

VT50/52/55

VT50A,B,H/52 ACCEPTANCE TEST
MD-11-DZVTC-E

EP-DZVTC-E-DL-E
COPYRIGHT © 74-77
FICHE 1 OF 1

DEC 1977
digital
MADE IN USA

This microfiche card contains a grid of frames, each displaying technical data. The data is organized into columns and rows, with some frames containing graphical elements like bar charts or waveforms. The text is small and dense, typical of microfiche storage. The frames are arranged in a grid that is approximately 10 frames wide and 15 frames high. The data appears to be related to the acceptance test mentioned in the header.

1. ABSTRACT

THIS PROGRAM IS AN ACCEPTANCE TEST OF THE VT50/52 VIDEO TERMINAL. THE PROGRAM CONSISTS OF FOUR PARTS, ALL OF WHICH REQUIRE OPERATOR INSPECTION OR INTERACTION. THE PROGRAM IS CAPABLE OF HANDLING MULTIPLE UNITS IN A SEQUENTIAL DL-11 FASHION (REF 8.2).

ONLY ONE VT50/52 IS TESTED AT ONE TIME.

THE PROGRAM WILL DEFAULT TO THE CONSOLE TTY (REF 5.0 AND 8.2). ALL CHARACTERS AND COMMANDS ARE TESTED. IN THE KEYBOARD CHARACTER TEST THE FOLLOWING "FUNCTION" KEYS ARE NOT TESTED:
BREAK, REPEAT, AUTO-PRINT, AND SCROLL.

PART 1 CONSISTS OF A SERIES OF TEST PATTERNS DISPLAYED ON THE VT50/52 SCREEN AND COPIER (REF 9, FOR DESCRIPTION). THE OPERATOR MUST VISUALLY INSPECT EACH TEST PATTERN FOR ERROR DETECTION.

PART 2 IS A KEYBOARD CHARACTER TEST. THIS TEST IS TO DETERMINE THAT THE TERMINAL IS GENERATING THE EXPECTED ASCII CODES. IN THIS TEST AN OPERATOR WILL BE REQUIRED TO FOLLOW THE INSTRUCTIONS DISPLAYED ON THE VT50/52 SCREEN AND EXECUTE THEM. DUE TO THE FLEXIBILITY OF DIFFERENT PROCESSORS OR OPTIONS, PARITY BIT TESTING MUST BE SELECTED BY THE OPERATOR. THE OPERATOR SELECTS THE TYPE OF PARITY TO BE TESTED BY SW 00-01

PART 3 IS A KEYBOARD OCTAL VALUE LOOP. WHEN A KEY IS DEPRESSED, THE OCTAL VALUE WILL BE DISPLAYED ON THE SCREEN. IF THE KEY DEPRESSED WAS PRINTABLE, IT WILL ALSO BE DISPLAYED ON THE SCREEN. IF A "DEFINED" CHARACTER, A TWO LETTER EQUIVALENT (IE. BL=BELL, ES=ESCAPE ETC.) WILL BE DISPLAYED.

PART 4 IS A KEYBOARD ECHO LOOP. WHEN A KEY IS DEPRESSED, THE CHARACTER IS ECHOED TO THE SCREEN. NO TESTING OF THE CHARACTER IS PERFORMED. THIS ALLOWS THE OPERATOR A "LOCAL" MODE OF OPERATION BETWEEN THE VT50/52 AND THE HOST COMPUTER.

2. REQUIREMENTS

2.1 EQUIPMENT

PDP-11 FAMILY COMPUTER WITH 8K WORDS OF MEMORY
VT50A, B, H, 52 VIDEO TERMINAL CONNECTED VIA A DL-11A B TYPE INTERFACE.

2.2 STORAGE

THIS PROGRAM USES 8K OF MEMORY.

3 LOADING PROCEDURE

PROCEDURE FOR NORMAL BINARY TAPES SHOULD BE FOLLOWED.

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

STANDARD PDP-11 FORMAT

SW 15 = 1	HALT ON ERROR
SW 14 = 1	LOOP ON TEST
SW 13 = 1	INHIBIT ERROR TYPEOUTS
SW 12 = 1	INHIBIT PROGRAM SUB-TEST DELAY
SW 11 = 1	INHIBIT COPIER TESTING
SW 10 = 1	ENABLE "SAVE COPIER PAPER" MODE
SW 08 = 1	LOOP ON TEST IN SWR (4:0)
SW 07 = 1	KEYBOARD CONTROL OF THE TEST
	(SW 8 AND SW 7 = 1 IS AN ERROR)

KEYBOARD CHARACTER TEST ONLY

SW02 =	1	ENABLE PARITY BIT TEST
SW00-01 =	00	EVEN PARITY CHECK
SW00-01 =	01	ODD PARITY CHECK
SW00-01 =	10	ALWAYS A 0
SW00-01 =	11	ALWAYS A 1

SPECIAL NOTE: IF THE COMPUTER UTILIZED IS A LSI 11 OR A COMPUTER WITHOUT A SWITCH REGISTER, THE PROGRAM WILL UTILIZE LOCATIONS 174 AND 176 AS A "DISPLAY" REGISTER AND A "SWITCH" REGISTER RESPECTIVELY. THE OPERATOR WILL BE RESPONSIBLE FOR THE LOADING OF THE "SWITCH" REGISTER LOCATION PRIOR TO STARTING OR RESTARTING THE PROGRAM.

4.2 STARTING ADDRESS OR ADDRESSES

200	IS THE STARTING ADDRESS OF THE ACCEPTANCE TEST
204	IS THE RESTART ADDRESS OF THE ACCEPTANCE TEST
210	IS THE STARTING ADDRESS OF THE KEYBOARD CHARACTER TEST
214	IS THE STARTING ADDRESS OF THE KEYBOARD OCTAL VALUE LOOP
220	IS THE STARTING ADDRESS OF THE KEYBOARD ECHO LOOP
224	IS THE SPECIAL STARTING ADDRESS FOR VT-50 PRODUCTION

S. OPERATING PROCEDURE

THE OPERATOR MUST INSERT THE CORRECT INFORMATION IN THE SWITCH REGISTER WHEN REQUIRED BY THE PROGRAM OR AN ERROR WILL OCCUR. ONCE STARTED, THE TEST WILL RUN IN ITS NORMAL MANNER WITHOUT OPERATOR INTERVENTION OR SWITCH CHANGES.

THIS PROGRAM ALLOWS THE OPERATOR TWO MODES OF TEST PATTERN SELECTION. THESE MODES ARE SELECTED BY THE STATE OF SW 07 AT THE BEGINNING OF THE PROGRAM. WHEN SW 07 IS A ZERO, THE PROGRAM IS UNDER SWITCH REGISTER CONTROL FOR TEST PATTERN SELECTION. IF SW07 IS EQUAL TO A ONE, THE PROGRAM IS UNDER KEYBOARD CONTROL OF THE TEST PATTERN SELECTION. IN THIS MODE THE OPERATOR WILL BE REQUIRED TO TYPE IN ON THE CONSOLE TTY THE FIRST AND LAST OCTAL BASE ADDRESS OF THE DL-11'S TO WHICH VT-50'S ARE CONNECTED.

IN THE KEYBOARD SELECT MODE, TWO CHARACTERS ARE USED TO SELECT THE "STARTING WITH" OR "LOOPING ON" A PARTICULAR TEST PATTERN BY "/" OR "\" RESPECTFULLY.

THE "/" KEY IS USED TO SUSPEND THE CURRENT TEST AND ASK THE OPERATOR A WHICH TEST PATTERN HE/SHE WISHES TO START. THE OPERATOR NOW DEPRESSES THE LETTER WHICH REPRESENTS THE TEST PATTERN TO BE STARTED WITH. REFER TO THE PROGRAM LISTING TABLE OF CONTENTS FOR THE TEST LETTER OF EACH PATTERN.

THE "\" KEY IS USED TO SUSPEND THE CURRENT TEST AND ASK THE OPERATOR WHICH TEST PATTERN HE/SHE WISHES TO LOOP ON. THE OPERATOR NOW DEPRESSES THE LETTER OF THE TEST TO LOOP ON.

IF DURING THE EXECUTION OF A TEST PATTERN, A KEY IS DEPRESSED AND SW 07 EQUALS A ZERO, AN ERROR WILL BE REPORTED TO THE CONSOLE TTY. IF SW 07 EQUALS A ONE, AND THE CHARACTER RECEIVED WAS NOT A "/" OR "\", AN ERROR WILL BE REPORTED. THE CODES "X-OFF" AND "X-ON" ARE THE ONLY EXCEPTIONS.

6. ERRORS

THIS PROGRAM USES THE DIAGNOSTIC 'SYSMAC' PACKAGE FOR
ERROR REPORTING AND TYPEOUT. REFER TO THE "ERROR POINTER TABLE"
FOR TYPE AND DESCRIPTION OF ERRORS.
THE ERROR INFORMATION CONSISTS OF THE FOLLOWING:

ERRPC - LOCATION AT WHICH AN ERROR WAS DETECTED
VTNOW - CURRENT DL-11 BUS ADDRESS OF VT50/52 UNDER TEST
TSTNUM- TEST PATTERN NUMBER OF FAILING TEST
EXPT - EXPECTED INPUT CHARACTER
RCVD - RECEIVED INPUT CHARACTER

7. RESTRICTIONS

- A. THE OPERATOR SHOULD SET SW 15 AND 13 IF THE VT50/52 UNDER TEST
IS THE CONSOLE TTY.
- B. ONLY ONE VT50/52 CAN BE TESTED AT ONE TIME.
- C. THE FIRST TIME AFTER LOADING THE PROGRAM, THE TERMINAL IDENTIFIER
MUST BE RUN.

8. MISCELLANEOUS

8.1 EXECUTION TIME

EXECUTION TIME WILL VARY WITH THE "BAUD" RATE, AND IF A COPIER IS CONNED
THE PROGRAM WILL TYPE 'END PASS' ON THE CONSOLE WHEN A PASS HAS BEEN COM
THE KEYBOARD LOOP AND CHARACTER TEST WILL NOT EXIT UNTIL
THE PROGRAM IS RESTARTED.

8.2 DEVICE ADDRESS PROGRAM LOCATIONS (AT APPROX. 1240)

THE LOCATION "FIRST" CONTAINS THE FIRST DL11 ADDRESS IF SEVERAL VT-50'S
ARE BEING TESTED. THE DEFAULT IS THE CONSOLE ADDRESS (177560)
THE LOCATION "LAST" CONTAINS THE LAST DL11 ADDRESS IF SEVERAL VT-50'S
ARE BEING TESTED. LOCATION VTNOW CONTAINS THE CURRENT DL11 BASE ADDRESS.

*NOTE: IF THESE LOCATIONS ARE CHANGED, THE OPERATOR MUST START
THE TEST AGAIN AT LOC. 200. THE PROGRAM WILL USE THE BASE
ADDRESS TO UPDATE THE ACTUAL PROGRAM VALUES.

8.3 COPIER SAVE PAPER SWITCH

IF SW 10 = 1 AND A COPIER IS INSTALLED, THE COPIER TESTS
WILL BE EXECUTED ON THE FIRST PASS AND THEN BYPASSED FOR
THE NEXT TEN PASSES. THIS REDUCES PAPER USAGE WHEN THE
PROGRAM IS RUN FOR AN EXTENDED PERIOD. (LOC. PTCT IS # OF PASSES

9. PROGRAM DESCRIPTION (SCREEN)

9.1 A TERMINAL IDENTIFICATION TEST

THIS TEST WILL INTERROGATE THE VT50/52 UNDER TEST AS TO IT'S TYPE.
THE RESPONSE RECIEVED IS USED TO DETERMINE THE MODES TO BE TESTED
(IE. COPIER, 12/24 LINES, D.C.A (DIRECT CURSOR ADDRESSING, ETC.
IF AN UNDEFINED OR AN INCORRECT RESPONSE IS RECIEVED, THE PROGRAM
ASSUMES VT50A MODEL (IE. 12 LINES, NO COPIER, NO D.C.A., NO EXTRA KEYPAC

9.2 B FULL SCREEN OF THE LETTER E

THIS TEST WILL FILL THE SCREEN WITH THE LETTER E.
THIS TEST WILL LOAD THE SCREEN RAM WITH A SINGLE
CHARACTER IN ALL LOCATIONS

9.3 C DATA PATH AND RAM NOISE TEST

THIS TEST WILL PRODUCE TWO FULL SCREENS OF A WORST CASE
NOISE PATTERN FOR THE SCREEN RAM AND THE DATA PATH
ALTERNATING LINES OF THE FOLLOWING
PATTERNS SHOULD BE DISPLAYED.

*U*U*U*U*U CODES 52 AND 125
2'2'2'2'2' CODES 77 AND 100
*U*U*U*U*U
2'2'2'2'2'

9.4 D SIMPLE CHARACTER SET

ONE FULL LINE OF EACH CHARACTER
(CODES 40 THRU 137) OF THE CHARACTER
SET. THIS WILL ALSO TEST THAT ALL WORDS
IN THE SCREEN RAM CAN BE LOADED.
THIS WILL ALSO EXERCISE THE SCROLLING FUNCTION.

IE: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD

9.5 E INCREMENTING AND SLIDING CHARACTER SET

ONE FULL LINE OF AN INCREMENTING
CHARACTER SET ACROSS THE FULL SCREEN
STARTING WITH THE CODE 40 (SPACE)
THE NEXT LINE SHOULD BEGIN WITH THE
"!" CHARACTER (CODE 41) ETC. CONTINUE
THE PATTERN BY INCREMENTING THE
FIRST COLUMN CHARACTER UNTIL THE
FULL CHARACTER SET HAS BEEN EXHAUSTED.

9.6 F CURSOR MOTION

IN THIS TEST THE BASIC CURSOR MOTIONS ARE
TESTED. THE FOLLOWING TEST PATTERN
IS GENERATED TO TEST THE CURSOR FUNCTIONS

BEFORE

```
-----  
+5+  
+  
+  
+  
+  
+ 2 1 +  
+  
+  
+  
+  
+ 3 4+  
+-----
```

UPON COMPLETION OF THE SETUP THE
BLINKING CURSOR WILL BE TO THE RIGHT
OF #4.

TO TEST "CURSOR UP": GENERATE CURSOR UP 6/12 TIMES AND
DISPLAY A "X"

TO TEST "CURSOR LEFT": GENERATE CURSOR LEFT (BACKSPACE)
FOURTY FOUR TIMES AND DISPLAY A "X"

TO TEST "CURSOR DOWN": GENERATE CURSOR DOWN 6/12 TIMES AND
DISPLAY A "X"

TO TEST "CURSOR RIGHT": GENERATE CURSOR RIGHT FOURTY TWO TIMES AND
DISPLAY A "X"

TO TEST "CURSOR HOME": GENERATE CURSOR HOME AND DISPLAY A "X"

AFTER

```
-----  
+X+  
+  
+  
+  
+  
+ X X +  
+  
+  
+  
+  
+ X 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
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
92
93
94
95
96
97
98
99
100

COPIER TEST PATTERNS

THE TEST PATTERNS USED TO TEST THE COPIER ARE THE SAME PATTERNS AS SOME OF THE SCREEN TESTS. TWO ADDITIONAL PATTERNS HAVE BEEN INCLUDED.

9.15 M GRAPHICS MODE/REVERSE LINE FEED TEST

THIS TEST IS EXECUTED ONLY IF UNIT IS A VT52. GRAPHICS MODE IS ENTERED AND 37 LINES OF GRAPHICS CHARACTERS (LINE LENGTH IS DECREMENTED BY ONE CHAR. AFTER A LINE IS PRINTED) ARE GENERATED. THE BOTTOM LINE OF THE WILL FORM A COMPLETE SERIES OF GRAPHIC CHARACTERS WHICH CAN BE VERIFIED FOR ACCURACY. IF REVERSE LINE FEED IS NOT OPERATIONAL A SINGLE LINE OF GRAPHICS CHARACTERS WILL BE GENERATED ON LINE 0 ONLY.

9.16 N COPIER - AUTO COPY MODE

THIS TEST IS USED TO DETERMINE IF THE AUTO COPY MODE IS FUNCTIONING CORR THE SUB-TEST TITLE AND A ROW OF THE LETTER 'E' WILL BE DISPLAYED. THE AUTO-COPY MODE IS ENABLED. THE COPIER SHOULD THEN START. A PROGRAM DELAY IS EXECUTED BETWEEN EACH LINE. THE RESULT IS THE COPIER SHOULD COPY ONE LINE AND THEN STOP FOR THE PROGRAM DELAY TIME. WHEN THE PROGRAM DELAY IS COMPLETED ANOTHER LINE OF 'E'S WILL BE DISPLAYED AND THE COPIER SHOULD START AGAIN. UPON COMPLETION OF THE 12 OR 24 LINES, DISABLE AUTO-COPY MODE IS SENT TO THE VT50. IF THE AUTO COPY MODE IS NOT DISABLED THE NEXT SUB-TEST WOULD PERFORM IN A SIMILAR MANNER.

9.17 O COPIER - FULL SCREEN OF THE LETTER E

SAME AS SCREEN PATTERN

9.20 P COPIER - DATA PATH TEST PATTERN

SAME AS SCREEN PATTERN

9.21 Q COPIER - SINGLE CHARACTER PER LINE

SAME AS SCREEN PATTERN

9.22 R COPIER - ROTATING CHARACTER TEST

SAME AS SCREEN PATTERN

579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618

9.30 X KEYBOARD CHARACTER TEST

THIS TEST IS DESIGNED TO VERIFY THAT CORRECT CHARACTER CODES AND PARITY BIT ARE GENERATED WHEN A KEY IS DEPRESSED. THIS TEST REQUIRES THE OPERATOR TO EXECUTE THE INSTRUCTIONS DISPLAYED ON THE SCREEN. THE OPERATOR SHOULD ONLY DEPRESS ONE KEY AT A TIME, WITH SOME EXCEPTIONS. THE OPERATOR WILL BE REQUIRED TO SKIP THOSE KEYS THAT ARE NOT IMPLEMENTED IF THE UNIT IS A VT50. THE PROGRAM WILL INFORM THE OPERATOR WHICH ROW TO TEST.

IN TESTING THE PARITY BIT, SW 0 AND 1 ARE USED TO INFORM THE PROGRAM OF THE EXPECTED PARITY. AN INCORRECT SWITCH SETTING WILL RESULT IN AN ERROR.

9.31 Y KEYBOARD OCTAL VALUE LOOP

THIS LOOP IS PROVIDED TO ENABLE THE OPERATOR TO EXAMINE THE OCTAL VALUE OF A CHARACTER. WHEN A KEY IS DEPRESSED, THE OCTAL VALUE WILL BE DISPLAYED. IF THE CHARACTER WAS A PRINTABLE CHARACTER, IT WILL BE DISPLAYED. THOSE CODES DEFINED AS "CONTROL" WILL BE DISPLAYED AS A TWO LETTER MNEMONIC (IE. DE=DELETE, BL=BELL, CL=CURSOR LEFT ETC.)

9.32 Z KEYBOARD ECHO LOOP

WHEN A KEY IS DEPRESSED, THE CHARACTER WILL BE DISPLAYED. NO MODIFICATION OR DATA TEST IS PERFORMED. THIS TEST CAN BE USED TO DETERMINE IF THERE IS A "UART" OR SERIAL LINE PROBLEM. WHEN THE VT50/52 IS IN LOCAL MODE (NO UART/SERIAL LINE), THE CHARACTERS SHOULD BE ECHOED CORRECTLY. IF NOT, THE PROBLEM IS IN THE VT50 UNIT. IF CHANGED TO REMOTE MODE AND THE CHARACTERS ARE IN ERROR, THERE IS A UART/SERIAL LINE PROBLEM.

630

```

TITLE MAINDEC-11-DZVTC-E
*COPYRIGHT (C) 1977
*DIGITAL EQUIPMENT CORP.
*MAYNARD, MASS. 01754
*
*PROGRAM BY TERMINAL DIAGNOSTICS
*
*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
*PACKAGE (MAINDEC-11-DZVTC-C3), JAN 19, 1977.
*

```

631

```

SBTTL BASIC DEFINITIONS

```

```

*INITIAL ADDRESS OF THE STACK POINTER *** 1100 ***

```

001100

```

STACK= 1100
EQUIV ENT.ERROR      ;; BASIC DEFINITION OF ERROR CALL
EQUIV TOT.SCOPE      ;; BASIC DEFINITION OF SCOPE CALL

```

```

*MISCELLANEOUS DEFINITIONS

```

 000011
 000012
 000013
 000200
 177776

```

HT= 11      ;; CODE FOR HORIZONTAL TAB
LF= 12      ;; CODE FOR LINE FEED
CR= 13      ;; CODE FOR CARRIAGE RETURN
CRLF= 200   ;; CODE FOR CARRIAGE RETURN-LINE FEED
PS= 177776  ;; PROCESSOR STATUS WORD

```

 177774
 177772
 177570
 177570

```

EQUIV PS.PSW
STKLMT= 177774 ;; STACK LIMIT REGISTER
PIRQ= 177772   ;; PROGRAM INTERRUPT REQUEST REGISTER
DSMR= 177570   ;; HARDWARE SWITCH REGISTER
DOISP= 177570  ;; HARDWARE DISPLAY REGISTER

```

```

*GENERAL PURPOSE REGISTER DEFINITIONS

```

 000000
 000001
 000002
 000003
 000004
 000005
 000006
 000007
 000006
 000007

```

R0= %0      ;; GENERAL REGISTER
R1= %1      ;; GENERAL REGISTER
R2= %2      ;; GENERAL REGISTER
R3= %3      ;; GENERAL REGISTER
R4= %4      ;; GENERAL REGISTER
R5= %5      ;; GENERAL REGISTER
R6= %6      ;; GENERAL REGISTER
R7= %7      ;; GENERAL REGISTER
SP= %6      ;; STACK POINTER
PC= %7      ;; PROGRAM COUNTER

```

```

*PRIORITY LEVEL DEFINITIONS

```

 000000
 000040
 000100
 000140
 000200
 000240
 000300
 000340

```

PR0= 0      ;; PRIORITY LEVEL 0
PR1= 40     ;; PRIORITY LEVEL 1
PR2= 100    ;; PRIORITY LEVEL 2
PR3= 140    ;; PRIORITY LEVEL 3
PR4= 200    ;; PRIORITY LEVEL 4
PR5= 240    ;; PRIORITY LEVEL 5
PR6= 300    ;; PRIORITY LEVEL 6
PR7= 340    ;; PRIORITY LEVEL 7

```

```

*"SWITCH REGISTER" SWITCH DEFINITIONS

```

 100000
 040000
 020000
 010000

```

SW15= 100000
SW14= 40000
SW13= 20000
SW12= 10000

```

```

(1) 004000 SW11= 4000
(1) 002000 SW10= 2000
(1) 001000 SW09= 1000
(1) 000400 SW08= 4000
(1) 000200 SW07= 2000
(1) 000100 SW06= 1000
(1) 000040 SW05= 4000
(1) 000020 SW04= 2000
(1) 000010 SW03= 1000
(1) 000004 SW02= 4000
(1) 000002 SW01= 2000
(1) 000001 SW00= 1000
(1) .EQUIV SW00, SW01
(1) .EQUIV SW01, SW02
(1) .EQUIV SW02, SW03
(1) .EQUIV SW03, SW04
(1) .EQUIV SW04, SW05
(1) .EQUIV SW05, SW06
(1) .EQUIV SW06, SW07
(1) .EQUIV SW07, SW08
(1) .EQUIV SW08, SW09
(1) .EQUIV SW09, SW10
(1) .EQUIV SW10, SW11
    
```

```

(1) 100000 .DATA BIT DEFINITIONS (BIT00 TO BIT15)
(1) 040000 BIT15= 100000
(1) 020000 BIT14= 40000
(1) 010000 BIT13= 20000
(1) 004000 BIT12= 10000
(1) 002000 BIT11= 4000
(1) 001000 BIT10= 2000
(1) 000400 BIT09= 1000
(1) 000200 BIT08= 400
(1) 000100 BIT07= 200
(1) 000040 BIT06= 100
(1) 000020 BIT05= 40
(1) 000010 BIT04= 20
(1) 000004 BIT03= 10
(1) 000002 BIT02= 4
(1) 000001 BIT01= 2
(1) .EQUIV BIT00, BIT01
(1) .EQUIV BIT01, BIT02
(1) .EQUIV BIT02, BIT03
(1) .EQUIV BIT03, BIT04
(1) .EQUIV BIT04, BIT05
(1) .EQUIV BIT05, BIT06
(1) .EQUIV BIT06, BIT07
(1) .EQUIV BIT07, BIT08
(1) .EQUIV BIT08, BIT09
(1) .EQUIV BIT09, BIT10
(1) .EQUIV BIT10, BIT11
(1) .EQUIV BIT11, BIT12
(1) .EQUIV BIT12, BIT13
(1) .EQUIV BIT13, BIT14
(1) .EQUIV BIT14, BIT15
    
```

```

(1) 000004 .BASIC "CPU" TRAP VECTOR ADDRESSES
(1) 000010 TRARVEC= 4 :: TIME OUT AND OTHER ERRORS
(1) 000014 RESVEC= 10 :: RESERVED AND ILLEGAL INSTRUCTIONS
(1) 000014 TBITVEC= 14 :: "T" BIT
(1) 000014 TRIVEC= 14 :: TRACE TRAP
    
```


640

.SBTTL COMMON TAGS

 : THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS
 : USED IN THE PROGRAM.

(1)	001100	001100	SCMTAG:	.WORD	0	START OF COMMON TAGS
(1)	001100	000000	SPASS:	.WORD	0	CONTAINS PASS COUNT
(1)	001102	000	STSTN:	.BYTE	00	CONTAINS THE TEST NUMBER
(1)	001103	000	SERFLG:	.BYTE	0000	CONTAINS ERROR FLAG
(1)	001104	000000	SICNT:	.WORD	0000	CONTAINS SUBTEST ITERATION COUNT
(1)	001106	000000	SLPAD:	.WORD	0000	CONTAINS SCOPE LOOP ADDRESS
(1)	001110	000000	SLPERR:	.WORD	0000	CONTAINS SCOPE RETURN FOR ERRORS
(1)	001112	000000	SERTL:	.WORD	0000	CONTAINS TOTAL ERRORS DETECTED
(1)	001114	000	SITEM:	.BYTE	0000	CONTAINS ITEM CONTROL BYTE
(1)	001115	001	SERMAX:	.BYTE	1	CONTAINS MAX. ERRORS PER TEST
(1)	001116	000000	SERRPC:	.WORD	0000	CONTAINS PC OF LAST ERROR INSTRUCTION
(1)	001120	000000	SGOADR:	.WORD	0000	CONTAINS ADDRESS OF 'GOOD' DATA
(1)	001122	000000	SBOADR:	.WORD	0000	CONTAINS ADDRESS OF 'BAD' DATA
(1)	001124	000000	SGO DAT:	.WORD	0000	CONTAINS 'GOOD' DATA
(1)	001126	000000	SBO DAT:	.WORD	0000	CONTAINS 'BAD' DATA
(1)	001130	000000		.WORD	0000	RESERVED--NOT TO BE USED
(1)	001132	000000		.WORD	0000	
(1)	001134	000	SAUTOB:	.BYTE	0000	CONTAINS AUTOMATIC MODE INDICATOR
(1)	001135	000	SINTAG:	.BYTE	0000	CONTAINS INTERRUPT MODE INDICATOR
(1)	001136	000000		.WORD	0000	
(1)	001140	177570	SWR:	.WORD	DSWR	CONTAINS ADDRESS OF SWITCH REGISTER
(1)	001142	177570	DISPLAY:	.WORD	DDISP	CONTAINS ADDRESS OF DISPLAY REGISTER
(1)	001144	177560	STKS:	177560		CONTAINS TTY KBO STATUS
(1)	001146	177562	STKB:	177562		CONTAINS TTY KBO BUFFER
(1)	001150	177564	STPS:	177564		CONTAINS TTY PRINTER STATUS REG. ADDRESS
(1)	001152	177566	STPB:	177566		CONTAINS TTY PRINTER BUFFER REG. ADDRESS
(1)	001154	000	SNULL:	.BYTE	0	CONTAINS NULL CHARACTER FOR FILLS
(1)	001155	002	SFILLS:	.BYTE	2	CONTAINS # OF FILLER CHARACTERS REQUIRED
(1)	001156	012	SFILLC:	.BYTE	12	CONTAINS INSERT FILL CHARS. AFTER A "LINE FEED"
(1)	001157	000	STPFLG:	.BYTE	0	CONTAINS "TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
(1)	001160	000000	SREGAD:	.WORD	0	CONTAINS THE ADDRESS FROM WHICH (\$REGO) WAS OBTAINED
(3)	001162	000000	SREGO:	.WORD	0	CONTAINS ((SREGAD)+0)
(3)	001164	000000	SREG1:	.WORD	0	CONTAINS ((SREGAD)+2)
(1)	001166	077	SQUES:	.ASCII	' '	CONTAINS QUESTION MARK
(1)	001167	015	SCRLF:	.ASCII	' '	CONTAINS CARRIAGE RETURN
(1)	001170	000012	SLF:	.ASCII	' '	CONTAINS LINE FEED

MAINDEC-11-DZVTC-E MACY11 30(1046) 31-AUG-77 09:17 PAGE 13-4
 DZVTC.P11 15-JUN-77 10:41 ERROR POINTER TABLE

SEG 0018

.SBTTL ERROR POINTER TABLE

;; THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR.
 ;; THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN
 ;; *LOCATION SITEMB. THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.
 ;; *NOTE1: IF SITEMB IS 0 THE ONLY PERTINENT DATA IS (\$ERRPC).
 ;; *NOTE2: EACH ITEM IN THE TABLE CONTAINS 4 POINTERS EXPLAINED AS FOLLOWS:

;; * EM :: POINTS TO THE ERROR MESSAGE
 ;; * DM :: POINTS TO THE DATA HEADER
 ;; * DT :: POINTS TO THE DATA
 ;; * DF :: POINTS TO THE DATA FORMAT

\$ERRTB:

645	001172	020345	: ITEM 1	EM1	: ERROR FLAG ON HOST TRANSMIT STATUS
646	001174	020570		DM1	: ERRPC VTNOW TSTNUM
647	001176	020734		DT1	: \$ERRPC VTNOW TSTNUM
648	001200	000000			
651	001202	020412	: ITEM 2	EM2	: NO HOST INPUT FLAG RECEIVED
652	001204	020570		DM1	: ERRPC VTNOW TSTNUM
653	001206	020734		DT1	: \$ERRPC VTNOW TSTNUM
654	001210	000000		C	
657	001212	020441	: ITEM 3	EM3	: INCORRECT I.D. RESPONSE
658	001214	020617		DM3	: ERRPC VTNOW 1ST WD 2ND WD 3RD WD
659	001216	020744		DT3	: \$ERRPC VTNOW SAVE4 SAVE2 SAVE3
660	001220	000000		0	
663	001222	020465	: ITEM 4	EM4	: INCORRECT OR UNEXPECTED INPUT CHARACTER
664	001224	020666		DM4	: ERRPC VTNOW TSTNUM EXPT RCVC
665	001226	020760		DT4	: \$ERRPC VTNOW TSTNUM \$GDDAT \$BDDAT
666	001230	000000		0	
669	001232	020530	: ITEM 5	EM5	: INVALID BUSS ADDRESS, TRY AGAIN
670	001234	000000		0	
671	001236	000000		0	
672	001240	000000		0	
673	001242	177560	FIRST:	177560	: FIRST DEVICE ADDRESS OF SEQUENTIAL DL-11-A/B TYPE DEVICE
674					: DEFAULT TO THE CONSOLE ADDRESS
675	001244	000000	LAST:	0	: LAST DEVICE ADDRESS OF DL-11-A/B TYPE
676	001246	177560	VTNOW:	177560	: CURRENT DEVICE BUSS ADDRESS
677	001250	100011	PTCT:	100011	: COPIER PAPER SAVE COUNT (LOW BYTE)
678	001252	000000	TSTNUM:	0	: ERROR PATTERN
680	001254	000300	TIMEO:	300	: CHARACTER FLAG TIMEOUT CONSTANT
681	001256	000005	SUBTST:	5	: SUBTEST DELAY CONSTANT
687	001260	000000	V*EX*:	0	: I.D. AND CHARASTICS

688	001262	000053			LASTLN: 53	:OR 67	:LAST VALID LINE # +40
689	001264	001700			TOTALC: 960.	:OR 1920.	:TOTAL CHARACTER COUNT
690	001266	000014			VHO: 12.	:24.	:VERTICAL LINE COUNT
691	001270	000006			VH1: 6.	:12.	:1/2 VERTICAL LINE COUNT
692	001272	000003			VH2: 3.	:6.	:1/4 VERTICAL LINE COUNT
693	001274	000000			PRICNT: 0		
694	001276	177560			VTIS: 177560		:DEVICE ADDRESSES
695	001300	177562			VTIB: 177562		:IN DATA
696	001302	177564			VTOS: 177564		:OUT STAT
697	001304	177566			VTOB: 177566		:OUT DATA
698	001306	000000			STCHAR: 0		:TEMP REG'S
699	001310	000140			LASTCH: 140	:OR 200	:FIRST NON-VALID CHARACTER
700	001312	000000			TEMP: 0		:TEMP REG'S
701	001314	000000			TEMPJ: 0		
702	001316	000204			PNTWID: 132.		:COLUMN COUNT FOR LINE PRINTER.
703	001320	000120			WIDTH: 80.		:COLUMN WIDTH
704	001322	000000			SAVE1: 0		:TEMP REG'S
705	001324	000000			SAVE2: 0		
706	001326	000000			SAVE3: 0		
707	001330	000000			SAVE4: 0		
708	001332	000000			WFTST: 0		:NON-ZERO IF SA = 224
709							
710							
711	001334	012737	002466	002402	BEGIN: MOV	#TST2,WHERE	:STARTING ACCEPTANCE TEST ADDRESS
712	001342	005037	001330		CLR	SAVE4	
713	001346	005037	001332		CLR	WFTST	
714	001352	005037	001274		CLR	PRICNT	
715	001356	000426			BR	GINA	
716	001360	005237	001332		MANFU: INC	WFTST	:SET W.F. PARAM.
717	001364	005037	001274		RBEGIN: CLR	PRICNT	:RESTART ADDRESS
718	001370	012737	002466	002402	MOV	#TST2,WHERE	
719	001376	000413			BR	GIN	
720	001400	012737	010100	002402	BEGIN3: MOV	#KRBECH,WHERE	:START AT ECHO LOOP
721	001406	000407			BR	GIN	
722	001410	012737	007404	002402	BEGIN1: MOV	#KRBTST,WHERE	:STARTING CHARACTER TEST ADDRESS
723	001416	000403			BR	GIN	
724	001420	012737	007110	002402	BEGIN2: MOV	#KRBECH,WHERE	:STARTING CHARACTER LOOP ADDRESS
725	001426	012737	000001	001330	GIN: MOV	#1,SAVE4	
726	001434	000005			GINA: RESET		
727					:SBTTL INITIALIZE THE COMMON TAGS		
(1)					:CLEAR THE COMMON TAGS (\$CMTAG) AREA		
(1)	001436	012706	001100		MOV	#SCMTAG,R6	:FIRST LOCATION TO BE CLEARED
(1)	001442	005026			CLR	(R6)+	:CLEAR MEMORY LOCATION
(1)	001444	022706	001140		CMP	#SWR,R6	:;DONE?
(1)	001450	001374			BNE	-6	:LOOP BACK IF NO
(1)	001452	012706	001100		MOV	#STACK,SP	:SETUP THE STACK POINTER
(1)					:INITIALIZE A FEW VECTORS		
(1)	001456	012737	023624	000020	MOV	#SCOPE,#IOTVEC	:IOT VECTOR FOR SCOPE ROUTINE
(1)	001464	012737	000340	000022	MOV	#340,#IOTVEC+2	:LEVEL 7
(1)	001472	012737	023742	000030	MOV	#ERROR,#EMTVEC	:EMT VECTOR FOR ERROR ROUTINE
(1)	001500	012737	000340	000032	MOV	#340,#EMTVEC+2	:LEVEL 7

MAINDEC-11-DZVTC-E MACY11 30(1046) 31-AUG-77 09:17 PAGE 13-7
DZVTCE.P11 15-JUN-77 10:41 INITIALIZE THE COMMON TAGS

SEQ 0021

767 002034 010120
768 002036 005721
769 002040 010110

MOV R1,(R0)+
TST (R1)+
MOV R1,(R0)

K02

MAINDEC-11-DZVTC-E

MACY11 30(1046)

31-AUG-77 09:17 PAGE 14-1

DZVTC.P11

15-JUN-77

10:41

T1

A

TERMINAL IDENTIFICATION TEST

SEQ 0023

```

828 002304 062737 000014 001262 ADD #12.,LASTLN ;ADJUST VALID LINE # +40
829
830 002312 013737 001266 001270 6$: MOV VMO,VH1 ;LOAD OTHER LINE COUNTS
831 002320 006237 001270 ASR VH1
832 002324 013737 001270 001272 MOV VH1,VH2
833 002332 006237 001272 ASR VH2
834 002336 012737 000140 001310 MOV #140,LASTCH ;LOAD FIRST NON-VALID CHARACTER
835 002344 032737 004000 001260 BIT #BIT11,VT5XX ;TEST IF UPPER-LOWER CASE TERM.
836 002352 001403 BEQ 7$ ;BR IF NOT
837 002354 062737 000037 001310 ADD #37,LASTCH ;UPDATE TO ALLOW UP-LW CASE CHARACTER SET
838 002362 7$: BR 12$ ;:BR AND START TESTING
839 002362 000403
840 002364 104002 2$: ERROR 2 ;NO RESPONSE FROM UUT ADTER ASKING FOR IDENTIFY
841 002366 000713 BR 11$
842 002370 000240 NOP
843 002372 004737 011262 12$: JSR PC,DELAY
844 002376 000177 000000 JMP @WHERE
845 002402 002466 WHERE: TST2
846
847 ;I.D. VALUES AND CHARACTERISTICS
848 : BIT15 = 1 24 LINES
849 : BIT14 = 1 COPIER CONNECTED
850 : BIT13 = 1 DIRECT CURSOR ADDRESSING (ESC Y) + "ESC-B" + "ESC-C"
851 : BIT12 = 1 VT50H KEYPAD
852 : BIT11 = 1 UPPER AND LOWER CASE CHARACTERS
853 : BIT10 = 1 PRINTER CONNECTED
854 : BIT09 = 1 VT52X MODEL
855
856 ;LOW BYTE CONTAINS THE I.D. FOR EACH KNOWN VT5??
857
858 002404 000101 TYPEPT: .WORD 000101 ;I.D. = 101 ;VT50A
859 002406 040102 .WORD 040102 ;I.D. = 102 ;VT50B COPIER
860 002410 140103 .WORD 140103 ;I.D. = 103 ;VT55 24 LINES, COPIER(VT50 BASED)
861 002412 070110 .WORD 070110 ;I.D. = 110 ;VT50H COPIER DCA VT50H KEYPAD
862 002414 135113 .WORD 135113 ;I.D. = 113 ;VT52
863 002416 175114 .WORD 175114 ;I.D. = 114 ;VT52 WITH COPIER
864 002420 137115 .WORD 137115 ;I.D. = 115 ;VT52 WITH PRINTER.
865 002422 175105 .WORD 175105 ;I.D. = 105 ;VT55.24LINES,WITH COPIER(VT52 BASED)
866 002424 000000 0
867 002426 000000 0
868 002430 000000 0
869 002432 000000 0
870 002434 000000 0
871
872 ;ASCII MESSAGE POINTERS
873
874 002436 016760 MSGTYP: VT50A ;VT50A NO COPIER
875 002440 017032 VT50B ;VT50B COPIER
876 002442 017101 VT55 ;VT55 COPIER
877 002444 017163 VT50H ;VT50H COPIER
878 002446 017241 VT52K ;VT52
879 002450 017324 VT52L ;VT52 WITH COPIER
880 002452 017405 VT52M ;VT52 WITH PRINTER
881 002454 017466 VT55E
882 002456 000000 0

```

L02

MAINDEC-11-DZVTC-E MACY11 30(1046) 31-AUG-77 09:17 PAGE 14-2
 DZVTC.P11 15-JUN-77 10:41 T1 A TERMINAL IDENTIFICATION TEST

SEG 0024

```

883 002460 000000      0
884 002462 000000      0
885 002464 000000      0
886      ;*****
(3)      ;TEST 2      B      FULL SCREEN OF A CHARACTER
(3)      ;*****
(2) 002466 000004      †ST2: SCOPE
887      JSR      R5,AMSG
888 002470 004537 011442      M91
889 002474 013245
890      JSR      PC,FILLWC      ;FILL SCREEN WITH A 'E'S
891 002476 004737 013010
892      JSR      PC,DELAY
893 002502 004737 011262
894
895      ;*****
(3)      ;TEST 3      C      DATA TRANSFER PATH TEST
(3)      ;*****
(2) 002506 000004      †ST3: SCOPE
896 002510 004537 011442      JSR      R5,AMSG      ;DISPLAY HEADING
897 002514 013313      M92
898 002516 013737 001270 001312      MOV      VHI,TEMP      ;SET-UP A COUNTER
899
900      JSR      R5,DTPSR      ;SET-UP BUFFER
901 002530 000077      77      ;OCTAL '2'
902 002532 000100      100      ;OCTAL '3'
903
904      JSR      R5,DTPSRB      ;SET-UP BUFFER
905 002540 000125      125      ;OCTAL 'U'
906 002542 000052      52      ;OCTAL '*'
907
908 002544 004737 010236      1$: JSR      PC,XPRNT      ;DISPLAY THIS LINE
909 002550 005337 001312      DEC      TEMP      ;COMPLETED FULL COUNT?
910 002554 001373      BNE      1$      ;BRANCH IF NOT COMPLETED
911
912      JSR      PC,DELAY      ;TEST DELAY SWITCH
913
914      ;*****
(3)      ;TEST 4      D      SINGLE CHARACTER ACROSS ALL COLS. 'ALL CHARACTERS'
(3)      ;*****
(2) 002562 000004      †ST4: SCOPE
914 002564 004537 011442      JSR      R5,AMSG      ;DISPLAY HEADING
915 002570 013342      M93
916 002572 012737 000040 001306      MOV      #40,STCHAR      ;SET-UP STARTING CHARACTER
917
918      MOV      VHI,TEMPO      ;LOAD COUNT
919 002606 013701 001306      1$: MOV      STCHAR,R1      ;LOAD R1= TO CHARACTER
920 002612 004737 010134      JSR      PC,FILBUF      ;LOAD A BUFFER WITH THAT CHARACTER
921
922      JSR      PC,XPRNT      ;DISPLAY A FULL LINE FROM THE BUFFER
923
924      DEC      TEMPO      ;DONE ?
925 002626 001005      BNE      2$      ;FINISHED
926 002630 004737 011262      JSR      PC,DELAY
927 002634 013737 001266 001314      MOV      VHI,TEMPO
928 002642 005237 001306      2$: INC      STCHAR      ;UPDATE THE CHARACTER
929 002646 023737 001310 001306      CMF      LASTCH,STCHAR      ;TEST FOR FINAL CHARACTER

```

MAINDEC-11-DZVTC-E MACY11 30(1046) 31-AUG-77 09:17 PAGE 14-3
 DZVTC.P11 15-JUN-77 10:41 T4 D SINGLE CHARACTER ACROSS ALL COLS. (ALL CHARACTERS)

SEQ 0025

```

930 002654 001354 BNE 1$ ;BRANCH IF NOT COMPLETED
931
932 002656 004737 011262 JSR PC,DELAY ;TEST DELAY SWITCH
933
934 *****
(3) ;*TEST 5 E ROTATING CHARACTERS ACROSS ALL COLS. (ALL CHARACTERS)
*****
(2) 002662 000004
935 002664 004537 011442 TST5: SCOPE
936 002670 013404 JSR R5,AMSG ;DISPLAY HEADING
937 002672 012737 000040 001306 MOV #40,STCHAR ;SET-UP STARTING CHARACTER
938
939 002700 013737 001266 001314 1$: MOV V#0,TEMPO ;LOAD TEMP
940 002706 013701 001306 MOV STCHAR,R1 ;LOAD R1=TO CHARACTER
941 002712 004537 010170 JSR R5,LIC ;LOAD A BUFFER STARTING WITH
942 002716 001320 WIDTH ; THAT CHARACTER AND WIDTH (BYTE)
943
944 002720 004737 010236 JSR PC,XPRNT ;DISPLAY A FULL LINE FROM THE BUFFER
945
946 002724 005337 001314 DEC TEMPO ;DONE ?
947 002730 001005 BNE 2$ ;BR IF YES
948 002732 004737 011262 JSR PC,DELAY
949 002736 013737 001266 001314 MOV V#0,TEMPO
950 002744 005237 001306 2$: INC STCHAR ;UPDATE THE STARTING CHARACTER
951 002750 023737 001310 001306 CMP LASTCH,STCHAR ;TEST FOR FINAL CHARACTER
952 002756 001353 BNE 1$ ;BRANCH IF NOT COMPLETED
953
954 002760 004737 011262 JSR PC,DELAY ;TEST DELAY SWITCH
955
956 *****
(3) ;*TEST 6 F CURSOR MOTION TEST
*****
(3)
(2) 002764 000004
956 002766 004537 011442 TST6: SCOPE
957 002772 013435 JSR R5,AMSG ;DISPLAY HEADING
958 M96
959 ;ROUTINE TO LOAD REFERENCE LINES FOR CURSOR MOTION TEST
960
961 002774 012700 025256 LBMT: MOV #BUFFER,RD ;LOAD POINTER
962 003000 012720 000065 MOV #65,(RD)+ ;LOAD #5
963 003004 013701 001270 MOV V#1,R1 ;LOAD R1
964 003010 005301 DEC R1
965 003012 004737 003110 JSR PC,MOVDN1 ;MOVE CURSOR DOWN
966 003016 004737 003126 JSR PC,MOVRIG ;MOVE RIGHT
967 003022 112720 000062 MOVB #62,(RD)+ ;LOAD #2
968 003026 004737 003126 JSR PC,MOVRIG ;MOVE RIGHT
969 003032 004737 003126 JSR PC,MOVRIG ;MOVE RIGHT
970 003036 112720 000040 MOVB #40,(RD)+
971 003042 112720 000061 MOVB #61,(RD)+ ;LOAD #1
972 003046 004737 003104 JSR PC,MOVDN1 ;MOVE DOWN
973 003052 004737 003126 JSR PC,MOVRIG ;MOVE RIGHT
974 003056 112720 000063 MOVB #63,(RD)+ ;LOAD #3
975 003062 004737 003126 JSR PC,MOVRIG
976 003066 004737 003126 JSR PC,MOVRIG ;MOVE RIGHT
977 003072 112720 000064 MOVB #64,(RD)+ ;LOAD #4
978 003076 112710 000377 MOVB #377,(RD) ;TERM
979 003102 000420 BR LBMT1 ;BR TO NEXT PART

```

```

980
981 003104 013701 001270      MOVDWN: MOV      VH1,R1
982 003110 112720 000015      MOVDN1: MOVB     #15,(R0)+      ;LOAD CR
983 003114 112720 000012      MOVB     #12,(R0)+      ;LOAD LF
984 003120 005301              DEC      R1
985 003122 001372              BNE      MOVDN1      ;LOOP UNTIL DONE
986 003124 000207              RTS       PC          ;EXIT
987
988 003126 012701 000024      MOVRIG: MOV      #20,R1      ;LOAD R1
989 003132 112720 000040      1$:      MOVB     #40,(R0)+      ;LOAD SPACES
990 003136 005301              DEC      R1
991 003140 100374              BPL      1$
992 003142 000207              RTS       PC          ;LOOP UNTIL DONE
993
994 003144 004737 010236      LBMT1: JSR      PC,XPRNT      ;DISPLAY THIS LINE
995 003150 004737 011262      JSR      PC,DELAY      ;TEST DELAY SWITCH
996
997      ;CURSOR MOTION SUBROUTINE
998      ;IF VT50H TYPE - USE "ESC-D" FOR CURSOR LEFT AND USE "ESC-B" FOR CURSOR DOWN
999 003154 013701 001270      LCM:      MOV      VH1,R1      ;LOAD COUNT
1000 003160 012700 025256      MOV      #BUFFER,R0      ;LOAD BUFFER POINTER
1001 003164 112720 000033      1$:      MOVB     #33,(R0)+      ;LOAD 'ESC'
1002 003170 112720 000101      MOVB     #101,(R0)+      ;LOAD 'A' CURSOR UP
1003 003174 005301              DEC      R1
1004 003176 001372              BNE      1$
1005 003200 112720 000130      MOVB     #130,(R0)+      ;LOAD 'X'
1006 003204 012701 000054      MOV      #44,R1      ;LOAD COUNT
1007 003210 032737 020000      BIT      #BIT13,VT5XX      ;TEST IF VT50H TYPE
1008 003216 001407              BEQ      20$      ;BR IF NOT
1009 003220 112720 000033      2$:      MOVB     #33,(R0)+      ;LOAD 'ESC'
1010 003224 112720 000104      MOVB     #104,(R0)+      ;LOAD 'CURSOR LEFT'
1011 003230 005301              DEC      R1
1012 003232 100372              BPL      2$
1013 003234 000404              BR       21$
1014 003236 112720 000010      20$:      MOVB     #10,(R0)+      ;LOAD "BACKSPACE"
1015 003242 005301              DEC      R1
1016 003244 100374              BPL      20$
1017 003246 112720 000130      21$:      MOVB     #130,(R0)+      ;LOAD 'X'
1018 003252 112720 000010      MOVB     #10,(R0)+      ;LOAD BACKSPACE
1019 003256 013701 001270      MOV      VH1,R1      ;LOAD COUNT
1020 003262 032737 020000      BIT      #BIT13,VT5XX      ;TEST IF VT50H TYPE
1021 003270 001407              BEQ      30$      ;BR IF NOT
1022 003272 112720 000033      3$:      MOVB     #33,(R0)+      ;LOAD 'ESC' CURSOR DOWN
1023 003276 112720 000102      MOVB     #102,(R0)+      ;
1024 003302 005301              DEC      R1
1025 003304 001372              BNE      3$
1026 003306 000404              BR       31$
1027 003310 112720 000012      30$:      MOVB     #12,(R0)+      ;LOAD CURSOR DOWN (LF)
1028 003314 005301              DEC      R1
1029 003316 001374              BNE      30$
1030 003320 112720 000130      31$:      MOVB     #130,(R0)+      ;LOAD 'X'
1031 003324 112720 000010      MOVB     #10,(R0)+      ;LOAD BACKSPACE
1032 003330 012701 000052      MOV      #42,R1      ;LOAD COUNT
1033 003334 112720 000033      4$:      MOVB     #33,(R0)+      ;LOAD 'ESC'
1034 003340 112720 000103      MOVB     #103,(R0)+      ;LOAD 'C' CURSOR RIGHT
1035 003344 005301              DEC      R1

```


1940	010660	000000	XOFFER: 0	
1941	010662	000000	XONARC: 0	
1942	010664	000000	ANESC: 0	
1943	010666	000000	IGNORE: 0	; WHEN SET IGNORE KEYBOARD FLAGS
1944				
1945			DETERMINE THE	TEST TO GO TO
1946	010670	004537	FNOW: JSR	RS,AMSC
1947	010674	017731	MO2	; ERROR ASK AGAIN
1948	010676	012706	FINDOT: MOV	STACK SP
1949	010702	004537	JSR	RS,AMSC
1950	010706	017643	FINDTA: MO1	
1951	010710	004737	JSR	PC,GETCHR
1952	010714	000770	BR	FINDOT
1953	010716	042700	BIC	#100600,RO ; MASK
1954	010722	122700	CMPS	#1A,RO ; TEST FOR NUMBER
1955	010726	101360	BHI	FNOA
1956	010730	122700	CMPS	#2,RO ; TEST FOR OTHERS
1957	010734	103755	BLO	FNOA
1958	010736	042700	BIC	#177740,RO ; MAKE 0-32
1959	010742	005300	DEC	RO
1960	010744	110037	MOVB	RO,\$TSTNM ; LOAD THAT TEST
1961	010750	006300	ASL	RO
1962	010752	005760	TST	DSPCH(RO) ; TEST IF VALID
1963	010756	001744	BEQ	FNOA ; BR IF NOT
1964	010760	000240	NOP	
1965	010762	000240	NOP	
1966	010764	016037	MOV	DSPCH(RO),SLPADR ; LOAD LOOP ADDRESS
1967	010772	000170	JMP	DSPCH(RO) ; GO TO THAT TEST
1968				
1969				
1970				
1971	010776	002044		
1972	011000	002470		
1973	011002	002510		
1974	011004	002564		
1975	011006	002664		
1976	011010	002766		
1977	011012	003430		
1978	011014	004000		
1979	011016	004156		
1980	011020	004304		
1981	011022	004342		
1982	011024	005102		
1983	011026	005252		
1984	011030	005364		
1985	011032	005534		
1986	011034	005570		
1987	011036	005660		
1988	011040	006012		
1989	011042	006146		
1990	011044	006270		
1991	011046	006346		
1992	011050	006460		
1993	011052	006514		
1994	011054	007110		
1995	011056	007404		

;SUBTEST DISPATCH TABLE

DSPCH: TST1+2
TST2+2
TST3+2
TST4+2
TST5+2
TST6+2
TST7+2
TST10+2
TST11+2
TST12+2
TST13+2
TST14+2
TST15+2
TST16+2
TST17+2
TST20+2
TST21+2
TST22+2
TST23+2
TST24+2
TST25+2
TST26+2
TST27+2
TST30+2
TST31+2


```

2052 011300 001413      BEQ      2$      ;BR IF NOT
2053 011302 006237 011364  ASR      10$     ;CHANGE DELAY TIMEP
2054 011306 006237 011364  ASR      10$
2055 011313 006237 011364  ASR      10$
2056 011316 000240      NOP
2057 011320 000240      NOP
2058 011322 000240      NOP
2059 011324 000240      NOP
2060 011326 000240      NOP
2061 011330 032777 010000 167602 2$:  BIT      #BIT12,JSWR  ;TEST SR
2062 011336 001006      BNE      3$      ;BR IF SET
2063 011340 005337 011366  DEC      11$     ;DELAY
2064 011344 001371      BNE      2$
2065 011346 005337 011364  DEC      10$
2066 011352 100366      BPL      2$      ;DELAY
2067 011354 000240      NOP
2068 011356 000240      NOP
2069 011360 000240      NOP
2070 011362 000207      RTS      PC      ;EXIT
2071
2072 011364 000002      10$:  2
2073 011366 000000      11$:  0
2074
2075 011370 013737 001256 011436 ADELAY: MOV      SUBTST,10$
2076 011376 005037 011440      CLR      11$
2077 011402 006237 011436      ASR      10$
2078 011406 006237 011436      ASR      10$
2079 011412 005337 011440      DEC      11$
2080 011416 001375      BNE      2$
2081 011420 005337 011436      DEC      10$
2082 011424 100372      BPL      2$
2083 011426 000240      NOP
2084 011430 000240      NOP
2085 011432 000240      NOP
2086 011434 000207      RTS      PC
2087 011436 000000      10$:  0
2088 011440 000000      11$:  0
2089
2090
2091      ;HEADER SUBROUTINE FOR VT-50
2092
2093 011442 012537 011452  AMSG:  MOV      (R5)+,10$  ;GET POINTER
2094 011446 004537 013130  1$:   JSR      R5,MT0B  ;MOVE TO BUFFER
2095 011452 000000      10$:  0
2096 011454 004737 010236  11$:  JSR      PC,XPRNT  ;DISPLAY IT
2097 011460 000205      RTS      R5      ;EXIT
2098
2099
2100      ;OCTAL - 3 BIT CONVERSION
2101
2102 011462 010001 177770  OCTAL:  MOV      R0,R1  ;LOAD R1
2103 011464 042701 000060  BIC      #177770,R1  ;MASK
2104 011470 062701 011556  ADD      #60,R1
2105 011474 110137  MOV      R1,01G2  ;SAVE LSD
2106 011500 010001  MOV      R0,R1
2107 011502 006001  ROR      R1

```


MAINDEC-11-DZVTC-E MACY11 30(1046) 31-AUG-77 09:17 PAGE 15-15
 DZVTC P11 :5-JUN-77 10:41 CONVERT BINARY TO DECIMAL AND TYPE ROUTINE

SEG 0053

013112	013124	DEC	TIME1	:FINISHED *
013116		BPL	IS	:LOOP TILL TIME EXPIRED
013120		ERROR	PC	:NO INPUT FLAG FROM DEVICE
013122		RTS		:EXIT
013124				
013126		TIME1:	0	
		TIME2:	0	
		:MOVE TO THE OUTPUT BUFFER		
013130	012500	MT0B:	MOV (RS)+,RO	:LOAD DEST.
013132	012701		MOV BUFFER,R1	:LOAD R1
013136	113021	IS:	MOVB (RO)+,(R1)+	:LOAD BYTE
013140	100376		BPL IS	:BR UNTIL DONE
013142	000205		RTS	:EXIT

