

TR79F

TIMING PROGRAM
MD-11-DZTRB-C

EP-DZTRB-C-DL-C
COPYRIGHT © 1977
FICHE 1 OF 1

JUN 1977
digital
MADE IN USA

The microfiche card contains 18 frames of data, arranged in 6 rows and 3 columns. Each frame displays a small table or chart with various data points and labels, typical of a timing program output. The data is too small to read clearly but appears to be organized in a structured format.

801

EOF10ZTEDASEQ

00010000

770526

PDP10 411

HDR10ZTRBCSEQ

00010000

770526

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZTRB-C-D
PRODUCT NAME: TR79 UTILITY PROGRAMS
DATE RELEASED: 1-APR-77
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: JOHN YASKO TELCO PRODUCT SUPPORT (EX 3922)

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS 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) 1976, 1977
DIGITAL EQUIPMENT CORPORATION, MAYNARD MASS.

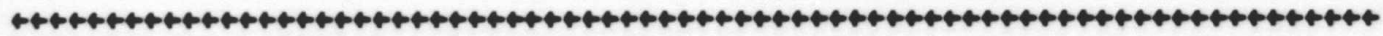
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50

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
92
93
94

TR79 UTILITY PROGRAM

TABLE OF CONTENTS

SECTION	SUBJECT
1.	ABSTRACT
2.	REQUIREMENTS
3.	LOADING PROCEDURE
4.	STARTING PROCEDURE
5.	RESTRICTIONS
6.	CONSOLE SETTINGS
7.	OPERATION
8.	LISTING



1. ABSTRACT

THIS PROGRAM IS IN TWO PARTS, AND IS INTENDED TO PROVIDE THE USER WITH A TOOL FOR TROUBLE-SHOOTING THE TR79 MAGTAPE SUBSYSTEM ON A POP-11 COMPUTER SYSTEM. THE FIRST PART OF THE PROGRAM ALLOWS THE USER TO GIVE THE MAGTAPE, COMMANDS, TO SIMULATE USER ROUTINES BY MERELY INSERTING THESE COMMANDS IN THE CORE LOCATIONS PROVIDED. THE USER MAY EXECUTE ONE OR SEVERAL INSTRUCTIONS IN ANY LEGAL SEQUENCE. WHILE THE CODE FOR THE DRIVER IS SIMPLE AND USES NO INTERRUPTS, DUE TO THE DESIGN OF THE HARDWARE CERTAIN ERROR CONDITIONS MUST BE IDENTIFIED IN ORDER TO PREVENT MISINTERPRITATION OF THE DESIRED RESULTS.

PART TWO OF THE PROGRAM CONSIST OF SELF CONTAINED ROUTINES TO PERMIT THE USER TO SET UP AND CHECK THE DELAYS CONTAINED WITHIN THE TR79 CONTROLLER, BY USING THE SWITCH REGISTER TO SELECT THE APPROPRIATE ROUTINE.

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
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146

2. REQUIREMENTS

2.1 HARDWARE

- A. PDP-11 PROCESSOR
- B. TR79 MAGTAPE TRANSPORT (HP-7970E DRIVE)
- C. TR79F MAGTAPE CONTROLLER

2.2 STORAGE

THIS PROGRAM REQUIRES A MINIMUM OF 4K OF CORE

3. LOADING

USE STANDARD BINARY LOADING PROCEDURE

4.0 STARTING PROCEDURE

THERE ARE TWO STARTING ADDRESSES THAT MAY BE USED

4.1 200 (8) LA 200 SR=0 A START AT THIS ADDRESS WILL RESULT IN A PROGRAMMED DEFAULT OPERATION OF A WRITE FORWARD WITH A WORD COUNT OF -20 AND A DATA PATTERN OF ALL 1'S. TO MODIFY THESE PARAMETERS SEE SECTION 7.1 PROGRAM OPERATION
NOTE: ALSO SEE SECTION 5.0 PROGRAM RESTRICTIONS, THE DEFAULT OF WRITE WILL NOT WORK IF TAPE IS AT B.O.T..

4.2 204 (8) LA 204 SR=0 A START AT THIS ADDRESS WILL EXECUTE THE SPECIALLY DESIGNED SETUP ROUTINES TO ALLOW THE USER TO SETUP OR VERIFY THE DELAYS WITHIN THE TR79 CONTROLLER.
NOTE: ALWAYS USE SCRATCH TAPES WHEN TAPE MOTION IS INDICATED.

148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200

5.0 RESTRICTIONS

- 5.1 A. A PSEUDO-OP OF A 20(8) HAS BEEN PROVIDED TO ALLOW THE USER TO POWER CLEAR BETWEEN OPERATIONS IF DESIRED, HOWEVER THE PROGRAM CAN RECOVER FROM ERRORS. THIS IS A POWER CLEAR AND TAKES 900 MILI SECONDS TO COMPLETE ANY ATTEMPTS TO ISSUE INSTRUCTIONS TO THE CONTROLLER WHILE A POWER CLEAR IS IN PROGRESS WILL RESULT IN ILLEGAL COMMAND BIT SETTING WHICH WILL INHIBIT ANY FURTHER INSTRUCTIONS FROM BEING EXECUTED. A POWER CLEAR IS ALSO GENERATED FROM A BUS INIT WHICH OCCURS FROM A RESET INSTRUCTION, THE DRIVER USES NO RESETS, (USE CAUTION IF YOU MODIFY THE DRIVER PACKAGE)
- B. THE TR79 CONTROLLER CHECKS FOR CERTAIN ILLEGAL FUNCTIONS DUE TO TAPE POSITION OR STATUS, THE DRIVER PACKAGE WILL CHECK THESE CONDITIONS AND HALT AT APPROPRIATE LOCATIONS WITH MEANINGFULL DATA DISPLAYED (SEE SECTION 7.2 ERROR CHECKS). THE LISTED CONDITIONS WILL PRODUCE ILLEGAL COMMAND ERRORS:
1. ATTEMPT TO WRITE DATA FROM LOAD POINT WITHOUT AN I.D.B.
 2. ATTEMPT TO WRITE A TAPE MARK FROM LOAD POINT
 3. ATTEMPT TO MOVE TAPE IN REVERSE FROM LOAD POINT
 4. ATTEMPT TO REWIND FROM LOAD POINT
 5. ATTEMPT TO WRITE AN I.D.B. AT OTHER THAN LOAD POINT
 6. ATTEMPT TO WRITE DATA WITH THE WRITE RING REMOVED
 7. ISSUE A COMMAND WHILE THE MAGTAPE IS NOT READY
 8. ISSUE A COMMAND WHILE THE CONTROLLER IS NOT READY
 9. ISSUE A COMMAND WITH INHIBIT BIT SET
 10. ILLEGAL FUNCTION CODES 00,03,05,06,11,12,14
- C. THE PROGRAM DOES NO DATA CHECKS ON READ OR WRITE DATA TRANSFERRED. IT IS THE RESPONSIBILITY OF THE OPERATOR TO MANUALLY EXAMINE THE BUFFER LOCATIONS TO DETERMINE IF THERE HAVE BEEN ANY PICKED OR DROPPED BITS IF DESIRED.
- D. NOTE: HARDWARE OPERATION OF THE TR-79 SPECIFIES THAT EACH CORE WORD LOCATION CONTAIN ONE BYTE (BITS 0-7) OF DATA AND PARITY (BIT 8). THEREFORE WHEN CALCULATING THE WORD COUNT FOR A TRANSFER THE ACTUAL NUMBER OF CORE LOCATIONS ACCESED IS EQUAL TO 2X THE NUMBER LOADED IN THE WORD COUNT REGISTER. ALSO NOTE THAT THE CONTROLLER DOES NOT APPEND PARITY TO THE BYTE BEFORE DOING A WRITE OPERATION. PARITY MUST BE CORRECT IN CORE OTHERWISE ERRORS WILL OCCUR ON THE TRANSFER (ODD PARITY) IS ALWAYS USED.

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
254
255

6.0 CONSOLE SWITCH SETTINGS

SW 15 = 1 STOP AFTER EACH OPERATION (ONLY WITH START 200)
 0 PROCEED

SW 14 = 1 STOP AT THE END OF EACH PROGRAM PASS (ONLY WITH START 200)
 0 PROCEED

SW 7 = 1 ENABLE FOR DELAY ROUTINES (EXECUTE ROUTINE ONLY WITH START 204)
 0 ALLOW SELECTION OF DELAY ROUTINES WITH SW 0-3

SW 0 THU 3 = DELAY ROUTINE TO BE EXECUTED (ONLY WITH START 204)

6.1 DELAY SETUP TABLE

SWITCH SETING	DELAY NAME	MODULE TYPE	LOCATION	PRINT PAGE	INPUT PIN	OUTPUT PIN	TIME
00	NO-OP						
01	P CLR	M-302	C-06	T02-2	H2	F2	20 MILI SEC.
02	P CLR OFF	M-306	D-09	T04-1	H2	T2	900 MILI SEC.
03	ERROR CLK	M-302	C-10	T04-2	H2	F2	200 NANO SEC.
04	WRITE ENAB	M-302	C-10	T09-3	H2	T2	40 MICRO SEC.
05	BUFF CONT	M-304	B-18	T11-1	E1	J1	1 MICRO SEC.
06	END WR DAT	M-302	A-16	T11-1	H2	T2	18 MICRO SEC.
07	1ST WD REQ	M-302	A-22	T11-2	H2	F2	100 MICRO SEC.
10	ERASE	M-304	B-18	T09-3	S1	M1	1 MICRO SEC.
11	WRITE IDB	M-302	A-16	T09-1	H2	F2	17 MILI SEC.
12	IDB TIMING	M-302	D-13	T09-1	H2	F2	75 MILI SEC.
13	ABORT	M-306	A-25	T09-3	H2	T2	1.5 SEC.
14	BUSY DELAY	M-304	B-18	T05-1	R1	P1	100 NANO SEC.
15	GO BIT DEL	M-304	B-18	T06-1	D1	H1	1 MICRO SEC.
16	M.S.D.	M-302	A-22	T09-2	H2	T2	900 MILI SEC.
17	NO-OP						

257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311

* DELAY CONDITION NOTES *

6.2 DELAY SS=FIXED DELAYS

- 00 NO OPERATION PERFORMED WAITING SWITCH SELECTION AND ENABLE
- 01 POWER CLEAR PERFORMED NO TAPE MOTION DELAY PULSE IS POSITIVE GOING
- 02 POWER CLEAR PERFORMED NO TAPE MOTION DELAY PULSE IS POSITIVE GOING
- 03 NO TAPE MOTION, PROGRAM FORCES AN ERROR WITH THE BGL BIT IN THE TR STATUS REGISTER (BIT 11) DELAY PULSE IS POSITIVE GOING
- 04 TAPE MOTION, PROGRAM DOES A SHORT ERASE WHILE MOVING TAPE TAPE MOTION IS NOT READILY NOTICIBLE WHILE EXECUTING THIS ROUTINE DELAY PULSE IS POSITIVE GOING
- SS 05 TAPE MOTION, PROGRAM DOES A 10 BYTE WRITE, PROGRAM CHECKS FOR LOAD POINT AND WILL WRITE AN I.D.B. BEFORE ENTERING THE DELAY LOOP. DELAY PULSE IS POSITIVE GOING
- 06 SAME CONDITIONS AS 05 DELAY PULSE IS POSITIVE GOING
- 07 SAME CONDITIONS AS 05 DELAY PULSE IS POSITIVE GOING
- SS 10 TAPE MOTION, PROGRAM WILL CHECK FOR LOAD POINT THEN DO A MAXIMUM ERASE TO MAKE THE OPERATION CONTINUOUS THE PROGRAM WILL CLEAR THE ERASE COUNT BEFORE THE OPERATION IS DONE. DELAY PULSE IS NEGATIVE GOING
- 11 TAPE MOTION, PROGRAM WILL CONTINUOUSLY WRITE THE I.D.B. DELAY PULSE IS POSITIVE GOING
- 12 SAME CONDITIONS AS DELAY 11. DELAY PULSE IS POSITIVE GOING
- 13 PROGRAM WILL REWIND TAPE TO L.P. AND FORCE AN ERROR BY DOING A WRITE DATA. DELAY PULSE IS POSITIVE GOING.
- SS 14 TAPE WILL MOVE TO L.P. , AND DO A MAXIMUM ERASE. WHILE THIS IS HAPPENING PROGRAM WILL LOAD THE COMMAND REGISTER TO PRODUCE A LD CTRL PULSE. DELAY PULSE IS POSITIVE GOING
- SS 15 SAME CONDITIONS AS DELAY 04. DELAY PULSE IS NEGATIVE GOING
- 16 PROGRAM WILL MOVE TAPE TO E.O.T. AND ATTEMPT TO DO A FAST FORWARD TO PRODUCE THE MOTION STOP DELAY. DELAY IS POSITIVE. NOTE: AFTER COMPLETION OF THIS ROUTINE A MANUAL REWIND SHOULD BE PERFORMED.
- 17 THIS IS A NO OPERATION SAME AS 00

313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351

7.0 OPERATION

THE PROGRAM IS QUITE SIMPLE HOWEVER IT DOES REQUIRE KNOWLEDGE OF THE OF THE TR-79 MAGTAPE SYSTEM AND AN UNDERSTANDING OF THE PROGRAM FUNCTIONS AND RESTRICTIONS. THE CODE HAS BEEN ASSEMBLED IN IMMEDIATE AND ABSOLUTE MODES USING PC ADDRESSING. IT IS RECOMMENDED THAT THE USER READ AND UNDERSTAND THE RESTRICTIONS AND OPERATIONS SECTIONS.

THE DRIVER PROGRAM (LOAD ADDRESS 200 START SWITCHES =0) CAN BE MADE TO EXECUTE ANY LEGAL SEQUENCE OF OPERATIONS (SEE SECTION 7.3) BY INSERTING THE COMMANDS IN THE OPERATIONS TABLE, (CORE LOCATIONS 722 THRU 766). EACH COMMAND SHOULD OCCUPY ONE CORE LOCATION BITS 0-4 ONLY. THE TOTAL NUMBER OF COMMANDS TO BE EXECUTED SHOULD THEN BE ENTERED IN LOCATION 720. THE PROGRAM PARAMETERS MAY BE ALTERED BY CHANGING THE APPROPRIATE CORE LOCATIONS (SEE SECTION 7.1). PROGRAM DEFAULT IS A SINGLE WRITE COMMAND OF 20 WORDS OF ALL 1'S FROM LOCATION 2700 WITH MINIMUM DELAY BETWEEN OPERATIONS. THIS DEFAULT WILL NOT WORK IF THE TAPE IS POSITIONED AT LOAD POINT.

THE DELAY PROGRAM (LOAD 204 START SWITCHES=0) WILL EXECUTE THE DELAY SET-UP ROUTINES TO ALLOW SET-UP OF ALL THE DELAYS IN THE TR-79 CONTROLLER THE PROGRAM HAS AN ACTIVE SWITCH REGISTER AFTER STARTING. BY SELECTING THE DESIRED DELAY ROUTINE IN SWITCH REGISTER 0 THRU 3, AND THEN SETTING BIT 7 =1 THE ROUTINE WILL BEGIN EXECUTION. TO CHANGE THE DELAY ROUTINE SET BIT 7=0. WAIT A FEW SECONDS FOR COMPLETION OF THE ROUTINE, THEN ENTER THE NEW ROUTINE NUMBER IN BITS 0-3 AND SET BIT 7=1. THE DELAY PROGRAM CONTAINS NO ERROR HALTS, HOWEVER IF ERRORS ARE DETECTED THE PROGRAM WILL INFORM THE USER BY OUTPUTTING A BELL CODE TO THE CONSOLE TERMINAL. THE PROGRAM WILL THEN DO A CONTROL RESET AND CONTINUE.

NOTE: THE PROGRAM BUILDS THE CORE DATA BUFFERS EACH TIME THE PROGRAM IS STARTED. THE PROGRAM DEFAULT IS LOCATION 2700 HOWEVER THIS MAY BE CHANGED BY MODIFYING LOCATION 242 IN THE CORE BUILD ROUTINE TO PUT THE BUFFERS ANYPLACE IN THE LOWER 28K. THIS PROGRAM DOES NOT PROGRAM THE KT AND DOES NOT RELOCATE ABOVE THE LOWER 28K OF MEMORY.

353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406

7.1 PROGRAM PARAMETER LIST AND CORE ADRESSES

PARAMETER	LOCATION	DESCRIPTION
EXTENDED CORE ADDRESS	700	BITS 12 AND 13 OF THIS LOCATION REPRESENT XBA 16 AND XBA 17 OF THE TR CONTROL REGISTER THESE BITS ALLOW RELOCATION OF THE DATA BUFFER.
UNIT SELECT	702	BITS 8+9 IN THIS LOCATION REPRESENT THE UNIT NUMBERS OF THE TAPE DRIVES. A MAXIMUM OF 4 DRIVES PER CONTROLLER DEFAULT IS UNIT 0.
WORD COUNT	704	THIS IS THE 2'S COMPLIMENT OF THE NUMBER OF WORDS TRANSFERRED. SINCE EACH BYTE OCCUPIES A WORD LOCATION THE NUMBER OF CORE LOCATIONS USED IS 2X THE WORD COUNT. PROGRAM DEFAULT IS -20 WORDS.
READ ADRESS	706	CONTAINS ADRESS OF THE READ BUFFER. PROGRAM DEFAULT IS LOCATION 6700.
WRITE ADRESS	710	CONTAINS ADRESS OF THE WRITE BUFFER. THE PROGRAM CONTAINS 4 WRITE PATTERNS CONTIGIOUS IN CORE. LOCATION 2700 = ALL 1'S PATTERN LOCATION 3700 = ALTERNATE 1 AND 0 BYTES LOCATION 4700 = ALTERNATE 1 AND 0 BITS LOCATION 5700 = SLIDING 1 BIT PATTERN PROGRAM DEFAULT IS LOCATION 2700
ERASE COUNT	712	CONTAINS A 2'S COMPLIMENT NUMBER PROPORTIONAL TO THE AMOUNT OF TAPE TO BE ERASED. THIS NUMBER IS LOADED INTO THE WORD COUNT REGISTER PRIOR TO AN ERASE COMMAND BIENG PERFORMED. PROGRAM DEFAULT IS 77777. EACH INCREMENT CAUSES .02 INCHES OF TAPE TO BE ERASED.
OPERATION DELAY	714	CONTAINS A NUMBER USED IN A TIMER BETWEEN OPERATIONS DEFAULT =000001 MINIMUM DELAY
OPERATION DELAY MULT.	716	THIS IS USED IN CONJUNCTION WITH LOC. 714 AS A MULTIPLIER IN THE DELAY TIMER. DEFAULT IS 000001 MINIMUM DELAY. INCREASING THIS NUMBER WILL ALLOW MORE TIME BETWEEN OPERATIONS.
OPERATIONS NUMBER	720	THIS LOCATION CONTAINS THE NUMBER OF OPERATIONS TO BE PERFORMED IN THE OP TABLE. DEFAULT = 1 .
OPERATIONS TABLE	722 THRU 766	THIS IS THE BEGINNING OF THE OPERATIONS TABLE. ALL OPERATIONS TO BE PERFORMED SHOULD BE ENTERED IN THE DESIRED SEQUENCE IN THIS TABLE. DEFAULT IS A WRITE OPERATION.

408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440

7.2 ERROR CHECKS AND HALTS

LOCATION DESCRIPTION

1320 HALT HERE IF THERE WAS AN ATTEMPT TO EXECUTE AN ILLEGAL FUNCTION, DUE TO TAPE POSITION OR SEQUENCE OF INSTRUCTIONS. THE ILLEGAL COMMAND IS DISPLAYED IN RD WHEN THE PROGRAM HALTS. SEE SECTION 5.18 FOR ILLEGAL FUNCTIONS

1332 HALT HERE IF THERE WAS A HARDWARE ERROR ON THE PREVIOUS OPERATION IF IT IS DESIRED TO BYPASS THE ERROR FLAG NOP THIS LOCATION. THE COMMAND AND STATUS REGISTER SHOULD BE EXAMINED AT THIS TIME TO DETERMINE THE PROBABLE CAUSE OF THE ERROR. PRESSING CONTINUE WILL CLEAR THE ERROR BY EXECUTING A CONTROL RESET.

1350 HALT HERE IF YOUR OPERATION TABLE LOC.722-766 HAS AN OPERATION THAT IS NOT DEFINED IN THE LEGAL FUNCTION CODES. RD HAS THE BAD CODE IN IT, CHECK YOUR TABLE IN LOCATIONS 722 THRU 766.

1406 HALT HERE IF BIT 15 OF THE SMR IS SET. THIS IS THE HALT BETWEEN INSTRUCTIONS.

1432 HALT HERE IF BIT 14 OF THE SMR IS SET. THIS IS THE HALT BETWEEN PASSES OF INSTRUCTIONS IN THE OP TABLE.

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

7.3 TABLE OF LEGAL FUNCTIONS AND CODES FOR USE IN OPERATIONS TABLE (LOC 722-766)

<u>CODE</u>	<u>FUNCTION</u>
00	**** ILLEGAL ****
01	WRITE DATA (ILLEGAL IF EXECUTED FROM LOAD POINT)
02	READ (DATA, TAPE MARK OR I.D.B.)
03	**** ILLEGAL ****
04	SPACE REVERSE (ILLEGAL IF ISSUED FROM LOAD POINT)
05	**** ILLEGAL ****
06	**** ILLEGAL ****
07	ERASE
10	REWIND (TAPE MOVES AT 160 I.P.S.) ILLEGAL IF ISSUED FROM LOAD POINT.
11	**** ILLEGAL ****
12	**** ILLEGAL ****
13	FAST FORWARD (TAPE MOVES FORWARD AT 160 I.P.S.)
14	**** ILLEGAL ****
15	WRITE I.D.B. (ILLEGAL IF ISSUED AT OTHER THAN LOAD POINT)
16	WRITE TAPE MARK (ILLEGAL IF ISSUED FROM LOAD POINT)
17	OFFLINE (REQUIRES MANUAL INTERVENTION)
20	CONTROL RESET (PROGRAM PSEUDO OP)

B. PROGRAM LISTING

.ENABLE ABS,AMA
.TITLE TR79 UTILITY DRIVER
.ASECT

000000

* GENERAL REGISTER DEFINITIONS *

R0=X0
R1=X1
R2=X2
R3=X3
R4=X4
R5=X5
SP=X6
PC=X7

000000
000001
000002
000003
000004
000005
000006
000007

* TR79 REGISTER DEFINITIONS *

TRCOM=164000
TRSTAT=164002
TRWC=164004
TRBA=164006

164000
164002
164004
164006

* PROCESSOR REGISTER DEFINITIONS *

PSW=177776
SWR=177570

177776
177570

* TTY REGISTERS *

TTSTAT=177564
TTBUF=177566

177564
177566

000000
000001
000002
000003
000004
000005
000006
000007
164000
164002
164004
164006
177776
177570
177564
177566


```

585
586
587 000300 001361
588 000302 005000
589 000304 005001
590 000306 005200
591 000310 005201
592 000312 010022
593 000314 022702 006700
594 000320 001405
595 000322 022701 000011
596 000326 001765
597 000330 006300
598 000332 000766
599 000334 005737 000356
600 000340 001404
601 000342 005037 000356
602 000346 000137 001462
603 000352 000137 001000
604 000356 000000
605
606
607
608
609
610 000500 000500
611 000500 000000
612
613
614
615 000600
616
617
618
619

```

```

:
:
SLB:      BNE      X2
          CLR      R0
          CLR      R1
          INC      R0
SLB1:     INC      R1
          MOV      R0,(R2)+
          CMP      #6700,R2
          BEQ      SLBDON
          CMP      #11,R1
          BEQ      SLB
          ASL      R0
          BR       SLB1
SLBDON:   TST      @#NORST
          BEQ      ALD
          CLR      @#NORST
          JMP      @#DRTN
          JMP      @#START
ALD:
NORST:   000000
:
:
          *****
          *   STACKS   *
          *****
:
SUBSTK:   000000      .=500
:
STACK:    000000      .=600
:

```

```

:NOT DONE YET
:CLEAR THE PATTERN GENERATOR
:CLEAR THE PATTERN COUNTER
:SET BIT IN PATTERN
:KEEP COUNT
:PUT IT IN CORE
:SEE IF HERE FINISHED
:YES JUMP OUT
:CHECK ON THE BIT POSITION
:RESET THE SLIDING BIT
:SHIFT THE BIT
:LOOP AGAIN
:SEE WHERE THE START CAME FROM
:IF = 0 MUST HAVE BEEN START 200
:CLEAR IT OUTFOR NEXT TIME
:NOT = GO TO 204 START
:GO TO A 200 START
:TEMPORARY LOCATION

```

```

;SUBROUTINE STACK
;SET STACK HERE

```



```

671
672
673
674
675
676
677 001000 001000 000600 START: MOV #600,SP      ;SET UP STACK AT LOC 600
678 001004 012737 000340 177776 MOV #340,PSW      ;SET PRIORITY LEVEL 7 NO INTERRUPTS ALLOWED
679 001012 032737 004000 164000 1S: BIT #4000,TRCOM ;WAIT FOR THE INITIAL 900 MS.POWER CLEAR TO SUBSIDE
680 001020 001374 000720 BNE 1S          ;LOOP UNTILL CONTROL IS READY
681 001022 013701 000720 MOV #OPNUM,R1    ;R1 CONTAINS HOW MANY OPERATIONS WILL BE DONE
682 001026 042737 176377 000702 UNUM: BIC #176377,UNIT ;STRIP OFF EVERYTHING EXCEPT BITS 8 AND 9 TO SET UP UNI
683
684
685
686
687
688
689 001034 012702 000722 LOOP: MOV #OPTBL,R2    ;R2 CONTAINS THE POINTER TO THE START OF THE OPERATINS
690 001040 012737 000011 000674 MOV #9,TEMP1    ;THE NUMBER OF LEGAL OPERATIONS
691 001046 012703 001440 MOV #LEGOPS,R3  ;POINTER TO THE BEGINNING OF THE LEGAL OPERATIONS COMPA
692 001052 012237 000676 MOV (R2)+,TEMP2 ;GET THE OPERATION AND PUT IT IN TEMP 2
693 001056 023723 000676 2S: CMP #TEMP2,(R3)+ ;CHECK IT AGAINST THE LEGAL FUNCTIONS
694 001062 001422 BEQ CLINH    ;IF ITS LEGAL CONTINUE TO EXECUTE IT
695 001064 005337 000674 DEC TEMP1      ;WASN'T THAT OPERATION SUBTRACT 1
696 001070 001372 BNE 2S        ;TRY NEXT ONE

```

HOUSE KEEPING AND INITIAL PROGRAM PARAMETERS #

.=1000

SET UP OPERATIONS COMPARITOR #

```

698
699
700
701
702
703
704
705 001072 022737 000020 000676      CMP  #20,#TEMP2      ;SEE IF ITS A CONTROL RESET (PSEUDO OP)
706 001100 001402                      BEQ  CRES             ;MUST BE A CONTROL RESET JUMP TO CRES
707 001102 000137 001344                      JMP  #ILFUT          ;IT MUST BE ILLEGAL FUNCTION STOP THE PROGRAM
708 001106 052737 004000 164000  CRES:  BIS  #4000,TRCOM    ;MUST BE A POWER CLEAR SO DO IT TAKES 900 MILI SECONDS
709 001114 032737 004000 164000  3S:   BIT  #4000,TRCOM    ;SEE IF DONE WITH POWER CLEAR YET
710 001122 001374                      BNE  3S              ;WAIT UNTILL DONE
711 001124 000137 001264                      JMP  FUDONE          ;GET BACK INTO PROGRAM
712
713
714
715
716
717
718
719 001130 005037 164002      CLINH: CLR  #TRSTAT      ;CLEAR OUT THE INHIBIT BIT
720 001134 162703 001442      SUB  #LEGOPS+2,R3     ;GET AN OFFSET VALUE
721 001140 060307                      ADD  R3,PC            ;ADD IT TO THE PC AND GO THERE
722
723
724
725
726
727
728
729 001142 000422      W:   BR  WRITE         ;WRITE INSTRUCTION
730 001144 000425      R:   BR  READ          ;READ INSTRUCTION
731 001146 000401      SR:  BR  .+4          ;SPACE REVERSE INSTRUCTION
732 001150 000432      ER:  BR  ERSE        ;ERASE COMMAND
733 001152 000240      REWD: NOP            ;REWIND COMMAND
734 001154 000240      FFOR: NOP           ;FAST FORWARD COMMAND
735 001156 000240      IDB:  NOP           ;WRITE ID BURST
736 001160 000240      WTM:  NOP           ;WRITE TAPE MARK

```

* CHECK FOR A CONTROL RESET *

* CLEAR INHIBIT BIT AND SET UP OPERATION *

* OPERATIONS DIRECTORY TABLE *

738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777

001162 006337 000676
001166 053737 000702 000676
001174 053737 000700 000676
001202 005237 000676
001206 000417

SGOB:

ASL @TEMP2
BIS @UNIT,@TEMP2
BIS @EXTCOR,@TEMP2
INC @TEMP2
BR EXECUT

:SHIFT THE FUNCTION INTO THE PROPER BIT POSITIONS
:SET THE UNIT # BITS
:SET THE MEMORY EXTENSION BITS
:SET THE GO BIT
:JUMP TO THE EXECUTION ROUTINE

* SET UP UNIT NUMBER AND GO BIT *

* OPERATIONS SETUP ROUTINES *

001210 013737 000710 164006
001216 000403
001220 013737 000706 164006
001226 013737 000704 164004
001234 000752
001236 013737 000712 164004
001244 000746

WRITE:

MOV @WADDR,@TRBA
BR WR1WC

:SET UP WRITE BUFFER AREA
:CONTINUE ON TO SET UP W.C.

READ:

MOV @RADDR,@TRBA
MOV @WCNT,@TRWC
BR SGOB

:SET UP READ BUFFER ADDRESS
:SET UP WORD COUNT
:SET UP GO BIT

ERSE:

MOV @ERSCNT,@TRWC
BR SGOB

:SET NUMBER TO INDICATE AMOUNT OF TAPE TO BE ERASED
:SET UP GO BIT

* ROUTINE TO EXECUTE THE FUNCTION AND CHECK FOR DONE *

001246 000240
001250 013737 000676 164000
001256 105737 164000
001262 100375

EXECUT:

NOP
MOV @TEMP2,@TRCOM
TSTB @TRCOM
BPL 7S

:DO THE FUNCTION
:SEE IF DONE YET
:WAIT FOR IT

7S:

;

```

779
780
781
782
783
784
785
786 001264 013737 164002 000672 FUDONE: MOV 2#TRSTAT,2#STEM ;SAVE STATUS
787 001272 013737 164000 000670 MOV 2#TRCOM,2#COMTEM ;SAVE COMMAND REGISTER
788 001300 032737 040000 000670 BIT 2#40000,2#COMTEM ;WAS IT AN ILLEGAL COMMAND DUE TO SEQUENCE OR TAPE POSI
789 001306 001405 BEQ ERDONE ;NO ERROR HERE
790 001310 006237 000676 ILLCOM: ASR 2#TEMP2 ;STRIP OFF THE GO BIT
791 001314 013700 000676 MOV 2#TEMP2,RO ;PUT BAD COMMAND IN RO
792 001320 000000 ERR14: HALT ;STOP WITH BAD COMMAND DISPLAYED
793 001322 032737 100000 000670 ERDONE: BIT 2#100000,2#COMTEM ;SEE IF ERROR BIT IS SET
794 001330 001410 BEQ OPDEL ;NO ERRORS CONTINUE
795 001332 000000 ERR15: HALT ;GOT AN AROR NOP THIS HALT TO CONTINUE
796 001334 000137 001106 JMP 2#CRES ;IF YOU GOT AN ERROR ONLY RECOVERY IS WITH A CONTROL RE
797 001340 006237 000676 ILLFUN: ASR 2#TEMP2 ;STRIP OFF GO BIT
798 001344 013700 000676 ILFUT: MOV 2#TEMP2,RO ;PUT THE BAD CODE IN RO TO DISPLAY WHEN HALTED
799 001350 000000 ERRIF: HALT ;GOT AN ILLEGAL FUNCTION CHECK YOUR PROGRAM LOCATION 72
800
801

```



```

803
804
805
806
807
808
809 001352 013737 000716 000666 OPDEL: MOV 2#OPDLX,2#TIMMUL ;SET UP OPERATIONS DELAY MULTIPLIER
810 001360 013700 000714          MOV 2#OPDLY,R0 ;SET UP OPERATIONS DELAY TIMER
811 001364 005300          BS:   DEC R0 ;TIMER IS TICKING
812 001366 001376          BNE BS ;GET MORE TIME
813 001370 005337 000666          DEC 2#TIMMUL ;COUNT DOWN THE MULTIPLIER
814 001374 001373          BNE BS ;GET MORE TIME
815 001376 032737 100000 177570 BIT 2#100000,2#SWR ;TIMES UP CHECK SWITCHES TO SEE IF WE HALT OR CONTINUE
816 001404 001401          BEQ .+4 ;DONT STOP NOW SKIP THE HALT
817 001406 000000          INSHLT: HALT ;STOP BETWEEN INSTRUCTIONS
818 001410 005301          DEC R1 ;-1 FROM THE NUMBER OF OPERATIONS IN R1
819 001412 001001          BNE 9S ;GO AND DO THE NEXT INSTRUCTION
820 001414 000401          BR .+4 ;SKIP THE JUMP
821 001416 000137 001040          9S:   JMP 2#LOOP ;DO THE LOOP AGAIN
822 001422 032737 040000 177570 BIT 2#40000,2#SWR ;CHECK SWITCHES TO SEE IF WE WANT TO STOP AT END OF PAS
823 001430 001401          BEQ REST ;DO THE NEXT PASS SKIP THE HALT
824 001432 000000          PASHLT: HALT ;STOP BETWEEN PASSES
825 001434 000137 001000          REST: JMP 2#START ;GO DO IT AGAIN (NEXT PASS)
826
827
828
829
830
831
832 001440 000001          LEGOPS: 00001 ;WRITE
833 001442 000002          00002 ;READ
834 001444 000004          00004 ;SPACE REVERSE
835 001446 000007          00007 ;ERASE
836 001450 000010          00010 ;REWIND
837 001452 000013          00013 ;FAST FORWARD
838 001454 000015          00015 ;WRITE IDB
839 001456 000016          00016 ;WRITE TAPE MARK
840 001460 000017          00017 ;OFFLINE
841

```

* OPERATION DELAY BETWEEN INSTRUCTIONS *

* LEGAL OPERATIONS COMPARIOR TABLE *

```

894
895
896
897
898 001646 004537 002452
899 001652 005000
900 001654 005300
901 001656 001376
902 001660 105737 177570
903 001664 100770
904 001666 000137 001510
905
906
907
908
909
910 001672 012700 000070
911 001676 052737 004000 164002
912 001704 005300
913 001706 001376
914 001710 005037 164002
915 001714 105737 177570
916 001720 100764
917 001722 004537 002452
918 001726 000137 001510
919
920
921
922
923
924 001732 004537 002472
925 001736 012737 177777 164004
926 001744 012737 000017 164000
927 001752 004537 002472
928 001756 004537 002510
929 001762 105737 177570
930 001766 100763
931 001770 000137 001510

```

```

*****
* ROUTINE FOR DELAYS 1 AND 2 *
*****
BR2: JSR R5, @PCL ;DO A POWER CLEAR
      CLR R0 ;CLEAR THE TIMER
BR2A: DEC R0 ;TIMER IS TICKING
      BNE BR2A ;WAIT TILL ITS DONE
      TSTB @SWR ;CHECK FOR A LOOP
      BMI BR2 ;DO IT AGAIN
      JMP @BR1 ;GET NEXT TEST

```

```

*****
* ROUTINE FOR DELAY 3 *
*****
BR3: MOV #70, R0 ;SET UP DELAY MULTIPLIER
      BIS #4000, @TRSTAT ;FORCE AN ERROR WITH B.G.L. BIT
13S: DEC R0 ;TIMER IS TICKING
      BNE 13S ;CHECK TIMER
      CLR @TRSTAT ;OK NOW CLEAR THE BIT
      TSTB @SWR ;SEE IF WE WANT TO DO IT AGAIN
      BMI BR3 ;OK LOOP BACK
      JSR R5, @PCL ;DONE HERE DO A POWER CLEAR AND GET THE NEXT ONE
      JMP @BR1 ;GET THE NEXT DELAY DIRECTIVE

```

```

*****
* ROUTINE FOR DELAY 4 AND 15 *
*****
BR4: JSR R5, @RDY ;CHECK FOR READY
BR4A: MOV #-1, @TRMC ;THIS NUMBER IS USED FOR AN ERASE COUNT
      MOV #17, @TRCOM ;DO THE ERASE
      JSR R5, @RDY ;WAIT TILL DONE
      JSR R5, @ERCK ;SEE IF WE ERRORED OUT
      TSTB @SWR ;SEE IF WE LOOPON TEST
      BMI BR4A ;LOOP HERE AND DO IT AGAIN
      JMP @BR1 ;GET OUT AND GET NEXT DELAY DIRECTIVE

```



```

933
934
935
936
937
938 001774 004537 002540
939 002000 005037 164002
940 002004 012737 177774 164004
941 002012 013737 000710 164006
942 002020 012737 000003 164000
943 002026 004537 002472
944 002032 004537 002510
945 002036 105737 177570
946 002042 100760
947 002044 000137 001510
948
949
950
951
952
953
954 002050 004537 002574
955 002054 004537 002624
956 002060 005037 164002
957 002064 005037 164004
958 002070 012737 000017 164000
959 002076 005037 164000
960 002102 105737 177570
961 002106 100004
962 002110 105737 164000
963 002114 100370
964 002116 000756
965 002120 004537 002452
966 002124 000137 001510

```

```

:
:
: *****
: ROUTINE FOR DELAY 5 , 6 AND 7
: *****
:
BR5: JSR RS, @OFLP ;SEE IF WE ARE AT LOAD POINT
      CLR @TRSTAT ;CLEAR INHIBIT BIT
BRSA: MOV @-4, @TRMC ;SET UP FOR A 10 BYTE WRITE
      MOV @WADDR, @TRBA ;SET UP THE WRITE ADDRESS BUFFER
      MOV @3, @TRCOM ;DO THE WRITE
      JSR RS, @RDY ;WAIT FOR READY
      JSR RS, @ERCK ;CHECK FOR ERRORS
      TSTB @SWR ;SEE IF WE WANT TO LOOP
      BMT BRSA ;LOOP HERE AND DO IT AGAIN
      JMP @BR1 ;GET OUT AND GET NEXT DELAY DIRECTIVE

```

```

:
:
: *****
: ROUTINE FOR DELAY 14
: *****
:
BR6: JSR RS, @REW ;DO A REMIND GET TO B.O.T.
BR6C: JSR RS, @ETS ;CHECK FOR END OF TAPE ANYWAY
      CLR @TRSTAT ;CLEAR THE INHIBIT BIT
      CLR @TRMC ;CLEAR THE WORD COUNT
      MOV @17, @TRCOM ;DO AN ERASE
BR6A: CLR @TRCOM ;FORCE A LOAD PULSE
      TSTB @SWR ;SEE IF WE WANT TO LOOP
      BPL BR6B ;ALL DONE
      TSTB @TRCOM ;SEE IF ERASE IS DONE YET
      BPL BR6A ;NOT DONE DO ANOTHER LOAD PULSE
      BR BR6C ;DO IT AGAIN
BR6B: JSR RS, @PCL ;DO A CLEAR AND EXIT
      JMP @BR1 ;GO BACK TO MAIN

```



```

992
993
994
995
996
997 002204 004537 002574
998 002210 012737 000033 164000
999 002216 004537 002472
1000 002222 004537 002510
1001 002226 105737 177570
1002 002232 100764
1003 002234 004537 002452
1004 002240 000137 001510
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014 002244 105737 164002
1015 002250 100407
1016 002252 005037 164002
1017 002256 012737 000027 164000
1018 002264 004537 002472
1019 002270 005037 164002
1020 002274 012737 000027 164000
1021 002302 004537 002472
1022 002306 105737 177570
1023 002312 100766
1024 002314 004537 002452
1025 002320 000137 001510
1026

```

```

.....
BR11:
.....
BR14:
BR14A:
;

```

```

*****
* ROUTINE FOR DELAY 11 AND 12 *
*****

```

```

JSR RS, @REW ;DO A REWIND
MOV #33, @TRCOM ;WRITE AN I.D.B.
JSR RS, @RDY ;CHECK FOR DONE
JSR RS, @ERCK ;SEE IF ANY ERRORS UP
TSTB @SWR ;SEE IF WE WANT TO LOOP
BMI BR11 ;LOOP BACK DO IT AGAIN
JSR RS, @PCL ;DO A POWER CLEAR
JMP @BR1 ;GET OUT AND GET NEXT DELAY DIRECTIVE

```

```

*****
* ROUTINE FOR DELAY 16 *
*****

```

```

TSTB @TRSTAT ;CHECK FOR EOT UP
BMI BR14A ;SKIP THE FAST FORWARD
CLR @TRSTAT ;CLEAR INHIBIT
MOV #27, @TRCOM ;DO A FAST FORWARD
JSR RS, @RDY ;WAIT TILL DONE
CLR @TRSTAT ;CLEAR INHIBIT
MOV #27, @TRCOM ;TRY A FAST FORWARD ,SHOULD PRODUCE ERROR
JSR RS, @RDY ;WAIT TILL DONE
TSTB @SWR ;SEE IF WE LOOP HERE
BMI BR14A ;YES LOOP HERE
JSR RS, @PCL ;DO A POWER CLEAR
JMP @BR1 ;GET OUT DO NEXT DELAY

```

```

1028
1029
1030
1031
1032
1033 002324 004537 002574 BR16: JSR RS, @AREW ; DO A REMIND
1034 002330 005037 164002 CLR @TRSTAT ; CLEAR INHIBIT
1035 002334 012737 177000 164004 MOV @177000, @TRMC ; SET UP ERASE COUNT
1036 002342 012737 000017 164000 MOV @17, @TRCOM ; DO AN ERASE
1037 002350 004537 002472 JSR RS, @RDY ; WAIT FOR IT
1038 002354 004537 002574 JSR RS, @AREW ; REMIND IT
1039 002360 004537 002540 JSR RS, @OFLP ; GET OFF LOAD POINT LEGALLY
1040 002364 012737 177760 164004 BR16A: MOV @-20, @TRMC ; SET UP M.C.
1041 002372 012737 003600 164006 MOV @3600, @TRBA ; SET UP CORE ADDRESS
1042 002400 005037 164002 CLR @TRSTAT ; CLEAR THE INHIBIT
1043 002404 012737 000003 164000 MOV @3, @TRCOM ; NOW TRY TO WRITE NOTHING, SHOULD ABORT
1044 002412 012701 000010 BR16B: MOV @10, R1 ; SET UP TIME MULTIPLIER
1045 002416 005000 BR16D: CLR R0 ; CLEAR TIMER
1046 002420 005200 BR16C: INC R0 ; TIMES WAITING
1047 002422 001376 BNE BR16C ; TIMER RUNNING
1048 002424 005301 DEC R1 ; -1 FROM MULTIPLIER
1049 002426 001373 BNE BR16D ; SEE IF DONE YET
1050 002430 004537 002452 JSR RS, @PCL ; CLEAR ERRORS
1051 002434 105737 177570 TSTB @@SWR ; LOOP ???
1052 002440 100751 BMI BR16A ; DO IT AGAIN
1053 002442 004537 002452 JSR RS, @PCL ; POWER CLEAR
1054 002446 000137 001510 JMP @@BR1 ; ALL DONE GET SOME MORE
1055

```



```

1098
1099
1100
1101
1102
1103
1104
1105 002540 032737 000040 164002 OFLP: BIT  #40,@#TRSTAT      ;SEE IF LOAD POINT IS UP
1106 002546 001411                BEQ  OFLP1          ;NO LP JUMP OUT
1107 002550 005037 164002        CLR  @#TRSTAT      ;CLEAR THE INHIBIT
1108 002554 012737 000033 164000    MOV  #33,@#TRCOM   ;WRITE AN I.D.B.
1109 002562 004537 002472        JSR  R5,@#RDY     ;WAIT FOR READY
1110 002566 004537 002510        JSR  R5,ERCK     ;CHECK FOR ERRORS
1111 002572 000205                RTS                    ;GO BACK TO MAIN
1112
1113
1114
1115
1116
1117
1118
1119 002574 032737 000040 164002 REW:  BIT  #40,@#TRSTAT      ;AT LOAD POINT??
1120 002602 001007                BNE  REW2         ;YES DON'T NEED REWIND
1121 002604 005037 164002        CLR  @#TRSTAT      ;CLR THE INHIBIT
1122 002610 012737 000021 164000    MOV  #21,@#TRCOM   ;DO A REWIND
1123 002616 004537 002472        JSR  R5,@#RDY     ;WAIT TILL DONE
1124 002622 000205                RTS                    ;GO BACK
1125
1126
1127
1128
1129
1130 002624 105737 164002        ETS:  TSTB @#TRSTAT ;IS END OF TAPE UP ?
1131 002630 100002                BPL  ETS1        ;NOT AT E.O.T.
1132 002632 004537 002574        JSR  R5,@#REW     ;DO A REWIND
1133 002636 000205                RTS                    ;GET BACK
1134
1135
1136

```

* SUBROUTINE TO GET OFF LOAD POINT LEGALLY *

* SUBROUTINE FOR REWIND AND L.P. *

* SUBROUTINE FOR E.O.T. *

1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187

* WRITE BUFFER PATTERNS *

.=2700
.REPT 1000 ;ALL 1'S
.WORD 000777 ;PATTERN = 0 000 000 111 111 111
.ENDR

.=3700
.REPT 1000 ;1'S AND 0'S ALTERNATE WORDS
.WORD 000400 ;PATTERN = 0 000 000 100 000 000
.WORD 000777 ;PATTERN = 0 000 000 111 111 111
.ENDR

.=4700
.REPT 1000 ;ALTERNATE BITS
.WORD 000525 ;PATTERN = 0 000 000 101 010 101
.WORD 000652 ;PATTERN = 0 000 000 110 101 010
.ENDR

.=5700
.REPT 1000 ;SLIDING 1 BIT
.WORD 000001 ;PATTERN = 0 000 000 000 000 001
.WORD 000002 ;PATTERN = 0 000 000 000 000 010
.WORD 000004 ;PATTERN = 0 000 000 000 000 100
.WORD 000010 ;PATTERN = 0 000 000 000 001 000
.WORD 000020 ;PATTERN = 0 000 000 000 010 000
.WORD 000040 ;PATTERN = 0 000 000 000 100 000
.WORD 000100 ;PATTERN = 0 000 000 001 000 000
.WORD 000200 ;PATTERN = 0 000 000 010 000 000
.WORD 000400 ;PATTERN = 0 000 000 100 000 000
.ENDR

* READ BUFFER AREA *

.=6700 ;1000 WORD LOCATIONS RESERVED FOR READ BUFFER

.END

000001

ALD	000352	BR6	002050	EXECUT	001246	PCL1	002460	TEMP1	000674
BEGIN	000200	BR6A	002076	EXTCOR	000700	PSW	= 177776	TEMP2	000676
BEGIN2	000204	BR6B	002120	FFOR	001154	R	001144	TIMMUL	000666
BR1	001510	BR6C	002054	FUDONE	001264	RADDR	000706	TRBA	= 164006
BR10	002130	CLINH	001130	IDB	001156	RDY	002472	TRCOM	= 164000
BR10A	002152	CONTEM	000670	IHB	002500	READ	001220	TRSTAT	= 164002
BR10B	002156	CORBIL	000230	ILFUT	001344	REST	001434	TRMC	= 164004
BR11	002204	CRES	001106	ILLCOM	001310	REW	002574	TTBUF	= 177566
BR14	002244	DROUTS	001500	ILLFUN	001340	REWD	001152	TTSTAT	= 177564
BR14A	002270	DRTN	001462	INSHLT	001406	REW1	002604	UNIT	000702
BR16	002324	ER	001150	LEGOPS	001440	REW2	002622	UNUM	001026
BR16A	002364	ERCK	002510	LOOP	001040	SG08	001162	W	001142
BR16B	002412	ERCK1	002524	NORST	000356	SLB	000302	WADDR	000710
BR16C	002420	ERCK2	002536	OFLP	002540	SLBOON	000334	WCNT	000704
BR16D	002416	ERDNE	001322	OFLP1	002572	SLB1	000310	WRITE	001210
BR2	001646	ERRIF	001350	OPDEL	001352	SR	001146	WR1MC	001226
BR2A	001654	ERR14	001320	OPDLX	000716	START	001000	WTM	001160
BR3	001672	ERR15	001332	OPDLY	000714	STEM	000672	X1	000240
BR4	001732	ERSCNT	000712	OPNUM	000720	SUBSTK	000500	X2	000244
BR4A	001736	ERSE	001236	OPTBL	000722	SWR	= 177570	X3	000250
BRS	001774	ETS	002624	PASHLT	001432	SWRTM	000664	X4	000264
BRSA	002004	ETS1	002636	PCL	002452	TABLE	001546	.	= 002640

. ABS. 002640 000

ERRORS DETECTED: 0
 DEFAULT GLOBALS GENERATED: 0

DSKZ:DZTRBC.BIN,DZTRBC.SEQ/DOC/NL:TOC=DZTRBC.P11
 RUN-TIME: 1 2 0 SECONDS
 RUN-TIME RATIO: 89/4=20.5
 CORE USED: 4K (8 PAGES)

DOCUMENT PAGES: 30

EOF1DZTRBCSEQ 00010000 770526 PDP10 411 0