PC              ;CONTINUE CHECKING


                                ;256(10) WORD TABLE THAT STORES A CORE IMAGE OF THE CONTENTS OF THE ROM




000336'                         BMTAB:          ;173000         BUTON1:
000336' 010037                          010037  ;173000 010037          MOV     R0,ROTOR7+0     ;SAVE R0 IN LOCATION 40
000340' 000040                          000040  ;173002 000040
000342' 013700                          013700  ;173004 013700          MOV     SWR,R0          ;GET SWITCH REGISTER
000344' 177570                          177570  ;173006 177570
000346' 032700                          032700  ;173010 032700          BIT     #BIT0,R0        ;IS LOW-ORDER BIT SET?
000350' 000001                          000001  ;173012 000001
000352' 001007                          001007  ;173014 001007          BNE     BUTONX          ;YES-- LOOK AT CONTENTS
000354' 000513                          000513  ;173016 000513          BR      RFGSAV          ;NO-- SAVE R1-R7 IN 42-5
                                                ;173020         BUTON3:
000356' 005000                          005000  ;173020 005000          CLR     R0              ;SAY LOAD FROM FLOPPY, U
000360' 000404                          000404  ;173022 000404          BR      BUTONX          ;GO TO COMMON CODE FOR 3
                                                ;173024                 FILLTO  24
000362' 173000                          173000  ;173024 173000          .WORD   ROMORG,PR7
```

```
000364' 000340                          000340  ;173026 000340
                                                ;173030         BUTON2:
000366' 012700                          012700  ;173030 012700          MOV     #BIT7,R0        ;BIT 7 MEANS LOAD FROM R
000370' 000200                          000200  ;173032 000200
                                                ;173034         BUTONX:
000372' 010005                          010005  ;173034 010005          MOV     R0,R5           ;SAVE PARAMETER FOR BOOT
000374' 106300                          106300  ;173036 106300          ASLB    R0              ;LEFT-ALIGN SPEED FIELD
000376' 122700                          122700  ;173040 122700          CMPB    #3*BIT4,R0      ;IS SPEED 0, 1, OR 2?
000400' 000060                          000060  ;173042 000060
000402' 101001                          101001  ;173044 101001          BHI     10$             ;YES-- UNIT IS UNIT TO U
000404' 005000                          005000  ;173046 005000          CLR     R0              ;NO-- USE UNIT #0
                                                ;173050         10$:
000406' 000300                          000300  ;173050 000300          SWAB    R0              ;GET UNIT # IN LOW BYTE
000410' 042700                          042700  ;173052 042700          BIC     #^C7,R0         ;TRIM TO 3 BITS 2, 1, 0
000412' 177770                          177770  ;173054 177770
000414' 105705                          105705  ;173056 105705          TSTB    R5              ;WHERE SHOULD WE BOOT FP
000416' 100553                          100553  ;173060 100553          BMI     RPBOOT          ;BIT 7 = 1 -- BOOT FROM
000420' 012737                          012737  ;173062 012737          MOV     #TCBOOT,4       ;SET TIMEOUT TRAP TO TRY
000422' 173304                          173304  ;173064 173304
000424' 000004                          000004  ;173066 000004
000426' 005037                          005037  ;173070 005037          CLR     6               ; . .
000430' 000006                          000006  ;173072 000006
                                                ;173074         RXBOOT:
000432' 012706                          012706  ;173074 012706          MOV     #RETRY,SP       ;SET RETRY COUNT
000434' 000014                          000014  ;173076 000014
000436' 012701                          012701  ;173100 012701          MOV     #RXEPA+RXCS,R1  ;ADDRESS CONTROL STATUS
000440' 177170                          177170  ;173102 177170
000442' 010003                          010003  ;173104 010003          MOV     R0,R3           ;COPY UNIT #
                                                ;173106         RXRTRY:
000444' 005705                          005705  ;173106 005705          TST     R5              ;INDEFINITE RETRY?
000446' 100402                          100402  ;173110 100402          BMI     RXRSET          ;YES-- TRY FAITHFULLY
000450' 005306                          005306  ;173112 005306          DEC     SP              ;NO-- DECREMENT RETRY CO
000452' 002445                          002445  ;173114 002445          BLT     RXEHLT          ;GIVE UP IF RUN OUT
                                                ;173116         RXRSET:
000454' 000005                          000005  ;173116 000005          RESET                   ;CLEAR THE WORLD
                                                ;173120         20$:
000456' 032711                          032711  ;173120 032711          BIT     #RXDONE,(R1)    ;WAIT UNTIL READY FOR FU
000460' 000040                          000040  ;173122 000040
000462' 001775                          001775  ;173124 001775          BEQ     20$             ;NOT YET-- WAIT
                                                ;173126         RXPERF:
000464' 010300                          010300  ;173126 010300          MOV     R3,R0           ;GET UNIT #
000466' 001402                          001402  ;173130 001402          BEQ     5$              ;ZERO-- USE ZERO
000470' 012700                          012700  ;173132 012700          MOV     #RXUNIT,R0      ;NON-ZERO-- ASSUME UNIT
000472' 000020                          000020  ;173134 000020
                                                ;173136         5$:
000474' 052700                          052700  ;173136 052700          BIS     #PXREAD+RXGO,R0 ;SET READ FUNCTION
000476' 000007                          000007  ;173140 000007
000500' 010102                          010102  ;173142 010102          MOV     R1,R2           ;COPY ADDRESS OF RXCS
000502' 010022                          010022  ;173144 010022          MOV     R0,(R2)+        ;START READ FUNCTION, R2
                                                ;173146         10$:
000504' 105711                          105711  ;173146 105711          TSTB    (R1)            ;READY?
000506' 100376                          100376  ;173150 100376          BPL     10$             ;NO-- WAIT
000510' 012712                          012712  ;173152 012712          MOV     #1,(R2)         ;SET SECTOR #
000512' 000001                          000001  ;173154 000001
                                                ;173156         20$:
000514' 105711                          105711  ;173156 105711          TSTB    (R1)            ;READY FOR TRACK?
```

```
   315 000516' 100376       100376  ;173160 100376             BPL     20$               ;NO-- WAIT
   316 000520' 012712       012712  ;173162 012712             MOV     #1,(R2)           ;SET TRACK #
   317 000522' 000001       000001  ;173164 000001
   318                              ;173166          30$:
   319 000524' 032711       032711  ;173166 032711             BIT     #RXERR!RXDONE,(R1) ;DONE OR ERROR?
   320 000526' 100040       100040  ;173170 100040
   321 000530' 001775       001775  ;173172 001775             BEQ     30$               ;NO-- WAIT
   322 000532' 100744       100744  ;173174 100744             BMI     RXRTRY            ;YES-- ERROR IN FUNCTION
   323                              ;173176          RXEMSL:
   324 000534' 012711       012711  ;173176 012711             MOV     #RXEMPT+RXGO,(R1) ;START EMPTY
   325 000536' 000003       000003  ;173170 000003
   326 000540' 005004       005004  ;173202 005004             CLR     R4                ;ALWAYS START TRANSFER A
   327                              ;173204          10$:
   328 000542' 132711       132711  ;173204 132711             BITB    #RXTREQ!RXDONE,(R1) ;READY FOR WORD, OR
   329 000544' 000240       000240  ;173206 000240
   330 000546' 001775       001775  ;173210 001775             BEQ     10$               ;NOT READY-- WAIT SOME M
   331 000550' 100153       100153  ;173212 100153             BPL     CLRPC             ;DONE-- GO TO LOCATION 0
   332 000552' 111224       111224  ;173214 111224             MOVB    (R2),(R4)+        ;NOT DONE-- GET A BYTE F
   333 000554' 000772       000772  ;173216 000772             BR      10$               ;WAIT FOR NEXT BYTE
   334                              ;173220                    FILLTO  224
   335 000556' 000000       000000  ;173220    000             .BYTE   0
   336                              ;173221    000             .BYTE   0
   337 000560' 000000       000000  ;173222    000             .BYTE   0
   338                              ;173223    000             .BYTE   0
   339 000562' 173000       173000  ;173224 173000             .WORD   ROMORG,PR7
   340 000564' 000340       000340  ;173226 000340
   341                              ;173230          RXEHLT:
   342 000566' 012711       012711  ;173230 012711             MOV     #RXERR+RXGO,(R1) ;DO A READ ERROR REGIS
   343 000570' 000017       000017  ;173232 000017
   344                              ;173234          10$:
   345 000572' 032711       032711  ;173234 032711             BIT     #RXDONE,(R1)      ;WAIT UNTIL ERROR ASSEMB
   346 000574' 000040       000040  ;173236 000040
   347 000576' 001775       001775  ;173240 001775             BEQ     10$               ; . .
   348 000600' 011200       011200  ;173242 011200             MOV     (R2),R0           ;GET ERROR REGISTER
   349 000602' 000541       000541  ;173244 000541             BR      HALTED            ;HALT AND DISPLAY ERRORS
   350                              ;173246          REGSAV:
   351 000604' 010037       010037  ;173246 010037             MOV     R0,ROTOR7+16      ;SAVE R0 AS PC IN 56
   352 000606' 000056       000056  ;173250 000056
   353 000610' 012700       012700  ;173252 012700             MOV     #ROTOR7+16,R0     ;R0 NOW POINTS TO 56
   354 000612' 000056       000056  ;173254 000056
   355 000614' 010640       010640  ;173256 010640             MOV     SP,-(R0)          ;SAVE SP IN 54
   356 000616' 010540       010540  ;173260 010540             MOV     R5,-(R0)          ;SAVE R5 IN 52
   357 000620' 010440       010440  ;173262 010440             MOV     R4,-(R0)          ;SAVE R4 IN 50
   358 000622' 010340       010340  ;173264 010340             MOV     R3,-(R0)          ;SAVE R3 IN 46
   359 000624' 010240       010240  ;173266 010240             MOV     R2,-(R0)          ;SAVE R2 IN 44
   360 000626' 010140       010140  ;173270 010140             MOV     R1,-(R0)          ;SAVE R1 IN 42
   361 000630' 014000       014000  ;173272 014000             MOV     -(R0),R0          ;RESTORE R0 FROM 40
   362 000632' 000177       000177  ;173274 000177             JMP     @ROTOR7+16        ;GO TO SAVED PC
   363 000634' 004556       004556  ;173276 004556
   364                              ;173300          TCBOTO:
   365 000636' 005000       005000  ;173300 005000             CLR     R0                ;HERE TO START WITH A FL
   366 000640' 005005       005005  ;173302 005005             CLR     R5                ; . .
   367                              ;173304          TCBOOT:
   368 000642' 012706       012706  ;173304 012706             MOV     #RETRY,SP         ;INIT RETRY COUNTER
   369 000644' 000014       000014  ;173306 000014
   370 000646' 012701       012701  ;173310 012701             MOV     #TCEPA+TCCM,R1    ;POINT TO COMMAND REGIST
```

```
   371 000650' 177342       177342  ;173312 177342
   372                              ;173314          TCRTRY:
   373 000652' 005705       005705  ;173314 005705             TST     R5                ;INDEFINITE RETRY?
   374 000654' 100402       100402  ;173316 100402             BMI     10$               ;YES-- TRY HARDER
   375 000656' 005306       005306  ;173320 005306             DEC     SP                ;NO-- DECREMENT COUNT
   376 000660' 002427       002427  ;173322 002427             BLT     TCEHLT            ;TOO MANY-- GIVE UP
   377                              ;173324          10$:
   378 000662' 000005       000005  ;173324 000005             RESET                     ;CLEAR TC11
   379 000664' 010003       010003  ;173326 010003             MOV     R0,R3             ;GET UNIT NUMBER
   380 000666' 000303       000303  ;173330 000303             SWAB    R3                ;TO BITS 10-8
   381 000670' 010304       010304  ;173332 010304             MOV     R3,R4             ;COPY FOR READ FUNCTION
   382 000672' 052703       052703  ;173334 052703             BIS     #TCREV+TCRNUM+TCGO,R3 ;START TAPE TOWARD
   383 000674' 004003       004003  ;173336 004003
   384 000676' 010311       010311  ;173340 010311             MOV     R3,(R1)           ; . .
   385                              ;173342          20$:
   386 000700' 005711       005711  ;173342 005711             TST     (R1)              ;ERROR?
   387 000702' 100376       100376  ;173344 100376             BPL     20$               ;NO-- WAIT FOR END-ZONE
   388 000704' 005761       005761  ;173346 005761             TST     TCST-TCCM(R1)     ;END-ZONE UP YET?
   389 000706' 177776       177776  ;173350 177776
   390 000710' 100360       100360  ;173352 100360             BPL     TCRTRY            ;NO-- MUST BE OTHER ERRO
   391 000712' 012761       012761  ;173354 012761             MOV     #-256.,TCWC-TCCM(R1) ;SET WORD COUNT
   392 000714' 177400       177400  ;173356 177400
   393 000716' 000002       000002  ;173360 000002
   394 000720' 052704       052704  ;173362 052704             BIS     #TCREAD+TCGO,R4   ;START READ, FORWARD
   395 000722' 000005       000005  ;173364 000005
   396 000724' 010411       010411  ;173366 010411             MOV     R4,(R1)           ; . .
   397                              ;173370          30$:
   398 000726' 105711       105711  ;173370 105711             TSTB    (R1)              ;TRANSFER DONE?
   399 000730' 100376       100376  ;173372 100376             BPL     30$               ;NO-- WAIT SOME MORE
   400 000732' 005711       005711  ;173374 005711             TST     (R1)              ;YES-- ERROR?
   401 000734' 100746       100746  ;173376 100746             BMI     TCRTRY            ;YES-- RETRY
   402 000736' 000460       000460  ;173400 000460             BR      CLRPC             ;NO-- DONE-- GOTO LOCATI
   403                              ;173402          TCEHLT:
   404 000740' 016100       016100  ;173402 016100             MOV     TCST-TCCM(R1),R0  ;GET STATUS REGISTER
   405 000742' 177776       177776  ;173404 177776
   406 000744' 000460       000460  ;173406 000460             BR      HALTED            ;AND STOP
   407                              ;173410          RPBOOT:
   408 000746' 012706       012706  ;173410 012706             MOV     #RETRY,SP         ;RETRY RETRY TIMES
   409 000750' 000014       000014  ;173412 000014
   410 000752' 012701       012701  ;173414 012701             MOV     #RPEPA+RPCS1,R1   ;ADDRESS RPCS1 IN R1
   411 000754' 176700       176700  ;173416 176700
   412 000756' 012702       012702  ;173420 012702             MOV     #RPECCI,R2        ;SET ECC INHIBIT, 20 SEC
   413 000760' 004000       004000  ;173422 004000
   414                              ;173424          RPRTRY:
   415 000762' 005705       005705  ;173424 005705             TST     R5                ;INFINITE RETRY?
   416 000764' 100402       100402  ;173426 100402             BMI     10$               ;YES-- TRY AGAIN
   417 000766' 005306       005306  ;173430 005306             DEC     SP                ;RETRY COUNT EXHAUSTED?
   418 000770' 002444       002444  ;173432 002444             BLT     RPEHLT            ;YES-- GIVE UP
   419                              ;173434          10$:
   420 000772' 000005       000005  ;173434 000005             RESET                     ;ZAP!!
   421 000774' 110061       110061  ;173436 110061             MOVB    R0,RPCS2(R1)      ;SELECT PROPER UNIT #
   422 000776' 000010       000010  ;173440 000010
   423 001000' 032711       032711  ;173442 032711             BIT     #RPDVA,(R1)       ;IS DRIVE AVAILABLE TO U
   424 001002' 004000       004000  ;173444 004000
   425 001004' 001766       001766  ;173446 001766             BEQ     RPRTRY            ;NO-- TRY AGAIN
   426 001006' 012711       012711  ;173450 012711             MOV     #RPPRST+RPGO,(R1) ;DO 'READ-IN PRESET' F
```

```
001010' 000021    000021 ;173452 000021
001012' 005061    005061 ;173454 005061           CLR     RPDC(R1)        ;SET CYLINDER 0
001014' 000034    000034 ;173456 000034
001016' 005061    005061 ;173460 005061           CLR     RPDA(R1)        ; TRACK 0, SECTOR 0
001020' 000006    000006 ;173462 000006
001022' 050261    050261 ;173464 050261           BIS     R2,RPOF(R1)     ;SET INHIBIT ECC, 22-SEC
001024' 000032    000032 ;173466 000032
001026' 012761    012761 ;173470 012761           MOV     #-256.,RPWC(R1) ;SET UP WORD COUNT TO PR
001030' 177400    177400 ;173472 177400
001032' 000002    000002 ;173474 000002
001034' 012711    012711 ;173476 012711           MOV     #RPREAD+RPGO,(R1) ;START READ FUNCTION
001036' 000071    000071 ;173470 000071
                         ;173502         20$:
001040' 105711    105711 ;173502 105711           TSTB    (R1)            ;READY?
001042' 100376    100376 ;173504 100376           BPL     20$             ;NO-- WAIT UNTIL IT IS
001044' 032761    032761 ;173506 032761           BIT     #RPFER,RPER1(R1) ;FORMAT ERROR?
001046' 000020    000020 ;173510 000020
001050' 000014    000014 ;173512 000014
001052' 001403    001403 ;173514 001403           BEQ     30$             ;NO-- TRY AGAIN
001054' 052702    052702 ;173516 052702           BIS     #RPFM22,R2      ;YES-- TRY FOR 22 SECTOR
001056' 010000    010000 ;173520 010000
001060' 000740    000740 ;173522 000740           BR      RPRTRY          ;TRY AGAIN
                         ;173524         30$:
001062' 032711    032711 ;173524 032711           BIT     #RPTRE!RPMCPE,(R1) ;TRANSFER OR MBC PARI
001064' 060000    060000 ;173526 060000
001066' 001335    001335 ;173530 001335           BNE     RPRTRY          ;YES-- ERROR-- TRY AGAIN
001070' 032761    032761 ;173532 032761           BIT     #RPATA!RPERR,RPDS(R1) ;ATTN OR OTHER ERR
001072' 140000    140000 ;173534 140000
001074' 000012    000012 ;173536 000012
001076' 001331    001331 ;173540 001331           BNE     RPRTRY          ;YES-- ERROR-- TRY AGAIN
                         ;173542         CLRPC:
001100' 005007    005007 ;173542 005007           CLR     PC              ;JMP 0
                         ;173544         RPEHLT:
001102' 016100    016100 ;173544 016100           MOV     RPDS(R1),R0     ;DISPLAY DRIVE STATUS
001104' 000012    000012 ;173546 000012
                         ;173550         HALTED:
001106' 000000    000000 ;173550 000000           HALT                    ;DIE
001110' 000776    000776 ;173552 000776           BR      HALTED          ;STAY DEAD
                         ;173554         BUTON4:
001112' 010037    010037 ;173554 010037           MOV     R0,ROTOR7+0     ;SAVE R0 IN 40
001114' 000040    000040 ;173556 000040
001116' 012700    012700 ;173560 012700           MOV     #10$,R0         ;SET RETURN ADDRESS IN R
001120' 173566    173566 ;173562 173566
001122' 000630    000630 ;173564 000630           BR      REGSAV          ;SAVE R1-R7
                         ;173566         10$:
001124' 005005    005005 ;173566 005005           CLR     R5              ;ADDRESS LOCATION ZERO
001126' 012500    012500 ;173570 012500           MOV     (R5)+,R0        ;SAVE 0 IN R0
001130' 012501    012501 ;173572 012501           MOV     (R5)+,R1        ;SAVE 2 IN R1
001132' 011502    011502 ;173574 011502           MOV     (R5),R2         ;SAVE 4 IN R2
001134' 012725    012725 ;173576 012725           MOV     #21$,(R5)+      ;SET NXM TRAP ADDRESS IN
001136' 173612    173612 ;173570 173612
001140' 011503    011503 ;173602 011503           MOV     (R5),R3         ;SAVE 6 IN R3
001142' 005015    005015 ;173604 005015           CLR     (R5)            ;SET PS FOR TRAP
                         ;173606         20$:
001144' 012704    012704 ;173606 012704           MOV     #DLYCNT-DTESIZ,R4 ;POINT TO DTE # -1'S D
001146' 174340    174340 ;173610 174340
```

```
                         ;173612         21$:
001150' 012706    012706 ;173612 012706           MOV     #4,SP           ;SET SP TO 4, STACK IS L
001152' 000004    000004 ;173614 000004
                         ;173616         22$:
001154' 062704    062704 ;173616 062704           ADD     #DTESIZ,R4      ;BUMP TO NEXT DTE'S EXTE
001156' 000040    000040 ;173620 000040
001160' 105704    105704 ;173622 105704           TSTB    R4              ;IS THIS THE END OF THE
001162' 100770    100770 ;173624 100770           BMI     20$             ;YES-- START ALL OVER, U
001164' 032764    032764 ;173626 032764           BIT     #TO11DB,STAT-DLYCNT(R4) ;DOORBELL RINGIN
001166' 004000    004000 ;173630 004000
001170' 000034    000034 ;173632 000034
001172' 001770    001770 ;173634 001770           BEQ     22$             ;NO-- TRY NEXT DTE
001174' 026417    026417 ;173636 026417           CMP     TO10BC-DLYCNT(R4),(PC) ;DOES THIS ONE HA
001176' 000014    000014 ;173640 000014
001200' 001365    001365 ;173642 001365           BNE     22$             ;NO-- TRY ANOTHER DTE
001202' 010315    010315 ;173644 010315           MOV     R3,(R5)         ;RESTORE LOCATION 6
001204' 010245    010245 ;173646 010245           MOV     R2,-(R5)        ; 4
001206' 010145    010145 ;173650 010145           MOV     R1,-(R5)        ; 2
001210' 010045    010045 ;173652 010045           MOV     R0,-(R5)        ; 0
001212' 012700    012700 ;173654 012700           MOV     #DTESAV,R0      ;POINT TO SAVE AREA
001214' 000130    000130 ;173656 000130
                         ;173660         29$:
001216' 012420    012420 ;173660 012420           MOV     (R4)+,(R0)+     ;SAVE A REGISTER
001220' 022700    022700 ;173662 022700           CMP     #TO11DT-DLYCNT+DTESAV,R0 ;FINISHED?
001222' 000156    000156 ;173664 000156
001224' 103374    103374 ;173666 103374           BHIS    29$             ;NO-- SAVE SOME MORE
                         ;173670                  ADDX    DIAG2-TO11DT-2,R4 ;R4 POINTS TO DIAG2 RE
001226' 005724    005724 ;173670 005724           TST     (R4)+
001230' 010401    010401 ;173672 010401           MOV     R4,R1           ; SO DOES R1
001232' 012700    012700 ;173674 012700           MOV     #DRESET,R0      ;SETUP R0 FOR 'DIAGNOSTI
001234' 000100    000100 ;173676 000100
001236' 010021    010021 ;173700 010021           MOV     R0,(R1)+        ;R1 POINTS TO STATUS REG
001240' 005061    005061 ;173702 005061           CLR     DLYCNT-STAT(R1) ;SET DTE20 FOR MAXIMUM D
001242' 177744    177744 ;173704 177744
001244' 005061    005061 ;173706 005061           CLR     TO10AD-STAT(R1) ;START DUMPING -11 MEMOR
001246' 177764    177764 ;173710 177764
                         ;173712         30$:
001250' 032711    032711 ;173712 032711           BIT     #TO11DB,(R1)    ;IS DOORBELL RINGING (TR
001252' 004000    004000 ;173714 004000
001254' 001775    001775 ;173716 001775           BEQ     30$             ;NO-- WAIT FOR DOORBELL
001256' 010014    010014 ;173720 010014           MOV     R0,(R4)         ;YES-- CLEAR DOORBELL AN
001260' 005061    005061 ;173722 005061           CLR     TO11AD-STAT(R1) ;START INPUT TO LOCATION
001262' 177766    177766 ;173724 177766
001264' 012761    012761 ;173726 012761           MOV     #IFLOP!<<-256.>&7777>,TO11BC-STAT(R1) ;2
001266' 107400    107400 ;173730 107400
001270' 177762    177762 ;173732 177762
                         ;173734         40$:
001272' 032711    032711 ;173734 032711           BIT     #TO11DN,(R1)    ;TRANSFER COMPLETE?
001274' 000200    000200 ;173736 000200
001276' 001775    001775 ;173740 001775           BEQ     40$             ;NO-- WAIT UNTIL DONE
001300' 005007    005007 ;173742 005007           CLR     PC              ;GO TO LOADED CODE, STAR
                         ;173744                  FILLTO  1000
001302' 000000    000000 ;173744    000           .BYTE   0
                         ;173745    000           .BYTE   0
001304' 000000    000000 ;173746    000           .BYTE   0
                         ;173747    000           .BYTE   0
```

```
BMFB DEC/X11 SYSTEM EXERCISER MODULE    MACY11 30A(1052)  12-OCT-78  16:20  PAGE 13
XBMFB0.P11    12-OCT-78 11:50                                                                        SEQ 0012

  539  001306' 000000                              000000  ;173750    000          .BYTE   0
  540                                                      ;173751    000          .BYTE   0
  541  001310' 000000                              000000  ;173752    000          .BYTE   0
  542                                                      ;173753    000          .BYTE   0
  543  001312' 000000                              000000  ;173754    000          .BYTE   0
  544                                                      ;173755    000          .BYTE   0
  545  001314' 000000                              000000  ;173756    000          .BYTE   0
  546                                                      ;173757    000          .BYTE   0
  547  001316' 000000                              000000  ;173760    000          .BYTE   0
  548                                                      ;173761    000          .BYTE   0
  549  001320' 000000                              000000  ;173762    000          .BYTE   0
  550                                                      ;173763    000          .BYTE   0
  551  001322' 000000                              000000  ;173764    000          .BYTE   0
  552                                                      ;173765    000          .BYTE   0
  553  001324' 000000                              000000  ;173766    000          .BYTE   0
  554                                                      ;173767    000          .BYTE   0
  555  001326' 000000                              000000  ;173770    000          .BYTE   0
  556                                                      ;173771    000          .BYTE   0
  557  001330' 000000                              000000  ;173772    000          .BYTE   0
  558                                                      ;173773    000          .BYTE   0
  559  001332' 000000                              000000  ;173774    000          .BYTE   0
  560                                                      ;173775    000          .BYTE   0
  561  001334' 000000                              000000  ;173776    000          .BYTE   0
  562                                                      ;173777    000          .BYTE   0
  563  001336' 177777                     TABEND:  177777  ;174000
  564
  565          000001                                      .END
```

```
BMFB DEC/X11 SYSTEM EXERCISER MODULE    MACY11 30A(1052)  12-OCT-78  16:20  PAGE 15
XBMFB0.P11    12-OCT-78 11:50           CROSS REFERENCE TABLE -- USER SYMBOLS                        SEQ 0013

ACSR     000102R            185#
ADDR     000006R            151#     204
ADDR22=  001000             203#
AGAIN    000230R            207#     225
ASB      000106R            189#     233*
ASTAT    000104R            187#
AWAS     000110R            190#     232*
BEGIN    000000R            148#     223      235
BIT0   = 000001             203#
BIT1   = 000002             203#
BIT10  = 002000             203#
BIT11  = 004000             203#
BIT12  = 010000             203#
BIT13  = 020000             203#
BIT14  = 040000             203#
BIT15  = 100000             203#
BIT2   = 000004             203#
BIT3   = 000010             203#
BIT4   = 000020             203#
BIT5   = 000040             203#
BIT6   = 000100             203#
BIT7   = 000200             203#
BIT8   = 000400             203#
BIT9   = 001000             203#
BMERR    000310R            215      218      230#
BMTAB    000336R            208      245#
BREAK$=  104407             203#
BR1      000012R            153#
BR2      000013R            154#
BTOD$ =  104421             203#
CDATA$=  104412             203#
CONFIG   000056R            173#
CSRA     000100R            183#
DATCK$=  104411             203#
DATER$=  104404             203#     235
DVID1    000014R            155#
ENDIT$=  104413             203#     223
END$  =  104410             203#
ERRTYP   000106R            188#
EXIT$ =  104400             203#
GETPA$=  104415             203#
GWBUF$=  104414             203#
HRDCNT   000044R            168#
HRDER$=  104405             203#
HRDPAS   000050R            170#
ICONT    000036R            165#
ICOUNT   000040R            166#
IDNUM    000122R            195#
INIT     000030R            162#
INTR     000120R            194#
MAP22$=  104416             203#
MODNAM   000000R            149#
MODSP    000224R            163      201#
MSGN$ =  104403             203#
MSGS$ =  104402             203#
MSG$  =  104401             203#
```

```
BMFB DEC/X11 SYSTEM EXERCISER MODULE      MACY11 30A(1052)  12-OCT-78  16:20  PAGE 16
XBMFB0.P11    12-OCT-78 11:50             CROSS REFERENCE TABLE -- USER SYMBOLS                        SEQ 0014

NULL   = 000000              203#
OPEN   = 000000              150      156      157      158      159      176      177      178      179      180      181      182      183
                             185      187      189      190      192      193      194      203#
OTOA$ = 104420               203#
PASCNT   000034R             164#
PIRQ$ = 000004               203#
POPSP = 005726               203#
POPSP2= 022626               203#
PRTY   = 000000              153      154      203#
PRTY0 = 000000               203#
PRTY1 = 000040               203#
PRTY2 = 000100               203#
PRTY3 = 000140               203#
PRTY4 = 000200               203#
PRTY5 = 000240               203#
PRTY6 = 000300               203#
PRTY7 = 000340               203#
PS     = 177776              203#
PSW    = 177776              203#
PUSH   = 005746              203#
PUSH2 = 024646               203#
RANDS = 104417               203#
RANNUM   000054R             172#
RESTRT   000230R             191      206#
RES1     000056R             174#
RES2     000060R             175#
RSTRT    000112R             191#
SBADR    000102R             184#     230*
SOFCNT   000042R             167#
SOFERS$= 104406              203#
SOFPAS   000046R             169#
SPOINT   000032R             163#
SPSIZ = 000040                 1#     196
SR1      000016R             156#
SR2      000020R             157#
SR3      000022R             158#
SR4      000024R             159#
START    000224R             162      204#
STAT     000026R             151#
SVR0     000062R             176#
SVR1     000064R             177#
SVR2     000066R             178#
SVR3     000070R             179#
SVR4     000072R             180#
SVR5     000074R             181#
SVR6     000076R             192#
SYSCNT   000052R             171#
TABEND   001336R             220      563#
TRPDFD= 000022               203#
VECTOR   000010R             152#
WASADR   000104R             186#     231*
WDFR     000116R             193#
WDTO     000114R             192#
XFLAG    000005R             150#

BMFB DEC/X11 SYSTEM EXERCISER MODULE      MACY11 30A(1052)  12-OCT-78  16:20  PAGE 17
XBMFB0.P11    12-OCT-78 11:50             CROSS REFERENCE TABLE -- USER SYMBOLS                        SEQ 0015

. ABS.   000000       000
         001340       001

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 XBMFB0,XBMFB0/SOL/CRF:SYM=DDXCOM,XBMFB0
 RUN-TIME: 1 1 .2 SECONDS
 RUN-TIME RATIO: 21/3=5.5
 CORE USED:  7K  (13 PAGES)
```