

TS03, TU10

TM A B 11 INSTR TST
CZTMAIO

AH 9395I MC
FICHE 1 OF 1

MAR 98
COPYR
MADE IN USA



Table with multiple columns and rows of data, likely a technical specification or test results table. The content is too faint to transcribe accurately but appears to contain various numerical and alphanumeric entries.



.REM X

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

IDENTIFICATION

PRODUCT CODE: AC-9393I-MC
PRODUCT TITLE: CZTMAIO TM,A,B-11 INSTR TST
PROGRAM DATE: SEPTEMBER 1979
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: JOHN RODENHISER

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.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.

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

COPYRIGHT (C) 1971, 1979 BY DIGITAL EQUIPMENT CORPORATION

38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91

1. ABSTRACT

THE TM,A,,B-11 INSTRUCTION TEST CONTAINS A SERIES OF BASIC TESTS THAT CHECK REGISTERS FOR PROPER OPERATION WHILE NOT INVOLVING TAPE MOTION, ALL TAPE MOTION FUNCTIONS, DATA TRANSFERS, EXTENDED MEMORY, AND MANUAL INTERVENTION TESTS OF THE TU10 OR TS03 TRANSPORTS. ***MANUAL INTERVENTION TESTS ARE SKIPPED IN CHAIN MODE***

2. REQUIREMENTS

2.1 EQUIPMENT

PDP-11 WITH TM,A.B-11 CONTROL UNIT AND 1 TS03,TU10,N,W TAPE UNIT.

2.2 STORAGE

2.2.1 PROGRAM STORAGE

THE ROUTINE REQUIRES 4K OF MEMORY.

3. LOADING PROCEDURE

3.1 METHOD

A. PROCEDURE FOR NORMAL BINARY TAPES SHOULD BE FOLLOWED.

B. PROGRAM IS LOADABLE AND CHAINABLE IN 8K OF MEMORY.

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

STARTING AT LOC. 200 ALL SWITCHES SHOULD BE DOWN OR ZERO.
SOFTWARE SWITCH REGISTER IS LOCATED AT LOC. 176 IF NEEDED.

4.2 STARTING ADDRESS

200

4.3 PROGRAM AND/OR OPERATOR ACTION

1. LOAD PROGRAM INTO MEMORY.
2. PLACE ONE TAPE UNIT, ON-LINE, AT LOAD POINT (BOT),WRITE ENABLED, UNIT 0 SELEC
3. LOAD STARTING ADDRESS
4. START PROGRAM
PROGRAM WILL TYPE 'SET SWO=1 IF 7 CHANNEL'.(IF NOT CHAIN MODE)
CHAIN MODE DEFAULT IS DRIVE 0 9TRK ONLY.
IF APPROPRIATE SET SWO AND THEN PRESS CONTINUE OR
***IF SOFTWARE SWITCH REGISTER IS USED TYPE CNTL G AND THEN CONTINUE
-THIS WILL ALLOW THE MODIFICATION OF THE SOFTWARE SWITCH REGISTER (REFER TO S
THE PROGRAM WILL BEGIN TESTING.

92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128

5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

IF THE DIAGNOSTIC IS RUN ON A CPU WITHOUT A SWITCH REGISTER THEN A SOFTWARE SWITCH REGISTER IS USED WHICH ALLOWS THE USER THE SAME SWITCH OPTIONS AS THE HARDWARE SWITCH REGISTER. IF THE HARDWARE SWITCH REGISTER DOES NOT EXIST OR IF ONE DOES AND IT CONTAINS ALL ONES (177777) THEN THE SOFTWARE SWITCH REGISTER (LOC. 176) IS USED.

CONTROL:

THIS PROGRAM ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER (LOC. 176) FROM THE TTY. THIS CAN BE ACCOMPLISHED BY DOING THE FOLLOWING:

- 1) TYPE CONTROL G <^G>; THIS WILL ALLOW THE TTY TO ENTER DATA INTO LOC. 176 AT SELECTED POINTS WITHIN THE PROGRAM.
- 2) THE MACHINE WILL THEN TYPE: SWR=XXXXXXNEW= (XXXXXX IS THE OCTAL CONTENTS OF THE SOFTWARE SWITCH REGISTER.)
- 3) AFTER THE ''NEW='' HAS BEEN TYPED THEN THE OPERATOR CAN DO ONE OF THE FOLLOWING AT THE TTY:
 - A) TYPE A NUMBER TO BE LOADED INTO LOC. 176 FOLLOWED BY A <CR>. (ONLY NUMBERS BETWEEN 0-7 WILL BE ACCEPTED AND ONLY 6 NUMBERS WILL BE ALLOWED)
IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH REGISTER CONTENTS WILL NOT BE CHANGED.
 - B) IF A CONTROL U <^U> IS DEPRESSED THEN THE PROGRAM WILL SEND YOU BACK TO STEP 2.

129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164

5.1.1 WITH ALL SWITCHES DOWN THE PROGRAM WILL PRINT OUT ON ERRORS AND CONTINUE IN TEST. (END OF PASS WILL PRINT ON EACH PASS)

5.1.2 SWITCH SETTINGS ARE:

SW15 (100000) = 1 OR UP ... HALT ON ERROR
SW14 (040000) = 1 OR UP ... SCOPE LOOP
SW13 (020000) = 1 OR UP ... INHIBIT PRINTOUT.
SW12 (010000) = 1 OR UP ... INHIBIT SUB-TEST INTERATION.
SW11 (004000) = 1 OR UP ... SINGLE PASS
SW10 (002000) = 1 OR UP ... INHIBIT MANUAL INTERVENTION TEST
SW9 (001000) = 1 OR UP ... FOR TS03 TAPE DRIVES
SW0 (000001) = 1 OR UP ... TEST 7 CHANNEL TAPE UNIT.

5.1.3 MANUAL INTERVENTION TEST

THIS TEST WILL REQUIRE THE OPERATOR TO PERFORM CERTAIN OPERATIONS WITH THE TRANSPORT AS DIRECTED BY MESSAGES PRINTED ON THE TELETYPE.

5.2 SUBROUTINE ABSTRACTS

5.2.1 SCOPE

THIS SUBROUTINE CALL IS PLACED BETWEEN EACH SUB-TEST IN THE INSTRUCTION SECTION. IT RECORDS THE STARTING ADDRESS OF EACH SUB-TEST AS IT IS BEING ENTERED. IF A SCOPE LOOP IS REQUESTED, IT WILL JUMP TO THE START OF THE SUB-TEST THAT THE SCOPE LOOP IS REQUESTING.
***CNTL G WILL BE RECOGNIZED IN THIS ROUTINE (REFER TO SECT 5.1)

5.2.2 HLT

THIS SUBROUTINE CALL PRINTS THE ADDRESS THAT TAGS THE FAILING SUBTEST AND THE CONTENTS OF ALL THE TM,A,B-11 REGISTERS.
***THIS ROUTINE RECOGNIZES CNTL G FUNCTION (REFER TO SECT. 5.1)

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

6.0 ERRORS

6.1 ERROR PRINTOUT FORMAT

WITH SW13=0 (OR DOWN) THE FOLLOWING PRINTOUT WILL APPEAR ON AN ERROR:

PC	STATUS	COMAND	BYTE	CA	DATA B	READ L	TEMP	CRC CAL
XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX

PC = ADDRESS OF TEST WHERE ERROR OCCURED
STATUS = CONTENTS OF STATUS REGISTER AT TIME OF ERROR
COMAND = CONTENTS OF COMMAND REGISTER AT TIME OF ERROR
BYTE = CONTENTS OF BYTE COUNTER AT TIME OF ERROR
CA = CONTENTS OF CURRENT MEMORY ADDRESS AT TIME OF ERROR
DATA B = CONTENTS OF DATA BUFFER AT TIME OF ERROR
READ L = CONTENTS OF READ LINES AT TIME OF ERROR
TEMP = CONTENTS OF ADDRESS 'TEMP' USED BY SOME TESTS
CRC CAL = CRC CHARACTER CALCULATED (USEFUL ONLY FOR CRC TEST)

NOTE THAT NOT ALL OF THE INFORMATION PRINTED IS INTENDED TO BE USEFUL FOR EVERY TYPE OF ERROR. THIS IS SIMPLY A STANDARD ERROR REPORT FOR ALL ERRORS. THE OPERATOR MUST REFER TO THE PROGRAM LISTING AT THE ADDRESS OF THE ERROR FOR A DESCRIPTION OF THE CAUSE OF THE ERROR. IT IS THEN UP TO HIM TO DETERMINE WHICH OF THE INFORMATION IS USEFUL.

6.2 ERROR RECOVERY

WITH SW15=1 OR UP THE PROGRAM WILL HALT ON AN ERROR. DEPRESS CONTINUE SWITCH TO RESTART TEST.

198
199
200
201
202
203
204
205
206
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

7. RESTRICTIONS

7.1 STARTING RESTRICTION

BEFORE STARTING PROGRAM THE OPERATOR MUST MAKE CERTAIN THAT THE TRANSPORT HAS DRIVE 0 SELECTED 'ON-LINE'.

7.2 OPERATIONAL RESTRICTIONS

MANUAL INTERVENTION TEST MUST BE PERFORMED ON EACH PASS THRU THE PROGRAM UNLESS INHIBITED WITH SW10=1 (OR UP).
IF UNIT IS A TS03 SW9 MUST BE 1 (OR UP).

8.0 MISCELLANEOUS

8.1 EXECUTION TIME

WITH MANUAL INTERVENTION TEST INHIBITED IT TAKES 1 MINUTE FOR ONE PASS THRU PROGRAM. MANUAL INTERVENTION TEST IS OPERATOR DEPENDENT BUT SHOULD TAKE APPROXIMATELY 2 MINUTES.
NOTE: A PDP11/10 TAKES ABOUT 3.5 MINUTES INSTEAD OF 1 MIN.

9.0 PROGRAM DESCRIPTION

10.0 LISTING

%

.TITLE CZTMA10 TM,A,B-11 INSTR TST
:COPYRIGHT 1971,1979 DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
:JOHN RODENHISER
:REVISED AUGUST 1972, JIM LACEY
:REVISED MARCH 1973, JIM KAPADIA
:REVISED JANUARY 1975, KEN LIND
:REVISED AUG 1975, R. B. BARNES
:REVISED MAR 1976, S. K. CARPENTER - SUPPORT SOFTWARE SWITCH REGISTER
:REVISED AUG 1976, R. SOLER - INCLUDE TU10W,N
:(A) MODIFIED TO SUPPORT SOFTWARE SWITCH REGISTER
:(B) ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER FROM TTY
: BY PRESSING A CNTL G
:(C) PROGRAM WILL ALLOW THE LOADING OF THE SOFTWARE SWITCH REGISTER AT START
: IF NO HARDWARE SWITCH REGISTER IS AVAILABLE OR IF THE
: HARDWARE SWITCH REGISTER CONTAINS ALL 1'S.
:REVISED SEPT 1979, LEN LORANGER
:CHGI1 - INSTALLED DEPO PATCH TO CHANGE ERROR BIT
: TEST FROM 100 TO 177701.
:CHGI2 - INSTALLED DEPO PATCH TO CHANGE NON-EXISTANT
: MEMORY ADDRESS FROM 173000 TO 176000.
:CHGI3 - DELETED DIRECT MEMORY REFERENCE AND MADE
: INDIRECT MEMORY REFERENCE.

254				.ENABL ABS,AMA
255				.=0
256		000000		.WORD 0,0 ;CATCH IMPROPERLY LOADED VECTORS
257	000000	000000	000000	;TRAP CATCHER 0-776
258				.=30
259		000030		TRAP30
260	000030	014106		340
261	000032	000340		.=34
262		000034		SCOPEC
263	000034	013404		340
264	000036	000340		EMT=TRAP30
265		014106		CC=177776
266		177776		NOP=240
267		000240		SCOPE=TRAP
268		104400		BUFF=776
269		000776		R0=%0
270		000000		R1=%1
271		000001		R2=%2
272		000002		R3=%3
273		000003		R4=%4
274		000004		R5=%5
275		000005		SP=%6
276		000006		PC=%7
277		000007		

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
315
316
317
318
319
320
321

```

;*****
;THIS PROGRAM SUPPORTS SOFTWARE SWITCH REGISTER WHICH IS LOCATED AT LOC. 176
;BEFORE STARTING REFER TO SECTION 5.1 IN DOCUMENT

```

```

;*****
;

```

```

.SBTTL ACT11 HOOKS

```

```

;*****
;HOOKS REQUIRED BY ACT11

```

```

000040      $SVPC=.           ;SAVE PC
000046      .=46
000046      $ENDAD          ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
000052      .=52
000000      .WORD 0         ;;2)SET LOC.52 TO ZERO
000040      .=$SVPC        ;; RESTORE PC
000176      .WORD 2000     ;SOFTWARE SWITCH REGISTER(9TRK=2000/7TRK=2001)
000200      .WORD 2000
000137      JMP START
001000      .=1000
MTV:        224           ; INTERRUPT VECTOR
MTVS:       226           ; INTERRUPT STATUS
MTS:        172520        ; STATUS REGISTER
MTC:        172522        ; COMMAND REGISTER
BC:         172524        ; BYTE COUNT
CA:         172526        ; CURRENT MEMORY ADDRESS
MTD:        172530        ; DATA BUFFER
MTRD:       172532        ; READ LINES
TPB:        177566
TPS:        177564
SWR:        177570
TKS:        177560
TKB:        177562
TEMP:       0
ICOUNT:     0
PCNTR:      0           ;PASS COUNTER

```

001040

```

322
323                ;PROGRAM START*****
324
325 001040 012706 000776 START: MOV #BUFF,SP
326 001044 012737 177570 001024 MOV #177570,SWR ;PRESET TO CONSOLE SWITCHES
327 001052 005037 001036 CLR PCNTR ;CLEAR PASS COUNTER
328 001056 122737 000004 000041 CMPB #4,@#41 ;SEE IF LOAD MEDIUM
329 001064 001006 BNE SUSWR ;IF NOT: BR
330 001066 012702 017117 MOV #MSG21,R2
331 001072 004737 013462 JSR PC, TOP ;PRINT NO TEST
332 001076 000137 012672 JMP TSTEND ;END TEST
333 001102 013746 000006 SUSWR: MOV @#6,-(SP) ;SAVE VECTORS
334 001106 013746 000004 MOV @#4,-(SP)
335 001112 012737 001140 000004 MOV #1$,@#4 ;SET UP FOR TIMEOUT
336 001120 022777 177777 177676 CMP #-1,@SWR ;REFERENCE HARDWARE SWITCH REGISTER
337 001126 001405 BEQ 2$
338 001130 005737 000042 TST @#42
339 001134 001002 BNE 2$ ;IF CHAIN MODE: BR
340 001136 000404 BR 3$
341 001140 022626 1$: CMP (SP)+,(SP)+ ;ADJUST STACK
342 001142 012737 000176 001024 2$: MOV #SWREG,SWR ;POINT TO SOFTWARE SWITCH REG
343 001150 012637 000004 3$: MOV (SP)+,@#4 ;RESTORE VECTORS
344 001154 012637 000006 MOV (SP)+,@#6
345 001160 012702 014204 MOV #MSG0,R2
346 001164 004737 013462 JSR PC, TOP
347 001170 005002 CLR R2
348 001172 005302 4$: DEC R2 ;DELAY
349 001174 001376 BNE 4$
350 001176 005737 000042 TST @#42 ;SEE IF CHAIN MODE
351 001202 001006 BNE BEGIN ;IF SO: BR
352 001204 012702 014242 MOV #MSG01,R2
353 001210 004737 013462 JSR PC, TOP ;PRINT 7 TRK SELECT
354 001214 000000 HALT
355 001216 104002 CKSWR
356 001220 012706 000776 BEGIN: MOV #BUFF,SP ;CHECK FOR A CNTL G
357 001224 012737 001220 013460 MOV #BEGIN,RETURN ;SET UP STACK FOR SCOPE LOOPS
358 001232 005037 013200 CLR PRINT1 ;SET UP RESTART OF PROGRAM
359 001236 005037 011244 CLR CRCWRT ;INITIALIZE ERROR PRINTOUT HEADING
360 001242 005037 000006 CLR 6 ;INITIALIZE CRC CALCULATED FOR PRINTOUT
361
362
363
364 ;*****
365 ;TEST ALL BITS OF COMMAND REGISTER (EXCEPT CU READY, BIT 7) TO BE CLEARED BY INIT
366 001246 104400 SCOPE
367 001250 000005 RESET
368 001252 032777 177577 177526 BIT #177577,@MTC
369 001260 001401 BEQ .+4
370 001262 104000 HLT ;ERROR, INIT DIDN'T CLEAR COMMAND REGISTER
371
372
373 ;*****
374 ;TEST BITS 7-13, 15 OF STATUS REGISTER TO BE CLEARED AFTER INIT
375 001264 104400 SCOPE
376 001266 000005 RESET
377 001270 032777 137600 177506 BIT #137600,@MTS

```

```

378 001276 001401          BEQ      .+4
379 001300 104000          HLT              ;ERROR, INIT DIDN'T CLEAR PROPER BITS IN STATUS REGISTER
380
381
382  ;*****
383  ;TEST INIT TO CLEAR BYTE RECORD COUNT
384 001302 104400          SCOPE
385 001304 000005          RESET
386 001306 005777 177476  TST      @BC
387 001312 001401          BEQ      .+4
388 001314 104000          HLT              ;ERROR, INIT DIDN'T CLEAR BYTE COUNT
389
390
391  ;*****
392  ;TEST INIT TO CLEAR CURRENT MEMORY ADDRESS REGISTER
393 001316 104400          SCOPE
394 001320 000005          RESET
395 001322 005777 177464  TST      @CA
396 001326 001401          BEQ      .+4
397 001330 104000          HLT              ;ERROR, INIT DIDN'T CLEAR CURRENT MEMORY ADDRESS REGISTE
398
399
400  ;*****
401  ;TEST INIT TO CLEAR DATA BUFFER
402 001332 104400          SCOPE
403 001334 000005          RESET
404 001336 005777 177452  TST      @MTD
405 001342 001401          BEQ      .+4
406 001344 104000          HLT              ;ERROR, INIT DIDN'T CLEAR DATA BUFFER
407
408
409  ;*****
410  ;TEST CU READY (BIT 7 COMMAND REGISTER) TO BE SET BY INIT.
411 001346 104400          SCOPE
412 001350 000005          RESET
413 001352 105777 177430  TSTB    @MTC
414 001356 100401          BMI      .+4
415 001360 104000          HLT              ;ERROR, INIT DIDN'T SET CU READY
416
417
418  ;*****
419  ;TEST BIT 14 OF TU10 READ LINES TO BE CLEARED BY INIT
420 001362 104400          SCOPE
421 001364 000005          RESET
422 001366 032777 040000 177422  BIT     #40000,@MTRD
423 001374 001401          BEQ      .+4
424 001376 104000          HLT              ;ERROR, INIT FAILED TO CLEAR BIT 14 OF MTRD
425
426
427  ;*****
428  ;TEST COMMAND REGISTER (EXCEPT CU READY, BIT 7) TO BE CLEARED BY POWER CLEAR (BIT 12)
429 001400 104400          SCOPE
430 001402 052777 010000 177376  BIS     #10000,@MTC          ;POWER CLEAR
431 001410 032777 177577 177370  BIT     #177577,@MTC
432 001416 001401          BEQ      .+4
433 001420 104000          HLT              ;ERROR, POWER CLEAR DIDN'T CLEAR COMMAND REGISTER
    
```

434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489

```

001422 104400
001424 052777 010000 177354
001432 032777 137600 177344
001440 001401
001442 104000

001444 104400
001446 052777 010000 177332
001454 005777 177330
001460 001401
001462 104000

001464 104400
001466 052777 010000 177312
001474 005777 177312
001500 001401
001502 104000

001504 104400
001506 052777 010000 177272
001514 005777 177274
001520 001401
001522 104000

001524 104400
001526 052777 010000 177252
001534 105777 177246
001540 100401
001542 104000

001544 104400
001546 052777 010000 177232
001554 032777 040000 177234
001562 001401
001564 104000
  
```

```

:*****
:TEST BITS 7-13, 15 OF STATUS REGISTER TO BE CLEARED BY POWER CLEAR (BIT12)
SCOPE
BIS #10000,@MTC ;POWER CLEAR
BIT #137600,@MTC
BEQ .+4
HLT ;ERROR, POWER CLEAR DIDN'T CLEAR PROPER BITS IN STATUS

:*****
:TEST POWER CLEAR (BIT 12) TO CLEAR BYTE RECORD COUNT
SCOPE
BIS #10000,@MTC ;POWER CLEAR
TST @BC
BEQ .+4
HLT ;ERROR, POWER CLEAR DIDN'T CLEAR BYTE COUNT

:*****
:TEST POWER CLEAR (BIT 12) TO CLEAR CURRENT MEMORY ADDRESS REGISTER
SCOPE
BIS #10000,@MTC ;POWER CLEAR
TST @CA
BEQ .+4
HLT ;ERROR, POWER CLEAR DIDN'T CLEAR CURRENT ADDRESS REGISTE

:*****
:TEST POWER CLEAR (BIT 12) TO CLEAR DATA BUFFER
SCOPE
BIS #10000,@MTC ;POWER CLEAR
TST @MTC
BEQ .+4
HLT ;ERROR, POWER CLEAR DIDN'T CLEAR DATA BUFFER

:*****
:TEST CU READY (BIT 7 COMMAND REGISTER) TO BE SET BY POWER CLEAR
SCOPE
BIS #10000,@MTC ;POWER CLEAR
TSTB @MTC
BMI .+4
HLT ;ERROR, POWER CLEAR DIDN'T SET CU READY

:*****
:TEST BIT 14 OF TU10 READ LINES TO BE CLEARED BY POWER CLEAR (BIT12)
SCOPE
BIS #10000,@MTC ;POWER CLEAR
BIT #40000,@MTRD
BEQ .+4
HLT ;ERROR, POWER CLEAR FAILED TO CLEAR BIT14 OF MTRD
  
```

```

490
491
492 ;*****
493 ;TEST FUNCTION BITS (1,2,3) OF COMMAND REGISTER CAN BE SET
494 001566 104400 SCOPE
495 001570 012777 000016 177210 MOV #16,@MTC
496 001576 122777 000216 177202 CMPB #216,@MTC
497 001604 001401 BEQ .+4
498 001606 104000 HLT ;ERROR, CU READY AND ALL FUNCTION BITS NOT SET
499

```

```

500 ;*****
501 ;TEST FUNCTION BITS (1,2,3) OF COMMAND REGISTER CAN BE CLEARED
502 001610 104400 SCOPE
503 001612 052777 000016 177166 BIS #16,@MTC
504 001620 042777 000016 177160 BIC #16,@MTC
505 001626 032777 000016 177152 BIT #16,@MTC
506 001634 001401 BEQ .+4
507 001636 104000 HLT ;ERROR, ALL FUNCTION BITS NOT CLEARED
508
509
510

```

```

511 ;*****
512 ;TEST FUNCTIONS BITS (1,2,3,) OF COMMAND REGISTER CAN BE SET AND CLEARED INDIVIDUALLY
513 001640 104400 SCOPE
514 001642 012777 000002 177136 MOV #2,@MTC
515 001650 122777 000202 177130 CMPB #202,@MTC
516 001656 001401 BEQ .+4
517 001660 104000 HLT ;ERROR, FUNCTION NOT =001 (READ)
518 001662 104400 SCOPE
519 001664 012777 000004 177114 MOV #4,@MTC
520 001672 122777 000204 177106 CMPB #204,@MTC
521 001700 001401 BEQ .+4
522 001702 104000 HLT ;ERROR, FUNCTION NOT =010 (WRITE)
523 001704 104400 SCOPE
524 001706 012777 000006 177072 MOV #6,@MTC
525 001714 122777 000206 177064 CMPB #206,@MTC
526 001722 001401 BEQ .+4
527 001724 104000 HLT ;ERROR, FUNCTION NOT =011 (WRITE EOF)
528 001726 104400 SCOPE
529 001730 012777 000010 177050 MOV #10,@MTC
530 001736 122777 000210 177042 CMPB #210,@MTC
531 001744 001401 BEQ .+4
532 001746 104000 HLT ;ERROR, FUNCTION NOT =100 (SPACE FORWARD)
533 001750 104400 SCOPE
534 001752 012777 000012 177026 MOV #12,@MTC
535 001760 122777 000212 177020 CMPB #212,@MTC
536 001766 001401 BEQ .+4
537 001770 104000 HLT ;ERROR, FUNCTION NOT =101 (SPACE REVERSE)
538 001772 104400 SCOPE
539 001774 012777 000014 177004 MOV #14,@MTC
540 002002 122777 000214 176776 CMPB #214,@MTC
541 002010 001401 BEQ .+4
542 002012 104000 HLT ;ERROR, FUNCTION NOT =110 (WRITE XIRG)
543 002014 104400 SCOPE
544 002016 012777 000016 176762 MOV #16,@MTC
545 002024 122777 000216 176754 CMPB #216,@MTC

```

```

CZTMAIO TM,A,B-11 INSTR TST          MACY11 30A(1052) 12-SEP-79 13:57 N 1 PAGE 14
CZTMAI.P11 12-SEP-79 13:54          TEST FUNCTIONS BITS (1,2,3,) OF COMMAND REGISTER CAN BE SET AND CLEARED INDIVIDUA SEQ 0013

546 002032 001401          BEQ      .+4          ;ERROR, FUNCTION NOT =111 (REWIND)
547 002034 104000          HLT
548
549
550
551 :*****
552 :TEST ADDRESS BITS (4,5) OF COMMAND REGISTER CAN BE SET
553 002036 104400          SCOPE
554 002040 012777 000060 176740      MOV      #60,@MTC
555 002046 122777 000260 176732      CMPB    #260,@MTC
556 002054 001401          BEQ      .+4
557 002056 104000          HLT          ;ERROR, CU READY AND ADDRESS BITS NOT SET
558
559 :*****
560 :TEST ADDRESS BITS (4,5) OF COMMAND REGISTER CAN BE CLEARED
561 002060 104400          SCOPE
562 002062 052777 000060 176716      BIS      #60,@MTC
563 002070 042777 000060 176710      BIC      #60,@MTC
564 002076 032777 000060 176702      BIT      #60,@MTC
565 002104 001401          BEQ      .+4
566 002106 104000          HLT          ;ERROR, ADDRESS BITS NOT CLEARED
567
568
569
570 :*****
571 :CHECK BYTE LOADING OF COMMAND REGISTER
572 :THIS 1ST SECTION WILL TEST THAT THE FUNCTION BITS CAN BE BYTE LOADED
573 002110 104400          SCOPE
574 002112
575 002112 012700 000002      CHGI3:
576 002116 005077 176664      CHKBYTE:MOV    #2,R0          ;INITIALIZE R0
577 002122 110077 176660      CLR      @MTC              ;CLEAR OUT COMMAND REGISTER
578 002126 017701 176654      1$:  MOVB    R0,@MTC         ;LOAD LOWER BYTE OF COMMAND REGISTER
579                                MOV      @MTC,R1          ;GET THE CONTENTS OF THE COMMAND
580                                ;REGISTER JUST LOADED
581                                BIC      #200,R1          ;MASK OUT READY BIT FOR COMPARE
582                                CMP      R1,R0           ;DID IT LOAD PROPERLY?
583                                BEQ      2$              ;BRANCH IF YES
584                                MOV      R0,@EXPECTED ;STORE WHAT SHOULD HAVE APPEARED IN
585                                ;THE COMMAND REGISTER
586                                ADD      #200,@EXPECTED ;ADD A READY BIT TO WHAT WAS BYTE LOADED
587                                MOV      @MTC,@RECEIVED ;STORE WHAT DID APPEAR IN THE
588                                ;COMMAND REGISTER
589                                MOV      #MSG14,R2        ;INDICATE INCORRECT BYTE LOAD
590                                JSR      PC,TOP
591                                HLT
592                                ;PROGRAM PC INDICATOR
593                                ;AT THIS POINT LOCATIONS 6$ AND 7$ BELOW WILL CONTAIN THE GOOD
594                                ;AND BAD DATA, RESPECTIVELY
595                                BR      3$
596                                ;GO TO START LOADING THE UPPER BYTE OF
597                                ;THE COMMAND REGISTER
598                                2$:  ADD      #2,R0          ;STEP UP VALUE TO BE BYTE LOADED
599                                CMP      #16,R0         ;OK TO LOAD IT?
600                                BMI      3$              ;BRANCH IF NO
601                                CLR      @MTC         ;CLEAR OUT THE COMMAND REGISTER
602                                BR      1$              ;GO BACK TO LOAD NEW VALUE
603                                ;THIS 2ND SECTION WILL TEST THAT ALL UPPER BYTE DATA COMBINATIONS
604                                ;CAN BE BYTE LOADED EXCEPT FOR THOSE REQUIRING BIT12 (POWER CLEAR)

```

```

602      :TO BE SET
603 002216 005000 3$: CLR R0 ;INITIALIZE R0
604 002220 005077 176562 CLR @MTC ;CLEAR OUT COMMAND REGISTER
605 002224 010003 4$: MOV R0,R3 ;SAVE INITIAL VALUE TO BE BYTE LOADED
606 002226 042700 000020 BIC #20,R0 ;MASK OUT POWER CLEAR BIT OF BYTE
607 ;TO BE LOADED
608 ;***** CHG13 *****
609 002232 013701 001006 MOV @MTC,R1 ;SET UP FOR COM REG UPPER BYTE
610 002236 110061 000001 MOV#B R0,1(R1) ;LOAD UPPER BYTE OF COMMAND REGISTER
611 ;*****
612 002242 000300 SWAB R0 ;MOVE LOWER BYTE INTO UPPER BYTE
613 ;POSITION FOR CHECKING
614 002244 017701 176536 MOV @MTC,R1 ;GET THE CONTENTS OF THE COMMAND
615 ;REGISTER JUST LOADED
616 002250 042701 000200 BIC #200,R1 ;MASK OUT READY BIT FOR COMPARE
617 002254 020100 CMP R1,R0 ;DID IT LOAD PROPERLY?
618 002256 001417 BEQ 5$ ;BRANCH IF YES
619 002260 010037 002342 MOV R0,@#GOODDATA ;STORE WHAT SHOULD HAVE APPEARED IN THE
620 ;COMMAND REGISTER
621 002264 062737 000200 002342 ADD #200,@#GOODDATA ;ADD A READY BIT TO WHAT WAS BYTE LOADED
622 002272 017737 176510 002344 MOV @MTC,@#BADDATA ;STORE WHAT DID APPEAR IN THE COMMAND
623 ;REGISTER
624 002300 012702 016552 MOV #MSG15,R2 ;INDICATE INCORRECT BYTE LOAD
625 002304 004737 013462 JSR PC,TOP
626 002310 104000 HLT ;PROGRAM PC INDICATOR
627 ;AT THIS POINT LOCATIONS 10$ AND 11$ BELOW WILL CONTAIN THE GOOD
628 ;AND BAD DATA, RESPECTIVELY
629 002312 000137 002346 JMP @#BYTEOPEND ;ALL DONE
630 002316 010300 5$: MOV R3,R0 ;OBTAIN INITIAL VALUE (LESS MASKING OF
631 ;POWER CLEAR BIT) THAT WAS TO BE BYTE
632 ;LOADED
633 002320 005200 INC R0 ;STEP UP VALUE TO BE BYTE LOADED
634 002322 022700 000157 CMP #157,R0 ;OK TO LOAD IT?
635 002326 100407 BMI BYTEOPEND ;BRANCH IF NO
636 002330 005077 176452 CLR @MTC ;CLEAR OUT THE COMMAND REGISTER
637 002334 000733 BR 4$ ;GO BACK TO LOAD NEW VALUE
638 ;THE FOLLOWING ARE THE GOOD AND BAD DATA HOLDERS FOR THE LOWER
639 ;AND UPPER BYTE LOAD TESTS PERFORMED ABOVE
640 6$:
641 002336 000000 EXPECTED: .WORD 0 ;HOLDS VALUE THAT SHOULD HAVE
642 ;APPEARED IN THE COMMAND REGISTER
643 ;ON LOW BYTE LOADING
644 002340 7$:
645 002340 000000 RECEIVED: .WORD 0 ;HOLDS VALUE THAT DID APPEAR IN
646 ;THE COMMAND REGISTER ON LOW
647 ;BYTE LOADING
648 002342 10$:
649 002342 000000 GOODDATA: .WORD 0 ;HOLDS VALUE THAT SHOULD HAVE
650 ;APPEARED IN THE COMMAND REGISTER
651 ;ON UPPER BYTE LOADING
652 002344 11$:
653 002344 000000 BADDATA: .WORD 0 ;HOLDS VALUE THAT DID APPEAR IN
654 ;THE COMMAND REGISTER ON UPPER
655 ;BYTE LOADING
656 002346 005077 176434 BYTEOPEND: CLR @MTC ;CLEAR OUT COMMAND REGISTER BEFORE GOING ON
657

```

```

658
659
660
661 ;*****
662 ;TEST ADDRESS BITS (4,5,6) OF COMMAND REGISTER CAN BE SET AND CLEARED INDIVIDUALLY
663 002352 104400 SCOPE
664 002354 012777 000020 176424 MOV #20,@MTC
665 002362 122777 000220 176416 CMPB #220,@MTC
666 002370 001401 BEQ .+4
667 002372 104000 HLT ;ERROR ADDRESS BITS NOT =01
668 002374 104400 SCOPE
669 002376 012777 000040 176402 MOV #40,@MTC
670 002404 122777 000240 176374 CMPB #240,@MTC
671 002412 001401 BEQ .+4
672 002414 104000 HLT ;ERROR, ADDRESS BITS NOT =10
673 002416 104400 SCOPE
674 002420 012777 000060 176360 MOV #60,@MTC
675 002426 122777 000260 176352 CMPB #260,@MTC
676 002434 001401 BEQ .+4
677 002436 104000 HLT ;ERROR, ADDRESS BITS NOT =11
678
679
680 ;*****
681 ;TEST UNIT SELECT BITS (8,9,10) OF COMMAND REGISTER CAN BE SET
682 002440 104400 SCOPE
683 002442 012777 003400 176336 MOV #3400,@MTC
684 002450 022777 003600 176330 CMP #3600,@MTC
685 002456 001401 BEQ .+4
686 002460 104000 HLT ;ERROR, CU READY AND ALL UNIT SELECT BITS NOT SET
687
688
689 ;*****
690 ;TEST UNIT SELECT BITS (8,9,10) OF COMMAND REGISTER CAN BE CLEARED
691 002462 104400 SCOPE
692 002464 052777 003400 176314 BIS #3400,@MTC
693 002472 042777 003400 176306 BIC #3400,@MTC
694 002500 032777 003400 176300 BIT #3400,@MTC
695 002506 001401 BEQ .+4
696 002510 104000 HLT ;ERROR, UNIT SELECT BITS NOT CLEARED
697
698
699 ;*****
700 ;TEST UNIT SELECT BITS (8,9,10) OF COMMAND REGISTER CAN BE SET AND CLEARED INDIVIDUALLY
701 002512 104400 SCOPE
702 002514 012777 000400 176264 MOV #400,@MTC
703 002522 022777 000600 176256 CMP #600,@MTC
704 002530 001401 BEQ .+4
705 002532 104000 HLT ;ERROR, UNIT SELECT NOT =001
706 002534 104400 SCOPE
707 002536 012777 001000 176242 MOV #1000,@MTC
708 002544 022777 001200 176234 CMP #1200,@MTC
709 002552 001401 BEQ .+4
710 002554 104000 HLT ;ERROR, UNIT SELECT NOT =010
711 002556 104400 SCOPE
712 002560 012777 001400 176220 MOV #1400,@MTC
713 002566 022777 001600 176212 CMP #1600,@MTC

```



```

714 002574 001401 BEQ .+4
715 002576 104000 HLT ;ERROR, UNIT SELECT NOT =011
716 002600 104400 SCOPE
717 002602 012777 002000 176176 MOV #2000,@MTC
718 002610 022777 002200 176170 CMP #2200,@MTC
719 002616 001401 BEQ .+4
720 002620 104000 HLT ;ERROR, UNIT SELECT NOT =100
721
722
723 002622 104400 SCOPE
724 002624 012777 002400 176154 MOV #2400,@MTC
725 002632 022777 002600 176146 CMP #2600,@MTC
726 002640 001401 BEQ .+4
727 002642 104000 HLT ;ERROR, UNIT SELECT NOT =101
728 002644 104400 SCOPE
729 002646 012777 003000 176132 MOV #3000,@MTC
730 002654 022777 003200 176124 CMP #3200,@MTC
731 002662 001401 BEQ .+4
732 002664 104000 HLT ;ERROR, UNIT SELECT NOT =110
733 002666 104400 SCOPE
734 002670 012777 003400 176110 MOV #3400,@MTC
735 002676 022777 003600 176102 CMP #3600,@MTC
736 002704 001401 BEQ .+4
737 002706 104000 HLT ;ERROR, UNIT SELECT NOT =111
738
739
740
741 ;*****
742 ;TEST PARITY BIT (BIT 11) CAN BE SET
743 002710 104400 SCOPE
744 002712 052777 004000 176066 BIS #4000,@MTC
745 002720 032777 004000 176060 BIT #4000,@MTC
746 002726 001001 BNE .+4
747 002730 104000 HLT ;ERROR, PARITY NOT SET
748
749
750 ;*****
751 ;TEST PARITY BIT (BIT 11) CAN BE CLEARED
752 002732 104400 SCOPE
753 002734 052777 004000 176044 BIS #4000,@MTC
754 002742 042777 004000 176036 BIC #4000,@MTC
755 002750 032777 004000 176030 BIT #4000,@MTC
756 002756 001401 BEQ .+4
757 002760 104000 HLT ;ERROR, PARITY BIT NOT CLEARED
758
759
760 ;*****
761 ;TEST DENSITY BITS (13,14) OF COMMAND REGISTER CAN BE SET
762 002762 104400 SCOPE
763 002764 012777 060000 176014 MOV #60000,@MTC
764 002772 022777 060200 176006 CMP #60200,@MTC
765 003000 001401 BEQ .+4
766 003002 104000 HLT ;ERROR, CU READY AND DENSITY BITS NOT SET
767
768
769
    
```

```
770 :*****
771 :TEST DENSITY BITS (13,14) OF COMMAND REGISTER CAN BE CLEARED
772 003004 104400 SCOPE
773 003006 052777 060000 175772 BIS #60000,@MTC
774 003014 042777 060000 175764 BIC #60000,@MTC
775 003022 032777 060000 175756 BIT #60000,@MTC
776 003030 001401 BEQ .+4
777 003032 104000 HLT
778
779
780
781
782 :*****
783 :TEST DENSITY BITS (13,14) OF COMMAND REGISTER CAN BE SET AND CLEARED INDIVIDUALLY
784 003034 104400 SCOPE
785 003036 012777 020000 175742 MOV #20000,@MTC
786 003044 022777 020200 175734 CMP #20200,@MTC
787 003052 001401 BEQ .+4
788 003054 104000 HLT ;ERROR, DENSITY NOT =01
789 003056 104400 SCOPE
790 003060 012777 040000 175720 MOV #40000,@MTC
791 003066 022777 040200 175712 CMP #40200,@MTC
792 003074 001401 BEQ .+4
793 003076 104000 HLT ;ERROR, DENSITY NOT =10
794 003100 104400 SCOPE
795 003102 012777 060000 175676 MOV #60000,@MTC
796 003110 022777 060200 175670 CMP #60200,@MTC
797 003116 001401 BEQ .+4
798 003120 104000 HLT ;ERROR DENSITY NOT =11
799
800
801 :*****
802 :TEST ALL BITS OF BYTE COUNT TO ACCEPT COUNT PATTERN
803 003122 104400 SCOPE
804 003124 005037 001034 CLR ICOUNT
805 003130 005037 001032 CLR TEMP
806 003134 013777 001032 175646 TBC: MOV TEMP,@BC
807 003142 023777 001032 175640 CMP TEMP,@BC
808 003150 001401 BEQ .+4
809 003152 104000 HLT ;ERROR, BYTE COUNT NOT =TEMP
810 003154 005237 001032 INC TEMP
811 003160 001365 BNE TBC
812
813
814 :*****
815 :TEST ALL BITS OF CURRENT MEMORY ADDRESS REGISTER TO ACCEPT COUNT PATTERN
816 003162 104400 SCOPE
817 003164 005037 001034 CLR ICOUNT
818 003170 005037 001032 CLR TEMP
819 003174 013777 001032 175610 TMA: MOV TEMP,@CA
820 003202 023777 001032 175602 CMP TEMP,@CA
821 003210 001401 BEQ .+4
822 003212 104000 HLT ;ERROR, CA NOT = TEMP
823 003214 005237 001032 INC TEMP
824 003220 001365 BNE TMA
825
```

```
826
827
828
829 003222 104400
830 003224 005037 001034
831 003230 005037 001032
832 003234 113777 001032 175552 TDB:
833 003242 123777 001032 175544
834 003250 001401
835 003252 104000
836 003254 105237 001032
837 003260 001365
838
839
840
841
842
843
844 003262 104400
845 003264 052777 040000 175524
846 003272 032777 040000 175516
847 003300 001001
848 003302 104000
849 003304 042777 040000 175504
850 003312 032777 040000 175476
851 003320 001401
852 003322 104000
853
854
855
856
857 003324 104400
858 003326 042777 003400 175452
859 003334 032777 000001 175442
860 003342 001001
861 003344 104000
862
863
864
865
866 003346 104400
867 003350 032777 000002 175426
868 003356 001401
869 003360 104000
870
871
872
873
874 003362 104400
875 003364 032777 000004 175412
876 003372 001401
877 003374 104000
878
879
880
881
```

;TEST BITS 0-7 OF DATA BUFFER TO ACCEPT COUNT PATTERN
SCOPE
CLR I COUNT
CLR
MOVE TFM @MTD
CMPF F M @MTD
BEQ
HLT ;ERROR, DATA BUFFER NOT=TEMP
INCB TFM
BNE I DB

;TEST BIT 14 OF MTRD CAN BE SET AND CLEARED
SCOPE
BIS #40000,@MTRD
BIT #40000,@MTRD
BNE .+4
HLT ;ERROR, BIT 14 OF MTRD NOT=1
BIC #40000,@MTRD
BIT #40000,@MTRD
BEQ .+4
HLT ;ERROR, BIT 14 OF MTRD NOT=0

;TEST FOR TAPE UNIT READY (BIT 0) SET
SCOPE
BIC #3400,@MTC ;SELECT DRIVE 0
BIT #1,@MTC
BNE .+4
HLT ;ERROR TU READY NOT SET

;TEST FOR REWIND STATUS (BIT 1) CLEARED
SCOPE
BIT #2,@MTC
BEQ .+4
HLT ;ERROR, REWIND STATUS IS SET

;TEST FOR WRITE LOCK (BIT 2) CLEARED
SCOPE
BIT #4,@MTC
BEQ .+4
HLT ;ERROR, WRITE LOCK IS SET

;TEST FOR SETTLEDOWN (BIT 3) CLEARED

```

882 003376 104400          SCOPE
883 003400 032777 000010 175376  BIT      #10,@MTS
884 003406 001401          BEQ      .+4
885 003410 104000          HLT
                                     ;ERROR, SETTLEDOWN IS SET
886
887
888
889
890 003412 132777 000001 175404  :*****
:TEST FOR 7 CHANNEL (BIT 4) CLEARED IF 9 CHANNEL SELECTED
891 003420 001007          BITB     #1,@SWR      ;IS SW0=1
892 003422 104400          BNE     T7CH        ;YES SKIP 9 CHANNEL TEST
893 003424 032777 000020 175352  SCOPE
894 003432 001401          BIT      #20,@MTS
895 003434 104000          BEQ      .+4
896 003436 000417          HLT
                                     ;ERROR, 7 CHANNEL SET WITH 9 TRACK DOWN SELECTED
897
898
899
900
901 003440 104400          T7CH:   SCOPE
902 003442 032777 000020 175334  BIT      #20,@MTS
903 003450 001001          BNE     .+4
904 003452 104000          HLT
                                     ;ERROR, 7 CHANNEL NOT SET
905
906
907
908
909 003454 104400          :*****
:TEST FOR BEGINNING OF TAPE (BIT 5) SET
910 003456 042777 003400 175322  SCOPE
911 003464 032777 000040 175312  BIC     #3400,@MTC  ;SELECT DRIVE 0
912 003472 001001          BIT      #40,@MTS
913 003474 104000          BNE     .+4
914
915
916
917
918 003476 104400          :*****
:TEST FOR SELECT/REMOTE (BIT 6) SET
919 003500 032777 000100 175276  TSR:   SCOPE
920 003506 001001          BIT      #100,@MTS
921 003510 104000          BNE     .+4
922
923
924
925
926
927
928
929
930
931 003512 104400          :*****
:WRITE 1 BYTE RECORD FROM BOT
932 003514 005077 175266          :BOT (BIT 5) SHOULD CLEAR, CU READY SHOULD SET, BYTE COUNT AND
933 003520 105777 175262          :CURRENT ADDRESS SHOULD INCREMENT
934 003524 100375          SCOPE
935 003526 006077 175252  CLR     @MTC        ;SELECT UNIT 0
936 003532 103375          TSTB   @MTC
937 003534 032777 000040 175242  BPL     .-4        ;WAIT FOR CU READY
                                     ROR     @MTS
                                     BCC     .-4        ;WAIT FOR TU READY
                                     BIT     #40,@MTS        ;SEE IF AT BOT

```

```

938 003542 001014          BNE      3$          ;IF SO: BR
939 003544 005737 000042   TST      @#42       ;SEE IF CHAIN MODE
940 003550 001410          BEQ      2$          ;IF NOT: BR
941 003552 012777 060017 175226 MOV     #60017,@MTC ;START REWIND
942 003560 032777 000001 175216 1$: BIT     #1,@MTS     ;SEE IF TUR
943 003566 001774          BEQ      1$          ;IF NOT:AWAIT TUR
944 003570 000401          BR       3$
945 003572 104000          HLT
946 003574 012777 177777 175206 2$: MOV     #-1,@BC    ;ERROR, DRIVE 0 NOT AT BOT
947 003602 012777 017202 175202 3$: MOV     #WBUF,@CA  ;SET BYTE COUNT=-1
948 003610 005000          CLR     R0          ;INIT CURRENT MEMORY ADDRESS
949 003612 012701 000020          MOV     #20,R1     ;INIT DELAY COUNTER
950 003616 012777 060005 175162 MOV     #60005,@MTC ;WRITE, 800 BPI, GO
951 003624 005200          INC     R0
952 003626 001376          BNE     .-2        ;DELAY LONG ENOUGH TO MOVE OFF BOT
953 003630 005301          DEC     R1
954 003632 001374          BNE     .-6
955 003634 032777 000040 175142 BIT     #40,@MTS   ;TEST FOR BOT
956 003642 001401          BEQ     .+4
957 003644 104000          HLT
958 003646 105777 175134   TSTB   @MTC        ;ERROR, BOT (BIT 5) NOT CLEARED
959 003652 100401          BMI     .+4        ;TEST FOR CU READY
960 003654 104000          HLT
961 003656 005777 175126   TST     @BC
962 003662 001401          BEQ     .+4        ;ERROR, CU READY NOT SET AFTER WRITE FINISHED
963 003664 104000          HLT
964 003666 022777 017203 175116 CMP     #WBUF+1,@CA ;TEST BYTE COUNT TO = 0
965 003674 001401          BEQ     .+4
966 003676 104000          HLT
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993

```

 ;TEST WRITE A 3 BYTE RECORD

```

SCOPE
MOV     #-3,@BC    ;SET BYTE COUNT = -3
MOV     #WBUF,@CA  ;INIT CURRENT MEMORY ADDRESS
CLR     R0          ;INIT DELAY COUNTER
CLR     @MTC       ;SELECT UNIT 0
TSTB   @MTC
BPL     .-4        ;WAIT FOR CU READY
ROR     @MTS
BCC     .-4        ;WAIT FOR TU READY
CLR     R0
MOV     #60005,@MTC ;WRITE, 800BPI, GO
TSTB   @MTC
BMI     .+10
INC     R0
BNE     .-10
HLT
;ERROR, CU READY DIDN'T SET AFTER 3 BYTE RECORD

CMP     #WBUF+3,@CA
BEQ     .+4
HLT
;ERROR, CURRENT MEMORY ADDRESS DIDN'T INCREMENT
TST     @BC
BEQ     .+4

```

```

994 004004 104000          HLT          ;ERROR, BYTE COUNT DIDN'T INCREMENT TO 0
995 004006 005777 174774   TST      @MTC
996 004012 100001          BPL      .+4
997 004014 104000          HLT          ;ERROR, BIT 15 SET IN COMMAND REGISTER
998
999
1000
1001
1002          ;*****
1002          ;TEST REWIND FUNCTION
1003          SCOPE
1004 004016 104400          MOV      #-3,@BC          ;WRITE SHORT RECORD TO BE CERTAIN THAT BOT NOT SE
1005 004020 012777 177775 174762 MOV      #WBUF,@CA
1006 004026 012777 017202 174756 TSTB    @MTC
1007 004034 105777 174746          BPL      .-4
1008 004042 012777 060005 174736 MOV      #60005,@MTC      ;WRITE, 800 BPI, GO
1009 004050 105777 174732          TSTB    @MTC
1010 004054 100375          BPL      .-4          ;WAIT FOR CU READY
1011 004056 006077 174722          ROR      @MTS
1012 004062 103375          BCC      .-4          ;WAIT FOR TU READY
1013 004064 012777 060017 174714 MOV      #60017,@MTC      ;REWIND, GO
1014 004072 105777 174710          TSTB    @MTC
1015 004076 100375          BPL      .-4
1016 004100 032777 000002 174676 BIT      #2,@MTS
1017 004106 001001          BNE      .+4
1018 004110 104000          HLT          ;ERROR, REWIND STATUS (BIT 1) NOT = 1 DURING REWIND
1019 004112 006077 174666          ROR      @MTS
1020 004116 103001          BCC      .+4
1021 004120 104000          HLT          ;ERROR, TU READY NOT = 0
1022 004122 032777 000002 174654 BIT      #2,@MTS
1023 004130 001374          BNE      .-6          ;WAIT FOR RWS TO CLEAR
1024 004132 032777 000010 174644 BIT      #10,@MTS
1025 004140 001774          BEQ      .-6          ;WAIT FOR SETTLEDOWN TO SET
1026 004142 032777 000040 174634 BIT      #40,@MTS
1027 004150 001001          BNE      .+4
1028 004152 104000          HLT          ;ERROR, BOT (BIT 5) NOT = 1 WHEN SDWN (BIT 3) SET ON REW
1029 004154 032777 000010 174622 BIT      #10,@MTS
1030 004162 001374          BNE      .-6
1031 004164 006077 174614          ROR      @MTS
1032 004170 103401          BCS      .+4
1033 004172 104000          HLT          ;ERROR, TU READY NOT SET AFTER SDWN CLEARED ON REWIND
1034
1035
1036
1037          ;*****
1038          ;READ 1 BYTE RECORD FROM BOT
1039          ;BOT (BIT 5) SHOULD CLEAR, CU READY SHOULD SET, BYTE COUNT AND
1040          ;CURRENT ADDRESS SHOULD INCREMENT
1041
1042          SCOPE
1043 004174 104400          CLR      @MTC          ;SELECT UNIT 0
1044 004176 005077 174604          TSTB    @MTC
1045 004202 105777 174600          BPL      .-4          ;WAIT FOR CU READY
1046 004206 100375          ROR      @MTS
1047 004210 006077 174570          BCC      .-4          ;WAIT FOR TU READY
1048 004214 103375          BIT      #40,@MTS
1049 004216 032777 000040 174560 BIT      #40,@MTS
1049 004224 001001          BNE      .+4

```

```
1050 004226 104000 HLT ;ERROR, DRIVE 0 NOT AT BOT
1051 004230 012777 177777 174552 MOV #-1,@BC ;SET BYTE COUNT=-1
1052 004236 012777 017202 174546 MOV #WBUF,@CA ;INIT CURRENT MEMORY ADDRESS
1053 004244 005000 CLR RO ;INIT DELAY COUNTER
1054 004246 012701 000020 MOV #20,R1
1055 004252 012777 060003 174526 MOV #60003,@MTC ;READ, 800 BPI, GO
1056 004260 005200 INC RO
1057 004262 001376 BNE -2 ;DELAY LONG ENOUGH TO MOVE OFF BOT
1058 004264 005301 DEC R1
1059 004266 001374 BNE -6
1060 004270 032777 000040 174506 BIT #40,@MTS ;TEST FOR BOT
1061 004276 001401 BEQ .+4
1062 004300 104000 HLT ;ERROR, BOT (BIT 5) NOT CLEARED
1063 004302 105777 174500 TSTB @MTC ;TEST FOR CU READY
1064 004306 100401 BMI .+4
1065 004310 104000 HLT ;ERROR, CU READY NOT SET AFTER WRITE FINISHED
1066 004312 005777 174472 TST @BC ;TEST BYTE COUNT TO = 0
1067 004316 001401 BEQ .+4
1068 004320 104000 HLT ;ERROR, BYTE COUNT DIDN'T INCREMENT
1069 004322 022777 017203 174462 CMP #WBUF+1,@CA ;TEST CURRENT MEMORY ADDRESS TO COUNT
1070 004330 001401 BEQ .+4
1071 004332 104000 HLT ;ERROR, CURRENT MEMORY ADDRESS DIDN'T INCREMENT TO 0
1072
1073
1074
1075 ;*****
1076 ;TEST READ A 3 BYTE RECORD
1077 004334 104400 SCOPE
1078 004336 012777 177775 174444 MOV #-3,@BC ;SET BYTE COUNT = -3
1079 004344 012777 017202 174440 MOV #WBUF,@CA ;INIT CURRENT MEMORY ADDRESS
1080 004352 005000 CLR RO ;INIT DELAY COUNTER
1081 004354 005077 174426 CLR @MTC ;SELECT UNIT 0
1082 004360 105777 174422 TSTB @MTC
1083 004364 100375 BPL -4 ;WAIT FOR CU READY
1084 004366 006077 174412 ROR @MTS
1085 004372 103375 BCC -4 ;WAIT FOR TU READY
1086 004374 005000 CLR RO
1087 004376 012777 060003 174402 MOV #60003,@MTC ;READ, 800BPI, GO
1088 004404 105777 174376 TSTB @MTC
1089 004410 100403 BMI .+10
1090 004412 005200 INC RO
1091 004414 001373 BNE -10
1092 004416 104000 HLT ;ERROR, CU READY DIDN'T SET AFTER 3 BYTE RECORD
1093 004420 022777 017205 174364 CMP #WBUF+3,@CA
1094 004426 001401 BEQ .+4
1095 004430 104000 HLT ;ERROR, CURRENT MEMORY ADDRESS DIDN'T INCREMENT
1096 004432 005777 174352 TST @BC
1097 004436 001401 BEQ .+4
1098 004440 104000 HLT ;ERROR, BYTE COUNT DIDN'T INCREMENT TO 0
1099 004442 005777 174340 TST @MTC
1100 004446 100001 BPL .+4
1101 004450 104000 HLT ;ERROR, BIT 15 SET IN COMMAND REGISTER
1102
1103
1104 ;*****
1105 ;TEST WRITE EOF
```

1106	004452	104400			SCOPE		
1107	004454	105777	174326		TSTB	@MTC	
1108	004460	100375			BPL	.-4	
1109	004462	012777	060017	174316	MOV	#60017,@MTC	;REWIND UNIT TO BOT
1110	004470	105777	174312		TSTB	@MTC	
1111	004474	100375			BPL	.-4	
1112	004476	012777	177777	174304	MOV	#-1,@BC	
1113	004504	012777	017202	174300	MOV	#WBUF,@CA	
1114	004512	012777	060007	174266	MOV	#60007,@MTC	;WRITE EOF
1115	004520	105777	174262		TSTB	@MTC	
1116	004524	100375			BPL	.-4	
1117	004526	005777	174256		TST	@BC	
1118	004532	001001			BNE	.+4	
1119	004534	104000			HLT		;ERROR, BYTE COUNT SHOULD NOT INCREMENT ON WRITE EOF
1120	004536	022777	017202	174246	CMP	#WBUF,@CA	
1121	004544	001401			BEQ	.+4	
1122	004546	104000			HLT		;ERROR, CURRENT ADDRESS SHOULD NOT INCREMENT ON WRITE EO

1123
1124
1125
1126

1127	004550	012777	060017	174230	MOV	#60017,@MTC	;REWIND
1128	004556	105777	174224		TSTB	@MTC	
1129	004562	100375			BPL	.-4	
1130	004564	012777	177776	174216	MOV	#-2,@BC	
1131	004572	012777	060011	174206	MOV	#60011,@MTC	;SPACE FORWARD 2 RECORDS
1132	004600	105777	174202		TSTB	@MTC	
1133	004604	100375			BPL	.-4	
1134	004606	032777	040000	174170	BIT	#40000,@MTS	
1135	004614	001001			BNE	.+4	
1136	004616	104000			HLT		;ERROR, EOF (BIT 14) NOT =1
1137	004620	005777	174162		TST	@MTC	
1138	004624	100401			BMI	.+4	
1139	004626	104000			HLT		;ERROR, (BIT 15) OF COMMAND REGISTER NOT=1 WITH EOF=1
1140	004630	022777	177777	174152	CMP	#-1,@BC	
1141	004636	001401			BEQ	.+4	
1142	004640	104000			HLT		;ERROR, BYTE COUNT SHOULD HAVE INCREMENTED FROM -2 TO -1
1143	004642	022777	017202	174142	CMP	#WBUF,@CA	
1144	004650	001401			BEQ	.+4	
1145	004652	104000			HLT		;ERROR, CURRENT ADDRESS REGISTER SHOULD NOT INCREMENT.
1146	004654	052777	010000	174124	BIS	#10000,@MTC	;PWR CLEAR
1147	004662	032777	040000	174114	BIT	#40000,@MTS	
1148	004670	001401			BEQ	.+4	
1149	004672	104000			HLT		;ERROR, PWR CLEAR DIDN'T CLEAR EOF (BIT 14)

1150
1151
1152
1153

1154					;TEST SPACE REVERSE TO STOP IN FIRST EOF		
1155					MOV	#-2,@BC	
1156	004674	012777	177776	174106	MOV	#WBUF,@CA	
1157	004702	012777	017202	174102	MOV	#60013,@MTC	;SPACE REVERSE 2 RECORDS
1158	004710	012777	060013	174070	TSTB	@MTC	
1159	004716	105777	174064		BPL	.-4	
1160	004722	100375			BIT	#40000,@MTS	
1161	004724	032777	040000	174052			


```

1162 004732 001001 BNE .+4
1163 004734 104000 HLT ;ERROR, EOF (BIT 14) NOT =1
1164 004736 032777 000040 174040 BIT #40,@MTS
1165 004744 001401 BEQ .+4
1166 004746 104000 HLT ; ERROR, BOT=1, SHOULD NOT HAVE REACHED BOT
1167 004750 022777 177777 174032 CMP #-1,@BC
1168 004756 001401 BEQ .+4
1169 004760 104000 HLT ;ERROR, BYTE COUNT SHOULD HAVE INCREMENTED FROM -2 TO-1
1170 004762 022777 017202 174022 CMP #WBUF,@CA
1171 004770 001401 BEQ .+4
1172 004772 104000 HLT ;ERROR, CURRENT ADDRESS REGISTER SHOULD NOT INCREMENT
1173
1174
1175
1176
1177
1178

```

```

:*****
:TEST SPACE FORWARD
:FIRST WRITE 2 RECORDS FOLLOWED BY EOF
: SPACE FORWARD 2 RECORDS, SHOULD NOT REACH EOF

```

```

1179 004774 104400 SCOPE
1180 004776 105777 174004 TSTB @MTC
1181 005002 100375 BPL .-4
1182 005004 012777 060017 173774 MOV #60017,@MTC ;REWIND TO BOT
1183 005012 105777 173770 TSTB @MTC
1184 005016 100375 BPL .-4
1185 005020 012777 177775 173762 MOV #-3,@BC
1186 005026 012777 017202 173756 MOV #WBUF,@CA
1187 005034 012777 060005 173744 MOV #60005,@MTC ; WRITE 1ST RECORD
1188 005042 105777 173740 TSTB @MTC
1189 005046 100375 BPL .-4
1190 005050 012777 177775 173732 MOV #-3,@BC
1191 005056 012777 017202 173726 MOV #WBUF,@CA
1192 005064 012777 060005 173714 MOV #60005,@MTC ; WRITE 2ND RECORD
1193 005072 105777 173710 TSTB @MTC
1194 005076 100375 BPL .-4
1195 005100 012777 060007 173700 MOV #60007,@MTC ; WRITE EOF
1196 005106 105777 173674 TSTB @MTC
1197 005112 100375 BPL .-4
1198 005114 012777 060017 173664 MOV #60017,@MTC ; REWIND
1199 005122 105777 173660 TSTB @MTC
1200 005126 100375 BPL .-4
1201 005130 012777 177776 173652 MOV #-2,@BC
1202 005136 012777 060011 173642 MOV #60011,@MTC ; SPACE FORWARD 2 RECORDS
1203 005144 105777 173636 TSTB @MTC
1204 005150 100375 BPL .-4
1205 005152 032777 040000 173624 BIT #40000,@MTS
1206 005160 001401 BEQ .+4
1207 005162 104000 HLT ;ERROR, EOF (BIT 14)=1, SHOULDN'T SPACE THIS FAR
1208
1209 ;NOW SPACE FORWARD TO EOF
1210 005164 012777 060011 173614 MOV #60011,@MTC ;SPACE FORWARD TO EOF WITH BYTE COUNT=0
1211 005172 105777 173610 TSTB @MTC
1212 005176 100375 BPL .-4
1213 005200 032777 040000 173576 BIT #40000,@MTS
1214 005206 001001 BNE .+4
1215 005210 104000 HLT ;ERROR, EOF NOT =1
1216 005212 022777 000001 173570 CMP #1,@BC
1217 005220 001401 BEQ .+4

```

```

1218 005222 104000          HLT          ;ERROR BYTE COUNT SHOULD =1
1219          ;NOW SPACE REVERSE 2 RECORDS (FIRST MUST BACKSPACE OVER EOF)
1220 005224 012777 177775 173556      MOV        #-3,@BC
1221 005232 012777 017202 173552      MOV        #RBUF,@CA
1222 005240 012777 060013 173540      MOV        #60013,@MTC
1223 005246 105777 173534          TSTB      @MTC
1224 005254 100375          BPL        -4
1225 005254 032777 040000 173522      BIT        #40000,@MTC
1226 005262 001001          BNE        +4
1227 005264 104000          HLT
1228 005266 005277 173514          INC        @MTC          ;ERROR, EOF (BIT 14) NOT =1 AFTER BACKSPACE OVER BOT
1229 005272 105777 173510          TSTB      @MTC          ;RESUME BACKSPACE
1230 005276 100375          BPL        -4
1231 005300 100377          BPL
1232 005302 105777 173502          TSTB      @BC
1233 005306 001401          BEQ        +4
1234 005310 104000          HLT          ;ERROR, BYTE COUNT NOT=0
1235 005312 001401          BEQ        +4
1236 005314 104000          HLT          ;ERROR, CURRENT MEMORY ADDRESS SHOULDN'T COUNT ON BACKSP
1237 005316 032777 000040 173460      BIT        #40,@MTC
1238 005324 001401          BEQ        +4
1239 005326 104000          HLT          ;ERROR, BACKSPACE SHOULD NOT HAVE REACHED BOT
1240
1241
1242          ;*****
1243          ;TEST READ TO FIND EOF
1244 005330 104400          SCOPE
1245 005332 105777 173450          TSTB      @MTC
1246 005336 100375          BPL        -4
1247 005340 012777 060007 173440      MOV        #60007,@MTC          ;WRITE EOF
1248 005346 105777 173434          TSTB      @MTC
1249 005352 100375          BPL        -4
1250 005354 012777 060013 173424      MOV        #60013,@MTC          ;BACKSPACE OVER EOF
1251 005362 105777 173420          TSTB      @MTC
1252 005366 100375          BPL        -4
1253 005370 005037 017346          CLR        RBUF
1254 005374 012777 177771 173406      MOV        #-7,@BC
1255 005402 012777 017346 173402      MOV        #RBUF,@CA
1256 005410 012777 060003 173370      MOV        #60003,@MTC          ;READ
1257 005416 105777 173364          TSTB      @MTC
1258 005422 100375          BPL        -4
1259 005424 032777 040000 173354      BIT        #40000,@MTC
1260 005432 001001          BNE        +4
1261 005434 104000          HLT          ;ERROR, EOF (BIT 14) NOT -1 DURING A READ OPERATION
1262 005436 132777 000001 173360      BITB      #1,@SWR          ;IS 7 CHANNEL SELECTED
1263 005444 001006          BNE        TEOF          ;YES
1264 005446 022737 011423 017346      CMP        #11423,RBUF
1265 005454 001401          BEQ        +4
1266 005456 104000          HLT          ;ERROR, EOF (23) NOT TRANSFERRED FOR 2 BYTES DURING READ
1267 005460 000405          BR        TRLE
1268 005462 022737 000377 017346      TEOF:    CMP        #377,RBUF
1269 005470 001401          BEQ        +4
1270 005472 104000          HLT          ;ERROR, EOF (17-7 CHANNEL) NOT TRANSFERRED DURING READ E
1271
1272
1273
    
```

```

1274
1275
1276
1277 005474 104400
1278 005476 105777 173304
1279 005502 100375
1280 005504 012737 177777 017202
1281 005512 012737 177777 017204
1282 005520 012777 177774 173262
1283 005526 012777 017202 173256
1284 005534 012777 060005 173244
1285 005542 105777 173240
1286 005546 100375
1287 005550 012777 177777 173232
1288 005556 012777 060013 173222
1289 005564 105777 173216
1290 005570 100375
1291 005572 005037 017346
1292 005576 005037 017350
1293 005602 012777 177775 173200
1294 005610 012777 017346 173174
1295 005616 012777 060003 173162
1296 005624 105777 173156
1297 005630 100375
1298 005632 032777 001000 173144
1299 005640 001001
1300 005642 104000
1301 005644 005777 173136
1302 005650 100401
1303 005652 104000
1304 005654 022737 177777 017346
1305 005662 001401
1306 005664 104000
1307 005666 022737 000377 017350
1308 005674 001401
1309 005676 104000
1310 005700 052777 010000 173100
1311 005706 032777 001000 173070
1312 005714 001401
1313 005716 104000
1314
1315
1316
1317
1318
1319
1320 005720 104400
1321 005722 105777 173060
1322 005726 100375
1323 005730 006077 173050
1324 005734 103375
1325 005736 012777 177775 173044
1326 005744 012777 017202 173040
1327 005752 012777 060007 173026
1328 005760 105777 173022
1329 005764 100775

```

```

:*****
:TEST RECORD LENGTH ERROR
TRLE:  SCOPE
      TSTB @MTC
      BPL  -4
      MOV  #-1,WBUF
      MOV  #-1,WBUF+2
      MOV  #-4,@BC
      MOV  #WBUF,@CA
      MOV  #60005,@MTC ;WRITE 4 BYTE RECORD
      TSTB @MTC
      BPL  -4
      MOV  #-1,@BC
      MOV  #60013,@MTC ;BACKSPACE
      TSTB @MTC
      BPL  -4
      CLR  RBUF
      CLR  RBUF+2
      MOV  #-3,@BC
      MOV  #RBUF,@CA
      MOV  #60003,@MTC ;READ 3 BYTE RECORD
      TSTB @MTC
      BIT  #1000,@MTS
      BNE  +4
      HLT ;ERROR, RECORD LENGTH ERROR (BIT 9) NOT =1
      TST @MTC
      BMI +4
      HLT ;ERROR, BIT 15 NOT =1 WHEN RLS (BIT 9) =1
      CMP  #-1,RBUF
      BEQ  +4
      HLT ;ERROR, BYTES 1+2 NOT READ PROPERLY
      CMP  #377,RBUF+2
      BEQ  +4
      HLT ;ERROR,BYTE 3 READ ERROR OR SOMETHING TRANSFERED TO BYTE
      BIS  #10000,@MTC ;PWR CLEAR
      BIT  #1000,@MTS
      BEQ  +4
      HLT ;ERROR PWR CLEAR DIDN'T RLE (BIT 9)

```

```

:*****
:TEST ILLEGAL COMMAND TO =1 ON A DATO OR DATOB TO MTC WITH CU READY=0
      SCOPE
      TSTB @MTC
      BPL  -4
      ROR  @MTS
      BCC  -4
      MOV  #-3,@BC
      MOV  #WBUF,@CA
      MOV  #60007,@MTC ;WRITE EOF
      TSTB @MTC
      BMI  -4 ;WAIT FOR CU READY TO CLEAR

```

```

1330 005766 012777 060017 173012      MOV      #60017,@MTC      ;DATO TO MTC WITH CU READY =0
1331 005774 105777 173006      TSTB    @MTC
1332 006000 100375      BPL     .-4
1333 006002 005777 172776      TST     @MTC
1334 006006 100401      BMI     .+4
1335 006010 104000      HLT     ;ERROR, ILLEGAL COMMAND (BIT 15) NOT =1
1336 006012 105777 172770      TSTB    @MTC
1337 006016 100401      BMI     .+4
1338 006020 104000      HLT     ;ERROR, (BIT 15) NOT =1 WITH ILLEGAL COMMAND
1339 006022 105777 172760      TSTB    @MTC
1340 006026 100375      BPL     .-4
1341 006030 006077 172750      ROR     @MTC
1342 006034 103375      BCC     .-4
1343
1344
1345
1346      ;*****
1347      ;TEST ILLEGAL COMMAND BY ISSUING A COMMAND TO TYPE A UNIT WITH SELECT REMOTE =0
1347 006036 104400      SCOPE
1348 006040 012777 003400 172740      MOV      #3400,@MTC      ;SELECT UNIT 7, SELECT REMOTE SHOULD =0
1349 006046 105777 172734      TSTB    @MTC
1350 006052 100375      BPL     .-4
1351 006054 032777 000100 172722      BIT      #100,@MTC
1352 006062 001401      BEQ     .+4
1353 006064 104000      HLT     ;ERROR, SELECT REMOTE (BIT 6) NOT =0 WITH NONEXISTENT U
1354 006066 052777 000017 172712      BIS      #17,@MTC      ;ISSUE REWIND
1355 006074 105777 172706      TSTB    @MTC
1356 006100 100375      BPL     .-4
1357 006102 005777 172676      TST     @MTC
1358 006106 100401      BMI     .+4
1359 006110 104000      HLT     ;ERROR, ILLEGAL COMMAND (BIT 15) NOT =1
1360 006112 052777 010000 172666      BIS      #10000,@MTC   ;PWR CLEAR
1361 006120 005777 172660      TST     @MTC
1362 006124 100001      BPL     .+4
1363 006126 104000      HLT     ;ERROR, POWER CLEAR DIDN'T CLEAR ILC (BIT 15)
1364
1365
1366
1367
1368
1369      ;*****
1370      ;TEST BACKSPACE WHILE AT BOT TO BE IGNORED
1370 006130 104400      SCOPE
1371 006132 032777 000040 172646      BIT      #40,@MTC      ;AT BOT ?
1372 006140 001003      BNE     .+10           ;YES
1373 006142 012777 060017 172636      MOV      #60017,@MTC   ;NO, REWIND
1374 006150 105777 172632      TSTB    @MTC
1375 006154 100375      BPL     .-4
1376 006156 006077 172622      ROR     @MTC
1377 006162 103375      BCC     .-4
1378 006164 012777 177777 172616      MOV      #-1,@BC
1379 006172 012777 000013 172606      MOV      #13,@MTC      ;BACKSPACE
1380 006200 105777 172602      TSTB    @MTC
1381 006204 100375      BPL     .-4
1382 006206 005777 172572      TST     @MTC
1383 006212 100001      BPL     .+4
1384 006214 104000      HLT     ;ERROR, ILC (BIT 15) =1 AFTER BACKSPACE WHILE AT BOT
1385 006216 032777 000040 172560      BIT      #40,@MTC
    
```

```
1386 006224 001001          BNE      .+4
1387 006226 104000          HLT              ;ERROR, NOT AT BOT AFTER BACKSPACE
1388
1389
1390 ;*****
1391 ;TEST REWIND WHILE AT BOT TO BE IGNORED
1392 006230 104400          SCOPE
1393 006232 032777 000040 172544  BIT      #40,@MTS      ;AT BOT?
1394 006240 001003          BNE      .+10         ;YES
1395 006242 012777 060017 172536  MOV      #60017,@MTC   ;NO, REWIND
1396 006250 105777 172532  TSTB    @MTC
1397 006254 100375          BPL      .-4
1398 006256 006077 172522  ROR      @MTS
1399 006262 103375          BCC      .-4
1400 006264 012777 060017 172514  MOV      #60017,@MTC   ;REWIND WHILE AT BOT
1401 006272 105777 172510  TSTB    @MTC
1402 006276 100375          BPL      .-4
1403 006300 005777 172500  TST      @MTS
1404 006304 100001          BPL      .+4
1405 006306 104000          HLT              ;ERROR, ILC(BIT15)=1 AFTER REWIND WHILE AT BOT
1406 006310 032777 000040 172466  BIT      #40,@MTS
1407 006316 001001          BNE      .+4
1408 006320 104000          HLT              ;ERROR, NOT BOT AFTER REWIND
1409
1410
1411
1412 ;*****
1413 ;TEST BAD TAPE ERROR (BIT 8) TO =1
1414 ;USE MAINTENANCE BIT 13 OF MTRD TO SET PREMATURE CU READY TO CAUSE BAD TAPE
1415 006322 104400          SCOPE
1416 006324 012777 177774 172456  MOV      #-4,@BC
1417 006332 012777 017202 172452  MOV      #WBUF,@CA
1418 006340 105777 172442  TSTB    @MTC
1419 006344 100375          BPL      .-4
1420 006346 012777 060005 172432  MOV      #60005,@MTC   ;WRITE, 800 BPI, GO
1421 006354 005777 172430  TST      @BC
1422 006360 001375          BNE      .-4
1423 006362 052777 020000 172426  BIS      #20000,@MTRD  ;SET PREMATURE CU READY
1424 006370 006077 172410  ROR      @MTS
1425 006374 103375          BCC      .-4
1426 006376 032777 000400 172400  BIT      #400,@MTS
1427 006404 001001          BNE      .+4
1428 006406 104000          HLT              ;ERROR, BAD TAPE ERROR (BIT 8) NOT =1
1429 006410 005777 172372  TST      @MTC
1430 006414 100401          BMI      .+4
1431 006416 104000          HLT              ;ERROR, BIT 15 NOT =1 WITH BTE=1
1432 006420 052777 010000 172360  BIS      #10000,@MTC
1433 006426 032777 000400 172350  BIT      #400,@MTS
1434 006434 001401          BEQ      .+4
```

```

1435 006436 104000          HLT          ;ERROR, POWER CLEAR DIDN'T CLEAR BTE (BIT 8)
1436
1437
1438
1439
1440          ;*****
1441          ;TEST NON-EXISTENT MEMORY (BIT 7) AND ERROR (BIT 15) TO =1.
1442 006440 104400          SCOPE
1443 006442 012777 177777 172340  MOV     #-1,@BC          ;INIT BYTE COUNTER
1444 006450 012777 176000 172334 CHG12: MOV     #176000,@CA      ;INIT CURRENT MEMORY ADDRESS FOR NON EXISTENT MEMORY
1445          ;*****
1446 006456 105777 172324          TSTB    @MTC
1447 006462 100375          BPL     .-4
1448 006464 012777 060063 172314  MOV     #60063,@MTC     ;READ, EA=3, 800 BPI, GO
1449 006472 105777 172310          TSTB    @MTC
1450 006476 100375          BPL     .-4
1451 006500 032777 000200 172276  BIT     #200,@MTC
1452 006506 001001          BNE     .+4
1453 006510 104000          HLT          ;ERROR, NON-EXISTENT MEMORY (BIT 7) NOT =1
1454 006512 005777 172270          TST     @MTC
1455 006516 100401          BMI     .+4
1456 006520 104000          HLT          ;ERROR, (BIT 15) NOT =1 WITH NXM (BIT 7) =1
1457 006522 052777 010000 172256  BIS     #10000,@MTC     ;PWR CLEAR
1458 006530 032777 000600 172246  BIT     #600,@MTC
1459 006536 001401          BEQ     .+4
1460 006540 104000          HLT          ;ERROR, POWER CLEAR DIDN'T CLEAR BTE (BIT 8) OR NXM (BIT
1461
1462
1463
1464
1465          ;*****
1466          ;TEST INTERRUPTS
1467          ;INTERRUPT TO 224 WITH PROCESSOR PRIORITY LEVEL 4, BY SETTING INT EN (BIT6)=1
1468 006542 104400          SCOPE
1469 006544 012737 000200 177776  MOV     #200,CC          ;SET PRIORITY LEVEL 4
1470 006552 005077 172224          CLR     @MTVS           ;CLEAR INTERRUPT VECTOR CC
1471 006556 012777 006606 172214  MOV     #IR1,@MTV       ;INIT INTERRUPT RETURN
1472 006564 005000          CLR     R0              ;INIT DELAY COUNT
1473 006566 012777 000100 172212  MOV     #100,@MTC       ;SET INT ENABLE
1474 006574 005200          INC     R0              ;WAIT FOR INTERRUPT
1475 006576 001376          BNE     .-2
1476 006600 005077 172202          CLR     @MTC           ;WAITED TOO LONG WITHOUT INTERRUPT, CLEAR INT ENABLE
1477 006604 104000          HLT          ;ERROR, INTERRUPT ENABLE FAILED TO CAUSE INT.
1478
1479
1480          ;*****
1481          ;TEST FOR PROCESSOR PRIORITY LEVEL 5 TO SUPPRESS INTERRUPT
1482 006606 012737 000240 177776  IR1:  MOV     #240,CC      ;SET PROCESSOR PRIORITY LEVEL 5
1483 006614 012777 006640 172156  MOV     #IR2,@MTV       ;INIT INTERRUPT RETURN
1484 006622 005000          CLR     R0              ;INIT DELAY COUNT
1485 006624 012777 000100 172154  MOV     #100,@MTC       ;SET INT ENABLE
1486 006632 005200          INC     R0              ;WAIT FOR INTERRUPT
1487 006634 001376          BNE     .-2
1488 006636 000401          BR      .+4
1489 006640 104000          IR2:  HLT          ;ERROR, SHOULDN'T HAVE INTERRUPT WITH PROCESSOR PRIORITY
1490

```

1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546

006642 104400
006644 012737 000200 177776
006652 012777 006710 172120
006660 012737 000001 006706
006666 005077 172114
006672 105777 172110
006676 100375
006700 012777 060107 172100
006706 000001
006710 012737 000240 006706
006716 105777 172064
006722 100401
006724 104000

006726 104400
006730 012737 000200 177776
006736 012777 007020 172034
006744 012737 000001 007016
006752 005077 172030
006756 105777 172024
006762 100375
006764 032777 000040 172012
006772 001403
006774 012777 000003 172004
007002 105777 172000
007006 100375
007010 012777 000117 171770
007016 000001
007020 012737 000240 007016
007026 105777 171754
007032 100401
007034 104000

007036 032777 000040 171740
007044 100001
007046 104000
007050 012737 000001 007064
007056 012777 007066 171714
007064 000001
007066 012737 000240 007064
007074 032777 000040 171702
007102 001001
007104 104000

```
*****  
:TEST CU READY TO CAUSE INTERRUPT WITH INT ENABLE =1  
:INT ENABLE (BIT 6) AND GO (BIT 0) SET AT SAME TIME SHOULDN'T CAUSE INTERRUPT  
SCOPE  
MOV #200,CC ;PROCESSOR PRIORITY LEVEL 4  
MOV #IR3,@MTV  
MOV #1,WAIT1  
CLR @MTC  
TSTB @MTC  
BPL -4  
MOV #60107,@MTC ;WRITE EOF, INT ENABLE, GO  
WAIT1: WAIT  
IR3: MOV #240,WAIT1  
TSTB @MTC  
BMI +4  
HLT ;ERROR, INTERRUPT NOT CAUSED BY CU READY  
  
*****  
:TEST REWIND TO CAUSE TWO INTERRUPTS  
:1ST AFTER CU READY AND 2ND AFTER REWIND COMPLETE  
SCOPE  
MOV #200,CC ;PROCESSOR PRIORITY LEVEL 4  
MOV #IR4,@MTV  
MOV #1,WAIT2  
CLR @MTC  
TSTB @MTC  
BPL -4  
BIT #40,@MTS ;AT BOT?  
BEQ +10 ;NO  
MOV #3,@MTC ;WRITE EOF TO MOVE OFF BOT  
TSTB @MTC  
BPL -4  
MOV #117,@MTC ;INT ENABLE, REWIND, GO  
WAIT2: WAIT  
IR4: MOV #240,WAIT2  
TSTB @MTC  
BMI +4  
HLT ;ERROR, INTERRUPT NOT CAUSED BY CU READY  
  
BIT #40,@MTS  
BPL +4  
HLT ;ERROR, SHOULDN'T BE AT BOT SO SOON AFTER 1ST INTERRUPT  
WAIT3: WAIT  
IR5: MOV #1,WAIT3  
MOV #IR5,@MTV  
MOV #240,WAIT3  
BIT #40,@MTS  
BNE +4  
HLT ;ERROR; 2ND INTERRUPT NOT CAUSED BY REWIND COMPLETE  
  
*****  
:DATA TRANSFER TEST
```

```

1547      ;WRITE RECORD, BACKSPACE, READ RECORD
1548      ;REPEAT FOR ALL BYTE PATTERNS FROM 0 THRU DATA PATTERN
1549 007106 005037 001032      CLR      TEMP      ;INITIALIZE DATA PATTERN
1550 007112 012700 017202      WBR:  MOV     #WBUF,RO
1551 007116 013720 001032      MOV     TEMP,(RO)+  ;SET UP WRITE BUFFER
1552 007122 022700 017226      CMP     #WBUF+24,RO
1553 007126 001373      BNE     WBR+4
1554 007130 012777 177754 171652  MOV     #-20.,@BC  ;INIT BYTE COUNT
1555 007136 012777 017202 171646  MOV     #WBUF,@CA  ;INIT CURRENT MEMORY ADDRESS
1556 007144 105777 171636      TSTB   @MTC
1557 007150 100375      BPL     -4
1558 007152 012777 060005 171626  MOV     #60005,@MTC ;WRITE, 800 BPI, GO
1559 007160 105777 171622      TSTB   @MTC
1560 007164 100375      BPL     -4
1561      ;AFTER WRITE, CHECK WRITE BUFFER TO MAKE CERTAIN IT WASN'T MODIFIED
1562 007166 012700 017202      MOV     #WBUF,RO
1563 007172 023720 001032      WBR1:  CMP     TEMP,(RO)+
1564 007176 001401      BEQ     .+4
1565 007200 104000      HLT
1566 007202 022700 017226      CMP     #WBUF+24,RO ;ERROR, DATA BUFFER MODIFIED DURING WRITE
1567 007206 001371      BNE     WBR1
1568      ;BACKSPACE 1 RECORD
1569 007210 012777 177777 171572  MOV     #-1,@BC
1570
1571 007216 012777 060013 171562  MOV     #60013,@MTC
1572 007224 105777 171556      TSTB   @MTC
1573 007230 100375      BPL     -4
1574 007232 012700 017346      MOV     #RBUF,RO
1575 007236 005020      WBR2:  CLR     (RO)+  ;CLEAR READ BUFFER
1576 007240 022700 017372      CMP     #RBUF+24,RO
1577 007244 001374      BNE     WBR2
1578      ;:READ RECORD
1579 007246 012777 177754 171534  MOV     #-20.,@BC  ;UNIT BYTE COUNT
1580 007254 012777 017346 171530  MOV     #RBUF,@CA  ;UNIT CURRENT MEMORY ADDRESS
1581 007262 012777 060003 171516  MOV     #60003,@MTC ;READ,800 BPI, GO
1582 007270 105777 171512      TSTB   @MTC
1583 007274 100375      BPL     -4
1584 007276 005777 171504      TST    @MTC
1585 007302 100001      BPL     .+4
1586 007304 104000      HLT
1587 007306 012700 017346      MOV     #RBUF,RO ;ERROR, ERROR (BIT 15) =1 AFTER READ
1588
1589
1590 007312 023720 001032      WBR3:  CMP     TEMP,(RO)+
1591 007316 001401      BEQ     .+4
1592 007320 104000      HLT
1593 007322 022700 017372      CMP     #RBUF+24,RO ;ERROR, DATA READ NOT EQUAL DATA WRITTEN
1594 007326 001371      BNE     WBR3
1595 007330 105237 001032      INCB   TEMP      ;DONE FOR ALL DATA PATTERNS?
1596 007334 001405      BEQ     WBR4      ;YES, EXIT
1597 007336 113737 001032 001033  MOVB   TEMP,TEMP+1 ;NO
1598 007344 000137 007112      JMP    WBR        ;REPEAT
1599
1600      ;WRITE AND READ A LONG RECORD
1601      ;USES MEMORY OCCUPIED BY THE PROGRAM AS A WRITE BUFFER
1602 007350 012777 177160 171432  WBR4:  MOV     #-400.,@BC
1603 007356 012777 002000 171426  MOV     #2000,@CA

```


CZTMAIO TM,A,B-11 INSTR TST
CZTMAI.P11 12-SEP-79 13:54

MACY11 30A(1052) 12-SEP-79 13:57 PAGE 33
DATA TRANSFER TEST

SEQ 0032

```
1603 007364 105777 171416          TSTB @MTC
1604 007370 100375          BPL   .-4
1605 007372 012777 060005 171406  MOV   #60005,@MTC      ;WRITE, 800 BPI, GO
1606 007400 105777 171402          TSTB @MTC
1607 007404 100375          BPL   .-4
1608 007406 012777 177777 171374  MOV   #-1,@BC
1609 007414 012777 060013 171364  MOV   #60013,@MTC     ;BACKSPACE 1 RECORD
1610 007422 105777 171360          TSTB @MTC
1611 007426 100375          BPL   .-4
1612 007430 012777 177160 171352  MOV   #-400.,@BC
1613 007436 012777 017346 171346  MOV   #RBUF,@CA
1614 007444 012777 060003 171334  MOV   #60003,@MTC     ;READ, 800 BPI, GO
1615 007452 105777 171330          TSTB @MTC
1616 007456 100375          BPL   .-4
1617 007460 012700 002000          MOV   #2000,R0
1618 007464 012701 017346          MOV   #RBUF,R1
1619 007470 022021          WBR5:  CMP   (R0)+,(R1)+  ;DO A DATA COMPARISON
1620 007472 001401          BEQ   .+4
1621 007474 104000          HLT
1622 007476 022701 020166          CMP   #RBUF+400.,R1  ;ERROR, DATA READ NOT EQUAL DATA WRITTEN
1623 007502 001372          BNE   WBR5           ;CHECKED WHOLE BUFFER
                          ;NO
```

```
1624
1625
1626
1627
1628
```

```
*****
;TEST PARITY
;WRITE 3 BYTE RECORD ODD PARITY, READ EVEN PARITY
;BIT 14 OF MTRD =1 SHOULD CAUSE LPC TO BE LOADED IN DATA BUFFER AFTER READ
PAR:  SCOPE
```

```
1631
1632 007504 104400
1633 007506 012737 177777 017202  MOV   #-1,WBUF
1634 007514 012737 177777 017204  MOV   #-1,WBUF+2
1635 007522 012777 177775 171260  MOV   #-3,@BC
1636 007530 012777 017202 171254  MOV   #WBUF,@CA
1637 007536 105777 171244          TSTB @MTC
1638 007542 100375          BPL   .-4
1639 007544 012777 060004 171234  MOV   #60004,@MTC     ;WRITE, 800 BPI, 9 TRACK
1640 007552 132777 000001 171244  BITB  #1,@SWR
1641 007560 001403          BEQ   .+10
1642 007562 042777 020000 171216  BIC   #20000,@MTC     ;MAKE COMMAND 7 TRACK
1643 007570 005277 171212          INC   @MTC           ;GO
1644 007574 105777 171206          TSTB @MTC
1645 007600 100375          BPL   .-4
1646 007602 012777 177777 171200  MOV   #-1,@BC
1647 007610 012777 060012 171170  MOV   #60012,@MTC     ;BACKSPACE, 9 TRACK
1648 007616 132777 000001 171200  BITB  #1,@SWR
1649 007624 001403          BEQ   .+10
1650 007626 042777 020000 171152  BIC   #20000,@MTC     ;MAKE COMMAND 7 TRACK
1651 007634 005277 171146          INC   @MTC           ;GO
1652 007640 105777 171142          TSTB @MTC
1653 007644 100375          BPL   .-4
1654 007646 052777 040000 171142  BIS   #40000,@MTRD
1655 007654 012777 177775 171126  MOV   #-3,@BC
1656 007662 012777 017346 171122  MOV   #RBUF,@CA
1657 007670 012777 064002 171110  MOV   #64002,@MTC
1658 007676 132777 000001 171120  BITB  #1,@SWR
```

1659	007704	001403			BEQ	.+10	
1660	007706	042777	020000	171072	BIC	#20000,@MTC	:MAKE COMMAND 7 TRACK
1661	007714	005277	171066		INC	@MTC	:GO
1662	007720	105777	171062		TSTB	@MTC	
1663	007724	100375			BPL	.-4	
1664	007726	032777	010000	171050	BIT	#10000,@MTC	
1665	007734	001001			BNE	.+4	
1666	007736	104000			HLT		:ERROR, PARITY ERROR (BIT 12) NOT =1
1667	007740	017700	171050		MOV	@MTC,R0	
1668	007744	042700	177000		BIC	#177000,R0	
1669	007750	132777	000001	171046	BITB	#1,@SWR	
1670	007756	001005			BNE	PAR1	
1671	007760	022700	000744		CMP	#744,R0	
1672	007764	001401			BEQ	.+4	
1673	007766	104000			HLT		:ERROR, LPC NOT =744 OR BIT 14 OF MTRD DID'T CAUSE LPC R
1674	007770	000404			BR	PAR2	
1675	007772	022700	000477		CMP	#477,R0	
1676	007776	001401			BEQ	.+4	
1677	010000	104000			HLT		:ERROR, LPC NOT =477 (7 CHANNEL) OR LPC NOT READ
1678							
1679							
1680							
1681	010002	012777	177775	171000			:WRITE EVEN PARITY, READ ODD PARITY
1682	010010	012777	017202	170774	PAR2: MOV	#-3,@BC	
1683	010016	012777	064004	170762	MOV	#RBUF,@CA	
1684	010024	132777	000001	170772	MOV	#64004,@MTC	:WRITE, 800 BPI, 9 TRACK
1685	010032	001403			BITB	#1,@SWR	
1686	010034	042777	020000	170744	BEQ	.+10	
1687	010042	005277	170740		BIC	#20000,@MTC	:MAKE 7 TRACK
1688	010046	105777	170734		INC	@MTC	:GO
1689	010052	100375			TSTB	@MTC	
1690	010054	012777	177777	170726	BPL	.-4	
1691	010062	012777	060012	170716	MOV	#-1,@BC	
1692	010070	132777	000001	170726	MOV	#60012,@MTC	:BACKSPACE
1693	010076	001403			BITB	#1,@SWR	
1694	010100	042777	020000	170700	BEQ	.+10	
1695	010106	005277	170674		BIC	#20000,@MTC	:MAKE COMMAND 7 TRACK
1696	010112	105777	170670		INC	@MTC	:GO
1697	010116	100375			TSTB	@MTC	
1698	010120	052777	040000	170670	BPL	.-4	
1699	010126	012777	177775	170654	BIS	#40000,@MTRD	
1700	010134	012777	017346	170650	MOV	#-3,@BC	
1701	010142	012777	060002	170636	MOV	#RBUF,@CA	
1702	010150	132777	000001	170646	MOV	#60002,@MTC	:READ, 800 BPI, 9 TRACK
1703	010156	001403			BITB	#1,@SWR	
1704	010160	042777	020000	170620	BEQ	.+10	
1705	010166	005277	170614		BIC	#20000,@MTC	:MAKE 7 TRACK
1706	010172	105777	170610		INC	@MTC	:GO
1707	010176	100375			TSTB	@MTC	
1708	010200	032777	010000	170576	BPL	.-4	
1709	010206	001001			BIT	#10000,@MTC	
1710	010210	104000			BNE	.+4	
1711	010212	017700	170576		HLT		:ERROR, PARITY ERROR (BIT 12) NOT =1
1712	010216	042700	177000		MOV	@MTC,R0	
1713	010222	132777	000001	170574	BIC	#177000,R0	
1714	010230	001005			BITB	#1,@SWR	
					BNE	PAR3	

```

1715 010232 022700 000004          CMP    #4,R0
1716 010236 001401          BEQ    .+4
1717 010240 104000          HLT                    ;ERROR, LPC NOT =004 OR LPC NOT READ PROPERLY
1718 010242 000404          BR     PAR4
1719 010244 022700 000077          PAR3: CMP    #77,R0
1720 010250 001401          BEQ    .+4
1721 010252 104000          HLT                    ;ERROR, LPC NOT =77 (7 TRACK)
1722 010254 052777 010000 170524  PAR4: BIS    #10000,@MTC ;PWR CLEAR
1723 010262 032777 010000 170514  BIT    #10000,@MTS
1724 010270 001401          BEQ    .+4
1725 010272 104000          HLT                    ;ERROR, POWER CLEAR DIDN'T CLEAR PARITY ERROR (BIT 11)
1726
1727
1728
1729

```

 ;TEST TIMER (BIT 15) TO BE COMPLIMENTING

```

1730 010274 104400          SCOPE
1731 010276 005000          CLR    R0
1732 010300 005777 170512          TST    @MTRD
1733 010304 001403          BEQ    .+10
1734 010306 005200          INC    R0              ;DELAY LONG TIME
1735 010310 001373          BNE    .-10
1736 010312 104000          HLT                    ;ERROR, TIMER (BIT 15) NEVER =0
1737 010314 005000          CLR    R0
1738 010316 005777 170474          TST    @MTRD
1739 010322 001003          BNE    .+10
1740 010324 005200          INC    R0
1741 010326 001373          BNE    .-10
1742 010330 104000          HLT                    ; ERROR, TIMER (BIT 15) NEVER -1
1743

```

```

1744 010332 132777 000001 170464  BITB   #1,@SWR        ;IS SW0=1 TO INDICATE 7 CHANNEL
1745 010340 001402          BEQ    .+6            ;NO
1746 010342 000137 011454          JMP    MIT            ;YES SKIP CRC TEST
1747
1748
1749

```

 ;TEST CRC GENERATION AND LPC CHARACTER
 ;PROCEDURE USED IS TO WRITE A 4 BYTE RECORD AND READ IT BACK.
 ;THEN THE CRC WRITTEN IS COMPARED WITH CRC CALCULATED.
 ;THEN RECORD IS READ AGAIN AND LPC SHOULD = CRC

```

1750
1751
1752
1753
1754
1755
1756
1757
1758 010346 105037 001032          ;TEST IS REPEATED FOR ALL DATA COMBINATIONS.
1759 010352 004737 011116          CRCTST: CLRB   TEMP    ;INITIALIZE DATA
1760 010356 013737 001032 011224  CRCT1: JSR    PC,CRCPAR ;GENERATE PARITY
1761 010364 013700 001032          MOV    TEMP,CRXOR1   ;SAVE 1ST DATA BYTE (+PARITY)
1762 010370 004737 011156          MOV    TEMP,R0
1763 010374 010037 011226          JSR    PC,CRCROT     ;ROTATE AND COMPLEMENT
1764 010400 013701 001032          MOV    R0,CRROT1    ;SAVE ROTATE
1765 010404 004737 011212          MOV    TEMP,R1
1766 010410 010137 011230          JSR    PC,CRCXOR     ;XOR 2ND BYTE
1767 010414 013700 011230          MOV    R1,CRXOR2
1768 010420 004737 011156          MOV    CRXOR2,R0
1769 010424 010037 011232          JSR    PC,CRCROT
1770 010430 013701 001032          MOV    R0,CRROT2
          MOV    TEMP,R1

```

```

1771 010434 004737 011212      JSR    PC,CRCXOR      ;XOR 3RD BYTE
1772 010440 010137 011234      MOV    R1,CRXOR3
1773 010444 013700 011234      MOV    CRXOR3,R0
1774 010450 004737 011156      JSR    PC,CRCROT
1775 010454 010037 011236      MOV    R0,CRROT3
1776 010460 013701 001032      MOV    TEMP,R1
1777 010464 004737 011212      JSR    PC,CRCXOR      ;XOR 4TH BYTE
1778 010470 010137 011240      MOV    R1,CRXOR4
1779 010474 013700 011240      MOV    CRXOR4,R0
1780 010500 004737 011156      JSR    PC,CRCROT
1781 010504 010037 011242      MOV    R0,CRROT4
1782 010510 010001 011242      MOV    R0,R1          ;COMPLEMENT ALL EXCEPT 4,6
1783 010512 042701 000727      BIC    #727,R1
1784 010516 005100 011242      COM    R0
1785 010520 042700 000050      BIC    #50,R0
1786 010524 050100 011242      BIS    R1,R0
1787 010526 010037 011244      MOV    R0,CRCWRT
1788 010532 042737 177000 011244  BIC    #177000,CRCWRT ;SAVE CRC CALCULATED
1789
1790
1791
1792      :WRITE A FOUR BYTE RECORD
1793      :ALL BYTES ARE = THEREFORE LPC SHOULD = CRC
1794      CWRITE: SCOPE
1795      MOV    TEMP,WBUF
1796      MOV    TEMP,WBUF+1
1797      MOV    WBUF,WBUF+2
1798      MOV    #WBUF,@CA
1799      MOV    #-4,@BC
1800      CLR    @MTC
1801      TSTB   @MTC
1802      BPL   .-4
1803      MOV    #60005,@MTC      ;WRITE, 4 BYTE RECORD, GO
1804      TSTB   @MTC
1805      BPL   .-4
1806      MOV    #-1,@BC
1807      BIC    #16,@MTC
1808      BIS    #13,@MTC      ;BACKSPACE
1809      TSTB   @MTC
1810      BPL   .-4
1811      MOV    #RBUF,@CA
1812      MOV    #-4,@BC
1813      BIC    #16,@MTC
1814      BIS    #3,@MTC      ;READ, 4 BYTE RECORD, GO
1815      TSTB   @MTC
1816      BPL   .-4
1817      CMP    WBUF,RBUF      ;WERE 1ST 2 BYTES WRITTEN AND READ OK?
1818      BEQ    .+4           ;YES
1819      HLT
1820      CMP    WBUF+2,RBUF+2 ;WERE 2ND 2 BYTES WRITTEN AND READ OK?
1821      BEQ    .+4           ;YES
1822      HLT
1823      MOV    @MTRD,R0      ;GET CRC
1824      MOV    @MTRD,R1      ;GET LPC ERROR
1825      BIC    #177000,R0    ;MASK CRC
1826      BIC    #177000,R1    ;MASK LPC ERROR
1827      BEQ    .+4

```

CZTMAIO TM,A,B-11 INSTR TST
CZTMAI.P11 12-SEP-79 13:54

MACY11 30A(1052) 12-SEP-79 13:57 PAGE 37
TEST IS REPEATED FOR ALL DATA COMBINATIONS.

SEQ 0036

```

1827 010762 104000          HLT                ;ERROR, LPC NOT = 0
1828 010764 020037 011244   CMP                RO,CRCWRT
1829 010770 001401          BEQ                .+4
1830 010772 104000          HLT                ;ERROR CRC WRITTEN NOT = CRC CALCULATED
1831 010774 012777 177777 170006  MOV                #-1,@BC
1832 011002 012777 000013 167776  MOV                #13,@MTC                ;BACKSPACE
1833 011010 105777 167772          TSTB               @MTC
1834 011014 100375          BPL                .-4
1835 011016 012777 177774 167764  MOV                #-4,@BC
1836 011024 012777 017346 167760  MOV                #RBUF,@CA
1837 011032 052777 040000 167756  BIS                #40000,@MTRD           ;ENABLE LPC READ
1838 011040 012777 060003 167740  MOV                #60003,@MTC           ;READ, 4 BYTE RECORD, GO
1839 011046 105777 167734          TSTB               @MTC
1840 011052 100375          BPL                .-4
1841
1842
1843 011054 017700 167734          MOV                @MTRD,RO
1844 011060 042700 177000          BIC                #177000,RO
1845 011064 020037 011244   CMP                RO,CRCWRT
1846 011070 001401          BEQ                .+4
1847 011072 104000          HLT                ;ERROR, LPC NOT=CRC
1848 011074 005037 011244   CLR                CRCWRT
1849 011100 005077 167712   CLR                @MTRD                ;ENABLE CRC READ
1850 011104 105237 001032   INCB               TEMP                ;+1 TO DATA PATTERN
1851 011110 001456          BEQ                ZEROCRC
1852 011112 000137 010352   JMP                CRCT1
1853                                     ;CALCULATE PARITY OF DATA TO BE WRITTEN IN CRC TEST (MAKE PARITY ODD)
1854 011116 112737 000001 001033  CRCPAR: MOVB        #1,TEMP+1           ;INITIALIZE ODD PARITY
1855 011124 113701 001032          MOVB               TEMP,R1
1856 011130 105701          CRCP1: TSTB        R1                ;IS DATA=0
1857 011132 001001          BNE                .+4                ;NO
1858 011134 000207          RTS                PC                ;YES, NOW TEMP+1 CONTAINS PARITY BIT
1859 011136 106301          ASLB               R1                ;SHIFT DATA BITS LEFT INTO C BIT
1860 011140 103002          BCC                .+6                ;WAS BIT=0?
1861 011142 105137 001033          COMB               TEMP+1           ;NO, COMPLEMENT PARITY
1862 011146 042737 177000 001032   BIC                #177000,TEMP
1863 011154 000765          BR                 CRCP1                ;DO AGAIN UNTIL DATA=0
1864                                     ;SIMULATE CRC ROTATE, IF CR1 GOES TO 1 COMPLEMENT 4,5,6, AND 7.
1865 011156 042700 177000  CRCR0T: BIC        #177000,RO
1866 011162 006000          ROR                RO
1867 011164 103011          BCC                CRCR1                ;NO EXIT
1868 011166 052700 000400   BIS                #400,RO            ;MAKE BIT1=1
1869 011172 010001          MOV                RO,R1
1870 011174 042701 000074   BIC                #74,R1
1871 011200 005100          COM                RO
1872 011202 042700 000703   BIC                #703,RO
1873 011206 050100          BIS                R1,RO                ;RECOMBINE COMPLEMENTED BITS
1874 011210 000207          CRCR1: RTS                PC                ;EXIT
1875                                     ;XOR RO WITH R1, SAVE RESULT IN R1
1876 011212 010103  CRCXOR: MOV        R1,R3
1877 011214 040001          BIC                RO,R1
1878 011216 040300          BIC                R3,RO
1879 011220 050001          BIS                RO,R1
1880 011222 000207          RTS                PC
1881 011224 000000  CRXOR1: 0
1882 011226 000000  CRROT1: 0

```

1883 011230 000000
 1884 011232 000000
 1885 011234 000000
 1886 011236 000000
 1887 011240 000000
 1888 011242 000000
 1889 011244 000000
 1890
 1891
 1892
 1893
 1894
 1895 011246 104400
 1896 011250 012777 011374 167534
 1897
 1898 011256 012777 177751 167524
 1899
 1900 011264 012777 060005 167514
 1901 011272 105777 167510
 1902 011276 100375
 1903 011300 012777 177777 167502
 1904
 1905 011306 012777 060013 167472
 1906 011314 105777 167466
 1907 011320 100375
 1908
 1909
 1910 011322 012777 011424 167462
 1911
 1912 011330 012777 177751 167452
 1913
 1914 011336 012777 060003 167442
 1915 011344 105777 167436
 1916 011350 100375
 1917 011352 017700 167436
 1918 011356 042700 177000
 1919
 1920 011362 001401
 1921 011364 104000
 1922 011366 104400
 1923 011370 000137 011454
 1924 011374
 1925 011374 020011 041056 052131
 1926 011402 004505 020040 041440
 1927 011410 026122 043114 046054
 1928 011416 026106 006460 000012
 1929 011424
 1930 011424 000014
 1931
 1932
 1933
 1934
 1935
 1936 011454 012777 060017 167324
 1937 011462 105777 167320
 1938 011466 100375

```

CRXOR2: 0
CRROT2: 0
CRXOR3: 0
CRROT3: 0
CRXOR4: 0
CRROT4: 0
CRCWRT: 0

:*****
:TEST FOR ZERO (0) CRC CHARACTER
ZEROCRC:
  MOV      #WCRCOBUFF,@CA ;SET CURRENT MEMORY ADDRESS TO BUFFER
                                ;CONTAINING '0' CRC PATTERN
  MOV      #-27,@BC        ;SET BYTE RECORD COUNTER TO 27(8) BYTES
                                ;TO BE WRITTEN
  MOV      #60005,@MTC     ;800 BPI,9-CHANNEL,UNIT 1,WRITE,GO
  TSTB    @MTC             ;CONTROL UNIT READY?
  BPL     .-4              ;NO - WAIT
  MOV      #-1,@BC        ;SET BYTE RECORD COUNTER FOR A
                                ;BACKSPACE OF 1 RECORD
  MOV      #60013,@MTC    ;BACKSPACE 1 RECORD!!
  TSTB    @MTC             ;CONTROL UNIT READY?
  BPL     .-4              ;NO - WAIT
:AT THIS POINT WE MUST READ THE DATA JUST WRITTEN IN ORDER TO
:GET THE CRC CHARACTER INTO THE MAGTAPE DATA BUFFER
  MOV      #RCRCOBUFF,@CA ;SET CURRENT MEMORY ADDRESS TO DUMP THE
                                ;PATTERN JUST WRITTEN
  MOV      #-27,@BC        ;SET BYTE RECORD COUNTER TO 27(8) BYTES
                                ;TO BE READ
  MOV      #60003,@MTC    ;800 BPI,9-CHANNEL,UNIT 0,READ,GO
  TSTB    @MTC             ;CONTROL UNIT READY?
  BPL     .-4              ;NO - WAIT
  MOV      @MTD,R0        ;GET THE GENERATED CRC CHARACTER
  BIC     #177000,R0      ;MASK UPPER BYTE TO SEGREGATE THE
                                ;TOTAL CRC CHARACTER
  BEQ     1$              ;BRANCH IF IT IS 0
                                ;CRC GENERATED NOT 0!!
  HLT
1$:      SCOPE
  JMP     @MIT            ;GO TO MANUAL INTERVENTION TESTS

WCRCOBUFF:
20011,41056,52131,4505,20040,41440
26122,43114,46054,26106,6460,00012

RCRCOBUFF:
.BLKW 14 ;RESERVE 12 WORDS FOR THE READ BUFFER

:*****
:MANUAL INTERVENTION TESTS
MIT:    MOV      #60017,@MTC ;REWIND
  TSTB    @MTC
  BPL     .-4

```

```

1939 011470 032777 002000 167326      BIT      #2000,@SWR
1940 011476 001402                    BEQ      .+6
1941 011500 000137 012672            JMP      TSTEND
1942 011504 012702 014411            MOV      #MSG3,R2
1943 011510 004737 013462            JSR      PC, TOP
1944 011514 000000                    HALT
1945 011516 104002                    CKSWR
1946 011520 032777 002000 167276      BIT      #2000,@SWR
1947 011526 001402                    BEQ      MITA
1948 011530 000137 012672            JMP      TSTEND
1949
1950
1951
1952
1953 011534 012702 014627      MITA:    MOV      #MSG3A,R2
1954 011540 004737 013462            JSR      PC, TOP
1955 011544 000000                    HALT
1956 011546 104400                    SCOPE
1957 011550 005077 167232            CLR      @MTC
1958 011554 105777 167226            TSTB    @MTC
1959 011560 100401                    BMI      .+4
1960 011562 104000                    HLT
1961 011564 104400                    SCOPE
1962 011566 032777 000040 167210      BIT      #40,@MTS
1963 011574 001001                    BNE     .+4
1964 011576 104000                    HLT
1965
1966
1967
1968
1969 011600 112737 000061 014722      MOV      #61,MSG4+16
1970 011606 012737 000400 001032      MOV      #400,TEMP
1971 011614 012702 014704      USS:    MOV      #MSG4,R2
1972 011620 004737 013462            JSR      PC, TOP
1973 011624 000000                    HALT
1974 011626 104400                    SCOPE
1975 011630 013777 001032 167150      MOV      TEMP,@MTC
1976 011636 005000                    CLR      R0
1977 011640 032777 000100 167136      USS1:   BIT      #100,@MTS
1978 011646 001003                    BNE     USS2
1979 011650 005200                    INC      R0
1980 011652 001372                    BNE     USS1
1981 011654 104000                    HLT
1982 011656 105237 014722      USS2:   INCB    MSG4+16
1983 011662 062737 000400 001032      ADD      #400,TEMP
1984 011670 032777 001000 167126      BIT      #1000,@SWR
1985 011676 001405                    BEQ     USS3
1986 011700 022737 001000 001032      CMP     #1000,TEMP
1987 011706 001342                    BNE     USS
1988 011710 000404                    BR      USS4
1989 011712 022737 004000 001032      USS3:   CMP     #4000,TEMP
1990 011720 001335                    BNE     USS
1991
1992
1993
1994

```

```

;PRINT INSTRUCTIONS TO SELECT OR INHIBIT TEST
;WAIT FOR OPERATOR TO CONTINUE
;CHECK FOR A CNTL G
;INHIBIT TESTS?
;NO
;YES

```

```

;*****

```

```

;MAKE SURE UNIT 0 SELECTED, ONLINE, AT BOT

```

```

;REQUEST TS03 RESPONSE
;SELECT UNIT 0
;ERROR, CU READY NOT SET, IS UNIT 0 SELECTED?
;ERROR, BOT AND TUR NOT SET, IS UNIT 0 ON LINE AT BOT?

```

```

;*****

```

```

;TEST UNIT SELECT SWITCH

```

```

;INITIALIZE TYPEOUT FOR UNIT 1
;INITIALIZE UNIT SELECT #1
;SELECT UNIT
;INIT DELAY
;IS SELECT REMOTE SET
;YES
;NO, HAVE WE WAITED LONG ENOUGH?
;NO, WAIT SOME MORE
;ERROR, PROPER UNIT NOT SELECTED
;INCREMENT UNIT #
;SEE IF TS03
;IF NOT: BR
;SEE IF DONE ALL TS03 UNITS
;IF NOT: BR
;DONE ALL UNITS?
;NO

```

```

;*****

```

```

;TEST ONLINE-OFFLINE SWITCH

```

1995	011722	012702	014744		USS4:	MOV	#MSG5,R2	
1996	011726	004737	013462			JSR	PC, TOP	
1997	011732	000000				HALT		
1998	011734	104400				SCOPE		
1999	011736	005077	167044			CLR	@MTC	;SELECT UNIT 0
2000	011742	032777	000100	167034		BIT	#100,@MTS	:
2001	011750	001401				BEQ	+.4	
2002	011752	104000				HLT		;ERROR, SELECT REMOTE SET, UNIT NOT OFF-LINE
2003								
2004								
2005								
2006								
2007								
2008								
2009								
2010	011754	012702	015016					
2011	011760	004737	013462			MOV	#MSG6,R2	
2012	011764	000000				JSR	PC, TOP	
2013	011766	104400				HALT		
2014	011770	005077	167012			SCOPE		
2015	011774	032777	000004	167002		CLR	@MTC	;SELECT UNIT 0
2016	012002	001001				BIT	#4,@MTS	;IS WRITE LOCK SET?
2017	012004	104000				BNE	+.4	;YES
2018						HLT		;ERROR, WRL (BIT 2) NOT SET WITH WRITE LOCK RING REMOVED
2019								
2020								
2021								
2022	012006	104400						
2023	012010	005077	166774			SCOPE		
2024	012014	005077	166772			CLR	@BC	
2025	012020	012777	060005	166760		CLR	@CA	
2026	012026	105777	166754			MOV	#60005,@MTC	
2027	012032	100375				TSTB	@MTC	
2028	012034	005777	166746			BPL	-.4	
2029	012040	100401				TST	@MTC	
2030	012042	104000				BMI	+.4	
2031	012044	005777	166734			HLT		;ERROR (BIT 15) NOT SET AFTER WRITE WITH WRITE LOCK SET
2032	012050	100401				TST	@MTS	
2033	012052	104000				BMI	+.4	
2034						HLT		;ERROR, ILLEGAL COMMAND (BIT 15) NOT SET AFTER WRITE WITH
2035								
2036								
2037								
2038	012054	104400						
2039	012056	012702	015151			SCOPE		
2040	012062	004737	013462			MOV	#MSG7,R2	
2041	012066	032777	001000	166730		JSR	PC, TOP	
2042	012074	001004				BIT	#1000,@SWR	;SEE IF TS03
2043	012076	012702	015236			BNE	1\$;IF SO: BR
2044	012102	004737	013462			MOV	#MSG7A,R2	
2045	012106	012702	015312			JSR	PC, TOP	
2046	012112	004737	013462		1\$:	MOV	#MSG7B,R2	
2047	012116	000000				JSR	PC, TOP	
2048	012120	104002				HALT		
2049	012122	012777	010000	166656		CKSWR		;CHECK FOR CNTL G
2050	012130	105777	166652			MOV	#10000,@MTC	;POWER CLEAR
						TSTB	@MTC	


```

CZTMAIO TM,A,B-11 INSTR TST MACY11 30A(1052) 12-SEP-79 13:57 B 4 PAGE 41
CZTMAI.P11 12-SEP-79 13:54 TEST OFFLINE FUNCTION TO SET UNIT OFFLINE AND REWIND TO BOT SEQ 0040

2051 012134 100375 BPL .-4
2052 :*****
2053 012136 032777 177701 166640 CHG11: BIT #177701,@MTS
2054 :*****
2055 012144 001001 BNE .+4
2056 012146 104000 HLT ;ERROR, UNIT 0 NOT ON LINE, OFF BOT
2057 012150 104400 SCOPE
2058 012152 012777 000001 166626 MOV #1,@MTC ;GO OFFLINE
2059 012160 105777 166622 TSTB @MTC
2060 012164 100375 BPL .-4
2061 012166 032777 000100 166610 BIT #100,@MTS
2062 012174 001401 BEQ .+4
2063 012176 104000 HLT ;ERROR, SELR (BIT 6) NOT CLEARED BY OFFLINE COMMAND
2064 :RE-SET UNIT
2065 012200 012702 015426 MOV #MSG8,R2
2066 012204 004737 013462 JSR PC, TOP
2067 012210 000000 HALI
2068 012212 104002 CKSWR ;CHECK FOR CNTL G
2069
2070
2071
2072 :*****
2073 :TEST BUS GRANT LATE (BIT 11) TO=1
2074 :HALT PROCESSOR DURING AN NPR SEQUENCE
2075 012214 012702 015507 MOV #MSG9,R2
2076 012220 004737 013462 JSR PC, TOP
2077 012224 000000 HALT
2078 012226 104002 CKSWR ;CHECK FOR CNTL G
2079 012230 032777 000002 166566 BIT #2,@SWR
2080 012236 001047 BNE BGL1
2081 012240 012702 015627 MOV #MSG10,R2
2082 012244 004737 013462 JSR PC, TOP
2083 012250 104400 SCOPE
2084 012252 005077 166530 CLR @MTC
2085 012256 105777 166524 TSTB @MTC
2086 012262 100375 BPL .-4
2087 012264 012777 177756 166516 MOV #-18.,@BC
2088 012272 012777 017202 166512 MOV #@BUF,@CA
2089 012300 012777 060005 166500 MOV #60005,@MTC ;WRITE, 800 BPI, GO
2090 012306 022777 017204 166476 CMP #@BUF+2,@CA
2091 012314 003774 BLE .-6 ;WAIT FOR NPR SEQUENCE TO START
2092 012316 000000 HALT ;CAUSE BGL, WAIT FOR CONTINUE
2093 012320 104002 CKSWR ;CHECK FOR CNTL G
2094 012322 032777 004000 166454 BIT #4000,@MTS
2095 012330 001001 BNE .+4
2096 012332 104000 HLT ;ERROR, BGL (BIT 11) NOT=1.
2097 012334 052777 010000 166444 BIS #10000,@MTC ;POWER CLEAR
2098 012342 032777 004000 166434 BIT #4000,@MTS
2099 012350 001401 BEQ .+4
2100 012352 104000 HLT ;ERROR, POWER CLEAR DIDN'T CLEAR BGL (BIT 11)
2101 012354 000443 BR LASTTEST
2102 012356 012702 015675 BGL1: MOV #MSG11,R2
2103 012362 004737 013462 JSR PC, TOP
2104 012366 104400 SCOPE
2105 012370 005077 166412 CLR @MTC
2106 012374 105777 166406 TSTB @MTC

```

```

2107 012400 100375          BPL      .-4
2108 012402 012777 177756 166400  MOV     #-18.,@BC
2109 012410 012777 017202 166374  MOV     #WBUF,@CA
2110 012416 000000          HALT
2111 012420 012777 060005 166360  MOV     #60005,@MTC      ;WRITE, 800 BPI, GO
2112 012426 000240          NOP
2113 012430 000240          NOP
2114 012432 032777 004000 166344  BIT     #4000,@MTS
2115 012440 001001          BNE     .+4
2116 012442 104000          HLT     ;ERROR, BGL (BIT 11) NOT= 1
2117 012444 052777 010000 166334  BIS     #10000,@MTC      ;POWER CLEAR
2118 012452 032777 004000 166324  BIT     #4000,@MTS
2119 012460 001401          BEQ     .+4
2120 012462 104000          HLT     ;ERROR, POWER CLEAR DIDN'T CLEAR BGL (BIT 11)
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
    ;*****
    ;TEST FOR ILC ERROR ON DELAYED UNIT CHANGE
    LASTTEST:
    MOV     #MSG12,R2      ;FIND OUT IF WE HAVE A 2ND TRANSPORT
    JSR     PC, TOP
    HALT
    CKSWR
    BIT     #4,@SWR        ;CHECK FOR CNTL G
    BEQ     TSTEND         ;DO WE HAVE A 2ND TRANSPORT?
    MOV     #MSG13,R2      ;BRANCH IF NO
    JSR     PC, TOP        ;GET OPERATOR TO SELECT UNIT 1 ON 1ST
    JSR     PC, TOP        ;TRANSPORT, ON-LINE
    BIT     #1000,@SWR     ;SEE IF TS03
    BNE     1$            ;IF SO: BR
    MOV     #MSG13A,R2
    JSR     PC, TOP
    MOV     #MSG13B,R2
    JSR     PC, TOP
    HALT
    CKSWR
    MOV     #MSG16,R2      ;CHECK FOR CNTL G
    JSR     PC, TOP        ;GET OPERATOR TO SELECT UNIT 0 ON 2ND
    JSR     PC, TOP        ;TRANSPORT, ON-LINE
    HALT
    CKSWR
    CLR     @MTC           ;CHECK FOR CNTL G
    MOV     #7777,R0       ;SELECT UNIT 0
    DEC     R0             ;IT SHOULD BE POWERED UP AND
    BEQ     4$            ;ON LINE
    JMP     @#3$           ;DELAY BETWEEN CHANGING UNITS
    MOV     #60417,@MTC    ;BRANCH AFTER DELAY
    TSTB   @MTC           ;KEEP DELAYING
    BPL     .-4           ;LOAD THE COMMAND REGISTER WITH
    TST    @MTS           ;800 BPI (7 OR 9-CHANNEL), REWIND, GO
    BMI     5$           ;THIS COMMAND WILL CHANGE UNITS AND REWIND
    BR     6$           ;CONTROL UNIT READY?
    ;NO - WAIT
    ;ILLEGAL COMMAND ERROR PRESENT?
    ;BRANCH IF YES
    
```

```
2163 012636 104000          5$:   HLT                ; ILC BIT SET - SHOULDN'T BE!!
2164 012640 052777 010000 166140   BIS      #10000,@MTC  ; ISSUE A POWER CLEAR!!
2165 012646 005777 166132          TST      @MTC        ; ILLEGAL COMMAND BIT CLEAR?
2166 012652 100001          BPL      .+4         ; BRANCH IF YES
2167 012654 104000          HLT                ; POWER CLEAR DIDN'T CLEAR ILC BIT
2168 012656 012702 016762          6$:   MOV      #MSG17,R2 ; GET OPERATOR TO RESELECT UNIT 1 TO 0
2169                                ; AND TURN 2ND TRANSPORT OFF-LINE
2170 012662 004737 013462          JSR      PC, TOP
2171 012666 000000          HALT
2172 012670 104400          SCOPE
2173                                ; END OF PASS MESSAGE
2174 012672 012702 017076          TSTEND: MOV     #MSG20,R2
2175 012676 004737 013462          JSR      PC, TOP      ; PRINT END OF PASS
2176 012702 013702 001036          MOV     PCNTR,R2
2177 012706 004737 013202          JSR     PC,OCTPRT    ; PRINT END OF PASS NUMBER
2178 012712 005237 001036          INC     PCNTR        ; BUMP PASS COUNTER
2179 012716 105777 166064          TSTB   @MTC
2180 012722 100375          BPL     .-4
2181 012724 012777 060017 166054   MOV     #60017,@MTC  ; REWIND UNIT
2182 012732 105777 166050          TSTB   @MTC
2183 012736 100375          BPL     .-4
2184 012740 006077 166040          ROR     @MTC
2185 012744 103375          BCC     .-4
2186 012746 013702 000042          MOV     @#42,R2
2187 012752 001405          BEQ    THERE
2188 012754 000005          RESET
2189 012756 004712          $ENDAD: JSR     PC,(R2)
2190 012760 000240          NOP
2191 012762 000240          NOP
2192 012764 000240          NOP
2193 012766 000240          THERE: NOP
2194 012770 032777 004000 166026   BIT     #4000,@SWR   ; SEE IF SINGLE PASS
2195 012776 001401          BEQ    1$           ; IF NOT: BR
2196 013000 000000          HALT
2197 013002 000137 001220          1$:   JMP     BEGIN      ; START OF TEST WITH TRACE OFF
2198
2199                                ; ENTERED WITH SYSTEM TRAP CALL(HLT)
2200                                ; PRINT PC, STATUS REGISTER, COMMAND REGISTER, BYTE COUNT, CURRENT ADDRESS, DATA BUFFER
2201 013006 037727 166012 020000   PRINT: BIT     @SWR,#20000 ; TEST FOR INHIBIT PRINT OUT
2202 013014 001401          BEQ    .+4         ; BRANCH TO PRINT
2203 013016 000466          BR     1$         ; INHIBIT, RETURN TO MAIN STREAM
2204 013020 012702 014274          MOV     #MSG1,R2
2205 013024 005737 013200          TST    PRINT1
2206 013030 001402          BEQ    .+6
2207 013032 012702 014406          MOV     #MSG2,R2
2208 013036 004737 013462          JSR     PC, TOP      ; PRINT ERROR HEADING
2209 013042 005237 013200          INC    PRINT1
2210 013046 011602          MOV    (SP),R2
2211 013050 162702 000002          SUB    #2,R2
2212 013054 004737 013202          JSR    PC,OCTPRT    ; PRINT PC
2213 013060 017702 165720          MOV    @MTC,R2
2214 013064 004737 013202          JSR    PC,OCTPRT    ; PRINT STATUS REGISTER
2215 013070 017702 165712          MOV    @MTC,R2
2216 013074 004737 013202          JSR    PC,OCTPRT    ; PRINT COMMAND REGISTER
2217 013100 017702 165704          MOV    @BC,R2
2218 013104 004737 013202          JSR    PC,OCTPRT    ; PRINT BYTE COUNT
```

```
2219 013110 017702 165676      MOV      @CA,R2
2220 013114 004737 013202      JSR      PC,OCTPRT      ;PRINT CURRENT ADDRESS
2221 013120 017702 165670      MOV      @MID,R2
2222 013124 042702 177000      BIC      #177000,R2
2223 013130 004737 013202      JSR      PC,OCTPRT      ;PRINT DATA BUFFER
2224 013134 017702 165656      MOV      @MTRD,R2
2225 013140 004737 013202      JSR      PC,OCTPRT      ;PRINT TU10 READ LINES
2226 013144 013702 001032      MOV      TEMP,R2
2227 013150 004737 013202      JSR      PC,OCTPRT      ;PRINT TEMP
2228 013154 013702 011244      MOV      CRCWRT,R2
2229 013160 004737 013202      JSR      PC,OCTPRT
2230 013164 005777 165634      TST      @SWR          ;CHECK @SWR FOR HALT SWITCH
2231 013170 100001      BPL      1$
2232 013172 000000      HALT
2233 013174 104002      1$:      CKSWR          ;HALT ON ERROR UP
2234 013176 000207      RTS      PC          ;CHECK FOR CNTL G
2235 013200 000000      PRINT1: 0          ;RETURN TO MAINLINE
2236      ;PRINT OCTAL VALUE IN REGISTER2
2237 013202 012737 000060 013314  OCTPRT: MOV      #'0,CHAR      ;INITIALIZE 1ST NUMBER AS 0
2238 013210 005702      TST      R2          ;IS VALUE POSITIVE
2239 013212 100003      BPL      OCT1        ;YES PRINT 0
2240 013214 012737 000061 013314  OCT1:  MOV      #'1,CHAR      ;NO PRINT 1
2241 013222 004737 013316      JSR      PC,OCTP
2242 013226 006102      ROL      R2
2243 013230 006102      ROL      R2
2244 013232 012737 177773 013312  OCT2:  MOV      #-5,OCT      ;COUNT 5 DIGITS
2245 013240 006102      ROL      R2
2246 013242 006102      ROL      R2
2247 013244 006102      ROL      R2
2248 013246 010237 013314      MOV      R2,CHAR      ;SAVE DIGIT
2249 013252 042737 177770 013314  BIC      #177770,CHAR  ;CLEAR OTHER BITS
2250 013260 052737 000060 013314  BIS      #60,CHAR      ;MAKE ASCII DIGIT
2251 013266 006002      ROR      R2
2252 013270 004737 013316      JSR      PC,OCTP      ;PRINT
2253 013274 006102      ROL      R2
2254 013276 005237 013312      INC      OCT          ;+1 TO DIGIT COUNT
2255 013302 001356      BNE      OCT2        ;NOT DONE
2256 013304 004737 013334      JSR      PC,SP3
2257 013310 000207      RTS      PC          ;EXIT
2258 013312 000000      OCT:      0
2259 013314 000000      CHAR:     0
2260 013316 105777 165500      OCTP:     TSTB      @TPS
2261 013322 100375      BPL      .-4          ;WAIT FOR READY
2262 013324 013777 013314 165466      MOV      CHAR,@TPB   ;PRINT
2263 013332 000207      RTS      PC
2264
2265      ;TYPE 3 SPACES
2266 013334 012702 013346      SP3:     MOV      #SP3A,R2
2267 013340 004737 013462      JSR      PC,TOP
2268 013344 000207      RTS      PC
2269 013346 020057 027440      SP3A:    .ASCII    ;/ /;
2270      .EVEN
2271
2272      ;SCOPE LOOP ROUTINE ENTERED BY USER TRAP
2273 013352 104002      SCOPEA: CKSWR          ;CHECK FOR CNTL G
2274 013354 032777 040000 165442      BIT      #40000,@SWR
```

```

2275 013362 001003          BNE    SCOPEB          ;SCOPE, BIT IS A ONE
2276 013364 011637 013460  MOV    @SP,RETURN     ;NO - SAVE PC FOR NEXT TIME
2277 013370 000002          RTI                    ;RETURN IN SEQUENCE
2278 013372 022606          SCOPEB: CMP    (SP)+,SP ;REPOSITION THE STACK
2279 013374 012637 177776  MOV    (SP)+,CC
2280 013400 000177 000054  JMP    @RETURN        ;SCOPE RETURN
2281
2282          ;SCOPE OR/AND ITERATION LOOP FOR EACH TEST 4000 TIMES
2283 013404 104002  SCOPEC: CKSWR        ;CHECK FOR CNTL G
2284 013406 032777 040000 165410 BIT    #40000,@SWR    ;TEST SWR FOR SCOPE
2285 013414 001366          BNE    SCOPEB          ;YES SCOPE
2286 013416 032777 010000 165400 BIT    #10000,@SWR   ;NO - TEST FOR ITERATION
2287 013424 001007          BNE    SCOPEG          ;INHIBIT ITERATION
2288 013426 023737 013456 001034 CMP    SCOPEF,ICOUNT
2289 013434 001403          BEQ    SCOPEG          ;EXIT - DONE
2290 013436 005237 013456  INC    SCOPEF          ;INCREMENT COUNT
2291 013442 000753          BR    SCOPEB           ;LOOP SOME MORE
2292 013444 005037 013456  SCOPEG: CLR   SCOPEF   ;CLEAR COUNT
2293 013450 011637 013460  MOV    @SP,RETURN     ;SAVE SCOPE RETURN POINTER
2294 013454 000002          RTI                    ;RETURN INLINE-NEXT TEST
2295 013456 000000          SCOPEF: 0              ;COUNT LOCATION FOR ITERATION LOOP
2296 013460 001220          RETURN: BEGIN         ;ADDRESS OF LAST TEST
2297          ;MOV ADDRESS OF MESSAGE TO REGISTER 2
2298          ;THEN JSR PC, TOP
2299 013462 142777 000177 165332 TOP:   BICB   #177,@TPS  ;CLR INT FLAG
2300 013470 112237 013562  MOVB   (R2)+,EOMK     ;MOVE IN EOM MARKER
2301 013474 121237 013562  TOP1:  CMPB   @R2,EOMK  ;COMPARE FOR EOM
2302 013500 001001          BNE    .+4            ;NO
2303 013502 000207          RTS    PC             ;YES, EXIT
2304 013504 121227 000100  CMPB   @R2,#'a
2305 013510 001406          BEQ    TOP2
2306 013512 105777 165304  TSTB   @TPS           ;CK TTY
2307 013516 100375          BPL    .-4            ;WAIT FOR DONE
2308 013520 112277 165274  MOVB   (R2)+,@TPB     ;MOVE CHARACTER
2309 013524 000763          BR    TOP1            ;BRANCH BACK
2310 013526 105777 165270  TOP2:  TSTB   @TPS
2311 013532 100375          BPL    .-4
2312 013534 112777 000215 165256 MOVB   #215,@TPB      ;SEND CARRIAGE RETURN
2313 013542 105777 165254  TSTB   @TPS
2314 013546 100375          BPL    .-4
2315 013550 112777 000212 165242 MOVB   #212,@TPB      ;SEND LINE FEED
2316 013556 005202          INC    R2             ;INCRMTN R2
2317 013560 000745          BR    TOP1            ;NO EOM, SO LOOP
2318 013562          000          EOMK: .BYTE 0
2319          013564          .EVEN
2320
2321
2322
2323          ;*****
2324          ;SOFTWARE SWITCH REGISTER CHANGE ROUTINE
2325
2326 013564 022737 000176 001024 CKSWRR: CMP    #SWREG,SWR ;SOFTWARE SWITCH REG PRESENT
2327 013572 001041          BNE    OUT            ;NO, GET OUT
2328
2329 013574 105777 165226  TSTB   @TKS           ;YES, WAIT FOR
2330 013600 100036          BPL    OUT            ;READY, GET CHARACTER
  
```

```

2331 013602 017737 165222 014104      MOV      @TKB,TIB          ;AND STRIP OFF
2332 013610 042737 177600 014104      BIC      #177600,TIB      ;THE GARBAGE
2333 013616 022737 000007 014104      CMP      #7,TIB          ;IS IT A <^G>
2334 013624 001024          BNE      OUT
2335 013626 012702 017154          MOV      #SCNTG,R2
2336 013632 004737 013462          JSR      PC, TOP
2337 013636 012702 017162          CNTLU:  MOV      #SMSWR,R2
2338 013642 004737 013462          JSR      PC, TOP
2339 013646 017702 165152          MOV      @SWR,R2
2340 013652 004737 013202          JSR      PC, OCTPRT
2341 013656 012702 017171          MOV      #SMNEW,R2
2342 013662 004737 013462          JSR      PC, TOP
2343 013666 005037 014102          CLR      @TEMPST
2344 013672 004737 013700          JSR      PC, $READ      ;GO READ A LINE
2345 013676 000207          OUT:    RTS             ;RETURN TO MAIN BODY OF PROGRAM
2346
2347 013700 005037 014102          $READ:  CLR      TEMPST
2348 013704 012737 000007 014100          MOV      #7,COUNT
2349 013712 004737 014132          1$:    JSR      PC, TTIN      ;GO READ A CHARACTER
2350 013716 042737 177600 014104          BIC      #177600,TIB      ;STRIP OFF GARBAGE
2351 013724 122737 000025 014104          CMPB     #25,TIB          ;IS IT A ^U?
2352 013732 001002          BNE      2$             ;BRANCH IF NOT
2353 013734 005726          3$:    TST      (SP)+      ;POP THE STACK
2354 013736 000737          BR      CNTLU          ;START OVER
2355 013740 122737 000015 014104          2$:    CMPB     #15,TIB      ;IS IT A <CR>?
2356 013746 001012          BNE      4$             ;BRANCH IF NOT
2357 013750 012702 014406          MOV      #MSG2,R2      ;DO A CRLF
2358 013754 004737 013462          JSR      PC, TOP
2359 013760 022737 000007 014100          CMP      #7,COUNT      ;WAS IT FIRST CHARACTER
2360 013766 001037          BNE      7$             ;CHANGE SWR IF NOT FIRST CR
2361 013770 005726          8$:    TST      (SP)+      ;POP THE STACK
2362 013772 000741          BR      OUT            ;GET OUT
2363 013774 122737 000060 014104          4$:    CMPB     #60,TIB
2364 014002 003004          BGT      5$
2365 014004 122737 000067 014104          CMPB     #67,TIB
2366 014012 002005          BGE      6$
2367 014014 012702 017177          5$:    MOV      #SQUEST,R2
2368 014020 004737 013462          JSR      PC, TOP
2369 014024 000743          BR      3$             ;START OVER IF NOT LEGAL CHARACTER
2370 014026 006337 014102          6$:    ASL      TEMPST
2371 014032 006337 014102          ASL      TEMPST
2372 014036 006337 014102          ASL      TEMPST
2373 014042 142737 000060 014104          BICB     #60,TIB          ;GET NITTY-GRITTY
2374 014050 153737 014104 014102          BISB     TIB,TEMPST
2375 014056 005337 014100          DEC      COUNT          ;ONLY WANT 6 DIGITS
2376 014062 001754          BEQ      5$
2377 014064 000712          BR      1$
2378 014066 013777 014102 164730          7$:    MOV      TEMPST,@SWR ;CHANGE SWITCH REGISTER CONTENTS
2379 014074 000735          BR      8$
2380
2381 014076 000000          RDSW:   0
2382 014100 000000          COUNT:  0
2383 014102 000000          TEMPST: 0
2384 014104 000000          TIB:    0
2385
2386

```

2387
2388
2389 014106 011666 000002
2390 014112 162716 000002
2391 014116 013646
2392 014120 062716 110126
2393 014124 013607
2394 014126 013006
2395 014130 013564
2396 104000
2397 104002
2398
2399
2400
2401
2402
2403 014132 005077 164670
2404 014136 005077 164666
2405 014142 005037 014104
2406 014146 005277 164654
2407 014152 105777 164650
2408 014156 100375
2409 014160 017737 164644 014104
2410 014166 105777 164630
2411 014172 100375
2412 014174 113777 014104 164616
2413 014202 000207
2414

:TRAP HANDLER
TRAP30: MOV @SP,2(6)
SUB #2,@SP
MOV @ (6)+,-(6)
ADD #TABLE-104000,@SP
MOV @ (SP)+,PC
TABLE: PRINT
CKSWR
HLT= 104000
CKSWR= 104002

:TTY READ SUBROUTINE
TTIN: CLR @TKS
CLR @TKB
CLR TIB
INC @TKS
TTIN1: TSTB @TKS
BPL TTIN1
MOV @TKB,TIB
TTIN2: TSTB @TPS
BPL TTIN2
MOVB TIB,@TPB
RTS PC

2415	014204	040057	055103	046524	MSG0:	.ASCII	;/@CZTMAIO TM,A,B-11 INSTR TST/;
2416	014212	044501	020060	046524			
2417	014220	040454	041054	030455			
2418	014226	020061	047111	052123			
2419	014234	020122	051524	027524			
2420	014242	040057	042523	020124	MSG01:	.ASCII	;/@SET SW0=1 IF 7 CHANNEL@/;
2421	014250	053523	036460	020061			
2422	014256	043111	033440	041440			
2423	014264	040510	047116	046105			
2424	014272	027500					
2425	014274	040057	020040	041520	MSG1:	.ASCII	;/@ PC STATUS COMAND BYTE CA DATA B READ L TEMP CRC CA
2426	014302	020040	020040	052123			
2427	014310	052101	051525	020040			
2428	014316	047503	040515	042116			
2429	014324	020040	041040	052131			
2430	014332	020105	020040	020040			
2431	014340	040503	020040	020040			
2432	014346	040504	040524	041040			
2433	014354	020040	042522	042101			
2434	014362	046040	020040	052040			
2435	014370	046505	020120	041440			
2436	014376	041522	041440	046101			
2437	014404	027500					
2438	014406	040057	057		MSG2:	.ASCII	;/@/;
2439	014411	057	052100	020117	MSG3:	.ASCII	;/@TO INHIBIT MANUAL INTERVENTION TEST: SET SW10=1 AND PRESS CONTINUE;
2440	014416	047111	044510	044502			
2441	014424	020124	040515	052516			
2442	014432	046101	044440	052116			
2443	014440	051105	042526	052116			
2444	014446	047511	020116	042524			
2445	014454	052123	020072	051440			
2446	014462	052105	051440	030527			
2447	014470	036460	020061	047101			
2448	014476	020104	051120	051505			
2449	014504	020123	047503	052116			
2450	014512	047111	042525				
2451	014516	047500	044124	051105			.ASCII ;@OTHERWISE SET SW10=0, SELECT UNIT 0, ON LINE, AT BOT AND PRESS CONTINU
2452	014524	044527	042523	051440			
2453	014532	052105	051440	030527			
2454	014540	036460	026060	051440			
2455	014546	046105	041505	020124			
2456	014554	047125	052111	030040			
2457	014562	020054	047117	046040			
2458	014570	047111	026105	040440			
2459	014576	020124	047502	020124			
2460	014604	047101	020104	051120			
2461	014612	051505	020123	047503			
2462	014620	052116	047111	042525			
2463	014626	057					
2464	014627	057	044500	020106	MSG3A:	.ASCII	;/@IF UNIT IS TS03, SET SW9=1, PRESS CONTINUE/;
2465	014634	047125	052111	044440			
2466	014642	020123	051524	031460			
2467	014650	020054	042523	020124			
2468	014656	053523	036471	026061			
2469	014664	050040	042522	051523			
2470	014672	041440	047117	044524			

2471	014700	052516	027505		
2472	014704	040057	042523	042514	MSG4: .ASCII ;/@SELECT UNIT 1, PRESS CONTINUE/;
2473	014712	052103	052440	044516	
2474	014720	020124	026061	050040	
2475	014726	042522	051523	041440	
2476	014734	047117	044524	052516	
2477	014742	027505			
2478	014744	040057	042523	042514	MSG5: .ASCII ;/@SELECT UNIT 0, OFF-LINE, PRESS CONTINUE/;
2479	014752	052103	052440	044516	
2480	014760	020124	026060	047440	
2481	014766	043106	046055	047111	
2482	014774	026105	050040	042522	
2483	015002	051523	041440	047117	
2484	015010	044524	052516	027505	
2485	015016	040057	044504	046523	MSG6: .ASCII ;/@DISMOUNT TAPE, REMOVE WRITE LOCK RING, MOUNT TAPE;
2486	015024	052517	052116	052040	
2487	015032	050101	026105	051040	
2488	015040	046505	053117	020105	
2489	015046	051127	052111	020105	
2490	015054	047514	045503	051040	
2491	015062	047111	026107	046440	
2492	015070	052517	052116	052040	
2493	015076	050101	105		
2494	015101	100	042523	042514	.ASCII ;/@SELECT UNIT 0, ON LINE, PRESS CONTINUE/;
2495	015106	052103	052440	044516	
2496	015114	020124	026060	047440	
2497	015122	020116	044514	042516	
2498	015130	020054	051120	051505	
2499	015136	020123	047503	052116	
2500	015144	047111	042525	057	
2501	015151	057	042100	051511	MSG7: .ASCII ;/@DISMOUNT TAPE, REPLACE WRITE LOCK RING, MOUNT TAPE/;
2502	015156	047515	047125	020124	
2503	015164	040524	042520	020054	
2504	015172	042522	046120	041501	
2505	015200	020105	051127	052111	
2506	015206	020105	047514	045503	
2507	015214	051040	047111	026107	
2508	015222	046440	052517	052116	
2509	015230	052040	050101	027505	
2510	015236	040057	047515	042526	MSG7A: .ASCII ;/@MOVE TAPE SHORT DISTANCE FORWARD FROM BOT/;
2511	015244	052040	050101	020105	
2512	015252	044123	051117	020124	
2513	015260	044504	052123	047101	
2514	015266	042503	043040	051117	
2515	015274	040527	042122	043040	
2516	015302	047522	020115	047502	
2517	015310	027524			
2518	015312	040057	042523	042514	MSG7B: .ASCII ;/@SELECT UNIT 0, ON LINE, PRESS CONTINUE;
2519	015320	052103	052440	044516	
2520	015326	020124	026060	047440	
2521	015334	020116	044514	042516	
2522	015342	020054	051120	051505	
2523	015350	020123	047503	052116	
2524	015356	047111	042525		
2525	015362	052500	044516	020124	.ASCII ;@UNIT SHOULD GO OFFLINE AND REWIND@/;
2526	015370	044123	052517	042114	

2527 015376 043440 020117 043117
2528 015404 046106 047111 020105
2529 015412 047101 020104 042522
2530 015420 044527 042116 027500
2531 015426 040057 042523 042514
2532 015434 052103 052440 044516
2533 015442 020124 026060 047440
2534 015450 020116 044514 042516
2535 015456 020054 052101 041040
2536 015464 052117 020054 051120
2537 015472 051505 020123 047503
2538 015500 052116 047111 042525
2539 015506 057
2540 015507 057 044500 020106
2541 015514 051120 041517 051505
2542 015522 047523 020122 051511
2543 015530 040440 050040 050104
2544 015536 032055 026065 051440
2545 015544 052105 051440 020127
2546 015552 036461 061
2547 015555 100 043111 040440
2548 015562 054516 047440 044124
2549 015570 051105 020054 042523
2550 015576 020124 053523 030440
2551 015604 030075 020054 051120
2552 015612 051505 020123 047503
2553 015620 052116 047111 042525
2554 015626 057
2555 015627 057 050100 047522
2556 015634 042503 051523 051117
2557 015642 053440 046111 020114
2558 015650 040510 052114 020054
2559 015656 051120 051505 020123
2560 015664 047503 052116 047111
2561 015672 042525 057
2562 015675 057 050100 047522
2563 015702 042503 051523 051117
2564 015710 020040 044527 046114
2565 015716 044040 046101 026124
2566 015724 050040 052125 023440
2567 015732 047105 041101 042514
2568 015740 044055 046101 023524
2569 015746 051440 020127 047117
2570 015754 023440 040510 052114
2571 015762 047
2572 015763 100 052520 020124
2573 015770 051447 044455 051516
2574 015776 026524 026523 052502
2575 016004 020123 054503 046103
2576 016012 023505 051440 020127
2577 016020 047117 023440 026523
2578 016026 052502 020123 054503
2579 016034 046103 023505
2580 016040 050100 042522 051523
2581 016046 023440 047503 052116
2582 016054 047111 042525 020047

MSG8: .ASCII ;@SELECT UNIT 0, ON LINE, AT BOT, PRESS CONTINUE/;

MSG9: .ASCII ;@IF PROCESSOR IS A PDP-45, SET SW 1=1;

.ASCII ;@IF ANY OTHER, SET SW 1=0, PRESS CONTINUE/;

MSG10: .ASCII ;@PROCESSOR WILL HALT, PRESS CONTINUE/;

MSG11: .ASCII ;@PROCESSOR WILL HALT, PUT 'ENABLE-HALT' SW ON 'HALT';

.ASCII ;@PUT 'S-INST-S-BUS CYCLE' SW ON 'S-BUS CYCLE';

.ASCII ;@PRESS 'CONTINUE' 6 TIMES;

2583	016062	020066	044524	042515	
2584	016070	123			
2585	016071	100	052520	020124	.ASCII ;@PUT SW'S BACK TO 'ENABLE' & 'S-INST', PRESS 'CONTINUE'@/;
2586	016076	053523	051447	020040	
2587	016104	040502	045503	052040	
2588	016112	020117	042447	040516	
2589	016120	046102	023505	023040	
2590	016126	023440	026523	047111	
2591	016134	052123	026047	050040	
2592	016142	042522	051523	023440	
2593	016150	047503	052116	047111	
2594	016156	042525	040047	057	
2595	016163	057	051500	052105	MSG12: .ASCII ;/@SET SW2 = 1 IF A 2ND TRANSPORT IS AVAILABLE;
2596	016170	051440	031127	036440	
2597	016176	030440	044440	020106	
2598	016204	020101	047062	020104	
2599	016212	051124	047101	050123	
2600	016220	051117	020124	051511	
2601	016226	040440	040526	046111	
2602	016234	041101	042514		
2603	016240	052100	042510	020116	.ASCII ;@THEN PRESS CONTINUE/;
2604	016246	051120	051505	020123	
2605	016254	047503	052116	047111	
2606	016262	042525	057		
2607	016265	057	051500	046105	MSG13: .ASCII ;/@SELECT UNIT 1, ON-LINE, ON 1ST TRANSPORT/;
2608	016272	041505	020124	047125	
2609	016300	052111	030440	020054	
2610	016306	047117	046055	047111	
2611	016314	026105	047440	020116	
2612	016322	051461	020124	051124	
2613	016330	047101	050123	051117	
2614	016336	027524			
2615	016340	040057	047101	020104	MSG13A: .ASCII ;/@AND MOVE UNIT 1 ON 1ST TRANSPORT OFF BOT/;
2616	016346	047515	042526	052440	
2617	016354	044516	020124	020061	
2618	016362	047117	030440	052123	
2619	016370	052040	040522	051516	
2620	016376	047520	052122	047440	
2621	016404	043106	041040	052117	
2622	016412	057			
2623	016413	057	052100	042510	MSG13B: .ASCII ;@THEN PRESS CONTINUE/;
2624	016420	020116	051120	051505	
2625	016426	020123	047503	052116	
2626	016434	047111	042525	057	
2627	016441	057	046100	053517	MSG14: .ASCII ;/@LOW BYTE OF COMMAND REGISTER LOADED INCORRECTLY;
2628	016446	041040	052131	020105	
2629	016454	043117	041440	046517	
2630	016462	040515	042116	051040	
2631	016470	043505	051511	042524	
2632	016476	020122	047514	042101	
2633	016504	042105	044440	041516	
2634	016512	051117	042522	052103	
2635	016520	054514			
2636	016522	052500	044523	043516	.ASCII ;@USING BYTE INSTRUCTION/;
2637	016530	041040	052131	020105	
2638	016536	047111	052123	052522	

2639	016544	052103	047511	027516	
2640	016552	040057	044510	044107	MSG15: .ASCII ;/HIGH BYTE OF COMMAND REGISTER LOADED INCORRECTLY;
2641	016560	041040	052131	020105	
2642	016566	043117	041440	046517	
2643	016574	040515	042116	051040	
2644	016602	043505	051511	042524	
2645	016610	020122	047514	042101	
2646	016616	042105	044440	041516	
2647	016624	051117	042522	052103	
2648	016632	054514			
2649	016634	052500	044523	043516	.ASCII ;@USING BYTE INSTRUCTION/;
2650	016642	041040	052131	020105	
2651	016650	047111	052123	052522	
2652	016656	052103	047511	027516	
2653	016664	040057	042523	042514	MSG16: .ASCII ;/SELECT UNIT 0 ON 2ND TRANSPORT, ON-LINE;
2654	016672	052103	052440	044516	
2655	016700	020124	020060	047117	
2656	016706	031040	042116	052040	
2657	016714	040522	051516	047520	
2658	016722	052122	020054	047117	
2659	016730	046055	047111	105	
2660	016735	100	044124	047105	.ASCII ;@THEN PRESS CONTINUE/;
2661	016742	050040	042522	051523	
2662	016750	041440	047117	044524	
2663	016756	052516	027505		
2664	016762	040057	042522	042523	MSG17: .ASCII ;/RESELECT UNIT 1 TO 0, AND TURN 2ND TRANSPORT OFF-LINE;
2665	016770	042514	052103	052440	
2666	016776	044516	020124	020061	
2667	017004	047524	030040	020054	
2668	017012	047101	020104	052524	
2669	017020	047122	031040	042116	
2670	017026	052040	040522	051516	
2671	017034	047520	052122	047440	
2672	017042	043106	046055	047111	
2673	017050	105			
2674	017051	100	044124	047105	.ASCII ;@THEN PRESS CONTINUE/;
2675	017056	050040	042522	051523	
2676	017064	041440	047117	044524	
2677	017072	052516	027505		
2678	017076	040057	042500	042116	MSG20: .ASCII ;/END OF PASS: /;
2679	017104	047440	020106	040520	
2680	017112	051523	020072	057	
2681	017117	057	040100	040503	MSG21: .ASCII ;/CANNOT TEST LOAD MEDIUM/;
2682	017124	047116	052117	052040	
2683	017132	051505	020124	047514	
2684	017140	042101	046440	042105	
2685	017146	052511	040115	027500	
2686	017154	040057	043536	027500	\$CNTG: .ASCII ;/^G@/;
2687	017162	040057	053523	036522	\$MSWR: .ASCII ;/@SWR=/;
2688	017170	057			
2689	017171	057	042516	036527	\$MNEW: .ASCII ;/NEW=/;
2690	017176	057			
2691	017177	057	027477		\$QUEST: .ASCII ;/?!/;
2692					.EVEN
2693	017202	000000			WBUF: 0
2694		017346			. =WBUF+100.

CZTMAIO TM,A,B-11 INSTR TST
CZTMAI.P11 12-SEP-79 13:54

MACY11 30A(1052) 12-SEP-79 13:57 N 4 PAGE 53
TTY READ SUBROUTINE

SEQ 0052

2695 017346 000000
2696 000001

RBUF: 0
.END

BADDAT	002344	622*	653#											
BC	001010	310#	386	450	806*	807	946*	961	972*	992	1004*	1051*	1066	1077*
		1096	1112*	1117	1130*	1140	1156*	1167	1185*	1190*	1201*	1216	1220*	1232
		1254*	1282*	1287*	1293*	1325*	1378*	1416*	1421	1442*	1554*	1569*	1579*	1601*
		1608*	1612*	1635*	1646*	1655*	1681*	1690*	1699*	1798*	1805*	1811*	1831*	1835*
		1898*	1903*	1912*	2023*	2087*	2108*	2217						
BEGIN	001220	351	356#	357	2197	2296								
BGL1	012356	2080	2102#											
BUFF =	000776	269#	325	356										
BYTEOP	002346	629	635	656#										
CA	001012	311#	395	459	819*	820	947*	964	973*	989	1005*	1052*	1069	1078*
		1092	1113*	1120	1143	1157*	1170	1186*	1191*	1221*	1255*	1283*	1294*	1326*
		1417*	1444*	1555*	1580*	1602*	1613*	1636*	1656*	1682*	1700*	1797*	1810*	1856*
		1896*	1910*	2024*	2088*	2090	2109*	2219						
		266#	1469*	1482*	1496*	1514*	2279*							
CC =	177776	2237*	2240*	2248*	2249*	2250*	2259#	2262						
CHAR	013314	2053#												
CHGI1	012136	1444#												
CHGI2	006450	574#												
CHGI3	002112	575#												
CHKBYT	002112	355	1945	2048	2068	2078	2093	2129	2142	2147	2233	2273	2283	2397#
CKSWR =	104002	2326#	2395											
CKSWRR	013564	2337#	2354											
CNTLU	013636	2348*	2359	2375*	2382#									
COUNT	014100	1759	1854#											
CRCPAR	011116	1856#	1863											
CRCP1	011130	1762	1768	1774	1780	1865#								
CRCROT	011156	1867	1874#											
CRCR1	011210	1758#												
CRCTST	010346	1759#	1852											
CRCT1	010352	359*	1787*	1788*	1828	1845	1848*	1889#	2228					
CRCWRT	011244	1765	1771	1777	1876#									
CRCXOR	011212	1763*	1882#											
CRROT1	011226	1769*	1884#											
CRROT2	011232	1775*	1886#											
CRROT3	011236	1781*	1888#											
CRROT4	011242	1760*	1881#											
CRXOR1	011224	1766*	1767	1883#										
CRXOR2	011230	1772*	1773	1885#										
CRXOR3	011234	1778*	1779	1887#										
CRXOR4	011240	1793#												
CWRITE	010540	265#												
EMT =	014106	2300*	2301	2318#										
EOPK	013562	583*	585*	641#										
EXPECT	002336	619*	621*	649#										
GOODDA	002342	370	379	388	397	406	415	424	433	443	452	461	470	479
HLT =	104000	488	497	507	517	522	527	532	537	542	547	556	566	590
		626	666	671	676	686	696	705	710	715	720	727	732	737
		747	757	766	777	788	793	798	809	822	835	848	852	861
		869	877	885	895	904	913	921	945	957	960	963	966	986
		991	994	997	1018	1021	1028	1033	1050	1062	1065	1068	1071	1091
		1094	1098	1101	1119	1122	1136	1139	1142	1145	1149	1163	1166	1169
		1172	1207	1215	1218	1227	1234	1236	1239	1261	1266	1270	1300	1303
		1306	1309	1313	1335	1338	1353	1359	1363	1384	1387	1405	1408	1428
		1431	1435	1453	1456	1460	1477	1489	1507	1530	1535	1542	1565	1586
		1592	1621	1666	1673	1677	1710	1717	1721	1725	1736	1742	1818	1821

		1827	1830	1847	1921	1960	1964	1981	2002	2017	2030	2033	2056	2063
ICOUNT	001034	2096	2100	2116	2120	2163	2167	2396#						
IR1	006606	320#	804*	817*	830*	2288								
IR2	006640	1471	1482#											
IR3	006710	1483	1489#											
IR4	007020	1497	1504#											
IR5	007066	1515	1527#											
LASTTE	012464	1537	1539#											
MIT	011454	2101	2125#											
MITA	011534	1746	1923	1936#										
MSG0	014204	1947	1953#											
MSG01	014242	345	2415#											
MSG1	014274	352	2420#											
MSG10	015627	2204	2425#											
MSG11	015675	2081	2555#											
MSG12	016163	2102	2562#											
MSG13	016265	2126	2595#											
MSG13A	016340	2132	2607#											
MSG13B	016413	2137	2615#											
MSG14	016441	2139	2623#											
MSG15	016552	588	2627#											
MSG16	016664	624	2640#											
MSG17	016762	2143	2653#											
MSG2	014406	2168	2664#											
MSG20	017076	2207	2357	2438#										
MSG21	017117	2174	2678#											
MSG3	014411	330	2681#											
MSG3A	014627	1942	2439#											
MSG4	014704	1953	2464#											
MSG5	014744	1969*	1971	1982*	2472#									
MSG6	015016	1995	2478#											
MSG7	015151	2010	2485#											
MSG7A	015236	2039	2501#											
MSG7B	015312	2043	2510#											
MSG8	015426	2045	2518#											
MSG9	015507	2065	2531#											
MTC	001006	2075	2540#											
		309#	368	413	430*	431	440*	449*	458*	467*	476*	477	485*	494*
		495	503*	504*	505	514*	515	519*	520	524*	525	529*	530	534*
		535	539*	540	544*	545	553*	554	562*	563*	564	576*	577*	578
		586	598*	604*	609	614	622	636*	656*	663*	664	668*	669	673*
		674	683*	684	692*	693*	694	702*	703	707*	708	712*	713	717*
		718	724*	725	729*	730	734*	735	744*	745	753*	754*	755	763*
		764	773*	774*	775	785*	786	790*	791	795*	796	858*	910*	932*
		933	941*	950*	958	975*	976	981*	982	995	1006	1008*	1009	1013*
		1014	1043*	1044	1055*	1063	1080*	1081	1086*	1087	1099	1107	1109*	1110
		1114*	1115	1127*	1128	1131*	1132	1137	1146*	1158*	1159	1180	1182*	1183
		1187*	1188	1192*	1193	1195*	1196	1198*	1199	1202*	1203	1210*	1211	1222*
		1223	1228*	1229	1245	1247*	1248	1250*	1251	1256*	1257	1259	1278	1284*
		1285	1288*	1289	1295*	1296	1301	1310*	1321	1327*	1328	1330*	1331	1336
		1339	1348*	1349	1354*	1355	1360*	1371	1373*	1374	1379*	1380	1395*	1396
		1400*	1401	1418	1420*	1429	1432*	1446	1448*	1449	1454	1457*	1473*	1476*
		1485*	1499*	1500	1502*	1505	1517*	1518	1522*	1523	1525*	1528	1556	1558*
		1559	1571*	1572	1581*	1582	1584	1603	1605*	1606	1609*	1610	1614*	1615
		1637	1639*	1642*	1643*	1644	1647*	1650*	1651*	1652	1657*	1660*	1661*	1662
		1683*	1686*	1687*	1688	1691*	1694*	1695*	1696	1701*	1704*	1705*	1706	1722*

		1799*	1800	1802*	1803	1806*	1807*	1808	1812*	1813*	1814	1832*	1833	1838*
		1839	1900*	1901	1905*	1906	1914*	1915	1936*	1937	1957*	1958	1975*	1999*
		2014*	2025*	2026	2028	2049*	2050	2058*	2059	2084*	2085	2089*	2097*	2105*
		2106	2111*	2117*	2148*	2155*	2158	2164*	2179	2181*	2182	2215		
MTD	001014	312#	404	468	832*	833	1667	1711	1822	1843	1917	2221		
MTRD	001016	313#	422	486	845*	846	849*	850	1423*	1654*	1698*	1732	1738	1823
		1837*	1849*	2224										
MTS	001004	308#	377	441	859	867	875	883	893	902	911	919	935*	937
		942	955	978*	1011*	1016	1019*	1022	1024	1026	1029	1031*	1046*	1048
		1060	1083*	1134	1147	1161	1164	1205	1213	1225	1237	1298	1311	1323*
		1333	1341*	1351	1357	1361	1376*	1382	1385	1393	1398*	1403	1406	1424*
		1426	1433	1451	1458	1520	1533	1540	1664	1708	1723	1962	1977	2000
		2015	2031	2053	2061	2094	2098	2114	2118	2160	2165	2184*	2213	
MTV	001000	306#	1471*	1483*	1497*	1515*	1537*							
MTVS	001002	307#	1470*											
NOP	= 000240	267#												
OCT	013312	2244*	2254*	2258#										
OCTP	013316	2241	2252	2260#										
OCTPRT	013202	2177	2212	2214	2216	2218	2220	2223	2225	2227	2229	2237#	2340	
OCT1	013222	2239	2241#											
OCT2	013240	2245#	2255											
OUT	013676	2327	2330	2334	2345#	2362								
PAR	007504	1632#												
PAR1	007772	1670	1675#											
PAR2	010002	1674	1681#											
PAR3	010244	1714	1719#											
PAR4	010254	1718	1722#											
PCNTR	001036	321#	327*	2176	2178*									
PRINT	013006	2201#	239#											
PRINT1	013200	358*	2205	2209*	2235#									
RBUF	017346	1253*	1255	1264	1268	1291*	1292*	1294	1304	1307	1574	1576	1580	1587
		1593	1613	1618	1622	1656	1700	1810	1816	1819	1836	2695#		
RCRCOB	011424	1910	1929#											
RDSW	014076	2381#												
RECEIV	002340	586*	645#											
RETURN	013460	357*	2276*	2280	2293*	2296#								
SCOPE =	104400	268#	366	375	384	393	402	411	420	429	439	448	457	466
		475	484	493	502	513	518	523	528	533	538	543	552	561
		573	662	667	672	682	691	701	706	711	716	723	728	733
		743	752	762	772	784	789	794	803	816	829	844	857	866
		874	882	892	901	909	918	931	971	1003	1042	1076	1106	1179
		1244	1277	1320	1347	1370	1392	1415	1441	1468	1495	1513	1632	1730
		1793	1895	1922	1956	1961	1974	1998	2013	2022	2038	2057	2083	2104
		2125	2172											
SCOPEA	013352	2273#												
SCOPEB	013372	2275	2278#	2285	2291									
SCOPEC	013404	263	2283#											
SCOPEF	013456	2288	2290*	2292*	2295#									
SCOPEG	013444	2287	2289	2292#										
SP3	013334	2256	2266#											
SP3A	013346	2266	2269#											
START	001040	304	325#											
SUSWR	001102	329	333#											
SWR	001024	316#	326*	336	342*	890	1262	1640	1648	1658	1669	1684	1692	1702
		1713	1744	1939	1946	1984	2041	2079	2130	2135	2194	2201	2230	2274
		2284	2286	2326	2339	2378*								

SSVPC = 000040
= 017350

295#	300												
256#	258	259#	262#	295	296#	298#	300#	301#	303#	305#	369	378	
387	396	405	414	423	432	442	451	460	469	478	487	496	
506	516	521	526	531	536	541	546	555	565	665	670	675	
685	695	704	709	714	719	726	731	736	746	756	765	776	
787	792	797	808	821	834	847	851	860	868	876	884	894	
903	912	920	934	936	952	954	956	959	962	965	977	979	
983	985	990	993	996	1007	1010	1012	1015	1017	1020	1023	1025	
1027	1030	1032	1045	1047	1049	1057	1059	1061	1064	1067	1070	1082	
1084	1088	1090	1093	1097	1100	1108	1111	1116	1118	1121	1129	1133	
1135	1138	1141	1144	1148	1160	1162	1165	1168	1171	1181	1184	1189	
1194	1197	1200	1204	1206	1212	1214	1217	1224	1226	1230	1231	1233	
1235	1238	1246	1249	1252	1258	1260	1265	1269	1279	1286	1290	1297	
1299	1302	1305	1308	1312	1322	1324	1329	1332	1334	1337	1340	1342	
1350	1352	1356	1358	1362	1372	1375	1377	1381	1383	1386	1394	1397	
1399	1402	1404	1407	1419	1422	1425	1427	1430	1434	1447	1450	1452	
1455	1459	1475	1487	1488	1501	1506	1519	1521	1524	1529	1534	1541	
1557	1560	1564	1573	1583	1585	1591	1604	1607	1611	1616	1620	1638	
1641	1645	1649	1653	1659	1663	1665	1672	1676	1685	1689	1693	1697	
1703	1707	1709	1716	1720	1724	1733	1735	1739	1741	1745	1801	1804	
1809	1815	1817	1820	1826	1829	1834	1840	1846	1857	1860	1902	1907	
1916	1930#	1938	1940	1959	1963	2001	2016	2027	2029	2032	2051	2055	
2060	2062	2086	2091	2095	2099	2107	2115	2119	2159	2166	2180	2183	
2185	2202	2206	2261	2302	2307	2311	2314	2319#	2694#				

CZTMA10 TM,A,B-11 INSTR TST
CZTMAI.P11 12-SEP-79 13:54

MACY11 30A(1052) 12-SEP-79 13:57 PAGE 62
CROSS REFERENCE TABLE -- MACRO NAMES

H 5

SEQ 0059

.\$READ 1#
.\$R2AZ 1#
.\$SAVE 1#
.\$SB2D 1#
.\$SB2O 1#
.\$SCOP 1#
.\$SIZE 1#
.\$SUPR 1#
.\$STRAP 1#
.\$TYPB 1#
.\$TYPD 1#
.\$TYPE 1#
.\$TYPO 1#
.\$4OCA 1#
.\$1170 1#

. ABS. 017350 000

ERRORS DETECTED: 0

CZTMAI.BIN,CZTMAI.LST/CRF/SOL/NL:TOC=CZTMAI.SML,CZTMAI.P11
RUN-TIME: 27 37 2 SECONDS
RUN-TIME RATIO: 153/67=2.2
CORE USED: 31K (61 PAGES)