

.REM

IDENTIFICATION

PRODUCT CODE: AC-F0668-MC
PRODUCT NAME: CXBMFR0 BM873-VH 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

BMFB 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:: BMFB 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-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-      3KMOD <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 WBNFF USAGE
000006-      173000 ADDR: 173000+0 ;1ST DEVICE ADDR.
000010-      000000 VECTOR: +0 ;1ST DEVICE VECTOR.
000012-      000 BR1: .BYTE PRTY+0 ;1ST RR LEVEL.
000013-      000 BR2: .BYTE PRTY+0 ;2ND RR 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: 40DSP ;MODULE STACK POINTER.
000034-      000000 PASCMT: 0 ;PASS COUNTER.
000036-      000100 ICOUNT: 100 ;# OF ITERATIONS PER PASS=100
000040-      000000 ICDUNT: 0 ;LOC TO COUNT ITERATIONS
000042-      000000 SOPCMT: 0 ;LOC TO SAVE TOTAL SOFT ERRORS
000044-      000000 HRDCMT: 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 SYSCMT: 0 ;# OF SYS ERRORS ACCUMULATED
000054-      000000 RANNUM: 0 ;Holds RANDOM # WHEN RAND MACRO IS CALLED
000056-      000000 COMFIC: ;RESERVED FOR MONITOR USE
000060-      000000 RES1: 0 ;RESERVED FOR MONITOR USE
000062-      000000 RES2: 0 ;RESERVED FOR MONITOR USE
000064-      000000 SVR6: OPEN ;LOC TO SAVE R6.
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.
000106-      000000 WASADR: OPEN ;ADDR OF BAD DATA, OR
000110-      000000 ASTAT: OPEN ;STATUS REG CONTENTS.
000112-      000230 ERRTYP: ;TYPE OF ERROR
000114-      000000 ASB: OPEN ;EXPECTED DATA.
000116-      000000 AWAS: OPEN ;ACTUAL DATA.
000118-      000230 RSTRT: RSTRT ;RESTART ADDRESS AFTER END OF PASS
000120-      000000 WDFR: OPEN ;WORDS TO MEMORY PER ITERATION
000122-      000155 INTR: OPEN ;WORDS FROM MEMORY PER ITERATION
          IDNUM: 155 ;# OF INTERRUPTS PER ITERATION
          -REPT SPSTZ ;MODULE IDENTIFICATION NUMBER=155
          -NLST ;MODULE STACK STARTS HERE.

```

```

          .WORD 0
          .LIST
          .ENDR
000224-      MODSP:
;*****

```

203
204 000224* 016705 177556
205
206 000230* 010501
207 000230* 022701 000336*
208 000236* 022701 173024
209 000242* 001483
210 000244* 027137 173224
211 000250* 001410
212 000252* 021112
213 000254* 001487
214 000254* 004767 000026
215 000262* 021112
216 000264* 001487
217 000266* 004767 000016
218 000272* 027137
219 000274* 022701 001336*
220 000300* 001356
221 000302* 104413 000000*
222 000306* 000750
223
224
225
226
227
228
229
230 000310* 010267 177566
231 000320* 011167 177564
232 000324* 011267 177564
233
234 000330* 104404 000000*
235
236 000334* 000207
237
238
239
240
241
242
243
244
245
246 000336* 010037
247 000336* 000940
248 000342* 013700
249 000344* 177570
250 000346* 032700
251 000348* 000001
252 000352* 001007
253 000354* 000513
254
255 000356* 005000
256 000360* 000404
257
258 000362* 173000

START: MOV ADDR,R5 ;GET FIRST ROM ADDRFS INTO R5
RSTR: MOV R5,R1 ;R1 POINTS TO ROM WORD
AGAIN: MOV RBTAB,R2 ;R2 POINTS TO ROM IMAGE IN CORE
1S: CMP #173024,R1 ;ROM ADDRESS = 173024 ??
REQ JSR ;BR IF YES
REQ #173224,R1 ;ROM ADDRESS = 173224 ??
REQ JSR ;BR IF YES
CMP (R1),(R2) ;CHECK ONE LOCATION
REQ PC,RMERR ;BR IF [CROM] = [CORE]
2S: JSR PC,RMERR ;GO SETUP AND REPORT ERROR
CMP (R1),(R2) ;CHECK IT AGAIN
REQ JSR ;BR IF [CROM] = [CORE]
3S: JSR PC,RMERR ;GO SETUP AND REPORT ERROR
CMP (R1)+(R2)+ ;ADD +2 TO BOTH POINTERS
BNE #TABEND,R2 ;DONE LAST WORD ??
4S: BNE IS ;BR IF NOT

THIS ROUTINE SETS UP AND REPORTS ALL DATA COMPARE ERRORS

BMERR: MOV R2,SBADR ;SAVE THE ADDR. OF GOOD DATA
MOV R1,ASADR ;SAVE ADDR. OF THE BAD DATA
MOV (R1),AWAS ;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 ROM

BMTAB: 010037 ;173000 010037 BUTON1: MOV R0,ROT07*0 ;SAVE R0 IN LOCATION 40
000440 ;173000 000040 MOV SWR,R0 ;GET SWITCH REGISTER
013700 ;173004 013700 MOV #R10,R0 ;IS LOW-ORDER BIT SET?
177570 ;173006 177570 BIT ;YES-- LOOK AT CONTENTS
032700 ;173010 032700 BNE BUTONX ;NO-- SAVE R1-R7 IN 42-5
000001 ;173012 000001 BR RFGSAV
001007 ;173014 001007 BUTON3: CLR R0 ;SAVE LOAD FROM PLOOPY, 0
000513 ;173016 000513 BR BUTONX ;GO TO COMMON CODE FOR 3
005000 ;173020 005000 FILL TO 24
000404 ;173022 000404 .WORD
173000 ;173024 173000 ROMORG,PR7

259 000364* 000340
260
261 000366* 012700
262 000370* 000200
263
264 000372* 010005
265 000374* 106300
266 000376* 107000
267 000400* 000000
268 000402* 001001
269 000404* 005000
270
271 000406* 000300
272 000410* 042700
273 000412* 177770
274 000414* 105705
275 000416* 100553
276 000420* 012737
277 000422* 173304
278 000424* 000004
279 000426* 005037
280 000430* 000006
281
282 000432* 012706
283 000434* 000014
284 000436* 012701
285 000440* 177170
286 000442* 010003
287
288 000444* 005705
289 000446* 100402
290 000450* 005306
291 000452* 002445
292
293 000454* 000005
294
295 000456* 032711
296 000460* 000040
297 000462* 001775
298
299 000464* 010300
300 000466* 001402
301 000470* 012700
302 000472* 000020
303
304 000474* 052700
305 000476* 000007
306 000500* 010102
307 000502* 010022
308
309 000504* 105711
310 000506* 100376
311 000510* 012712
312 000512* 000001
313
314 000514* 105711

000340 ;173026 000340 BUTON2: MOV #BIT7,R0 ;BIT 7 MEANS LOAD FROM R
012700 ;173030 012700 BUTONX: MOV R0,R5 ;SAVE PARAMETER FOR BOOT
000200 ;173032 000200 ASLR R0 ;LEFT-ALIGN SPEED FIELD
010005 ;173034 010005 CMPR #3*BIT4,R0 ;IS SPEED 0, 1, OR 2?
106300 ;173036 106300 BHI 10S ;YES-- UNIT IS UNIT TO 0
107000 ;173038 107000 CLR R0 ;NO-- USE UNIT #0
000000 ;173040 000000
101001 ;173044 101001
005000 ;173046 005000
000300 ;173050 000300
042700 ;173052 042700
177770 ;173054 177770
105705 ;173056 105705
100553 ;173058 100553
012737 ;173062 012737
173304 ;173064 173304
000004 ;173066 000004
005037 ;173068 005037
000006 ;173072 000006
012706 ;173074 012706 RXBOOT: MOV #RETRY,SP ;SET RETRY COUNT
000014 ;173076 000014 MOV #RXEPA+RXCS,R1 ;ADDRESS CONTROL STATUS
012701 ;173100 012701
177170 ;173102 177170
010003 ;173104 010003
005705 ;173106 005705 RXRTRY: TST R5 ;INDEFINITE RETRY?
00402 ;173110 00402 BHI RKRSET ;YES-- TRY FAITHFULLY
005306 ;173112 005306 DEC SP ;NO-- DECREMENT RETRY CO
002445 ;173114 002445 BLT RKEHLT ;GIVE UP IF RUN OUT
000005 ;173116 000005 RXRSET: RESET ;CLEAR THE WORLD
032711 ;173120 032711 20S: BIT #RXDONE,(R1) ;WAIT UNTIL READY FOR FU
000040 ;173122 000040 BEQ 20S ;NOT YET-- WAIT
001775 ;173124 001775
010300 ;173126 010300 RXPERF: MOV R3,R0 ;GET UNIT #
001402 ;173130 001402 BEQ 5S ;ZERO-- USE ZERO
012700 ;173132 012700 MOV #RXUNIT,R0 ;NON-ZERO-- ASSUME UNIT
000020 ;173134 000020
052700 ;173136 052700 5S: BIS #PXREAD+RXGO,R0 ;SET READ FUNCTION
000007 ;173140 000007 MOV R1,R2 ;COPY ADDRESS OF RXCS
010102 ;173142 010102 MOV R0,(R2)+ ;START READ FUNCTION, R2
010022 ;173144 010022 10S: TSTB (R1) ;READY?
105711 ;173146 105711 BPL 10S ;NO-- WAIT
100376 ;173150 100376 MOV #1,(R2) ;SET SECTOR #
012712 ;173152 012712
000001 ;173154 000001
105711 ;173156 105711 20S: TSTB (R1) ;READY FOR TRACK?

315	000516	100376	100376	173160	100376	BPL	205		;NO-- WAIT
316	000520	012712	012712	173162	012712	MOV	#1,(R2)		;SET TRACK #
317	000522	000001	000001	173166	000001				
318				173166					
319	000524	032711	032711	173166	032711	30S:	BIT	#RXERRIRXDONE,(R1)	;DONE OR ERROR?
320	000526	100040	100040	173170	100040				
321	000530	001745	001745	173172	001745	REQ	30S		;NO-- WAIT
322	000532	100744	100744	173174	100744	BMI	RXRTRY		;YES-- ERROR IN FNCTION
323				173176					
324	000534	012711	012711	173176	012711	RXEMSL:	MOV	#RXFMPT+RXGO,(R1)	;START EMPTY
325	000536	000003	000003	173176	000003				
326	000540	000004	000004	173176	000004	CLR	R4		;ALWAYS START TRANSFER A
327				173204					
328	000542	132711	132711	173204	132711	10S:	BITB	#RXTREQIRXDONE,(R1)	;READY FOR WORD, OR
329	000544	000240	000240	173206	000240				
330	000546	001775	001775	173210	001775	REQ	10S		;NOT READY-- WAIT SOME M
331	000550	100153	100153	173212	100153	BPL	CLRPC		;DONE-- GO TO LOCATION 0
332	000552	111224	111224	173214	111224	MOVW	{R2},{R4}+		;NOT DONE-- GET A BYTE F
333	000554	000772	000772	173216	000772	BR	10S		;WAIT FOR NEXT BYTE
334				173220					
335	000556	000000	000000	173220	000000	BLLTO	0		
336				173221	000000		0		
337				173221	000000		0		
338	000560	000000	000000	173223	000000		0		
339				173224	173000		0		
340	000564	000340	000340	173226	000340		0		
341				173226			0		
342	000566	012711	012711	173226	012711	RXEHLT:	MOV	#RXRERR+RXGO,(R1)	;DO A READ ERROR REGIS
343	000570	000017	000017	173232	000017				
344				173234					
345	000572	032711	032711	173234	032711	10S:	BIT	#RXDONE,(R1)	;WAIT UNTIL ERROR ASSEMB
346	000574	000040	000040	173236	000040				
347	000576	001775	001775	173240	001775	REQ	10S		;GET ERROR REGISTER
348	000600	011200	011200	173242	011200	MOV	{R2},R0		;HALT AND DISPLAY ERRORS
349	000602	000541	000541	173242	000541	BR	HALTED		
350				173246					
351	000604	010037	010037	173246	010037	REGSAV:	MOV	R0,R0TOR7+16	;SAVE R0 AS PC IN 56
352	000606	000056	000056	173250	000056				
353	000610	010014	010014	173252	010014	MOV	#R0TOR7+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	173264	010440	MOV	R4,-(R0)		;SAVE R4 IN 50
358	000622	010340	010340	173264	010340	MOV	R3,-(R0)		;SAVE R3 IN 48
359	000624	010240	010240	173266	010240	MOV	R2,-(R0)		;SAVE R2 IN 46
360	000626	010140	010140	173270	010140	MOV	R1,-(R0)		;SAVE R1 IN 44
361	000630	014000	014000	173270	014000	MOV	-(R0),R0		;RESTORE R0 FROM 40
362	000632	000177	000177	173274	000177	JMP	@R0TOR7+16		;GO TO SAVED PC
363	000634	004556	004556	173276	004556				
364				173300					
365	000636	005000	005000	173300	005000	TCBOTO:	CLR	R5	;HERE TO START WITH A FL
366	000640	005005	005005	173302	005005	TCBOOT:	CLR	R5	;HERE TO START WITH A FL
367				173304					
368	000642	012706	012706	173304	012706	MOV	#RETRY,SP		;INIT RETRY COUNTER
369	000644	012701	012701	173310	012701	MOV	#TCEPA+TCCM,R1		;POINT TO COMMAND REGIST
370	000646	012701	012701	173310	012701				

371	000650	177342	177342	173312	177342	TCRTRY:	TST	R5		;INDEFINITE RETRY?
372				173314			BMI	10S		;YES-- TRY HARDER
373	000652	005705	005705	173314	005705		DEC	SP		;NO-- DECREMENT COUNT
374	000654	100402	100402	173316	100402	BLT	TCEHLT			;TOO MANY-- GIVE UP
375	000656	005306	005306	173322	005306					
376	000660	002427	002427	173322	002427	10S:	RESET	R0,R3		;CLEAR TC11
377				173324			MOV	R0,R3		;GET UNIT NUMBER
378	000662	000005	000005	173324	000005	SWAB	R3			;TO BITS 10-8
379	000664	010003	010003	173330	010003	MOV	R3,R4			;COPY FOR READ FUNCTION
380	000666	000303	000303	173330	000303	BIS	#TCPEV+TCRNM+TCGO,R3			;START TAPE TOWARD
381	000670	010304	010304	173332	010304					
382	000672	052703	052703	173334	052703					
383	000674	040003	040003	173334	040003					
384	000676	010311	010311	173340	010311	MOV	R3,(R1)			; . .
385				173342						
386	000700	005711	005711	173342	005711	20S:	TST	(R1)		;ERROR?
387	000702	100376	100376	173342	100376	BPL	20S			;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	BIS	#TCREAD+TCGO,R4			;START READ, FORWARD
394	000720	052704	052704	173362	052704					
395	000722	000005	000005	173364	000005					
396	000724	010411	010411	173366	010411	MOV	R4,(R1)			; . .
397				173370						
398	000726	105711	105711	173370	105711	30S:	TSTB	(R1)		;TRANSFER DONE?
399	000730	005711	005711	173372	005711	BPL	30S			;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						
404	000740	016100	016100	173402	016100	TCEHLT:	MOV	TCST-TCCM(R1),R0		;GET STATUS REGISTER
405	000742	177776	177776	173404	177776					
406	000744	000460	000460	173406	000460	RPBOOT:	BP	HALTED		;AND STOP
407				173410						
408	000746	012706	012706	173410	012706	MOV	#RETRY,SP			;RETRY RETRY TIMES
409	000750	000014	000014	173412	000014	MOV	#RPEPA+RPCS1,R1			;ADDRESS RPCS1 IN R1
410	000752	012701	012701	173414	012701	MOV	#RPECCI,R2			;SET ECC INHIBIT, 20 SEC
411	000754	176700	176700	173416	176700					
412	000756	012702	012702	173420	012702					
413	000760	004000	004000	173422	004000					
414				173424						
415	000762	005705	005705	173424	005705	RPRTRY:	TST	R5		;INFINITE RETRY?
416	000764	100402	100402	173426	100402	BPL	10S			;YES-- TRY AGAIN
417	000766	005306	005306	173430	005306	BMI	SP			;RETRY COUNT EXHAUSTED?
418	000770	002444	002444	173432	002444	BLT	RPEHLT			;YES-- GIVE UP
419				173434						
420	000772	000005	000005	173434	000005	10S:	RESET			;ZAPII
421	000774	110061	110061	173436	110061	MOVW	R0,RPCS2(R1)			;SELECT PROPER UNIT #
422	000776	000010	000010	173440	000010					
423	010000	032711	032711	173442	032711	BIT	#PPDVA,(R1)			;IS DRIVE AVAILABLE TO U
424	010002	004000	004000	173444	004000					
425	010004	001766	001766	173446	001766	REQ	RPRTRY			;NO-- TRY AGAIN
426	010006	012711	012711	173450	012711	MOV	#RPPRST+RPGO,(R1)			;DO "READ-IN PRESET" F

427 001010* 000021 000021 ;173452 000021 CLR RPD(R1) ;SET CYLINDER 0
428 001012* 005061 005061 ;173454 005061 CLR RPD(R1) ; TRACK 0, SECTOR 0
429 001014* 000034 000034 ;173456 000034 CLR RPD(R1) ; TRACK 0, SECTOR 0
430 001016* 005061 005061 ;173458 005061 CLR RPD(R1) ; TRACK 0, SECTOR 0
431 001018* 000006 000006 ;173460 000006 CLR RPD(R1) ; TRACK 0, SECTOR 0
432 001020* 000006 000006 ;173462 000006 CLR RPD(R1) ; TRACK 0, SECTOR 0
433 001022* 050261 050261 ;173464 050261 BIT R2,RPD(R1) ;SET INHIBIT ECC, 22-SEC
434 001024* 000032 000032 ;173466 000032 MOV #-256.,RPWC(R1) ;SET UP WORD COUNT TO PR
435 001026* 174400 174400 ;173468 174400 MOV #-256.,RPWC(R1) ;SET UP WORD COUNT TO PR
436 001028* 000002 000002 ;173470 000002 MOV #-256.,RPWC(R1) ;SET UP WORD COUNT TO PR
437 001030* 012711 012711 ;173472 012711 MOV #RPREAD+RPGO,(R1) ;START READ FUNCTION
438 001032* 000071 000071 ;173474 000071 MOV #RPREAD+RPGO,(R1) ;START READ FUNCTION
439 001034* 000071 000071 ;173476 000071 MOV #RPREAD+RPGO,(R1) ;START READ FUNCTION
440 001040* 105711 105711 ;173502 105711 20\$: TSTB (R1) ;READY?
441 001042* 100376 100376 ;173504 100376 BPL 20\$;NO-- WAIT UNTIL IT IS
442 001044* 032751 032751 ;173506 032751 BIT #RPFER,RPERI(R1) ;FORMAT ERROR?
443 001046* 000020 000020 ;173508 000020 BR ;TRY AGAIN
444 001048* 000014 000014 ;173510 000014 BR ;TRY AGAIN
445 001050* 000014 000014 ;173512 000014 BR ;TRY AGAIN
446 001052* 001403 001403 ;173514 001403 BEQ 30\$;NO-- TRY AGAIN
447 001054* 052702 052702 ;173516 052702 BIS #RPFM22,R2 ;YES-- TRY FOR 22 SECTOR
448 001056* 010000 010000 ;173518 010000 BR ;TRY AGAIN
449 001058* 000740 000740 ;173520 000740 BR ;TRY AGAIN
450 001060* 000740 000740 ;173522 000740 BR ;TRY AGAIN
451 001062* 032711 032711 ;173524 032711 30\$: BIT #RPTREIRPMCPE,(R1) ;TRANSFER OR M9C PARI
452 001064* 060000 060000 ;173526 060000 BNE RPRTRY ;YES-- ERROR-- TRY AGAIN
453 001066* 001335 001335 ;173528 001335 BNE RPRTRY ;YES-- ERROR-- TRY AGAIN
454 001068* 032761 032761 ;173530 032761 BIT #RPTAIRPERR,RPDS(R1) ;ATTN OR OTHER ERR
455 001070* 000012 000012 ;173532 000012 BIT #RPTAIRPERR,RPDS(R1) ;ATTN OR OTHER ERR
456 001072* 000012 000012 ;173534 000012 BIT #RPTAIRPERR,RPDS(R1) ;ATTN OR OTHER ERR
457 001074* 000012 000012 ;173536 000012 BIT #RPTAIRPERR,RPDS(R1) ;ATTN OR OTHER ERR
458 001076* 001331 001331 ;173538 001331 BIT #RPTAIRPERR,RPDS(R1) ;ATTN OR OTHER ERR
459 001100* 005007 005007 ;173542 005007 CLRPC: CLR PC ;JMP 0
460 001102* 016100 016100 ;173544 016100 RPEHLT: MOV RPDS(R1),R0 ;DISPLAY DRIVE STATUS
461 001104* 000012 000012 ;173546 000012 RPEHLT: MOV RPDS(R1),R0 ;DISPLAY DRIVE STATUS
462 001106* 000000 000000 ;173550 000000 HALTED: HALT ;DIE
463 001108* 000776 000776 ;173552 000776 BR ;DIE
464 001110* 000776 000776 ;173554 000776 BR ;DIE
465 001112* 010037 010037 ;173556 010037 BTON4: MOV RO,ROTOR7+0 ;SAVE RO IN 40
466 001114* 000040 000040 ;173558 000040 MOV #10S,R0 ;SET RETURN ADDRESS IN R
467 001116* 012700 012700 ;173560 012700 MOV #10S,R0 ;SET RETURN ADDRESS IN R
468 001118* 012700 012700 ;173562 012700 MOV #10S,R0 ;SET RETURN ADDRESS IN R
469 001120* 173566 173566 ;173564 173566 BR REGSAV ;SAVE R1-R7
470 001122* 000630 000630 ;173566 000630 BR REGSAV ;SAVE R1-R7
471 001124* 005005 005005 ;173568 005005 10\$: CLR R5 ;ADDRESS LOCATION ZERO
472 001126* 012500 012500 ;173570 012500 MOV (R5),R0 ;SAVE 0 IN R0
473 001128* 011502 011502 ;173572 011502 MOV (R5),R1 ;SAVE 1 IN R1
474 001130* 011502 011502 ;173574 011502 MOV (R5),R2 ;SAVE 2 IN R2
475 001132* 012725 012725 ;173576 012725 MOV (R5),R3 ;SAVE 3 IN R3
476 001134* 012725 012725 ;173578 012725 MOV #21S,(R5)+ ;SET NXM TRAP ADDRESS IN
477 001136* 173612 173612 ;173580 173612 MOV (R5),R3 ;SAVE 6 IN R3
478 001138* 011502 011502 ;173582 011502 CLR (R5),R3 ;SET PS FOR TRAP
479 001140* 005013 005013 ;173604 005013 CLR (R5),R3 ;SET PS FOR TRAP
480 001142* 005013 005013 ;173606 005013 CLR (R5),R3 ;SET PS FOR TRAP
481 001144* 012704 012704 ;173608 012704 20\$: MOV #DLVCNT-DTESI7,R4 ;POINT TO DTE # -1'S D
482 001146* 174340 174340 ;173610 174340 MOV #DLVCNT-DTESI7,R4 ;POINT TO DTE # -1'S D

483 001150* 012706 012706 ;173612 012706 21\$: MOV #4,SP ;SET SP TO 4, STACK IS L
484 001152* 000004 000004 ;173614 000004 MOV #4,SP ;SET SP TO 4, STACK IS L
485 001154* 062704 062704 ;173616 062704 22\$: ADD #DTESI2,R4 ;RUMP TO NEXT DTE'S EXT
486 001156* 000040 000040 ;173618 000040 ADD #DTESI2,R4 ;RUMP TO NEXT DTE'S EXT
487 001158* 105704 105704 ;173620 105704 TSTB R4 ;IS THIS THE END OF THE
488 001160* 105704 105704 ;173622 105704 BMT 20\$;YES-- START ALL OVER, U
489 001162* 100770 100770 ;173624 100770 BIT #T011DB,STAT-DLYCNT(R4) ;DOORBELL RINGIN
490 001164* 004000 004000 ;173626 004000 BFC 22\$;NO-- TRY NEXT DTE
491 001166* 000034 000034 ;173628 000034 CMP T010BC-DLYCNT(R4),(PC) ;FINISHED?
492 001168* 001770 001770 ;173630 001770 BNE 22\$;NO-- TRY ANOTHER DTE
493 001170* 000034 000034 ;173632 000034 MOV R3,(R5) ;RESTORE LOCATION 6
494 001172* 001770 001770 ;173634 001770 MOV R3,(R5) ;RESTORE LOCATION 6
495 001174* 026417 026417 ;173636 026417 MOV R2,(R5) ; 4
496 001176* 000014 000014 ;173638 000014 MOV R1,(R5) ; 2
497 001178* 001365 001365 ;173640 001365 MOV R0,(R5) ; 0
498 001180* 010315 010315 ;173642 001365 MOV #DTESAV,R0 ;POINT TO SAVE AREA
499 001182* 010315 010315 ;173644 001365 MOV #DTESAV,R0 ;POINT TO SAVE AREA
500 001184* 010145 010145 ;173646 010145 MOV #DTESAV,R0 ;POINT TO SAVE AREA
501 001186* 010045 010045 ;173648 010045 MOV #DTESAV,R0 ;POINT TO SAVE AREA
502 001188* 012700 012700 ;173650 012700 MOV #DTESAV,R0 ;POINT TO SAVE AREA
503 001190* 000130 000130 ;173652 000130 MOV #DTESAV,R0 ;POINT TO SAVE AREA
504 001192* 000130 000130 ;173654 000130 MOV #DTESAV,R0 ;POINT TO SAVE AREA
505 001216* 012420 012420 ;173660 012420 29\$: MOV (R4)+(R0)+ ;SAVE A REGISTER
506 001218* 022700 022700 ;173662 022700 CMP #T011DT-DLYCNT+DTESAV,R0 ;FINISHED?
507 001220* 000156 000156 ;173664 000156 BHS 29\$;NO-- SAVE SOME MORE
508 001222* 103374 103374 ;173666 103374 ADDX DIAG2-T011DT-2,R4 ;P4 POINTS TO DIAG2 RE
509 001224* 005724 005724 ;173668 005724 TST (R4)+ ; DOES R1
510 001226* 010401 010401 ;173670 010401 MOV R4,R1 ; SETUP R0 FOR "DIAGNOSTI
511 001228* 012700 012700 ;173672 012700 MOV #DRESRT,R0 ; SETUP R0 FOR "DIAGNOSTI
512 001230* 000100 000100 ;173674 000100 MOV R0,(R1)+ ; R1 POINTS TO STATUS REC
513 001232* 010021 010021 ;173676 010021 CLR DLYCNT-STAT(R1) ;SET DTE20 FOR MAXIMUM D
514 001234* 010021 010021 ;173678 010021 CLR DLYCNT-STAT(R1) ;SET DTE20 FOR MAXIMUM D
515 001236* 005061 005061 ;173680 005061 CLR T010AD-STAT(R1) ;START INPUT TO LOCATION
516 001238* 017744 017744 ;173682 017744 CLR T010AD-STAT(R1) ;START INPUT TO LOCATION
517 001240* 005061 005061 ;173684 005061 CLR T010AD-STAT(R1) ;START INPUT TO LOCATION
518 001242* 017744 017744 ;173686 017744 CLR T010AD-STAT(R1) ;START INPUT TO LOCATION
519 001244* 005061 005061 ;173688 005061 CLR T010AD-STAT(R1) ;START INPUT TO LOCATION
520 001246* 017744 017744 ;173690 017744 CLR T010AD-STAT(R1) ;START INPUT TO LOCATION
521 001250* 032711 032711 ;173712 032711 30\$: BIT #T011DB,(R1) ;IS DOORBELL RINGING (TR
522 001252* 004000 004000 ;173714 004000 BIT #T011DB,(R1) ;IS DOORBELL RINGING (TR
523 001254* 001775 001775 ;173716 001775 BEQ 30\$;NO-- WAIT FOR DOORBELL
524 001256* 010014 010014 ;173718 010014 MOV R3,(R4) ;YES-- CLEAR DOORBELL
525 001258* 005061 005061 ;173720 005061 CLR T011AD-STAT(R1) ;START INPUT TO LOCATION
526 001260* 177766 177766 ;173722 177766 CLR T011AD-STAT(R1) ;START INPUT TO LOCATION
527 001262* 012761 012761 ;173724 177766 MOV #IFLOPI<<-256.>&7777>,T011BC-STAT(R1) ;2
528 001264* 012761 012761 ;173726 012761 MOV #IFLOPI<<-256.>&7777>,T011BC-STAT(R1) ;2
529 001266* 107400 107400 ;173728 107400 MOV #IFLOPI<<-256.>&7777>,T011BC-STAT(R1) ;2
530 001268* 177762 177762 ;173730 177762 MOV #IFLOPI<<-256.>&7777>,T011BC-STAT(R1) ;2
531 001270* 177762 177762 ;173732 177762 MOV #IFLOPI<<-256.>&7777>,T011BC-STAT(R1) ;2
532 001272* 032711 032711 ;173734 032711 40\$: BIT #T011DN,(R1) ;TRANSFER COMPLETE?
533 001274* 000200 000200 ;173736 000200 BEQ 40\$;NO-- WAIT UNTIL DONE
534 001276* 001775 001775 CLR PC ;GO TO LOADED CODE, STAR
535 001278* 005007 005007 CLR PC ;GO TO LOADED CODE, STAR
536 001280* 000000 000000 ;173744 000000 FILL TO 1000
537 001282* 000000 000000 ;173746 000000 FILL TO 1000
538 001284* 000000 000000 ;173748 000000 FILL TO 1000
539 001286* 000000 000000 ;173750 000000 FILL TO 1000
540 001288* 000000 000000 ;173752 000000 FILL TO 1000
541 001290* 000000 000000 ;173754 000000 FILL TO 1000
542 001292* 000000 000000 ;173756 000000 FILL TO 1000
543 001294* 000000 000000 ;173758 000000 FILL TO 1000
544 001296* 000000 000000 ;173760 000000 FILL TO 1000
545 001298* 000000 000000 ;173762 000000 FILL TO 1000
546 001300* 005007 005007 ;173764 005007 FILL TO 1000
547 001302* 000000 000000 ;173766 000000 FILL TO 1000
548 001304* 000000 000000 ;173768 000000 FILL TO 1000
549 001306* 000000 000000 ;173770 000000 FILL TO 1000
550 001308* 000000 000000 ;173772 000000 FILL TO 1000
551 001310* 000000 000000 ;173774 000000 FILL TO 1000
552 001312* 000000 000000 ;173776 000000 FILL TO 1000
553 001314* 000000 000000 ;173778 000000 FILL TO 1000
554 001316* 000000 000000 ;173780 000000 FILL TO 1000
555 001318* 000000 000000 ;173782 000000 FILL TO 1000
556 001320* 000000 000000 ;173784 000000 FILL TO 1000
557 001322* 000000 000000 ;173786 000000 FILL TO 1000
558 001324* 000000 000000 ;173788 000000 FILL TO 1000
559 001326* 000000 000000 ;173790 000000 FILL TO 1000
560 001328* 000000 000000 ;173792 000000 FILL TO 1000
561 001330* 000000 000000 ;173794 000000 FILL TO 1000
562 001332* 000000 000000 ;173796 000000 FILL TO 1000
563 001334* 000000 000000 ;173798 000000 FILL TO 1000
564 001336* 000000 000000 ;173800 000000 FILL TO 1000
565 001338* 000000 000000 ;173802 000000 FILL TO 1000
566 001340* 000000 000000 ;173804 000000 FILL TO 1000
567 001342* 000000 000000 ;173806 000000 FILL TO 1000
568 001344* 000000 000000 ;173808 000000 FILL TO 1000
569 001346* 000000 000000 ;173810 000000 FILL TO 1000
570 001348* 000000 000000 ;173812 000000 FILL TO 1000
571 001350* 000000 000000 ;173814 000000 FILL TO 1000
572 001352* 000000 000000 ;173816 000000 FILL TO 1000
573 001354* 000000 000000 ;173818 000000 FILL TO 1000
574 001356* 000000 000000 ;173820 000000 FILL TO 1000
575 001358* 000000 000000 ;173822 000000 FILL TO 1000
576 001360* 000000 000000 ;173824 000000 FILL TO 1000
577 001362* 000000 000000 ;173826 000000 FILL TO 1000
578 001364* 000000 000000 ;173828 000000 FILL TO 1000
579 001366* 000000 000000 ;173830 000000 FILL TO 1000
580 001368* 000000 000000 ;173832 000000 FILL TO 1000
581 001370* 000000 000000 ;173834 000000 FILL TO 1000
582 001372* 000000 000000 ;173836 000000 FILL TO 1000
583 001374* 000000 000000 ;173838 000000 FILL TO 1000
584 001376* 000000 000000 ;173840 000000 FILL TO 1000
585 001378* 000000 000000 ;173842 000000 FILL TO 1000
586 001380* 000000 000000 ;173844 000000 FILL TO 1000
587 001382* 000000 000000 ;173846 000000 FILL TO 1000
588 001384* 000000 000000 ;173848 000000 FILL TO 1000
589 001386* 000000 000000 ;173850 000000 FILL TO 1000
590 001388* 000000 000000 ;173852 000000 FILL TO 1000
591 001390* 000000 000000 ;173854 000000 FILL TO 1000
592 001392* 000000 000000 ;173856 000000 FILL TO 1000
593 001394* 000000 000000 ;173858 000000 FILL TO 1000
594 001396* 000000 000000 ;173860 000000 FILL TO 1000
595 001398* 000000 000000 ;173862 000000 FILL TO 1000
596 001400* 000000 000000 ;173864 000000 FILL TO 1000
597 001402* 000000 000000 ;173866 000000 FILL TO 1000
598 001404* 000000 000000 ;173868 000000 FILL TO 1000
599 001406* 000000 000000 ;173870 000000 FILL TO 1000
600 001408* 000000 000000 ;173872 000000 FILL TO 1000

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

ACSP	000102R	185#	
ADDR	000006R	151#	204
ADDR2=	001000R	203#	
AGAIN	000230R	207#	225
ASB	000106R	189#	233*
ASSTAT	000104R	187#	
AWAS	000110R	190#	232*
BEGYN	000000R	140#	223
BIT0	= 000001	203#	235
BIT1	= 000000	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
BMTAB	000336R	208#	245#
BREAKS=	104400	203#	
BRI	000012R	153#	
BR2	000013R	154#	
BTDOS	= 104421	203#	
CDATA=	104412	203#	
CONFIG	000056R	173#	
CSRI	000100R	183#	
DITCK=	104411	203#	
DATER=	104404	203#	235
DVID1	000014R	155#	
ENDIT=	104413	203#	223
ENDS	= 104410	203#	
ERR7YP	000106R	188#	
EXITS	= 104400	203#	
GETPAS=	104415	203#	
GWBDF=	104414	203#	
HRDCHT	000044R	168#	
HRDFRS=	104405	203#	
HRDPAS	000050R	170#	
ICONT	000036R	165#	
ICOUNT	000040R	162#	
IDNUM	000122R	195#	
INIT	000030R	162#	
IWTS	000120R	194#	
MAP22=	104416	203#	
MODNAM	000000R	149#	
MODSP	000224R	163#	201#
MSGMS	= 104403	203#	
MSGSS	= 104402	203#	
MSG	= 104401	203#	

NULL	=	000000	203#																		
OPEN	=	000000	150#	156																	
			185#	187	157	158	159	176	177	178	179	180	181	182	183						
			203#		189	190	192	193	194	203#											
OTQAS	=	104420	203#																		
PASCNT	=	000034R	164#																		
PIRGS	=	000004	203#																		
POPS	=	005726	203#																		
POPS2	=	022626	203#																		
PRTV	=	000000	153#	154																	
PRTV0	=	000000	203#																		
PRTV1	=	000040	203#																		
PRTV2	=	000100	203#																		
PRTV3	=	000140	203#																		
PRTV4	=	000200	203#																		
PRTV5	=	000240	203#																		
PRTV6	=	000300	203#																		
PRTV7	=	000340	203#																		
PS	=	177776	203#																		
PSW	=	177776	203#																		
PUSH	=	005746	203#																		
PUSH2	=	024646	203#																		
RANUS	=	104417	203#																		
RANUM	=	00054R	177#																		
RESTR	=	000230R	191#	206#																	
RES1	=	000056R	174#																		
RES2	=	000060R	175#																		
RSTRT	=	00011R	191#																		
SBADR	=	000102R	184#	230*																	
SDFCNT	=	000042R	167#																		
SDFERS	=	104406	203#																		
SDFPAS	=	000046R	163#																		
SPOINT	=	000032R	163#																		
SPSTZ	=	000040	1#	196																	
SR1	=	000016R	156#																		
SR2	=	000020R	157#																		
SR3	=	000022R	158#																		
SR4	=	000024R	159#																		
START	=	000224R	162#	204#																	
SV0	=	000062R	176#																		
SVR1	=	000064R	177#																		
SVR2	=	000066R	178#																		
SVR3	=	000070R	179#																		
SVR4	=	000072R	180#																		
SVR5	=	000074R	181#																		
SVR6	=	000076R	182#																		
SVSCNT	=	000052R	171#																		
TABEND	=	001336R	220#	563#																	
TRPDPD	=	000022	203#																		
VECTOR	=	000010R	152#																		
WASADR	=	000104R	186#	231*																	
WDFR	=	000116R	193#																		
WDFD	=	000114R	192#																		
XFLAG	=	000005R	150#																		

. ABS. 000000 000
001340 001

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0
XBMPFB,XMPFB/SOL/CRF:SYM=DDXCOM,XMPFB0
RUN-TIME: 11.2 SECONDS
RUN-TIME RATIO: 21/3=5.5
CORE USED: 7K (13 PAGES)