

.REM

IDENTIFICATION

PRODUCT CODE: AC-F0698-MC
PRODUCT NAME: CXBMG80 BM873-YJ 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

BMGB IS A BACKGROUND MODULE THAT EXERCISES A SINGLE BM873-YJ 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-YJ OPTION

STORAGE:: BMGB 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 BMGB 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.

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-ROUTINE 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 <BMGB > 173000,0,0,0,100,156
000000- MODULE 40020, BMGB, 173000,0,0,0,100,156
          - TITLE BMGB DEC/X11 SYSTEM EXERCISER MODULE
          , DDXCOM VERSION 6 23-MAY-78
          *****LIST BIN*****
000000- BEGIN:
000000- 046502 041107 040 MODNAM: - ASCII /BMGB / ;MODULE NAME
000000- 000000 000 RPLAC: - BYTE OPEN ;USED TO KEEP TRACK OF WBUFF USAGE
000000- 173000 ADDR: 173000+0 ;1ST DEVICE ADDR.
000010- 000000 VECTOR: 0+0 ;1ST DEVICE VECTOR.
000012- 000 BR1: - BYTE PRTO+0 ;1ST BR LEVEL.
000014- 000 BR2: - BYTE PRTO+0 ;2ND BR LEVEL.
000016- 000001 DVID1: 0+1 ;DEVICE INDICATOR 1.
000020- 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 ICDNT: 100 ;# OF ITERATIONS PER PASS=100
000040- 000000 ICONF: 0 ;LOC TO COUNT ITERATIONS
000042- 000000 SOFCNT: 0 ;LOC TO SAVE TOTAL SOFT ERRORS
000044- 000000 HRDCNT: 0 ;LOC TO SAVE 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 RANDVC: 0 ;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- 000000 SBADR: OPEN ;ADDR OF GOOD DATA, OR
000104- 000000 ACSR: OPEN ;CONTENTS OF CSR.
000104- 000000 WABADR: OPEN ;ADDR OF BAD DATA, OR
000106- 000000 ASTAT: OPEN ;STATUS REG CONTENTS.
000106- 000000 ERRDTP: OPEN ;TYPE OF ERROR
000108- 000000 ASB: OPEN ;EXPECTED DATA.
000110- 000000 AVAS: OPEN ;ACTUAL DATA.
000112- 000230- RSTRT: RSTRT ;RESTART ADDRESS AFTER END OF PASS
000114- 000000 WMDT: OPEN ;WORDS TO MEMORY PER ITERATION
000116- 000000 WDFR: OPEN ;WORDS FROM MEMORY PER ITERATION
000120- 000000 INTR: OPEN ;# OF INTERRUPTS PER ITERATION
000122- 000156 IONUM: 156 ;MODULE IDENTIFICATION NUMBR=156
          -REPT SPSIZ ;MODULE STACK STARTS HERE.
          -NLIST

```

```

          .WORD 0
          .LIST
          .ENDR
000224-
          *****

```

199
200 000224 016705 177556
201
202 000230
203 000230 010501
204 000232 012702 000336
205 000232 022701 173024
206 000242 001413
207 000244 022701 173224
208 000250 001410
209 000252 021112
210 000254 001404
211 000256 004767 000026
212 000262 021112
213 000264 001402
214 000266 004767 000016
215 000272 022172
216 000274 052162 001336
217 000300 001356
218 000302
219 000302 104413 000000
220
221 000306 000750
222
223
224
225
226 000310 010267 177566
227 000310 012767 177564
228 000320 011167 177564
229 000324 011267 177556
230
231 000330 104404 000000
232
233 000334 000207
234
235
236
237
238
239
240 000336
241 000336 010037
242 000340 000040
243 000342 013700
244 000344 177570
245 000346 032700
246 000350 000001
247 000352 001007
248 000354 000510
249
250 000356 005000
251 000360 000404
252
253 000362 173000
254 000364 000340

```

START:  MOV      ADDR,R5          ;GET FIRST ROW ADDRESS INTO R5
RESTART:
AGAIN:  MOV      R5,R1            ;R1 POINTS TO ROW WORD
        MOV      R2,R1            ;R2 POINTS TO ROW TRACE IN CORP
        CMP      #173024,R1      ;ROW ADDRESS = 173024 ??
        BEQ     J5               ;BR IF YES
        CMP      #173224,R1      ;ROW ADDRESS = 173224 ??
        BEQ     J5               ;BR IF YES
        MOV      (R1),(R2)       ;CHECK ONE LOCATION
        BR      IF [CORN] = [CORE]
        JSR     PC,BMERR         ;GO SETUP AND REPORT ERROR
        CMP      (R1),(R2)       ;CHECK IT AGAIN
        BR      IF [CORN] = [CORE]
        JSR     PC,BMERR         ;GO SETUP AND REPORT ERROR
        ADD     #2,RO             ;ADD +2 TO BOTH POINTERS
        CMP      #TABEND,R2      ;DONE LAST WORD ??
        BNE     J5              ;BR IF NOT

4$:     ENDITS,BEGIN            ;SIGNAL END OF ITERATION.
        BR      AGAIN           ;MONITOR SHALL TEST END OF PASS

;THIS ROUTINE SETS UP AND REPORTS ALL DATA COMPARE ERRORS
BMERR:  MOV      R2,SBADR        ;SAVE THE ADDR. OF GOOD DATA
        MOV      R1,WASADR       ;SAVE ADDR. OF THE BAD DATA
        MOV      (R1),AMAS       ;GET WAS DATA
        MOV      (R2),ASB        ;GET THE S/B DATA
        *****
        DATERS,BEGIN           ;DATA ERROR!!!
        *****
        RTS     PC              ;CONTINUE CHECKING

;256(10) WORD TABLE THAT STORES A CORE IMAGE OF THE CONTENTS OF THE ROW
BMTAB:  010037  ;173000  010037  BUTON1:  MOV      RO,RSSGCV+0  ;SAVE RO IN LOCA
        010040  ;173000  013700  MOV      @#SWR,RO      ;GET SWITCH REGI
        013700  ;173000  032700  BIT      @BIT0,RO     ;IS LOW-ORDFR BI
        032700  ;173010  000001  MOV      @#BIT0,RO     ;IS LOW-ORDFR BI
        000001  ;173012  000001  MOV      @#BIT0,RO     ;IS LOW-ORDFR BI
        000001  ;173014  001007  BUTON3:  CLR      RO          ;SAVE LOAD FROM P
        001007  ;173016  000510  BR       REGSAV        ;NO-- SAVE R1-R7
        000510  ;173020  005000  BR       REGSAV        ;NO-- SAVE R1-R7
        005000  ;173022  000404  CLR      RO          ;SAVE LOAD FROM P
        000404  ;173022  000464  BR       REGSAV        ;NO-- SAVE R1-R7
        000464  ;173024  000464  BR       REGSAV        ;NO-- SAVE R1-R7
        173000  ;173024  173000  FILLTO  24          ;GO TO COMMON CF
        000340  ;173026  000340  .WORD  BM873,PP7

```

255
256 000366 012700
257 000370 000200
258
259 000372 010005
260 000374 106300
261 000376 106300
262 000400 000000
263 000402 101001
264 000404 005000
265
266 000406 000300
267 000410 042700
268 000412 177770
269 000414 105705
270 000416 100551
271 000420 012737
272 000422 173274
273 000424 000004
274 000426 005037
275 000430 000006
276
277 000432 012706
278 000434 000020
279 000436 012701
280 000440 177170
281 000442 005711
282
283 000444 005705
284 000446 100402
285 000450 005306
286 000452 002531
287
288 000454 000005
289
290 000456 032711
291 000460 000040
292 000462 001775
293 000464 111704
294 000466 010002
295 000470 001402
296 000472 012702
297 000474 000020
298
299 000476 052702
300 000500 000007
301 000502 010103
302 000504 010223
303
304 000506 105711
305 000510 100376
306 000512 012713
307 000514 000001
308 000516 005304
309 000520 001372
310

```

012700  ;173030  012700  BUTON2:  MOV      #BIT7,RO  ;BIT 7 MEANS LOA
000200  ;173032  000200  MOV      @#SWR,RO      ;GET SWITCH REGI
010005  ;173034  010005  BUTONX:  MOV      RO,R5          ;SAVE PARAMETER
106300  ;173036  106300  ASLR     #3,BIT4,RO    ;LEFT-ALIGN SPEE
122700  ;173042  122700  CMPR    #3,BIT4,RO    ;IS SPEED 0, 1,
101001  ;173044  101001  BHI     R0             ;YES-- UNIT IS U
005000  ;173046  005000  CLR     R0             ;NO-- USE UNIT #
000300  ;173050  000300  10$:    SWAB     RO          ;GET UNIT # IN L
042700  ;173052  042700  BIC     #C7,RO        ;TRIM TO 3 BITS
105705  ;173056  105705  TSTB   R5             ;WHERE SHOULD WE
100551  ;173060  100551  BMI     RPB00T        ;BIT 7 = 1 -- BO
012737  ;173062  012737  MOV     #TCBOOT,#4    ;BOOT FR
005037  ;173064  005037  CLR     @#6           ;SET PS OF TINED
000006  ;173072  000006  RXBOOT:  MOV      #RSSTRY,SP ;SET RETRY COUNT
012706  ;173074  012706  MOV     #R$X11+RXCS,R1 ;ADDRESS CONTROL
000020  ;173076  000020  TST     (R1)          ;RX11 EXIST?
012701  ;173102  012701  TXRTRY:  TST     R5          ;INDEFINITE RETR
005705  ;173106  005705  BMI     RXRSET        ;YES-- TRY FAITH
100402  ;173110  100402  DEC     SP             ;NO-- DECREMENT
005306  ;173112  005306  BLT     RXEHLT        ;GIVE UP IF RUN
002531  ;173114  002531  RXRSET:  RESET     #6          ;CLEAR THE WORLD
000005  ;173116  000005  20$:    BIT      #RXDONE,(R1) ;WAIT UNTIL READ
032711  ;173120  032711  BEQ     20$          ;NOT YET-- WAIT
001775  ;173122  001775  MOV     (PC),R4       ;SET TRACK/SECTO
111704  ;173126  111704  MOV     R0,R2         ;GET UNIT # ***
010002  ;173130  010002  BEQ     30$          ;ZERO-- USE ZERO
001402  ;173132  001402  BEQ     30$          ;ZERO-- USE ZERO
012702  ;173134  012702  MOV     #RXUNIT,R2   ;NOW--ZERO-- ASSU
000020  ;173136  000020  30$:    BIS     #RXREAD+RXGO,P2 ;SET READ FUNCTI
052702  ;173140  052702  MOV     R1,R3         ;COPY ADDRESS OF
000007  ;173142  000007  MOV     R2,(R3)+     ;START READ FUNC
010103  ;173144  010103  40$:    TSTR    (R1)         ;READY?
010223  ;173146  010223  BPL     40$          ;NO-- WAIT
012713  ;173152  012713  MOV     #1,(R3)      ;SET SECTOR #, T
000001  ;173156  000001  50$:    DEC     R4           ;COUNT DOWN SECT
005304  ;173160  005304  BNE     40$          ;TRACK TO SET ST
001372  ;173162  001372

```


535	001276	177744	177744	173740	177744				
536	001300	005061	005061	173742		005061	CLR	TOLOAD-STAT(R1)	START DUMPING -
537	001302	177764	177764	173744	177764				
538				173746					
539	001304	032711	032711	173746		032711	30S:	BIT	STO11DB,(R1) ST DOORBELL RIN
540	001306	004000	004000	173750	004000				
541	001310	001775	001775	173752		001775	REQ	30S	NO-- WAIT FOR D
542	001312	010014	010014	173754		010014	MOV	R0,(R4)	YES-- CLEAR DOO
543	001314	005061	005061	173756		005061	CLR	TOLOAD-STAT(R1)	START INPUT TO
544	001316	177766	177766	173760	177766				
545	001320	012761	012761	173762		012761	MOV	#IFLOP1<<-256.>&BCOUNT>,TO11BC-S	
546	001322	107400	107400	173780	107400				
547	001324	177782	177782	173782	177782				
548				173770					
549	001326	105711	105711	173770		105711	40S:	TSTB	(R1) TRANSFER COMPLE
550	001330	100376	100376	173772		100376	RPL	30S	NO-- WAIT SOME
551	001332	005007	005007	173774		005007	CLR	0	GO TO LOADED CO
552				173776					
553	001334	000000	000000	173778		000	FILL TO	1000	
554				173778		000	-BYTE	0	
555	001336	177777	177777	173778		000	-BYTE	0	
556				173778		000			
557	000001								

TABEND: 177777

.END

ACSR	000102R	181#	
ADDR	000006R	147#	200
ADDR22=	001000	199#	
AGAIN	000230R	203#	221*
ASB	000106R	185#	229*
ASTAT	000104R	183#	
AWAS	000110R	186#	228*
BEGTN	000000R	144#	219
BIT0 =	000001	199#	
BIT1 =	000002	199#	
BIT10 =	002000	199#	
BIT11 =	004000	199#	
BIT12 =	010000	199#	
BIT13 =	020000	199#	
BIT14 =	040000	199#	
BIT15 =	100000	199#	
BIT2 =	000004	199#	
BIT3 =	000010	199#	
BIT4 =	000020	199#	
BIT5 =	000040	199#	
BIT6 =	000100	199#	
BIT7 =	000200	199#	
BIT8 =	000400	199#	
BIT9 =	001000	199#	
BMERR	000310R	211#	214 226#
BMTAB	000336R	204#	240#
BREAKS=	104407	199#	
BPI	000012R	149#	
BW	000013R	150#	
BTODS =	104421	199#	
CDATA=	104412	199#	
CONFIG	000056R	169#	
CSRA	000100R	178#	
DATCKS=	104411	199#	
DATFRS=	104404	199#	231
DVIO1	000014R	151#	
ENDITS=	104413	199#	219
ENDS =	104410	199#	
ERRTPP	000106R	184#	
EXITS =	104400	199#	
GTPAS =	104415	199#	
GWBOP=	104414	199#	
HRDCNT	000044R	164#	
HRDRS =	104405	199#	
HRDPAS	000050R	166#	
ICONT	000036R	161#	
ICOUNT	000040R	162#	
INMM	000122R	191#	
INIT	000030R	158#	
INTR	000120R	190#	
MAP22S=	104416	199#	
MDNAM	000000R	145#	
MDSP	000224R	159#	197#
MSGNS =	104403	199#	
MSGSS =	104402	199#	
MSGS =	104401	199#	

NULL = 000000	199#													
OPEN = 000000	146#	152	153	154	155	172	173	174	175	176	177	178	179	
	181#	183	185	186	188	189	190	199#						
QTOAS = 104420	199#													
PASCNT = 000034R	160#													
PTRQS = 000004	199#													
PDFS = 025720	199#													
PDFSP2 = 025720	199#													
PRTY = 000000	199#													
PRTY0 = 000000	149#	150	199#											
PRTY1 = 000040	199#													
PRTY2 = 000100	199#													
PRTY3 = 000140	199#													
PRTY4 = 000200	199#													
PRTY5 = 000240	199#													
PRTY6 = 000300	199#													
PRTY7 = 000340	199#													
PS = 177770	199#													
PSM = 177770	199#													
PUSH = 005746	199#													
PUSH2 = 024646	199#													
RANDS = 104417	199#													
RANNUM = 000054R	168#													
RPSRT = 000230R	187#	202#												
RES1 = 000058R	170#													
RES2 = 000060R	171#													
RSTRT = 000112R	187#													
SBADR = 000102R	180#	226*												
SDFCNT = 000042R	163#													
SDFERS = 104406	199#													
SDFPAS = 000046R	165#													
SPDINT = 000032R	159#													
SPSIZ = 000040	192													
SR1 = 000016R	152#													
SR2 = 000020R	153#													
SR3 = 000022R	154#													
SR4 = 000024R	155#													
START = 000224R	158#	200#												
STAT = 000025R	157#													
SVR0 = 000062R	172#													
SVR1 = 000064R	173#													
SVR2 = 000066R	174#													
SVR3 = 000070R	175#													
SVR4 = 000072R	176#													
SVR5 = 000074R	177#													
SVR6 = 000076R	178#													
SVSCT = 000052R	167#													
TABFND = 001336R	216#	555#												
TRPDFD = 000022	199#													
VECTOR = 000010R	148#													
WASADR = 000104R	182#	227*												
WDFR = 000116R	189#													
WDT0 = 000104R	188#													
XPLAG = 000005R	146#													

. ARS- 000000 000
001340 001

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

XRMGB0, XRMGB0/SOL/CRF:SYM=DDXCON, XRMGB0
RUN-TIME: 1 1 .1 SECONDS
RUN-TIME RATIO: 38/3=10.2
CORE USED: 7K (13 PAGES)