

.REM %

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48

IDENTIFICATION

PRODUCT CODE: AC-T207B-MC
PRODUCT NAME: CJM9AB0 11/24 ROM M9312
PRODUCT DATE: SEPTEMBER,1982
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: J. HAMEL

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 DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C): 1982,1983 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	DECX/11

49
50
51
52
53
54
55
56
57
58

HISTORY

CJM9AA - DOCUMENT RELEASED APRIL 1982
CJM9AB - CHANGES MADE SO DIAGNOSTIC WOULD INTERFACE CLEANLY WITH
CONSOLE EMULATOR, AND FIXED TROUBLE OF LOSING LAST TWO
CHARACTERS OF MEMORY SIZE PRINTOUT.

TABLE OF CONTENTS

59	
60	
61	
62	
63	
64	
65	
66	1.0 GENERAL PROGRAM INFORMATION
67	1.1 ABSTRACT
68	1.2 SYSTEM REQUIREMENTS
69	1.3 RELATED DOCUMENTS AND STANDARDS
70	1.4 DIAGNOSTIC HIERARCHY PREREQUISITES
71	1.5 ASSUMPTIONS
72	
73	2.0 OPERATING INSTRUCTIONS
74	2.1 LOADING AND STARTING PROCEDURE
75	2.2 PROGRAM OPTIONS
76	2.3 EXECUTION TIMES
77	
78	3.0 ERROR INFORMATION
79	3.1 ERROR REPORTING PROCEDURES
80	3.2 ERROR HALTS
81	
82	4.0 PERFORMANCE AND PROGRESS REPORTS
83	4.1 PERFORMANCE REPORTS
84	4.2 PROGRESS REPORTS
85	
86	5.0 DEVICE INFORMATION TABLES
87	
88	6.0 PROGRAM DESCRIPTION
89	6.1 PROGRAM EXECUTION CHARACTERISTICS
90	6.2 SUBTEST SUMMARIES
91	6.3 SPECIAL SUBROUTINE DESCRIPTION
92	
93	
94	7.0 HISTORY
95	
96	8.0 LISTING

1.5 ASSUMPTIONS

THE DIAGNOSTIC ASSUMES PROPER SETUP OF THE DIP SWITCHPAK ON THE M9312 BOOTSTRAP MODULE (SEE M9312 USER'S MANUAL AND TABLE BELOW FOR THIS INFORMATION) AND PRESENCE OF A BOOT ROM IN THE M9312.

BOOTSTRAP:	DIAGNOSTICS	S1 (1-10)		
		FIRST DEVICE (ALL ROMS)	VIRTUAL ADDRESS	SWITCHPAK S1 SWITCHES ON
ODT	NO YES	2004 2006	165004 165006	1,9 1,9,10
DEVICE ROM #1	NO YES	0004 0006	173004 173006	9 9,10
DEVICE ROM #2	NO YES	0204 0206	173204 173206	4,9 4,9,10
DEVICE ROM #3	NO YES	0404 0406	173404 173406	3,9 3,9,10
DEVICE ROM #4	NO YES	0604 0606	173604 173606	3,4,9 3,4,9,10

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURE

IF DIAGNOSTIC IS SELECTED BY M9312 SWITCHPAK, IT WILL BE RUN ON POWER UP, BUT CAN ALSO SELECTED FROM MICRO-ODT BY COMMAND:

165000G

TO BOOT A DEVICE ROM DIRECTLY, USE THE VIRTUAL ADDRESS COLUMN OF THE ABOVE TABLE. FOR EXAMPLE, TO BOOT DEVICE ROM #3 WITH DIAGNOSTICS, USE 173406G (FROM MICRO-ODT).

2.2 PROGRAM OPTIONS

NONE

151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206

CJM9
CJM9
.SSL
.STR
.STY
.STY
.STY
.STY
\$40
.117
. AB
ERR
CJM
RUN
RUN
COR

207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262

2.3 EXECUTION TIMES

- A. FIRST PASS (QV)
A PASS OF THIS CODE TAKES APPROXIMATELY 6 SECONDS TO COMPLETE INCLUDING THE PRINTING OF THE MEMORY SIZE. THIS TIMING IS BASED ON A 11/24 CPU WITH 128K WORD OF MOS MEMORY.
- B. LONGEST TEST
THE LONGEST SINGLE TEST IS THE MEMORY TEST WHICH TAKES APPROXIMATELY 5 SECONDS PER 128K WORDS OF MEMORY.
- C. ADDITIONAL TIME FOR UNITS
APPROXIMATELY 5 SECONDS IS ADDED TO THE TEST TIME FOR EVERY ADDITIONAL 128K WORDS OF MEMORY. THIS TIMING IS BASED ON AN 11/24 WITH MOS MEMORY.
SAMPLE TEST TIMES BASED ON ABOVE FIGURES:
 - 128KW= 6 SECONDS
 - 256KW= 11 SECONDS
 - 512KW= 21 SECONDS
 - 1024KW= 41 SECONDS
 - 1536KW= 61 SECONDS
 - 1920KW= 76 SECONDS
- D. FULL PASS TIME (ITERATIONS)
THIS PROGRAM DOES NOT DO ANY ITERATIONS ON ANY OF THE TESTS AND IN FACT ONLY MAKES ONE PASS THRU THE CODE FOR EACH START.

3.0 ERROR INFORMATION

3.1 ERROR REPORTING PROCEDURES

SINCE THIS DIAGNOSTIC IS A GO/NOGO, LOW-LEVEL TEST, NO ERROR REPORTING, AS SUCH, IS IMPLEMENTED; HOWEVER, IF THE MICRO-ODT AND CONSOLE TERMINAL ARE OPERATIONAL THE ERROR HALT ADDRESS+2 WILL BE TYPED FOR THE OPERATOR.

3.2 ERROR HALTS

BADADD = 165144 THIS ERROR IS CAUSED BY TRAPPING TO LOCATION 4 AT ANY TIME PRIOR TO EXECUTING THE MEMORY TEST ON THE FIRST 4K OF MEMORY. THE PROGRAM DOES ACCESSES TO SOME OF THE MEMORY MANAGEMENT REGISTERS DURING THIS TIME. IT MAY BE HELPFUL TO EXAMINE THE STACK BUT SINCE THE PROGRAM HAS NOT SET IT UP THE INFORMATION RECIEVED MAY NOT BE VALID.

CPUERR = 165146 THIS ERROR INDICATES A FAILURE WITH EITHER THE BASE INSTRUCTION SET OR THE EIS INSTRUCTION SET. FIRST SUSPECT THE DCF11-A HYBRID OR THE CPU BOARD.

MEMERR = 165550 THIS ERROR INDICATES A MEMORY SYSTEM FAILURE. FIRST SUSPECT THE MEMORY THEN THE KTF11-A. TO LOCATE THE FAILING BANK DIVIDE THE CONTENTS OF PAR0 (1772342) BY 200(8) THEN MULTIPLY BY 4.

SLUERR = 165702 THIS HALT INDICATES A DATA ERROR IN THE CONSOLE SLU. THE GOOD DATA IS IN R2.

263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS

NONE

4.2 PROGRESS REPORTS

THE PROGRAM REPORTS ITS PROGRESS USING TWO LEDS LOCATED ON THE 11/24 CPU BOARD. THEY ARE LOCATED ON THE HANDLE EDGE OF THE BOARD ALONG WITH THE RUN LIGHT. THEY ARE DRIVEN THROUGH BITS 0 AND 1 OF THE DISPLAY REGISTER WITH BIT 0 DRIVING THE LED FURTHEST FROM THE FRONT OF THE BOX AND BIT 1 DRIVING THE LED IN THE CENTER OF THE THREE LIGHTS. THE LIGHT CLOSEST TO THE FRONT OF THE BOX IS THE RUN LIGHT. THE PROGRAM FIRST WRITES AN OCTAL 3 TO THE REGISTER (TURNING BOTH LEDS ON) TO SIGNAL THE START OF THE CPU TEST. THE COUNT IS DECREMENTED BY ONE AT THE START OF EACH OF THE NEXT THREE TEST SECTIONS.

LIGHT COUNT	LAST TEST COMPLETED
3	NONE-SUCCESSFUL ENTRY TO PROGRAM
2	CPU TEST
1	MEMORY TEST
0	SLU TEST

5.0 DEVICE INFORMATION TABLES

POOL (LOC 165102) CONTAINS DATA USED BY DOUBLE OPERAND SECTION.

6.0 PROGRAM DESCRIPTION

6.1 PROGRAM EXECUTION CHARACTERISTICS

PROPER EXECUTION OF THE DIAGNOSTIC RESULTS IN PRINTOUT OF THE MEMORY SIZE AND BOOTING A PERIPHERAL OR ENTERING MICRO-ODT (DEPENDING ON M9312 SWITCH SETTINGS).

6.2 SUBTEST SUMMARIES

A. CPUTST - THIS SUBTEST VERIFIES THE PDP-11 INSTRUCTION SET, INCLUDING EIS, FOR BOTH WORD AND BYTE FORMATS. IT CONSISTS OF FIVE SECTIONS - SINGLE OPERAND (DESTINATION MODE 0), DOUBLE OPERAND (ALL SOURCE MODES, DESTINATION MODE 0), CONDITIONAL BRANCHES, BYTE INSTRUCTIONS (ALL DESTINATION MODES), AND JSR/RTS WITH EIS.

315
316
317 B. MEMTST - THIS SUBTEST CHECKS ALL OF MEMORY IN 4K PAGES USING
318 THE MMU, DETERMINES MEMORY SIZE, AND CONSTRUCTS A PHYSICAL
319 ADDRESS FROM THE VIRTUAL ADDRESS WHICH CAUSED A TIMEOUT TRAP.
320 LOOPING THROUGH EACH 4K PAGE IS CONTROLLED BY THE 1\$ LOOP, AND;
321 WITHIN THIS LOOP, THE 2\$ LOOP LOADS EACH LOCATION WITH ITS
322 ADDRESS, THE 3\$ LOOP CHECKS THE DATA AND COMPLEMENTS IT, THE 4\$
323 LOOP ADDS THE CONTENTS OF EACH LOCATION TO ITS COMPLEMENT AND
324 INCREMENTS THE RESULT TO PRODUCE ZERO. SECTION TIMEOUT TURNS OFF
325 MEMORY MANAGEMENT AND BUILDS A PHYSICAL ADDRESS IN R0 AND R1.
326

327 NOTE: BECAUSE OF SPACE LIMITATIONS THERE
328 IS A KNOWN FLAW IN THE MEMORY
329 SIZING ROUTINE. IF MAXIMUM MEMORY
330 IS CONFIGURED ON THE SYSTEM(1920KW)
331 THE SIZING ROUTINE WILL REPORT A
332 MEMORY SYSTEM SIZE OF 2044KW. THIS
333 IS BECAUSE OF THE UNIBUS MAP WILL
334 MAP THE LAST 124K OF VIRTUAL
335 ADDRESS SPACE TO THE LOWER 124K
336 OF MAIN MEMORY.
337

338
339 C. SLUTST - THIS SUBTEST PLACES SLU1 IN MAINTENANCE MODE (SERIAL
340 OUT OF UART TIED TO SERIAL IN OF UART) AND TESTS THAT ALL 8-BIT
341 PATTERNS CAN BE TRANSMITTED AND RECEIVED. SECTION PRINT PRINTS
342 THE MEMORY SIZE CALCULATED BY MEMTST, AND TRANSFERS CONTROL BACK
343 TO ODT OR THE APPLICABLE BOOT ROM.
344
345

346
347 6.3 SPECIAL SUBROUTINE DESCRIPTION
348

349 THE PRINT ROUTINE TYPES THE LAST 22-BIT MEMORY ADDRESS+2
350 FOUND BY THE MEMORY SIZE ROUTINE. THE 16 HIGH ORDER BITS
351 ARE SAVED IN R0 AND THE 6 LOW ORDER BITS ARE SAVED IN R1.
352

353 7.0 HISTORY
354

355 CJM9AA - DOCUMENT RELEASED APRIL 1982
356 CJM9AB - CHANGES MADE SO DIAGNOSTIC WOULD INTERFACE CLEANLY WITH
357 CONSOLE EMULATOR, AND FIXED TROUBLE OF LOSING LAST TWO
358 CHARACTERS OF MEMORY SIZE PRINTOUT.
359

360
361 8.0 LISTING
362


```
363
364
365
366 .TITLE CJM9AB
367 .SBTTL ROM AREA 165000-165776
368 .SBTTL GO/NOGO MINIMUM DIAGNOSTIC
369 .TITLE CJM9ABO 11/24 ROM M9312
370 :*COPYRIGHT (C) SEPTEMBER,1982
371 :*DIGITAL EQUIPMENT CORP.
372 :*MAYNARD, MASS. 01754
373 :*
374 :*PROGRAM BY D.SOBIK
375 :*
376 :*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
377 :*PACKAGE (MAINDEC-11-DZQAC-C5), JAN, 1981.
378 000001 $TN=1
379 160000 $SWR=160000 ;:HALT ON ERROR, LOOP ON TEST, INHIBIT ERROR TYP0UT
380
381 .SBTTL REGISTER DEFINITIONS
382 177570 SWR= 177570 ;SWITCH REGISTER (DIAGNOSTIC LIGHTS)
383 177776 PS= 177776 ;PROCESSOR STATUS WORD
384 :SLU1 REGISTERS
385 177560 RCSR= 177560 ;RECEIVER CSR
386 177562 RBUF= 177562 ;RECEIVER BUFFER
387 177564 XCSR= 177564 ;TRANSMITTER CSP
388 177566 XBUF= 177566 ;TRANSMITTER BUFFER
389
390 :MMU REGISTERS
391 172340 PAR0= 172340 ;KERNAL PAGE ADDRESS REGISTERS
392 172342 PAR1= 172342
393 172356 PAR7= 172356
394 172300 PDR0= 172300 ;KERNAL PAGE DESCRIPTOR REGISTERS
395 172302 PDR1= 172302
396 172316 PDR7= 172316
397 177640 UPAR0= 177640 ;USER PAGE ADDRESS REGISTERS
398 177642 UPAR1= 177642
399 177572 SR0= 177572 ;STATUS REGISTER 0
400 172516 SR3= 172516 ;STATUS REGISTER 3 (22-BIT)
```

```

1      165000 000177 006020      .=165000
2      165004 000000      START: JMP @173024      ;TRANSFER TO SELECTED BOOT ROM OR ODT
3                                          HALT      ;ENTRY POINT FOR ODT, NO DIAGNOSTICS
4                                          ;ENTRY POINT FOR ODT WITH DIAGNOSTICS
5      165006 012704 165002      MOV #165002,R4      ;SET UP RETURN ADDRESS-2 IN R4
6
7      .SBTTL CPU TEST
8      ;*****
9
10     ; BASIC CPU TEST
11     ;*****
12     165012 012737 000003 177570 CPUTST: MOV #3, @#SWR      ;LIGHTS = 3, INDICATING CPUTST
13     165020 012737 165144 000004      MOV #BADADD,@#4      ;SET UP TIME OUT VECTOR INCASE OF TRAP
14     165026 005037 000006      CLR @#6      ;CLEAR PRIORITY OF TRAP ROUTINE
15     165032 010037 177640      MOV R0, @#UPARO      ;WE ARE USING THE UPAR'S HERE SIMPLY BECAUSE
16     165036 010137 177642      MOV R1, @#UPAR1      ;THEY ARE AVAILABLE UNUSED INTERNAL REGS.
17                                          ;WE ARE STORING PARAMETERS FROM THE BOOT ROM
18
19     ;R1 CONTENTS
20     165042 005001      CLR R1      ;000000      N Z V C
21     165044 005201      INC R1      ;1      0 0 0 0
22     165046 005101      COM R1      ;177776      1 0 0 1
23     165050 006201      ASR R1      ;177777      1 0 1 0
24     165052 006301      ASL R1      ;177776      1 0 0 1
25     165054 006001      ROR R1      ;177777      1 0 1 0
26     165056 005701      TST R1      ;177777      1 0 0 0
27     165060 005401      NEG R1      ;1      0 0 0 1
28     165062 005301      DEC R1      ;0      0 1 0 1
29     165064 005601      SBC R1      ;177777      1 0 0 1
30     165066 005501      ADC R1      ;0      0 1 0 1
31     165070 001026      BNE CPUERR      ;ERROR IF NOT ZERO
32
33     ; SECTION FOR DOUBLE OPERAND, ALL SOURCE MODES, DEST MODE 0
34
35     165072 012702 165126      MOV #POOL, R2      ;SET UP ADDRESS OF DATA TABLE
36     165076 011201      MOV (R2), R1      ;R1/POOL, SMODE 1
37     165100 022201      CMP (R2)+, R1      ;DATA CORRECT? SMODE 2
38     165102 001021      BNE CPUERR
39     165104 063201      ADD @-(R2)+, R1      ;R1/POOL + 1, SMODE 3
40     165106 165201      SUB @-(R2), R1      ;R1/POOL, SMODE 5
41     165110 044201      BIC -(R2), R1      ;R1/0, SMODE 4
42     165112 056201 000004      BIS 4(R2), R1      ;R1/177777, SMODE 6
43     165116 037201 000006      BIT @6(R2), R1      ;RESULT IS 177777, SMODE 7
44     165122 001411      BEQ CPUERR
45     165124 000411      BR CONT      ;BRANCH AROUND DATA TABLE
46
47     POOL: .WORD POOL
48     165126 165126      .WORD DATA1
49     165130 165136      .WORD 177777
50     DATA2: .WORD DATA2
51     165134 165132      .WORD 1
52     DATA1: .WORD 500
53     165136 000001      .WORD 500
54     165140 000500      .WORD 500
55     165142 000500      .WORD 500
56     165144 000000      BADADD: HALT
57     165146 000000      CPUERR: HALT
    
```

; CHECK CONDITIONAL BRANCHES

165150	000277	CONT: SCC		:SET ALL CONDITION CODES
165152	001375	BNE	CPUERR	:BR IF Z=0
165154	100374	BPL	CPUERR	:... .. N=0
165156	102373	BVC	CPUERR	:... .. V=0
165160	103372	BCC	CPUERR	:... .. C=0
165162	002771	BLT	CPUERR	:... .. N XOR V=1
165164	003370	BGT	CPUERR	:... .. Z OR (N XOR V)=0
165166	101367	BHI	CPUERR	:... .. C OR Z=0
165170	000257	CCC		:CLR ALL CONDITION CODES
165172	001765	BEQ	CPUERR	:BR Z=1
165174	100764	BMI	CPUERR	:BR N=1
165176	102763	BVS	CPUERR	:BR V=1
165200	103762	BCS	CPUERR	:BR C=1
165202	003761	BLE	CPUERR	:BR Z OR (N XOR V)=1
165204	101760	BLOS	CPUERR	:BR C OR Z=1
165206	000270	SEN		:N=1
165210	002356	BGE	CPUERR	:BR N XOR V=0

; CHECK BYTE INSTRUCTIONS, ALL DEST MODES

				:R1 CONTENTS	N	Z	V	C
165212	105001	CLRB	R1	:177400	0	1	0	0
165214	105201	INCB	R1	:177401	0	0	0	0
165216	105101	COMB	R1	:177776	1	0	0	1
165220	106201	ASRB	R1	:177777	1	0	1	0
165222	106301	ASLB	R1	:177776	1	0	0	1
165224	106001	RORB	R1	:177777	1	0	1	0
165226	105401	NEGB	R1	:177401	0	0	0	1
165230	105301	DECB	R1	:177400	0	1	0	1
165232	105601	SBCB	R1	:177777	1	0	0	1
165234	106101	ROLB	R1	:177777	1	0	0	1
165236	105501	ADCB	R1	:177400	0	1	0	1
165240	000301	SWAB	R1	:000377	1	0	0	0
165242	012703	MOV	#500, R3	:SETUP FOR DMODE TESTING				
165246	105063	CLRB	1(R3)	:CLR LOC 501, DMODE 6				
165252	110113	MOVB	R1, (R3)	:500/000 377, DMODE 1				
165254	120123	CMPB	R1, (R3)+	:SHOULD COMPARE, DMODE 2				
165256	001333	BNE	CPUERR					
165260	105143	COMB	-(R3)	:500/000 000, DMODE 4				
165262	012703	MOV	#DOUBLE, R3	:SETUP DEFERRED DMODES				
165266	153733	BISB	@#DATA1, @ (R3)+	:500/1, DMODE 3				
165272	143753	BICB	@#DATA2, @-(R3)	:500/0, DMODE 5				
165276	130173	BITB	R1, @2(R3)	:Z=1, DMODE 7				
165302	001321	BNE	CPUERR					

; CHECK JSR/RTS AND EIS

165304	012706	000500	MOV	#500,	SP		;SET UP STACK
165310	004767	000004	JSR	PC,	EISTST		
165314	000714		BR	CPUERR			
165316	000421		BR	MEMTST			;EXIT CPU TEST
165320	012701	000040	EISTST: MOV	#40,	R1		:R1/ 40
165324	070127	000010	MUL	#10,	R1		:R1/ 400
165330	006700		SXT	RO			:RO/ 0
165332	072127	000006	ASH	#6,	R1		:R1/ 40000
165336	073127	000071	ASHC	#71,	R1		:R1/ 200
165342	071027	000200	DIV	#200,	RO		:RO/1 R1/0
165346	005201		INC	R1			:R1/1
165350	074001		XOR	RO,	R1		:R1/0
165352	001275		BNE	CPUERR			
165354	062716	000002	ADD	#2,	(SP)		;FIX RETURN ADDRESS TO BYPASS ERROR BR
165360	000207		RTS	PC			;EXIT TO MEMORY TEST

::*****

.SBTTL MEMTST

::*****

:+
:
:
:
:
:-

THIS TEST SIZES MEMORY AND CHECKS MEMORY FROM LOC 1000 TO END OF MEMORY BY WRITING IN EACH LOCATION THE ADDRESS OF THE LOCATION AND COMPARING THE LOCATION AND ITS CONTENTS. THE PROCEDURE IS REPEATED USING THE COMPLEMENTS OF THE ADDRESS IN EACH LOCATION. MEMORY IS CLEARED ON EXIT FROM THIS TEST IN 4K BLOCKS; THAT IS, IF 12K IS PRESENT, 120K GETS CLEARED, THE REST HAS THE ADDRESS WRITTEN IN IT.

::*****

65362	012737	000012	177566	MEMTST: MOV	#12,@#XBUF	:OUTPUT A LINE FEED .
65370	012702	000002		MOV	#2, R2	:WORD INCREMENT
65374	010237	177570		MOV	R2, @#SWR	:LIGHTS = 2, INDICATING MEMTST
65400	005037	172340		CLR	@#PAR0	:MAP VECTOR SPACE TO PAR0
65404	005037	172342		CLR	@#PAR1	:PAR1 IS MOVABLE WINDOW INTO MEMORY
55410	012737	177600	172356	MOV	#177600,@#PAR7	:MAP PAR7 TO I/O PAGE
65416	012701	077406		MOV	#77406,R1	:SET UP R1 AS CONSTANT TO SETUP PDR'S
65422	010137	172300		MOV	R1,@#PDR0	:PAGE0=4K R/W
65426	010137	172302		MOV	R1,@#PDR1	:PAGE1=4K R/W
65432	010137	172316		MOV	R1,@#PDR7	:PAGE7=4K R/W
65436	005037	177776		CLR	@#PS	:ENSURE KERNEL MODE
65442	005237	177572		INC	@#SRO	:TURN ON KT
65446	012737	000020	172516	MOV	#20, @#SR3	:SET UP FOR 22-BIT RELOCATION
65454	012737	165552	000004	1\$: MOV	#TIMOUT,@#4	:SETUP TIMEOUT VECTOR FOR MEMORY SIZE
65462	012703	020000		MOV	#20000, R3	:START AT VIRTUAL ZERO
65466	012705	010000		MOV	#10000, R5	:PAGE LENGTH = 4K
65472	010301			MOV	R3, R1	:WORKING COPY OF FIRST ADDRESS
65474	010500			MOV	R5, R0	:COPY PAGE LENGTH
65476	010111			2\$: MOV	R1, (R1)	:WRITE ADDRESS INTO LOCATION
65500	060201			ADD	R2, R1	:INCREMENT ADDRESS
65502	077003			SOB	R0, 2\$:LOOP TILL PAGE DONE
65504	010301			MOV	R3, R1	:RESTORE INITIAL CONDITIONS FOR
65506	010500			MOV	R5, R0	: DATA CHECK AND COMPLEMENTING
65510	020111			3\$: CMP	R1, (R1)	:GOOD DATA?
65512	001014			BNE	MEMERR	:NO, HALT
65514	005121			COM	(R1)+	:COM DATA AND INC ADDRESS
65516	077004			SOB	R0, 3\$:LOOP TILL PAGE DONE
65520	010301			MOV	R3, R1	:START AGAIN TO TEST COM DATA
65522	010500			MOV	R5, R0	
65524	060111			4\$: ADD	R1, (R1)	:RESULT SHOULD BE -1
65526	005221			INC	(R1)+	:RESULT=0 AND SETUP NEXT ADDRESS
65530	001005			BNE	MEMERR	
65532	077004			SOB	R0, 4\$:FINISH THE PAGE
65534	062737	000200	172342	ADD	#200, @#PAR1	:RELOC TO A NEW PAGE
65542	000744			BR	1\$:CHECK OUT A NEW PAGE
65544	005037	177572		MEMERR: CLR	@#SRO	:TURN OFF KT
65550	000000			HALT		:BAD MEMORY

```

579 165552 005037 177572          TIMEOUT: CLR    @#SRO          :TURN OFF KT
580 165556 005037 172516          CLR    @#SR3          :RESTORE 18-BIT RELOCATION
581 165567 000402          BR     1$             :BRANCH AROUND ENTRY POINT
582 165564 000167 177222          JMP    CPUTST         :BOOT ROM ENTRY POINT
583 165570 012737 000006 000004 1$:  MOV    #6,    @#4     :TIMEOUT TRAPCATCHER
584 165576 160301          SUB    R3,    R1     :GET RID OF VIRTUAL OFFSET
585 165600 005000          CLR    R0          :CLR HIGH HALF OF DOUBLEWORD
586 165602 073027 000012          ASHC   #12,   R0     :LSH 10, TO BUILD PHYSICAL ADDRESS
587 165606 063700 172342          ADD   @#PAR1, R0     :ADD PAGE BASE ADDRESS
588 165612 073027 177776          ASHC   #-2,   R0     :ACCJUNT FOR 1-BIT MSD
589          .SBTTL  SLU1 TEST
590
591          ;:*****
592
593          :      CHECK SLU1 VIA MAINTENANCE MODE
594
595          ;:*****
596
597 165616 012737 000001 177570  SLUTST: MOV    #1,    @#SWR    :LIGHTS = 1, INDICATING SLUTST
598 165624 005002          CLR    R2          :FIRST ASCII CODE TO BE CHECKED
599 165626 012737 000004 177564  MOV    #4,    @#XCSR  :ENABLE MAINTENANCE MODE
600 165634 105737 177564          2$:  TSTB   @#XCSR    :TRANSMITTER READY?
601 165640 100375          BPL    2$          :NO, WAIT FOR READY
602 165642 110237 177566          MOVB   R2,    @#XBUF  :TRANSMIT CHAR
603 165646 105737 177560          3$:  TSTB   @#RCSR    :DATA RECEIVED?
604 165652 100375          BPL    3$          :NO
605 165654 120237 177562          CMPB   R2,    @#RBUF  :DID DATA LOOP AROUND CORRECTLY?
606 165660 001006          BNE    SLUERR      :NO
607 165662 005202          INC    R2          :SET UP NEXT PATTERN
608 165664 105702          TSTB   R2          :DONE ALL PATTERNS?
609 165666 001362          BNE    2$          :NO, CONTINUE
610 165670 005037 177564          CLR    @#XCSR      :EXIT MAINTENANCE MODE
611 165674 000403          BR     PRINT        :PRINT LAST ADDRESS
612 165676 005037 177564          SLUERR: CLR   @#XCSR  :EXIT MAINT MODE
613 165702 000000          HALT
    
```

```

614
615
616
617
618
619
620 165704 005037 177570
621 165710 012705 000010
622 165714 012703 000003
623 165720 005002
624 165722 073027 000001
625 165726 006102
626 165730 077304
627 165732 062702 000060
628 165736 105737 177564
629 165742 100375
630 165744 110237 177566
631 165750 077517
632 165752 012705 120000
633 165756 077501
634
635
636
637
638
639
640
641 165760 013700 177640
642 165764 013701 177642
643 165770 000164 000002
644 165774 042060
645 165776 051560
646
647 165000
    
```

```

;*****
; PRINT MEMORY SIZE (LAST ADDRESS + 2) AND EXIT
;*****
PRINT: CLR @#SWR ;LIGHTS = 0, ALL TESTS PASSED
MOV #10, R5 ;PRINT 8. DIGITS
1$: MOV #3, R3 ;SHIFT COUNT FOR DIGIT ASSEMBLY
CLR R2 ;CLR DIGIT ASSEMBLY AREA
2$: ASHC #1, R0 ;PUT BIT IN C-BIT
ROL R2 ;MOVE C-BIT TO R2
SOB R3, 2$ ;BUILD A DIGIT IN R2
ADD #0, R2 ;CNVRT TO ASCII
3$: TSTB @#XCSR ;PRINTER READY?
BPL 3$
MOVB R2, @#XBUF ;PRINT IT
SOB R5, 1$ ;PRINT 8. DIGITS
MOV #120000, R5 ;GIVE PRINTER TIME TO PRINT
4$: SOB R5, 4$ ;OUT THE LAST TWO DIGITS

;////////////////////////////////////
; EXIT TO ODT OR TO BOOT ROM
;////////////////////////////////////

MOV @#UPAR0, R0 ;RESTORE BOOT ROM PARAMETERS
MOV @#UPAR1, R1
JMP 2(R4) ;RETURN TO BOOT ROM OR ODT
.CRC11 "OD" ;#4 CPU ROM FOR M9312
CRC: .WORD 051560 ;CRC CHECKWORD FOR ROM

.END START
    
```


CJM9AB0 11/24 ROM M9312 MACY11 30(1046) 15-SEP-82 17:04 PAGE 21 F 2
CJM9AB.P11 15-SEP-82 17:03 CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0018

.\$SUPR 1#
.\$STRAP 1#
.\$STYPB 1#
.\$STYPD 1#
.\$STYPE 1#
.\$STYPO 1#
.\$40CA 1#
.\$1170 1#

. ABS. 166000 000

ERRORS DETECTED: 0

CJM9AB,CJM9AB/CRF/NL:TOC/SOL=SYSMAC.SML,CJM9AB.P11
RUN-TIME: 56.2 SECONDS
RUN-TIME RATIO: 39/12=3.1
CORE USED: 34K (68 PAGES)