

VT20/DL11

PDP-11/VT20 HOST DIAGNOSTIC
MD-11-DZVTE-B

EP-DZVTE-B-DL-A

OCT 1976

COPYRIGHT ©1976

digital

FICHE 1 OF 1

Made in U.S.A.

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

.REM %

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZVTE-B-D
PRODUCT NAME: VT20 HOST COMPUTER PROGRAM
DATE CREATED: OCTOBER 17, 1974
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: EARL L. BOUSE

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS MANUAL.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITAL'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

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

VT20 HOST DIAGNOSTIC PROGRAM P11

CO1

POP-11 VT20 HOST DIAGNOSTIC PROGRAM
DEVTEB.P11

MACY11 27(732) 08-SEP-76 09:00 PAGE 2

57

COPYRIGHT (C) 1974, BY DIGITAL EQUIPMENT CORPORATION.

09
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
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

1. ABSTRACT

THIS PROGRAM IS A COMBINATION DL11 DIAGNOSTIC AND DATA HANDLING ROUTINE. IT IS USED IN CONJUNCTION WITH MAINDEC-11-DBVTA (VT20 SYSTEM DIAGNOSTIC), TEST 21. THE PROGRAM IS COMPLETELY OPERATOR INTERACTIVE BY MEANS OF A CONSOLE DEVICE. THIS DEVICE IS ALWAYS ACTIVE AND WILL RESPOND TO ANY OPERATOR INPUTS. BY MEANS OF THE CONSOLE DEVICE, SPECIFIED LINES CAN BE HELD FROM TRANSMITTING. RECEIVER BUFFERS CAN BE EXAMINED, DATA CAN BE ORIGINATED AND TRANSMITTED, SYSTEM STATUS MONITORED AND ACTIVE DL11'S LISTED. THE PROGRAM ALSO HAS A PROVISION TO BOOT (TRANSFER) PROGRAMS I.E. VT20 DIAGNOSTIC, FROM THE HOST READER TO ANY SPECIFIED VT20 SYSTEM.

THE PROGRAM HAS BEEN WRITTEN TO EXERCISE UP TO 16 DL11 LINES SIMULTANEOUSLY. HOWEVER, BECAUSE OF SOFTWARE OVERHEAD LIMITATIONS, ONLY 10 LINES CAN BE RUN SUCCESSFULLY AT 9600 BAUD. ANY INCREASE OVER 10 LINES MAY RESULT IN OVERRUN ERRORS. THEREFORE, VT20'S SHOULD BE TESTED IN GROUPS OF FIVE (TEN LINES).

ALL LINE NUMBERS ARE TO BE ENTERED AS DECIMAL VALUES. THE PROGRAM WILL RESPOND TO ALL ILLEGAL INPUTS BY TYPING '"'. THIS INDICATES THAT THE PREVIOUSLY INPUTTED CHARACTER WAS IGNORED. USE (CR) TO TERMINATE ALL INPUTS. RUBOUT MAY BE USED TO DELETE PREVIOUSLY INPUTTED CHARACTERS WHEN ENTERING ADDRESSES AND LINE NUMBERS. THE RUBOUT WILL HAVE NO EFFECT IN THE SEND MODE SINCE THE PREVIOUSLY INPUTTED CHARACTER WILL HAVE ALREADY BEEN TRANSMITTED.

2. REQUIREMENTS (EQUIPMENT & MEMORY)

A. ANY PDP-11 FAMILY COMPUTER WITH A CONSOLE DEVICE AND AT LEAST 4K OF MEMORY. THIS IS THE MINIMUM CONFIGURATION TO SUPPORT TESTING OF TWO VT20 SYSTEMS. THEREAFTER, AT LEAST 1K OF ADDITIONAL MEMORY IS REQUIRED FOR EACH VT20 SYSTEM TO BE TESTED.

3. LOADING PROCEDURE

A. USE STANDARD PROCEDURE FOR LOADING BINARY TAPES.

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
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183

4. STARTING PROCEDURE

LOAD AND START PROGRAM AT LOCATION 200. THE PROGRAM WILL RESPOND BY TYPING THE PROGRAM HEADER AND THEN ASK FOR THE FIRST DL11 LINE ADDRESS.

1. RESPOND WITH (CR) IF A DEFAULT ADDRESS OF 175610 IS TO BE USED.
2. OTHERWISE, RESPOND WITH THE ADDRESS OF THE FIRST DL11 LINE ADDRESS TO BE USED.

5. PROGRAM ACTION

AFTER RECEIVING THE DL11 LINE ADDRESS, THE PROGRAM TESTS ALL AVAILABLE DL11 RECEIVER ADDRESSES. THESE AND THEIR CORRESPONDING TRANSMITTER ADDRESSES ARE THEN LOADED INTO AN ACTIVE DEVICE BUFFER AREA. ANY ERRORS ENCOUNTERED WHILE MAPPING THESE DEVICES WILL RESULT IN AN ERROR PRINTOUT. THESE ARE CONSIDERED 'FATAL ERRORS' AND MUST BE CORRECTED BEFORE THE PROGRAM WILL CONTINUE. AFTER THE DL11 ADDRESSES HAVE BEEN SET UP, THE PROGRAM PRINTS A DOT AND ENTERS A MONITOR MODE.

IN THE MONITOR MODE, THE PROGRAM IS READY TO AUTOMATICALLY RECEIVE AND TRANSMIT DATA RECEIVED FROM ANY INITIALIZED VT20 SYSTEM. ALSO IN THE MONITOR MODE, THE PROGRAM RUNS BACKGROUND JOBS OF PRINTING ERRORS, KEEPING ACCOUNT OF SYSTEM STATUS AND EXECUTING OPERATOR REQUESTS.

6. MONITOR COMMANDS

THERE ARE SEVERAL MONITOR COMMANDS WHICH ENABLE THE OPERATOR TO CONTROL AND COMMAND THE PROGRAM. THESE COMMANDS ARE ESSENTIALLY DIAGNOSTIC TOOLS WHICH CAN BE USED TO DEBUG RECEIVER/TRANSMITTER LINES AND TO MONITOR SYSTEM STATUS. THERE ARE TWO TYPES OF COMMANDS: MONITOR RESPONSE COMMANDS I.E. ↑A, ↑C, ↑D, ↑E & ↑L, AND LINE NO. DEPENDENT COMMANDS, I.E. ↑B, ↑H, ↑M, ↑R & ↑S. ON RECEIPT OF A MONITOR RESPONSE COMMAND, THE PROGRAM WILL IMMEDIATELY EXECUTE THE COMMAND. ON RECEIPT OF A LINE DEPENDENT COMMAND THE PROGRAM WILL WAIT FOR A LINE NUMBER AND A CARRIAGE RETURN (CR) BEFORE EXECUTING THE COMMAND. THE FORM OF THIS TYPE OF COMMAND IS "COMMAND & LINE NO. (CR)". IN SOME CASES, THE COMMAND WILL ALLOW FOR MULTIPLE LINE NUMBERS TO BE ACCEPTED. IN THESE CASES THE LINE NUMBERS ARE TO BE SEPERATED BY COMMAS AND TERMINATED BY (CR).

184
185
186
187
188
189
190
191
192
193
194
195
196
197
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

AN EXAMPLE WOULD BE: "↑HO 4,6,7(CR)". ON RECEIPT OF THIS COMMAND, LINES 0,4,6 & 7 WOULD BE HELD FROM TRANSMITTING (REFER TO THE ↑H COMMAND).

THE OMISSION OF A LINE NUMBER I.E. (CR) ONLY, WILL RESULT IN LINE '0' BEING SERVICED. IN ALL CASES, THE LINE NUMBERS MAY BE INPUTTED IN ANY ORDER. ALL CONTROL CHARACTERS I.E. ↑A,↑B ETC. ARE OBTAINED BY TYPING THE 'CNTRL & CHARACTER SPECIFIED' KEYS SIMULTANEOUSLY.

A. ↑A (ABSOLUTE SYSTEM RESTART)

ONE RECEIPT OF THIS COMMAND, THE PROGRAM WILL BE RESTARTED. THIS WILL ENABLE FOR A NEW SET OF DL11 DEVICES ADDRESSES TO BE ENTERED.

B. ↑B (BOOT)

THE PURPOSE OF THIS COMMAND IS TO SELECT A BOOT ROUTINE TO TRANSFER PROGRAMS FROM THE READER OF THE HOST COMPUTER TO SELECTED VT20 SYSTEMS. UPON RECEIPT OF THE (↑B), THE PROGRAM WILL WAIT FOR THE LINE OR LINES NUMBERS OVER WHICH THE PROGRAM IS TO BE TRANSFERED. AFTER RECEIVING THE LINE NUMBER(S), THE PROGRAM WILL REQUEST THE READER DEVICE & VECTOR ADDRESSES. THIS QUESTION WILL ONLY BE ASKED ON THE INITIAL USE OF THE BOOT ROUTINE, BUT THESE ADDRESSES CAN BE CHANGED BY TYPING A (↑E). THEN ON THE NEXT OCCURANCE OF THE (↑B) COMMAND, A NEW READER DEVICE ADDRESS WILL BE REQUESTED.

IN ORDER FOR PROGRAMS TO BE BOOTED TO A SELECTED VT20 SYSTEM, THE BOOTSTRAP LOADER IN THE PDP-11/05 OF THE VT20 MUST BE MODIFIED. THIS WILL ENABLE THE PDP-11 TO ACCEPT THE BOOTED PROGRAM. TO DO THIS, SIMPLY REPLACE THE PDP-11 OR TTY 'CSR' ADDRESS IN LOCATION '37776' WITH A DL11 'CSR' ADDRESS OF EITHER '175610 OR 175620'. IF THE ABSOLUTE LOADER ISN'T RESIDENT, THIS WILL HAVE TO BE THE FIRST PROGRAM BOOTED OVER. THIS WOULD THEN BE FOLLOWED BY THE VT20 DIAGNOSTIC.

THE BOOT ROUTINE IS CAPABLE OF BOOTING UP TO '16' DL LINES SIMULTANEOUSLY.

C. ↑C (CLEAR SOFTWARE SWITCHES)

THIS COMMAND CAN BE EXECUTED AT ANYTIME TO TERMINATE OPERATOR REQUESTS I.E. SEND, HOLD, BOOT MODES ETC., AND RESET THE SYSTEM STATUS TO A KNOWN STATE. THE (↑C) WILL NOT EFFECT THE STATE OF ACTIVE RECEIVERS AND TRANSMITTERS. HOWEVER, ALL LINES THAT WERE BEING HELD, WILL BE RELEASED AND RE-ACTIVATED.

236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
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

D. ↑D (DIAGNOSTIC MODE)

THIS COMMAND ENABLES ALL DATA BEING RECEIVED BY THE HOST PROGRAM, REGARDLESS OF LINE NUMBER, TO BE PRINTED. THIS IS A DOUBLE FUNCTION COMMAND WHICH ON THE FIRST RECEIPT OF (↑D) WILL ENABLE THE DIAGNOSTIC MODE. THIS WILL RESULT IN THE MESSAGE "DIAGNOSTIC MODE ENABLE" TO BE TYPED. ON THE SECOND RECEIPT OF A '↑D', THE DIAGNOSTIC MODE WILL BE DISABLED.

THE USE OF THIS COMMAND SHOULD BE RESTRICTED TO RUNNING ONE LINE AND THEN ONLY IF THERE ARE NO ERRORS BEING REPORTED BY THAT LINE. IT IS RECOMMENDED THAT THE (↑P) FEATURE BE USED IN LIEU OF THE DIAGNOSTIC MODE IF MULTIPLE LINES ARE BEING EXERCISED.

E. ↑E (ESCAPE RESTART)

THIS COMMAND IS TO BE USED IF MULTIPLE RECEIVER AND/OR TRANSMITTER ERRORS ARE BEING REPORTED AND THE USE OF (↑C) HAS NO APARENT EFFECT. ON RECEIPT OF A (↑E), ALL SYSTEM SOFTWARE AND HARDWARE FLAGS ARE RESET. THIS WILL RESULT IN TERMINATING THE OPERATION OF ANY LINES WHICH WERE ACTIVE UPON THE RECEIPT OF THE (↑E). HENCEFORTH, ALL VT20 SYSTEMS WILL HAVE TO BE REINITIALIZED. THIS COMMAND WILL ALSO RESET THE MONITOR TRANSFER AND ERRORS COUNTERS. THESE ARE THE COUNTS DISPLAYED WHEN LISTING MONITOR STATUS (REFER TO ↑M).

F. ↑H (HOLD MODE)

THIS COMMAND WILL ENABLE FOR A SELECTED LINES' BUFFER TO BE HELD FROM BEING TRANSMITTED. THE COMMAND MAY BE USED SOLEY TO HOLD LINE TRANSMISSION OR USED IN CONJUNCTION WITH THE (↑S) FEATURE. THIS WILL ENABLE A USER TO CREATE A BUFFER WHICH CAN BE RELEASED, ON COMMAND, IN A BURST AT TRANSMITTER BAUD RATE. REFER TO THE MONITOR COMMAND (↑S) FOR INSTRUCTIONS ON CREATING THIS BUFFER. THIS COMMAND WILL FACILITATE HOLDING UP TO '16' DL LINES SIMULTANEOUSLY IN ANY ORDER. THE LINES CAN THEN BE RELEASED (REFER TO ↑R) INDIVIDUALLY OR SIMULTANEOUSLY USING (↑C).

G. ↑L (LIST)

THIS COMMAND IS USED TO LIST THE ACTIVE DL11 RECEIVER ADDRESSES THAT WERE MAPPED BY THE PROGRAM.

H. ↑M (MONITOR)

THIS COMMAND WILL ENABLE FOR THE CURRENT SYSTEM STATUS TO BE MONITORED. THE (↑M) OPTION CAN BE USED IN ONE OF TWO WAYS: (1) TYPE "↑M(CR)" TO PRINT THE STATUS OF ALL DL11 LINES. (2) TYPE "↑M & LINE NO., LINE NO....(CR)" TO PRINT THE STATUS OF SPECIFIED LINE(S). THE FOLLOWING IS AN EXAMPLE AND EXPLANATION OF THE MONITOR PRINTOUT. REFER TO SECTION

IO1

FDP-11/VT20 HOST DIAGNOSTIC PROGRAM
DZVTEB.P11

MACY11 27(732) 08-SEP-76 09:00 PAGE 8

292

7. (ERRORS) FOR A FURTHER EXPLANATION OF THE ERROR DATA.

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
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

EXAMPLE:

LINE	IN	OUT	OR	PAR.	FRAM	REC	TRAN	ST.	HELD
A	B	C	D	E	F	G	H	I	J

A= NO. OF LINE BEING MONITORED
B= NO. OF BLOCKS OF DATA RECEIVED
C= NO. OF BLOCKS OF DATA TRANSMITTED
D= NO. OF OVERRUN ERRORS INCURRED
E= NO. OF PARITY ERRORS INCURRED
F= NO. OF FRAMING ERRORS INCURRED
G= NO. ILLEGAL RECEIVER INTERRUPTS INCURRED.
H= NO. OF ILLEGAL TRANSMITTER INTERRUPTS INCURRED.
I= NO. OF ILLEGAL START CODES INCURRED.
J= TO '1' IF LINE IS CURRENTLY BEING HELD.

I. ↑P (PRINT)

THIS COMMAND IS USED TO PRINT THE CONTENTS OF A SELECTED LINES' BUFFER. IT SHOULD BE NOTED THAT THE START CODE (377) IS DETECTED AND PRINTED AS AN UP-ARROW (↑). UPON COMPLETION OF PRINTING A BUFFER OR IF THE BUFFER IS EMPTY, A DOT WILL BE PRINTED INDICATING A RETURN TO THE MONITOR.

J. ↑R (RELEASE)

THIS COMMAND IS USED TO RELEASE LINES THAT WERE HELD USING THE (↑H) FEATURE. LINES MAY BE RELEASED IN ANY ORDER AND EITHER INDIVIDUALLY OR COLLECTIVELY. THE (↑M) COMMAND CAN BE USED TO DETERMINE IF A SELECTED LINE IS BEING HELD. ANY ATTEMPT TO RELEASE AN ACTIVE LINE WHICH WASN'T BEING HELD, MAY RESULT IN AN 'ILLEGAL TRANSMITTER INTERRUPT' ERROR. THE (↑R) COMMAND WILL FACILITATE RELEASING UP TO 16 LINES SIMULTANEOUSLY.

K. ↑S (SEND)

THIS COMMAND IS USED TO SEND DATA, ORGINATED ON THE HOST CONSOLE DEVICE, TO SELECTED TRANSMITTER LINE(S). IN THIS MODE, AS EACH CHARACTER IS RECEIVED IT IS SIMULTANEOUSLY TRANSFERED. AN ALTERNATE TO THIS SINGLE CHARACTER TRANSMISSION IS TO BUFFER THE DATA AND THEN RELEASE IT IN A BURST AT THE TRANSMITTER BAUD RATE. THIS IS DONE USING THE (↑H) FEATURE TO HOLD SELECTED LINE(S) AND THEN ENTERING THE (↑S) MODE. ALL DATA THEN WILL BE BUFFERED INSTEAD OF BEING TRANSMITTED. WHEN THE DESIRED BUFFER HAS BEEN CREATED, EXIT THE (↑S) MODE BY TYPING (ALT). THE PROGRAM WILL RESPOND BY A DOT, INDICATING A RETURN TO THE MONITOR. THE DATA BUFFER(S) CAN THEN BE PRINTED (REFER TO ↑P) AND/OR RELEASED (REFER TO ↑R).

WHEN USING THE (↑S) FEATURE, SW00 MUST BE SET TO A '1' (UP) ON THE VT20 SYSTEM RECEIVING THE DATA. OTHERWISE, THE DATA

K01

PDP-11/VT20 HOST DIAGNOSTIC PROGRAM
DZVTEB.P11

MACY11 27(732) 08-SEP-76 09:00 PAGE 10

349
350

WILL LOOK AND BE HANDLED AS DATA, OR RECEIVER ERRORS. THIS
SWITCH SHOULD BE RESET (DOWN) WHEN NOT USING THE SEND MODE.

351
352
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. ERROR REPORTER

THE PROGRAM HANDLES ERRORS IN TWO PHASES (1) FATAL ERRORS (INCURRED WHILE MAPPING THE ACTIVE DL11 DEVICES) AND (2) NON-FATAL SYSTEM ERRORS (INCURRED WHILE RECEIVING AND TRANSFERRING DATA). FATAL ERRORS MUST BE CORRECTED BEFORE THE PROGRAM WILL CONTINUE.

A. FATAL ERRORS

THESE ERRORS CAN RESULT FROM THE USER ENTERING ILLEGAL DEVICE ADDRESSES OR SIMPLY BY HAVING BAD DL11'S PRESENT. FATAL ERRORS OF THIS NATURE WILL RESULT IN ONE OF TWO TYPEOUTS:

1. "NO DL11 ADDRESSES PRESENT"

THIS PRINTOUT WILL RESULT IF THE ADDRESS ENTERED BY THE USER DIDN'T RETURN A 'SLAVE SYNC' WHEN ADDRESSED.

2. "NO RESPONSE FROM DEVICE NNNNNN"

THIS PRINTOUT WILL RESULT AFTER MAPPING IF A DL TRANSMITTER FAILS TO INTERRUPT WHEN ENABLED.

B. SYSTEM ERRORS

SYSTEM ERRORS ARE HANDLED AS BACKGROUND JOBS. WHEN AN ERROR IS INCURRED IT IS CATEGORIZED BY GIVING IT A UNIQUE NUMBER AND PUSHED INTO AN ERROR BUFFER.

THESE ERRORS ARE THEN PRINTED (IN THE ORDER OF INCURRENCE) BY THE MONITOR. A MAXIMUM OF SIX(6) ERRORS PER/LINE, REGARDLESS OF TYPE, ARE SAVED IN THE ERROR BUFFER AND PRINTED. THE SYSTEM DOES, HOWEVER, KEEP A RUNNING COUNT OF ALL ERRORS. THIS INFORMATION CAN BE ACCESSED BY USE OF THE (↑M) COMMAND. THESE ERROR COUNTERS ARE RESET ONLY ON SYSTEM START UP AND BY THE (↑E) & (↑A) COMMANDS. EACH ERROR PRINTOUT WILL CONSIST OF THE LINE NUMBER AND A MESSAGE DESCRIBING THE ERROR TYPE.

FOLLOWING IS A LIST AND DESCRIPTION OF THE POSSIBLE ERROR MESSAGES THAT MAY OCCUR:

1. "ILLEGAL RECVR. INTERRUPT"

THIS ERROR WILL RESULT IF THE LINE TRANSMITTER IS ACTIVE OR BEING HELD AND A RECEIVER INTERRUPT IS SERVICED.

2. "ILLEGAL TRANS. INTERRUPT"

THIS ERROR WILL RESULT IS THE LINE RECEIVER IS ACTIVE AND A TRANSMITTER INTERRUPT IS SERVICED.

MO1

PDP-11/VT20 HOST DIAGNOSTIC PROGRAM
DZVTEB.P11

MACY11 27(732) 08-SEP-76 09:00 PAGE 12

407

3. "OVERRUN ERROR"

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
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462

4. "FRAMING ERROR"

5. "PARITY ERROR"

6. "ILLEGAL START CODE"

THIS ERROR IS A RESULT OF THE FIRST CHARACTER RECEIVED, OTHER THAN A NULL CODE, NOT BEING '377'. (REFER TO DATA FORMAT SECTION FOR A FURTHER EXPLANATION.)

7. "ILLEGAL READER INTERRUPT"

8. DATA FORMAT

ALL DATA RECEIVED FROM THE VT20 SYSTEMS IS ESPECIALLY FORMATTED. THIS FORMAT IS CHECKED AND TRANSMITTED EXACTLY AS IT WAS RECEIVED. THE FORMAT OF THIS DATA IS: 4 NULL CHARACTERS (000), A START CODE(377), DATA (UP TO 384 CHARACTERS ORIGINATED BY USER), AND A EOP (END OF PARAGRAPH CODE-14).

NOTE - THE DATA ORIGINATED FROM A VT20 MAY BE IN ONE OF THREE FORMATS:

- A. RANDOM, GENERATED FROM THE KEYBOARD
- B. INCREMENTAL, GENERATED BY ↑A
- C. WORST CASE, GENERATED BY ↑W

REFER TO MAINDEC-11-DBVTA WRITEUP FOR DETAILS.

WHEN DATA IS RECEIVED, THE PROGRAM VERIFIES THAT THE FIRST CHARACTER, OTHER THAN NULLS), IS A START CODE. ON RECEIPT OF THE START CODE, THE RECEIVER SERVICE ROUTINE IS INITIALIZED. ALL DATA FROM THIS POINT, UNTIL THE RECEIPT OF AN EOP CODE, IS STORED IN THE APPROPRIATE LINES' BUFFER. ON RECEIPT OF THE EOP, THE RECEIVER SERVICE ROUTINE IS TERMINATED. THE TRANSMITTER SERVICE ROUTINE IS THEN INITIALIZED. THUS, NO DL11 TRANSMITTER AND RECEIVER ARE ACTIVE SIMULTANEOUSLY. THE RECEIVED DATA IS THEN TRANSMITTED, EXACTLY AS IT WAS RECEIVED.

IT SHOULD BE NOTED THAT WHEN A LINES' BUFFER IS PRINTED, THE START CODE CHARACTER IS DETECTED AND PRINTED AS AN UP-ARROW (↑). THIS ALSO HOLDS TRUE ON THE VT20 END, WHERE THE START CODE IS DISPLAYED AS AN (↑) ON THE SCREEN.

9. DISPLAY REGISTER OPTION

IF THIS PROGRAM IS BEING RUN ON A PDP-11/45 THE ACTIVITY OF THE DL11 LINES MAY BE VISUALLY MONITORED. THIS IS DONE BY SETTING THE DATA DISPLAY SELECT SWITCH, ON THE /45 CONSOLE, TO THE "DISPLAY REGISTER" POSITION. THEN EVERY TIME A DL11 TRANSMITTER BECOMES ACTIVE, IT'S CORRESPONDING LINE NUMBER WILL BE REFLECTED BY LIGHTING A LIGHT IN THE DATA LIGHT REGISTER. IT CAN BE NOTED, THAT IF A SELECTED LINE IS HELD, THE LIGHT REFLECTING THAT LINE WILL BE LIT IF THAT LINE IS READY TO BE RELEASED.

10. LISTING

%.
:TITLE PDP-11/VT20 HOST DIAGNOSTIC PROGRAM
:ABS
:ENABLE AMA
:MAINDEC-11-DZVTE-8-D
:COPYRIGHT OCTOBER 17, 1974
:DIGITAL EQUIPMENT CORP. MAYNARD, MASS. 01754
:PROGRAMMER: EARL L. BOUSE

;REGISTER DEFINITIONS

000000
000001
000002
000003
000004
000005
000006
000007

R0=%0
R1=%1
R2=%2
R3=%3
R4=%4
R5=%5
SP=%6
PC=%7

;INSTRUCTION DEFINITION

005726
022626
024646
000240
001100
000014

POP1SP=05726
POP2SP=22626
PUSH2SP=24646
NOP=240
STACK=1100
EOP=14

;LOAD TRAP CATCHER INTO LOC 0-1000

000000

.REPT =0
200
4
.ENDR

;LOAD TRAP WITH 'IOT' TRAP VECTOR

Vertical column of characters on the left margin, possibly a scan artifact or a specific control sequence.

Vertical characters at the bottom left, possibly a signature or page marker.

000019	000020	000020	. =20	
000020	000022	000340	MAPVEC	:MAPPER TRAP ROUTINE
000024	000336	340	PWRFAL	:POWER FAIL HANDLER
000026	000340	340	EMTSRV	:EMT SERVICE ROUTINE
000030	006442	340	. =60	
000032	000340	KEYSRV	KEYSRV	:TTY KEYBOARD SERVICE ROUTINE
000060	002500	200	. =200	
000062	000200	JMP	START	:PROGRAM STARTING ADDRESS
000200	000137	001314		

:REGISTER ADDRESSES

PSW: 1200
TKS: 177776
TKB: 177560
TPS: 177562
TPB: 177564
SWR: 177566
177570

:ADDRESS OF PROCESSOR STATUS REG.
:ADDRESS OF KEYBOARD STATUS REG.
: " " BUFFER
: " " PRINTER STATUS REG.
: " " PRINTER BUFFER REG.
: " " SWITCH REG.

:TRAP EQUIVALENCE TABLE:

PRINT=EMT
PRTOCT=PRINT+1
BINDEC=PRTOCT+1

:SUBROUTINE TO PRINT ASCII MESSAGES.
:SUBROUTINE TO PRINT A 6 DIGIT OCTAL NO
:SUBROUTINE TO CONVERT OCTAL TO BINARY & PRINT IT

:*****
:DL11 DEVICE ADDRESS LIST
:TABLE IS TERMINATED WITH '-1' AFTER MAPPING '16' DEVICES
:*****

RXCR0: 175610
RXCR1: 175620
RXCR2: 175630
RXCR3: 175640
RXCR4: 175650
RXCR5: 175660
RXCR6: 175670
RXCR7: 175700
RXCR10: 175710
RXCR11: 175720
RXCR12: 175730
RXCR13: 175740
RXCR14: 175750
RXCR15: 175760
RXCR16: 175770
RXCR17: 176000
RXCR20: 176010
RXCR21: 176020
RXCR22: 176030
RXCR23: 176040
RXCR24: 176050
RXCR25: 176060
RXCR26: 176070
RXCR27: 176100
RXCR30: 176110
RXCR31: 176120
RXCR32: 176130
RXCR33: 176140
RXCR34: 176150
RXCR35: 176160
RXCR36: 176170
RXEND: 177777

:LINE 0 DEVICE ADDRESS (RXCSR)
:LINE 1 DEVICE ADDRESS (RXCSR)
:LINE 2 DEVICE ADDRESS (RXCSR)
:LINE 3 DEVICE ADDRESS (RXCSR)
:LINE 4 DEVICE ADDRESS (RXCSR)
:LINE 5 DEVICE ADDRESS (RXCSR)
:LINE 6 DEVICE ADDRESS (RXCSR)
:LINE 7 DEVICE ADDRESS (RXCSR)
:LINE 10 DEVICE ADDRESS (RXCSR)
:LINE 11 DEVICE ADDRESS (RXCSR)
:LINE 12 DEVICE ADDRESS (RXCSR)
:LINE 13 DEVICE ADDRESS (RXCSR)
:LINE 14 DEVICE ADDRESS (RXCSR)
:LINE 15 DEVICE ADDRESS (RXCSR)
:LINE 16 DEVICE ADDRESS (RXCSR)
:LINE 17 DEVICE ADDRESS (RXCSR)
:LINE 20 DEVICE ADDRESS (RXCSR)
:LINE 21 DEVICE ADDRESS (RXCSR)
:LINE 22 DEVICE ADDRESS (RXCSR)
:LINE 23 DEVICE ADDRESS (RXCSR)
:LINE 24 DEVICE ADDRESS (RXCSR)
:LINE 25 DEVICE ADDRESS (RXCSR)
:LINE 26 DEVICE ADDRESS (RXCSR)
:LINE 27 DEVICE ADDRESS (RXCSR)
:LINE 30 DEVICE ADDRESS (RXCSR)
:LINE 31 DEVICE ADDRESS (RXCSR)
:LINE 32 DEVICE ADDRESS (RXCSR)
:LINE 33 DEVICE ADDRESS (RXCSR)
:LINE 34 DEVICE ADDRESS (RXCSR)
:LINE 35 DEVICE ADDRESS (RXCSR)
:LINE 36 DEVICE ADDRESS (RXCSR)
:LINE XX DEVICE ADDRESS (RXCSR)

001200
001208
001202
001204
001206
001210
001212

104000
104001
104002

081214
081216
081220
081222
081224
081226
081228
081230
081232
081234
081236
081240
081242
081244
081246
081250
081252
081254
081256
081260
081262
081264
081266
081270
081272
081274
081276
081300
081302
081304
081306
081310
081312

```

593                                     :*****
594                                     :SET UP AN 'ACTIVE' DL11 ADDRESS BUFFER FOR ALL EXISTING DL11'S
595                                     :*****
596
597 001314 012777 000340 177656 START: MOV #340, @PSW ;SET PROCESSOR PRIORITY '07:
598 001322 012706 001100 MOV #STACK, SP ;INITIALIZE STACK POINTER
599 001326 005077 177650 CLR @TKS ;CLR KEYBOARD INTERRUPT ENABLE
600 001332 004737 010712 JSR PC, OVRLAY ;OVERLAY TRAP AREA.
601 001336 012737 001354 000004 MOV #CORSIZ, @#4 ;RESET TIMEOUT
602 001344 012701 020000 MOV #20000, R1 ;TEST CORE SIZE
603 001350 005721 TST (R1)+
604 001352 000776 BR -2
605 001354 162701 001000 CORSIZ: SUB #1000, R1 ;SAVE THIS ADDRESS AS ERROR BUFFER LIMIT
606 001360 010137 012144 MOV R1, MEMSIZ
607 001364 012737 000006 000004 MOV #6, @#4
608 001372 012737 000004 000006 MOV #4, @#6 ;RESET '6' TO TRAP
609 001400 012700 012206 MOV #RCSRO, R0 ;SETUP TO CLR BUFFER & STORAGE AREA
610 001404 005020 CLR (R0)+
611 001406 023700 012144 CMP MEMSIZ, R0 ;DONE?
612 001412 001374 BNE -6 ;NO
613 001414 005737 012142 TST MONFLG ;HAS THE HEADER BEEN TYPED?
614 001420 001004 BNE START1 ;YES, SKIP RE-TYPING IT
615 001422 005237 012142 INC MONFLG ;NO, SET FLAG
616 001426 104000 011111 PRINT, TITLE ;TEXT 'DL11 DATA HANDLING ROUTINE'
617 001432 104000 011403 START1: PRINT, MF1AD ;ASK FOR FIRST DL11 ADDRESS
618 001436 004737 004050 JSR PC, GETLN1 ;GET & DECODE THE LINE ADDRESS
619 001442 005737 013206 TST DEVADR ;WAS AN ADDRESS INPUTTED?
620 001446 001003 BNE START2 ;YES, SET IT UP.
621 001450 012737 175610 013206 MOV #175610, DEVADR ;NO, USE DEFAULT 175610 AS FIRST
622 001456 013701 013206 START2: MOV DEVADR, R1 ;GET READY TO ASSEMBLE
623 001462 012704 001214 MOV #RXCRO, R4 ;SET UP POINTER TO SAVE ACTIVE DL'S
624 001466 005037 012626 CLR HERE
625 001472 012737 001522 000004 MOV #MAPNE, @#4 ;SET UP TIME-OUT ADDRESS
626 001500 005037 000006 START3: CLR @#6
627 001504 005711 TST (R1) ;TEST IF ADDRESS IS PRESENT
628 001506 000240 NOP
629 001510 000240 NOP
630 001512 005237 012626 INC HERE ;YES, KEEP TRACK
631 001516 010124 MOV R1, (R4)+ ;SAVE ADDRESS IN TABLE
632 001520 000401 BR +4
633 001522 022626 MAPNE: POP2SP ;RESET STACK
634 001524 062701 000010 ADD #10, R1 ;SET UP NEXT DL ADDRESS
635 001530 022701 176200 CMP #176200, R1 ;TESTED ALL DL ADDRESSES?
636 001534 003404 BLE 3$ ;YES
637 001536 022737 000020 012626 CMP #16, HERE ;MAPPED '16' ACTIVE DL'S?
638 001544 001357 BNE START3 ;NO
639 001546 012714 177777 3$: MOV #-1, (R4) ;YES, TERMINATE TABLE
640 001552 012737 000006 000004 MOV #6, @#4 ;RESET TRAP
641 001560 012737 000004 000006 MOV #4, @#6
642 001566 005737 012626 TST HERE ;ANY DEVICES PRESENT
643 001572 001003 BNE LOADRS ;YES, CONTINUE
644 001574 104000 PRINT ;NO, REPORT ERROR
645 001576 011222 MES1 ;TEXT 'NO DL11 ADDRESSES PRESENT'.
646 001600 000645 BR START

```

```

637
638
639
640
641
642
643 001602 012700 012206
644 001606 012701 001214
645 001612 032711 0000C1
646 001616 001015
647 001620 011120
648 001622 011110
649 001624 052720 000002
650 001630 011110
651 001632 052720 000004
652 001636 012110
653 001640 052720 000006
654 001644 062700 000010
655 001650 000760
656
657
658
659
660
661
662 001652 012700 012212
663 001656 012701 012224
664 001662 013737 012626 013324
665 001670 005737 013324
666 001674 001444
667 001676 005337 013324
668 001702 005077 177272
669 001706 052737 000001 013352
670 001714 042770 000100 000000
671 001722 052770 000100 000000
672 001730 000240
673 001732 000240
674 001734 012777 000340 177236
675 001742 005711
676 001744 001010
677 001746 104000
678 001750 011257
679 001752 011037 013326
680 001756 104001
681 001760 013326
682 001762 000137 001314
683 001766 042770 000100 000000
684 001774 062700 000020
685 002000 062701 000026
686 002004 000731

```

```

;*****
;AT THIS POINT THE 'ACTIVE' DL11 ADDRESSES BEEN FOUND. THIS NEXT
;SUBROUTINE FORMS THE ENTIRE TRANSMITTER AND RECEIVER ADDRESSES FOR THESE
;DL11'S AND SAVES THEM IN AN 'ACTIVE' DL11 BUFFER.
;*****
LOADRS: MOV #RCSR,RO ;SETUP BUFFER ADDRESS POINTER
        MOV #RXCR,RI ;SETUP ACTIVE DEVICE TABLE POINTER
NXADRS: BIT #1,(R1) ;IS DEVICE PRESENT?
        BNE FINVEC ;NO, DONE
        MOV (R1),(RO)+ ;LOAD THE 'RCSR' ADDRESS
        MOV (R1),(RO)
        SIS #2,(RO)+ ;LOAD THE 'REJF' ADDRESS
        MOV (R1),(RO)
        BIS #4,(RO)+ ;LOAD THE 'XCSR' ADDRESS
        MOV (R1),(RO)+
        BIS #6,(RO)+ ;LOAD THE 'XBUF' ADDRESS
        ADD #10,RO ;MOVE POINTER TO NEXT 'RCSR' ADDRESS
        BR NXADRS ;FORM NEXT GROUP

;*****
;NOW AN 'ACTIVE' DEVICE TABLE HAS BEEN SETUP, AN INTERRUPT IS FORCED
;FOR EACH DL11 AND IT'S VECTOR IS MAPPED.
;*****
FINVEC: MOV #XCSR,RO ;SET UP 'XCSR' ADDRESS POINTER
        MOV #XLVLO,RI ;SETUP 'XMITTER' BR LEVEL POINTER
        MOV HERE,TEMP ;SETUP CNTR.
FOUNDV: TST TEMP ;DONE MAPPING?
        BEQ LDVECT ;YES
        DEC TEMP ;NO, DECREMENT CNTR
        CLR @PSW ;SET PROC. PRIORITY 20
        BIS #1,FMAP ;SET MAPPING FLAG
        BIC #100,@(RO)
        BIS #100,@(RO) ;CAUSE TRANSMITTER INTERRUPT
        NOP
        NOP
        MOV #340,@PSW
        TST (R1) ;DID INTERRUPT OCCUR?
        BNE TSTVA ;YES,
        PRINT ;NO, REPORT ERROR
        MES2 ;TEXT 'NO DEVICE PRESENT FOR THIS _LINE_'
        MOV (RO),KSTOR1 ;PRINT 'RCSR' ADDRESS
        PRTCT
        KSTOR1
        JMP START
TSTVA: BIC #100,@(RO) ;CLR INTERRUPT ENABLE
        ADD #20,RO ;SETUP OFFSET TO ADDR. NEXT GROUP
        ADD #26,R1 ;SETUP NEXT BR LEVEL POINTER
        BR FOUNDV ;MAP NEXT VECTOR

```

```

687
688
689
690
691
692
693
694 002006 012700 012216
695 002012 012701 006510
696 002016 012702 007612
697 002022 012703 012206
698 002026 012704 012212
699 002032 012705 006664
700 002036 162705 006620
701 002042 013737 012626 013324
702 002050 005737 013324
703 002054 001436
704 002056 005337 013324
705 002062 010130
706 002064 012730 000240
707 002070 010230
708 002072 012770 000240 000000
709 002100 052773 000100 000000
710 002106 052774 000100 000000
711 002114 062703 000020
712 002120 062704 000020
713 002124 062700 000012
714 002130 060501
715 002132 060502
716 002134 000745
717 002136 012701 012630
718 002142 005021
719 002144 022701 012672
720 002150 001374
721
722
723
724
725
726
727
728
729 002152 012777 000340 177020
730 002160 012701 013354
731 002164 005021
732 002166 023701 012144
733 002172 001374

```

```

;*****
; AT THIS POINT ALL VECTOR ADDRESSES HAVE BEEN MAPPED AND LOADED.
; THE FOLLOWING SUBROUTINE LOADS THE VECTOR ADDRESSES WITH THEIR
; RESPECTIVE SERVICE ROUTINE ADDRESSES AND ENABLES THE DL11 INTERRUPTS.
;*****
LDVECT: MOV      #RVTR0,R0      ;SETUP VECTOR ADDRESS POINTER
        MOV      #RECVD0,R1    ;SETUP RECEIVER SERVICE ROUTINE POINTER
        MOV      #XFER00,R2    ;SETUP TRANSMITTER SERVICE ROUTINE POINTER
        MOV      #RCSR0,R3     ;SETUP 'RCSR' ADDRESS POINTER
        MOV      #XCSR0,R4     ;SETUP 'XCSR' ADDRESS POINTER
        MOV      #RECVD3,R5
        MOV      #RECVD2,R5    ;GET SERVICE ROUTINE OFFSET
        SUB      HERE,TEMP
LDNXTV: TST      TEMP          ;SETUP ALL SERVICE ROUTINES?
        BEQ      SERVICE       ;YES, START SERVICE ROUTINE
        DEC      TEMP
        MOV      R1,@(R0)+      ;LOAD RECEIVER SERVICE ADDRESS.
        MOV      #240,@(R0)+    ;SET BR LEVEL 25
        MOV      R2,@(R0)+      ;LOAD TRANSMITTER SERVICE ADDRESS
        MOV      #240,@(R0)     ;SET BR LEVEL 25
        BIS      #100,@(R3)     ;ENABLE RECEIVER INTERRUPTS
        BIS      #100,@(R4)     ;ENABLE TRANSMITTER INTERRUPTS
        ADD      #20,R3         ;SETUP NEXT 'RCSR' ADDRESS
        ADD      #20,R4         ;SETUP NEXT 'XCSR' ADDRESS
        ADD      #12,R0         ;SETUP NEXT RECEIVER VECTOR ADDR.
        ADD      R5,R1          ;SETUP NEXT RECEIVER SERVICE ADDR.
        ADD      R5,R2          ;SETUP NEXT TRANSMITTER SERVICE ADDR.
        BR       LDNXTV        ;LOAD NEXT VECTOR ADDRESS
        MOV      #INTRAN,R1
25:     CLR      (R1)+
        CMP      #HLDLO,R1
        BNE     25
SERVICE: MOV     #340,@PSW     ;SET PROC. PRIORITY 27
        MOV     #RECSWH,R1     ;SET UP TO CLEAR ALL SOFTWARE SW'S.
15:     CLR     (R1)+
        CMP     MEMSIZ,R1      ;DONE?
        BNE     15             ;NO

```

```

;*****
;PROGRAM ENTERED HERE TO INITIALIZE ALL SOFTWARE SWITCHES (BOTH FOR
;USER OPTIONS I.E. '↑S, ↑P, ETC. AND PROGRAM SWITCHES. THIS IS ENTER-
;ED EITHER FROM THE MONITOR ON PROGRAM LOADS OR BY TYPING A '↑E'.
;*****

```

```

734
735
736
737
738
739
740 002174 012706 001100
741 002200 012701 012672
742 002204 005021
743 002206 022701 013354
744 002212 001374
745 002214 012700 014146
746 002220 010005
747 002222 052777 000100 176752
748 002230 104000 012135
749 002234 005077 176740
750
751
752
753
754
755
756
757 002240 013777 013164 176744
758 002246 022700 014754
759 002252 101403
760 002254 005737 013342
761 002260 001004
762 002262 012705 014146
763 002266 010500
764 002270 000763
765
766
767
768
769
770
771
772 002272 005337 013342
773 002276 100412
774 002300 005737 013336
775 002304 001043
776 002306 112537 013330
777 002312 112537 013326
778 002316 005337 013342
779 002322 100003
780 002324 005037 013342
781 002330 000743
782 002332 104000 011312
783 002336 104002 013326
784 002342 013704 013330
785 002346 042704 177770
786 002352 006304
787 002354 022704 000007
788 002360 001727
789 002362 016437 002376 002372

```

```

;*****
;RESTART ENTRY POINT WHEN 'IC' IS TYPED.
;ENTRY HERE ONLY CLEARS USERS SOFTWARE SWITCHES AND RESETS THE ERROR
;BUFFER POINTERS TO INHIBIT PRINTING FOR PREVIOUSLY BUFFERED ERRORS.
;*****
RESTR: MOV    #STACK,SP      ;RESET STACK POINTER
2$:  MOV    #HLDLO,R1      ;CLEAR 'IC' SOFTWARE SW.'S
3$:  CLR    (R1)+
      CMP    #RECSWH,R1    ;DONE?
      BNE   3$
      MOV    #ERRBUF,R0    ;SETUP ERROR BUFFER
      MOV    R0,R5
      BIS   #100,2TKS      ;ENABLE KEYBOARD INTERRUPTS
      PRINT, DOT           ;TO INDICATE READY
      CLR   2PSW          ;SET PROC. PRIORITY 20
;*****
;PROGRAM RUNS IN THE FOLLOWING SUBROUTINE BREAKING OUT TO SERVICE
;DL11 & KEYBOARD INTERRUPTS AND RUNNING A BACKGROUND JOB OF LOOKING
;FOR AND PRINTING ANY ERRORS ENCOUNTER BY THE SERVICE ROUTINES.
;*****
TSTERR: MOV    ACTIVE,2SWR  ;DISPLAY SYSTEM STATUS
      CMP    #ERRBUF+390.,R0 ;TEST THAT THE ERROR BUFFER ISN'T EXCEEDED
      BLOS  1$             ;IF IT DOES, RESET BUFFER POINTER
      TST   ERRFLG         ;ANY ERRORS PENDING?
      BNE   SRVERR        ;YES, SERVICE ERROR
1$:  MOV    #ERRBUF,R5     ;NO, RE-SET BUFFER POINTERS
      MOV    R5,R0
      BR    TSTERR       ;RE-SERVICE
;*****
;ENTERED HERE TO REPORT THE SYSTEM ERRORS. ALL ERRORS ARE BUFFERED
;IN THE SERVICE ROUTINES I.E. RECEIVER, READER AND TRANSMITTER.
;THESE ERRORS ARE THEN REPORTED AS BACKGROUND JOBS.
;*****
SRVERR: DEC    ERRFLG
      BMI    1$           ;COUNT SHOULD NEVER GO NEGATIVE
      TST   RMODE        ;DATA REPORT MODE REQUESTED?
      BNE   DATA        ;YES, GO TO DATA REPORT ROUTINE
      MOVB  (R5)+,KSTOR2  ;GET THE ERROR CODE
      MOVB  (R5)+,KSTOR1  ;GET FAILING UNIT NO.
      DEC   ERRFLG
      BPL   2$           ;CHECK ERROR COUNT NEVER GOES NEGATIVE
1$:  CLR    ERRFLG       ;RESET IF IT DOES
      BR   TSTERR
2$:  PRINT, MES3        ;TEXT 'LINE'
      BINDEC, KSTOR1    ;PRINT FAILING UNIT NO.
      MOV   KSTOR2,R4   ;PICK UP OFFSET TO PRINT ERROR TYPE
      BIC   #177770,R4
      ASL  R4
      CMP  #7,R4
      BEQ  TSTERR
      MOV  ERRTABL(R4),ERRMES+2

```

790 002370 104000 000000
 791 002374 000721
 792
 793 002376 011666
 794 002400 011723
 795 002402 011745
 796 002404 011767
 797 002406 012010
 799 002410 012037
 799 002412 012074
 800
 801
 802
 803
 804

ERRMES: PRINT, HALT ;MODIFIED TO PRINT ERROR MESSAGE
 BR TSTERR ;LOOK FOR FUTHER ERRORS
 ERRRTL: CODE00 ;= ILLEGAL RECEIVER INTERRUPT
 CODE01 ;= OVERRUN ERROR
 CODE02 ;= FRAMING ERROR
 CODE03 ;= PARITY ERROR
 CODE04 ;= ILLEGAL START CODE RECEIVED
 CODE05 ;= ILLEGAL READER INTERRUPT
 CODE06 ;= ILLEGAL TRANSMITTER INTERRUPT

 ;ENTERED HERE WHEN IN DIAGNOSTIC MODE TO TYPE DATA RECEIVED FROM THE VT20

805 002414 105777 176566
 806 002420 100375
 807 002422 122715 000377
 808 002426 001010
 809 002430 112777 000336 176552
 810 002436 005037 013340
 811 002442 104000 012133
 812 002446 000413
 813

DATA: TSTB @TPS
 BPL -4
 CMPB #377,(RS) ;CHAR. = TO START CODE?
 BNE PRTAS1 ;NO, CHECK FOR 'LF' CODE
 MOV #336,@TPB ;YES, CHANGE CODE TO '1'.
 PRTAS0: CLR PRTCNT ;CLR PRINT COUNT.
 PRINT, CRLF
 BR EXIT

814 002450 122715 000012
 815 002454 001770
 816 002456 111577 176526
 817 002462 005237 013340
 818 002466 022737 000100 013340
 819 002474 001760
 820 002476 105725
 821 002500 009657
 822

PRTAS1: CMPB #12,(RS) ;CHAR. = TO 'LF'
 BEQ PRTAS0 ;YES, PRINT 'CR-LF'
 MOVB (RS),@TPB ;NO, PRINT CHAR. AS IS.
 INC PRTCNT ;INC. PRINT COUNT
 CMP #64,PRTCNT ;LINE FULL?
 BEQ PRTAS0 ;YES, PRINT CRLF.
 EXIT: TSTB (RS)+ ;INC. BUFFER POINTER.
 BR TSTERR ;RETURN

 ;ENTERED HERE TO SERVICE KEYBOARD INTERRUPTS

827 002502 010146
 828 002504 010246
 829 002506 010346
 830 002510 010446
 831 002512 117701 176466
 832 002516 042701 177600
 833 002522 010137 013170
 834 002526 022701 000015
 835 002532 001003
 836 002534 004737 005042
 837 002540 000520
 838 002542 022701 000012
 839 002546 001504
 840 002550 020127 000033
 841 002554 002101
 842 002556 012701 000136
 843 002562 004737 005042
 844 002566 013701 013170
 845 002572 052701 000100

KEYSRV: MOV R1,-(SF) ;SAVE WORKING REG.'S
 MOV R2,-(SP)
 MOV R3,-(SP)
 MOV R4,-(SP)
 MOV @T(R,R1) ;GET CHAR.
 BIC #17600,R1 ;STRIPE OFF PARITY BIT
 MOV R1,SCHAR ;SAVE THE CHAR.
 CMP #15,R1 ;CHAR. = TO 'CR'?
 BNE 4\$;NO
 JSR PC,TYPEIT ;YES, ECHO IT
 BR EXIT ;EXIT
 4\$: CMP #12,R1 ;CHAR. = TO 'LF'?
 BEQ GETLN ;YES
 CMP R1,#33 ;CHAR. PRINTABLE?
 BGE GETLN ;YES
 MOV #136,R1 ;NO, PRINT AS A CONTROL CHAR.
 JSR PC,TYPEIT
 MOV SCHAR,R1
 BIS #100,R1 ;MAKE CHAR. PRINTABLE

846	002576	004737	005042		JSR	PC,TYPEIT	
847	002602	122701	000101		CMPB	#101,R1	;CHAR. = TO '↑A'?
848	002606	001002			BNE	.+6	;NO
849	002610	000137	001314		JMP	START	;YES, RESTART PROGRAM
850	002614	122701	000103		CMPB	#103,R1	;CHAR. = TO '↑C'?
851	002620	001524			BEQ	CONC	
852	002622	122701	000105		CMPB	#105,R1	;↑E FOR ESCAPE AND RESTART
853	002626	001002			BNE	.+6	
854	002630	000137	002152		JMP	SERVICE	;YES, DO A COMPLETE RESTART
855	002634	005737	013154		TST	SYSSWH	;SYSTEM ACTIVE?
856	002640	001042			BNE	QMARK	;YES, IGNORE REQUEST
857	002642	005237	013154		INC	SYSSWH	;NO, SET REQUEST SW.
858	002646	122701	000102		CMPB	#102,R1	;CHAR. = TO '↑B'?
859	002652	001002			BNE	.+6	;NO
860	002654	000137	004452		JMP	BOOT	;YES, BOOT TAPE TO VT20.
861	002660	122701	000104		CMPB	#104,R1	;CHAR. = '↑D'?
862	002664	001466			BEQ	COND	;YES, DIAGNOSTIC MODE
863	002666	122701	000110		CMPB	#110,R1	;↑H (HOLD)?
864	002672	001450			BEQ	CONH	
865	002674	122701	000122		CMPB	#122,R1	;↑R (RELEASE)?
866	002700	001517			BEQ	CONR	
867	002702	122701	000123		CMPB	#123,R1	;↑S (SEND)?
868	002706	001571			BEQ	CONS	
869	002710	122701	000114		CMPB	#114,R1	;↑L (LIST ACTIVE RECEIVERS)
870	002714	001002			BNE	1\$;NO
871	002716	000137	004134		JMP	CONL	;YES
872	002722	122701	000115	1\$:	CMPB	#115,R1	;↑M (MONITOR SYSTEM STATUS)
873	002726	001002			BNE	2\$;NO
874	002730	000137	004216		JMP	CONM	;YES
875	002734	122701	000120	2\$:	CMPB	#120,R1	;↑P (PRINT)?
876	002740	001531			BEQ	CONP	
877	002742	005337	013154		DEC	SYSSWH	;CLR SYSTEM SWITCH ON ILLEGAL ENTRY
878	002746	112701	000077	QMARK:	MOVB	#77,R1	;ILLEGAL CHAR.
879	002752	004737	005042		JSR	PC,TYPEIT	;TYPE '?'
880	002756	000411			BR	EXITKS	;IGNORE IT
881	002760	005737	013202	GETLN:	TST	CONSFL	;ARE WE IN SEND MODE?
882	002764	001161			BNE	SENDLN	;YES, GO TO SEND ROUTINE
883	002766	000137	003520		JMP	GETLN2	;NO TO LINE NO. INPUT ROUTINE
884							
885	002772	005037	013154	PRTDOT:	CLR	SYSSWH	
886	002776	104000	012135		PRINT,	DOT	
887	003002	012604		EXITKS:	MOV	(SP)+,R4	;RESTORE THE WORKING REG.'S
888	003004	012603			MOV	(SP)+,R3	
889	003006	012602			MOV	(SP)+,R2	
890	003010	012601			MOV	(SP)+,R1	
891	003012	000002			RTI		;EXIT

```

892
893
894
895
896 003014 004737 003730
897 003020 005263 012672
898 003024 005337 013102
899 003030 001403
900 003032 004737 003734
901 003036 000770
902 003040 000754
903
904
905
906
907
908 003042 005137 013336
909 003046 001403
910 003050 104000 011567
911 003054 000405
912 003056 012700 014146
913 003062 010005
914 003064 005037 013342
915 003070 000740
916
917
918
919
920
921 003072 005002
922 003074 005003
923 003076 005037 013102
924 003102 004737 003152
925 003106 062702 000020
926 003112 062703 000002
927 003116 022703 000040
928 003122 001367
929 003124 012604
930 003126 012603
931 003130 012602
932 003132 012601
933 003134 000137 002174
934
935
936
937
938
939
940 003140 004737 003730
941 003144 004737 003152
942 003150 000710
943
944 003152 005763 012672
945 003156 001413
946 003160 005063 012672
947 003164 005063 012734

```

```

;*****
;ENTERED HERE ON RECEIPT A 'H' TO HOLD A SPECIFIED TRANSMISSION LINE.
;*****

```

```

CONH: JSR PC,FORMIT ;FORM OFFSETS
      INC HLDLO(R3) ;SET HOLD SWITCH FOR THIS LINE
      DEC BCDCTR ;ANY MORE LINES TO BE HELD?
      BEQ CONH1 ;NO
      JSR PC,FORMON ;YES, FORM OFFSET FOR LINE
      BR CONH+4 ;HOLD NEXT LINE
CONH1: BR PRDPT ;EXIT

```

```

;*****
;SUBROUTINE ENTERED ON RECEIPT OF A 'D' TO ENTER DIAGNOSTIC MODE
;*****

```

```

COND: COM RMODE ;SET/CLEAR DIAGNOSTIC SW.
      BEQ 1$ ;RESET POINTERS IF CLEARED
      PRINT, MES9 ;TEXT 'DIAG. MODE ENABLED'
      BR 2$ ;EXIT
1$: MOV #ERRBUF,RO ;RESET BUFFER POINTERS
    MOV RO,R5
    CLR ERAFLG ;CLEAR THE ERROR COUNTER
    BR PRDPT ;EXIT
2$:

```

```

;*****
;ENTERED ON RECEIPT OF A 'C' TO CLEAR ALL USER SOFTWARE SWITCHES.
;*****

```

```

CONC: CLR R2
      CLR R3
      CLR BCDCTR
CONC1: JSR PC,RELEASE ;RELEASE HELD LINES
      ADD #16,R2
      ADD #2,R3
      CMP #32,R3 ;RELEASED ALL LINES?
      BNE CONC1 ;NO
      MOV (SP)+,R4 ;RESTORE THE WORKING REG.'S
      MOV (SP)+,R3
      MOV (SP)+,R2
      MOV (SP)+,R1
      JMP RESTR

```

```

;*****
;ENTERED HERE ON RECEIPT OF A 'R' TO RELEASE A SPECIFIED TRANSMISSION
;LINE.
;*****

```

```

CONR: JSR PC,FORMIT ;FORM THE OFFSETS
      JSR PC,RELEASE ;RELEASE HELD LINES
      BR PRDPT ;EXIT
RELEASE: TST HLDLO(R3) ;ACTIVE?
        BEQ CONR2 ;NO, EXIT
CONR1: CLR HLDLO(R3) ;CLR HOLD SW FOR THIS LINE
      CLR SENDSW(R3) ;CLEAR THE SEND SW.

```



```

948 003170 005263 013422          INC    TRNSWH(R3)      ;SET TRANSMITTER SWITCH
949 003174 056337 012146 013164    BIS    UNITNO(R3),ACTIVE ;SET ACTIVE WHEN TRANSMITTING
950 003202 005072 012214          CLR    QXBUFO(R2)     ;START TRANSMISSION OPERATION
951 003206 005337 013102    CONR2: DEC    BCDCTR      ;ANY MORE LINES TO BE RELEASED?
952 003212 003403          BLE    CONR3          ;NO
953 003214 004737 003734          JSR    PC,FORMON      ;YES, FORM OFFSET FOR NEXT LINE
954 003220 000754          BR    RELEASE        ;RELEASE NEXT LINE
955 003222 000207    CONR3: RTS    PC      ;RETURN

```

```

;*****
;ENTERED HERE ON RECEIPT OF A 'P' TO PRINT THE DATA IN A SPECIFIED
;LINES BUFFER. THIS CAN EITHER BE DATA RECEIVED FROM A SLAVE VT20
;OR DATA ENTERED WHILE IN THE SEND MODE.
;*****

```

```

963 003224 005077 175750    CONP:  CLR    QPSW      ;ENABLE FURTHER INTERRUPTS
964 003230 005037 013336    CLR    RMODE          ;CLR DIAGNOSTIC MODE
965 003234 004737 003730    JSR    PC,FORMIT      ;FORM THE OFFSETS
966 003240 010437 003246    CONP1: MOV    R4,CONP2+2 ;SET UP BUFFER TO BE PRINTED
967 003244 104000 000000    CONP2: PRINT, HALT
968 003250 005337 013102    DEC    BCDCTR          ;DONE PRINTING ALL REQUESTS
969 003254 003405          BLE    CONP3          ;YES
970 003256 004737 003734    JSR    PC,FORMON      ;NO, FORM NEXT OFFSET
971 003262 104000 012133    PRINT, CRLF
972 003266 000764          BR    CONP1
973 003270 000640    CONP3: BR    PRDPT     ;EXIT

```

```

;*****
;SUBROUTINE TO SETUP TO SEND DATA FROM KB TO SPECIFIED LINE
;*****

```

```

979 003272 004737 003730    CONS:  JSR    PC,FORMIT ;FORM THE OFFSETS
980 003276 005237 013202    INC    CONSFL          ;SET SOFTWARE SW.
981 003302 010463 013464    CONS1: MOV    R4,SAVR4(R3) ;SET UP THE BUFFER POINTER
982 003306 005263 012734    INC    SENDSW(R3)      ;SET SEND SW.
983 003312 005337 013102    DEC    BCDCTR          ;ANY MORE LINES TO BE SETUP?
984 003316 003403          BLE    CONS2          ;NO
985 003320 004737 003734    JSR    PC,FORMON      ;YES, SET THEM UP
986 003324 000766          BR    CONS1
987 003326 000625    CONS2: BR    EXITKS    ;EXIT
989 003330 004737 005042    SENDLN: JSR    PC,TYPEIT ;ECHO CHAR.
990 003334 005001          CLR    R1
991 003336 005002          CLR    R2
992 003340 005003          CLR    R3
993 003342 005763 012734    CONSO: TST    SENDSW(R3) ;SEND TO THIS LINE?
994 003346 001424          BEQ    TAGC           ;NO, CHECK NEXT LINE
995 003353 016304 013464    MOV    SAVR4(R3),R4   ;SET UP BUFFER POINTER
996 003354 122737 000033 013170    CMPB  #33,SCHAR      ;TERMINATE SEND MODE?
997 003362 001427          BEQ    TAGA           ;YES
998 003364 005763 012672    TST    HLDLC(R3)      ;IS IT BEING HELD?
999 003370 001003          BNE    SENDBF        ;YES-BUFFER DATA
1000 003372 113772 013170 012214    MOVB  SCHAR,QXBUFO(R2) ;SEND CHARACTER
1001
1002 003400 113724 013170    SENDBF: MOVB  SCHAR,(R4)+ ;SAVE CHARACTER IN BUFFER
1003 003404 112714 000014    MOVB  #EOP,(R4)      ;TERIMATE THE BUFFER

```

```

1004 003410 010463 013464
1005 003414 062701 000500
1006 003420 062702 000020
1007 003424 062703 000002
1008 003430 022703 000040
1009 003434 001342
1010 003436 000137 003002
1011 003442 005037 013202
1012 003446 112714 000014
1013 003452 005763 012734
1014 003456 001407
1015 003460 005063 012734
1016 003464 012704 015036
1017 003470 060104
1018 003472 010463 013464
1019 003476 062701 000764
1020 003502 062703 000002
1021 003506 022703 000040
1022 003512 001357
1023 003514 000137 002772
1024
1025
1026
1027
1028
1029 003520 022701 000177
1030 003524 001446
1031 003526 005737 013166
1032 003532 001402
1033 003534 104000 012140
1034 003540 005037 013166
1035 003544 004737 005042
1036 003550 120127 000054
1037 003554 001410
1038 003556 120127 000060
1039 003562 002403
1040 003564 120127 000071
1041 003570 003402
1042 003572 000137 002746
1043 003576 110177 007354
1044 003602 005237 013156
1045 003606 105077 007344
1046 003612 042701 177770
1047 003616 006337 013206
1048 003622 006337 013206
1049 003626 006337 013206
1050 003632 060137 013206
1051 003636 000137 003002
1052
1053 003642 005737 013166
1054 003646 001002
1055 003650 104000 012140
1056 003654 005237 013166
1057 003660 005337 013156
1058 003664 117701 007266
1059 003670 105077 007262

```

```

MOV R4,SAVR4(R3) ;SAVE BUFFER POINTER
TAGC: ADD #500,R1
ADD #16,R2
ADD #2,R3
CMP #32,R3 ;DONE ALL LINES?
BNE CONSO ;NO
JMP EXITKS ;YES, EXIT
TAGA: CLR CONSFL
MOV #EOP,(R4) ;TERMINATE BUFFER
TAGB: TST SENDSW(R3) ;SENDING ON THIS LINE?
BEQ TAGD ;NO
CLR SENDSW(R3)
MOV #BUFFER+4,R4
ADD R1,R4
TAGD: MOV R4,SAVR4(R3) ;SET UP BUFFER POINTER
ADD #500,R1
ADD #2,R3
CMP #32,R3
BNE TAGB
JMP PRDCT ;EXIT

;*****
;SUBROUTINE ENTERED TO FORM ADDRESS OR LINE NUMBER
;*****

GETLN2: CMP #177,R1 ;CHAR. = RUBOUT?
BEQ RUBOUT ;YES
TST RUBSWH ;IS RUBOUT SW. SET?
BEQ 1$ ;NO
PRINT, SLASH ;YES, PRINT '/'
CLR RUBSWH ;CLR SW.
JSR PC,TYPEIT ;ECHO CHAR.
CMPB R1,#54 ;CHAR. = TO ','?
BEQ 3$ ;YES, SAVE IT
CMPB R1,#60 ;LEGAL NO.?
BLT 2$ ;NO
CMPB R1,#71
BLE 3$ ;YES
JMP QMARK ;NO, TYPE '?'
3$: MOVB R1,@TTYPTR ;SAVE CHAR. IN TTY BUFFER
INC TTYPTR ;UPDATE POINTER
CLRB @TTYPTR ;TERMINATE BUFFER WITH NULL
BIC #177770,R1 ;STRIP
ASL DEVADR ;LEFT JUSTIFY '3' PLACES
ASL DEVADR
ASL DEVADR
ADD R1,DEVADR ;THEN ADD NEW DIGIT
JMP EXITKS ;EXIT

RUBOUT: TST RUBSWH ;IS THE RUBOUT SW. SET?
BNE 1$ ;YES
PRINT, SLASH
1$: INC RUBSWH ;SET SW.
DEC TTYPTR ;BACK UP BUFFER POINTER
MOVB @TTYPTR,R1 ;PICK UP PREVIOUS CHAR.
CLRB @TTYPTR ;TERMINATE BUFFER

```

```

1060 003674 004737 005042
1061 003700 013701 013206
1062 003704 042701 177770
1063 003710 006237 013206
1064 003714 006237 013206
1065 003720 006237 013206
1066 003724 000137 003002

```

```

JSR PC,TYPEIT ;ECHO CHAR.
MOV DEVAOR,R1
BIC #177770,R1
ASR DEVAOR
ASR DEVAOR
ASR DEVAOR
JMP EXITKS ;EXIT

```

```

1067
1068
1069
1070
1071

```

```

;*****
;SUBROUTINE ENTERED TO FORM THE ADDRESS AND REGISTER OFFSETS
;*****

```

```

1072 003730 004737 004050
1073 003734 005001
1074 003736 005002
1075 003740 005003
1076 003742 017704 007136
1077 003746 062737 000002 013104
1078 003754 005704
1079 003756 001420
1080 003760 022704 000017
1081 003764 002005
1082 003766 012701 000077
1083 003772 004737 005042
1084 003776 000754
1085 004000 062701 000764
1086 004004 062702 000020
1087 004010 062703 000002
1088 004014 005304
1089 004016 001370
1090 004020 010137 013200
1091 004024 010237 013176
1092 004030 010337 013174
1093 004034 012704 015032
1094 004040 060104
1095 004042 005024
1096 004044 005024
1097 004046 000207

```

```

FORMIT: JSR PC,GETLN1 ;FORM LINE NO.
FORMON: CLR R1
CLR R2
CLR R3
MOV @BCDPTR,R4 ;PICK UP LINE NUMBER
ADD #2,BCDPTR ;UPDATE POINTER FOR NEXT ENTRY
TST R4 ;IS THE LINE NO. '0'?
BEQ 1$ ;YES, NO WORK NEEDED
CMP #15.,R4 ;LEGAL LINE NO?
BGE 2$ ;YES
MOV #77,R1
JSR PC,TYPEIT
BR FORMIT
2$: ADD #500.,R1 ;FORM THE BUFFER OFFSET
ADD #16.,R2 ;FORM THE REGISTER OFFSET
ADD #2.,R3 ;FORM THE ADDRESS OFFSET
DEC R4 ;DONE?
BNE 2$ ;NO
1$: MOV R1,BUFOFF
MOV R2,REGOFF
MOV R3,ADDOFF
MOV #BUFFER,R4
ADD R1,R4 ;ADD BUFFER OFFSET
CLR (R4)+
CLR (R4)+
RTS PC ;RETURN

```

```

1098
1099
1100
1101
1102 004050 005037 013206
1103 004054 005037 013170
1104 004060 005037 014770
1105 004064 012737 014770 013156
1106 004072 052777 000100 175102
1107 004100 005077 175074
1108 004104 013777 013164 175100
1109 004112 023727 013170 000015
1110 004120 001371
1111 004122 004737 006072
1112 004126 104000 012133
1113 004132 000207
1114
1115
1116
1117
1118
1119
1120 004134 005077 175040
1121 004140 104000 011445
1122 004144 005037 013160
1123 004150 012702 001214
1124 004154 104000 012133
1125 004160 104002 013160
1126 004164 012237 013324
1127 004170 104001 013324
1128 004174 005237 013160
1129 004200 032712 000001
1130 004204 001763
1131 004206 005037 013154
1132 004212 000137 002772

```

```

:*****
:SUBROUTINE TO ENABLE FORMING OF A LINE NUMBER
:*****

```

```

GETLN1: CLR      DEVADR      :SET UP TO GET LINE NUMBER
        CLR      SCHAR
        CLR      TTYBUF
        MOV      #TTYBUF, TTYPTR :SET UP BUFFER POINTER
        BIS      #100, @TKS      :ENABLE TTY INTERRUPTS
        CLR      @PSW           :GET LINE NO.
IS:     MOV      ACTIVE, @SWR    :DISPLAY SYSTEM STATUS
        CMP      SCHAR, #15     :EXIT ON CARRIAGE RETURN
        BNE      IS
        JSR      PC, BCOBIN     :CONVERT LINE NO. TO OCTAL
        PRINT   , CRLF
        RTS      PC

```

```

:*****
:ENTERED HERE ON RECEIPT OF A 'L' TO LIST THE ACTIVE DL11 RECEIVER
:ADDRESSES THAT WERE MAPPED AND SET UP BY THE PROGRAM.
:*****

```

```

CONL:  CLR      @PSW          :ALLOW FURTHER INTERRUPTS
        PRINT   , MES7        :TEXT "LIST OF ACTIVE DL11 RECEIVERS"
        CLR      CNTR
        MOV      @RXCRC, R2   :SET UP POINTER TO PICK UP ADDRESSES
IS:     PRINT   , CRLF
        BINDEC  , CNTR        :PRINT THE LINE NO.
        MOV      (R2)+, TEMP   :PICK UP THE DL11 ADDRESS
        PRTOCT , TEMP         :PRINT IT
        INC     , CNTR        :UPDATE THE LINE NUMBER
        BIT     #1, (R2) ;DONE?
        BEQ    IS            :NO
        CLR    SYSSWH
        JMP    PRTOCT       :EXIT

```

```

1133
1134
1135
1136
1137
1138
1139
1140 004216 005077 174756
1141 004222 004737 003730
1142 004226 013702 012626
1143 004232 104000 011504
1144 004236 010337 013172
1145 004242 006237 013172
1146 004246 104000 012133
1147 004252 104002 013172
1148 004256 016337 013526 013324
1149 004264 104002 013324
1150 004270 016337 013570 013324
1151 004276 104002 013324
1152 004302 016337 013674 013324
1153 004310 104002 013324
1154 004314 016337 014000 013324
1155 004322 104002 013324
1156 004326 016337 013736 013324
1157 004334 104002 013324
1158 004340 016337 013632 013324
1159 004346 104002 013324
1160 004352 016337 014104 013324
1161 004360 104002 013324
1162 004364 016337 014042 013324
1163 004372 104002 013324
1164 004376 016337 012672 013324
1165 004404 104002 013324
1166 004410 005237 013172
1167 004414 062703 000002
1168 004420 105737 014770
1169 004424 001002
1170 004426 005302
1171 004430 001306
1172 004432 005337 013102
1173 004436 003403
1174 004440 004737 003734
1175 004444 000674
1176 004446 000137 002772

```

```

:*****
:ENTERED HERE ON RECEIPT OF A 'M' TO MONITOR SYSTEM STATUS. THE 'M'
:OPTION CAN BE USED IN ONE OF TWO WAYS: (1) TYPE 'M (CR)' TO PRINT THE
:STATUS OF ALL DL11 LINES. (2) TYPE 'M & LINE NO.' TO PRINT THE STATUS
:OF SPECIFIED LINE(S).
:*****

```

```

CONM: CLR      QPSW      ;ENABLE FURTHER INTERRUPTS
      JSR      PC,FORMIT ;FORM THE LINE NUMBER
      MOV      HERE,R2   ;PICK UP THE NO. OF ACTIVE DL'S
      PRINT,   MESB      ;PRINT HEADER
15:   MOV      R3,LINNO   ;SET UP THE LINE NO.
      ASR      LINNO
      PRINT,   CRLF      ;PRINT THE LINE NO.
25:   BINDEC,  LINNO
      MOV      RECNR(R3),TEMP ;PRINT NO. OF BLOCKS RECEIVED
      BINDEC,  TEMP
      MOV      XFERCT(R3),TEMP ;PRINT NO. OF BLOCKS TRANSFERED
      BINDEC,  TEMP
      MOV      OR(R3),TEMP    ;PRINT NO. OF OVERRUN ERRORS
      BINDEC,  TEMP
      MOV      PAR(R3),TEMP   ;PRINT NO. OF PARITY ERRORS
      BINDEC,  TEMP
      MOV      FRM(R3),TEMP   ;PRINT NO. OF FRAMING ERRORS
      BINDEC,  TEMP
      MOV      REC(R3),TEMP   ;PRINT NO. OF RECEIVER ERRORS
      BINDEC,  TEMP
      MOV      TRN(R3),TEMP   ;PRINT NO. OF TRANSMITTER ERRORS
      BINDEC,  TEMP
      MOV      ST(R3),TEMP    ;PRINT NO. OF START CODE ERRORS
      BINDEC,  TEMP
      MOV      HLDLD(R3),TEMP ;PRINT STATUS IF HELD
      INC      LINNO        ;UPDATE THE LINE NO.
      ADD      #2,R3        ;UPDATE THE OFFSET NO.
      TSTB    TTYBUF       ;WAS A SPECIFIED LINE REQUESTED?
      BNE     35           ;YES, EXIT
      DEC     R2           ;NO, DONE ALL LINES?
      BNE     25           ;NO
35:   DEC     BCDCTR       ;DONE ALL LINES?
      BLE     45           ;YES, EXIT
      JSR     PC,FORMON    ;NO SET UP NEXT LINE
      BR     15
45:   JMP     PRD DOT

```

```

1177
1178
1179
1190
1191
1182
1183
1184
1185
1186
1187
1188
1189 004452 004737 003730
1190 004456 013737 013102 013334
1191 004464 012737 013262 013212
1192 004472 012737 013220 013214
1193 004500 010277 006506
1194 004504 010377 006504
1195 004510 062737 000002 013212
1196 004516 062737 000002 013214
1197 004524 005263 012776
1198 004530 005337 013102
1199 004534 001403
1200 004536 004737 003734
1201 004542 000756
1202 004544 005737 013416
1203 004550 001027
1204 004552 104000 011614
1205 004556 004737 004050
1206 004562 013701 013206
1207 004566 005701
1208 004570 001770
1209 004572 010137 013416
1210 004576 062701 000002
1211 004602 010137 013420
1212 004606 104000 011645
1213 004612 004737 004050
1214 004616 013701 013206
1215 004622 012721 004646
1216 004626 005011
1217 004630 012777 000101 006560
1218 004636 005037 013154
1219 004642 000137 003002
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229 004646 010146
1230 004650 010246
1231 004652 010346
1232 004654 010446

```

```

:*****
:BOOT SUBROUTINE
:THIS SUBROUTINE IS ENTERED ON RECEIPT OF A '1B' FROM THE KEYBOARD.
:ITS PURPOSE IS TO BOOT (TRANSFER) THE VT20 PROGRAM FROM THE READER
:OF THE HOST COMPUTER TO A SELECTED VT20 SYSTEMS. WHEN ENTERED, THE
:THE PROGRAM WILL WAIT FOR THE LINE OR LINES OVER WHICH THE PROGRAM
:IS TO BE TRANSFERED. THE PROGRAM WILL THEN CHECK TO SEE IF A READER
:DEVICE ADDRESS HAS BEEN PREVIOUSLY SETUP. IF IT HASN'T, THE PROGRAM
:WILL REQUEST THE DEVICE & VECTOR ADDRESSES. AFTER COMPLETION OF
:LOADING, THE PROGRAM WILL RETURN TO THE MONITOR.
:*****

```

```

BOOT: JSR PC,FORMIT ;WAIT FOR LINE NUMBERS
      MOV BCDCTR,KSTOR4 ;SAVE NO. OF LINE SET UP
      MCV #BOOTLN,BOOTP1 ;SET UP POINTERS TO SAVE LINE NO.'S
      MOV #BOOTOF,BOOTP2
BOOT1: MOV R2,#BOOTP1
      MOV R3,#BOOTP2
      ADD #2,BOOTP1
      ADD #2,BOOTP2
      INC BOOTSW(R3) ;SET BOOT SOFTWARE SW.
      DEC BCDCTR ;SET UP ALL LINES?
      BEQ BOOT2 ;YES,
      JSR PC,FORMON ;NO, FORM OFFSETS FOR NEXT LINE
      BR BOOT1
BOOT2: TST RCSR ;HAS READER ADDRESS BEEN SETUP?
      BNE BOOT4 ;YES
BOOT3: PRINT, MES11 ;REQUEST READER DEVICE ADDRESS
      JSR PC,GETLN1 ;GET IT
      MOV DEVADR,R1
      TST R1 ;WAS ONE ENTERED?
      BEQ BOOT3 ;NO, MAKE HIM DO IT
      MOV R1,RCSR ;YES, SET IT UP
      ADD #2,R1
      MOV R1,ROBR
      PRINT, MES13 ;ASK FOR VECTOR ADDRESS
      JSR PC,GETLN1 ;GET IT
      MOV DEVADR,R1
      MOV #READER,(R1)+ ;SET UP READER SERVICE ROUTINE
      CLR (R1) ;SET PRIORITY 20
BOOT4: MOV #101,RCSR ;SET GO & INTERRUPT ENABLE FOR READER
      CLR SYSSW
      JMP EXITKS ;EXIT

```

```

:*****
:READER INTERRUPT SERVICE ROUTINE
:THIS ROUTINE READS '1' CHARACTER FROM THE PAPERTAPE READER AND
:TRANSMITS IT TO THE SPECIFIED LINE OR LINES ENTERED BY (1B).
:READER ERROR CODES ARE AS FOLLOWS:
:05 = ILLEGAL READER INTERRUPT
:*****

```

```

READER: MOV R1,-(SP) ;SAVE WORKING REGISTERS
        MOV R2,-(SP)
        MOV R3,-(SP)
        MOV R4,-(SP)

```

```

1233 004656 013704 013334      MOV      KSTOR4,R4      ;SET UP NO. OF LINES TO BE SERVICED
1234 004662 005037 013216      CLR      BOOTAD
1235 004666 017737 006524 013152  MOV      @RCSR,RSTAT   ;SAVE READER STATUS
1236 004674 017737 006520 013150  MOV      @RDBR,RCHAR   ;READ & SAVE CHAR.
1237 004702 013701 013216      READIT: MOV      BOOTAD,R1
1238 004706 016102 013262      MOV      BOOTLN(R1),R2
1239 004712 016103 013220      MOV      BOOTOF(R1),R3 ;SET UP BOOT OFFSET
1240 004716 005763 012776      TST      BOOTSW(R3)   ;IS BOOT SW. SET?
1241 004722 001010      BNE      READ1        ;YES, LEGAL INTERRUPT
1242 004724 062737 000002 013342  ADD      #2,ERRFLG     ;NO, SET ERROR FLAG
1243 004732 112720 000005      MOVVB   #05,(R0)+    ;CODE FOR ILLEGAL READER INTERRUPT
1244 004736 110310      MOVVB   R3,(R0)
1245 004740 106220      ASRB   (R0)+
1246 004742 000417      BR      READ2
1247 004744 005737 013152      READ1:  TST      RSTAT   ;EXIT
1248 004750 100414      BMI      READ2        ;END OF TAPE FLAG SET?
1249 004752 113772 013150 012214  MOVVB   RCHAR,@XBUFO(R2) ;YES, EXIT
1250 004760 062737 000002 013216  ADD      #2,BOOTAD    ;TRANSMIT THE CHARACTER
1251 004766 005304      DEC     R4            ;SET UP TO ADDRESS NEXT LINE
1252 004770 001344      BNE     READIT       ;DONE ALL LINES?
1253 004772 012777 000101 006416  MOV      #101,@RCSR   ;NO
1254 005000 000413      BR      READ3        ;START NEXT READ
1255 005002 005077 006410      READ2:  CLR      @RCSR   ;EXIT
1256 005006 005063 012776      CLR     BOOTSW(R3)   ;DISABLE READER INTERRUPT
1257 005012 062737 000002 013216  ADD      #2,BOOTAD    ;CLEAR THE SOFTWARE SW.
1258 005020 005304      DEC     R4            ;SET UP TO SERVICE NEXT LINE
1259 005022 001327      BNE     READIT       ;DONE
1260 005024 104000 012135      PRINT,  DOT         ;NO
1261 005030 012604      READ3:  MOV      (SP)+,R4
1262 005032 012603      MOV      (SP)+,R3
1263 005034 012602      MOV      (SP)+,R2
1264 005036 012601      MOV      (SP)+,R1
1265 005040 000002      RTI
1266
1267      ;*****
1268      ;SUBROUTINE TO TYPE THE CHARACTER IN 'R1'
1269      ;*****
1270
1271 005042 013777 013164 174142  TYPEIT: MOV      ACTIVE,@SWR ;DISPLAY SYSTEM STATUS
1272 005050 105777 174132      TSTB   @TPS         ;WAIT FOR PRINTEF'
1273 005054 100372      BPL    TYPEIT
1274 005056 110177 174126      MOVVB  R1,@TPB     ;OUTPUT CHAR.
1275 005062 000207      RTS     PC

```

```

1276 :*****
1277 :SUBROUTINE ENTERED TO SERVICE ALL DL11 RECEIVER INTERRUPTS.
1278 :R0=ERROR ADDRESS POINTER
1279 :R1=DATA BUFFER ADDRESS OFFSET (BUFFERS ARE 512 BYTES APART)
1280 :R2=DEVICE REGISTER ADDRESS OFFSET
1281 :R3=UNIT ADDRESS OFFSET
1282 :R4=DATA BUFFER ADDRESS POINTER
1283 :RECEIVER ERROR CODES ARE AS FOLLOWS:
1284 :00 = ILLEGAL RECEIVER INTERRUPT
1285 :01 = OVERRUN ERROR
1286 :02 = FRAMING ERROR
1287 :03 = PARITY ERROR
1288 :04 = ILLEGAL START CODE
1289 :*****
1290
1291 005064 017237 012210 013204 RECVER: MOV      DRBUF0(R2), SAVCHR      ;READ & SAVE CHARACTER
1292 005072 005763 013422          TST      TRNSWH(R3)      ;CURRENTLY TRANSMITTING DATA?
1293 005076 001410          BEQ      RECVR1        ;NO, SERVICE AS LEGAL INTERRUPT
1294 005100 005263 013632          INC      REC(R3)       ;KEEP TRACK OF RECEIVER ERRORS
1295 005104 022763 000005 013040      CMP      #5, ERACR(R3) ;HAS UNIT EXCEEDED ERROR LIMIT?
1296 005112 002553          BLT      RECEXT        ;YES, EXIT
1297 005114 105020          CLRB    (R0)+         ;YES, POST AS ILLEGAL RECVR. INTERRUPT
1298 005116 000542          BR      RECERR        ;REPORT ERROR
1299
1300 005120 005737 013204      RECVR1: TST      SAVCHR      ;RECEIVER ERROR FLAG SET?
1301 005124 100043          BPL      RECVR2        ;NO, VALID CHAR.
1302 005126 032737 040000 013204      BIT      #40000, SAVCHR ;YES, IS IT AN OVERRUN ERROR?
1303 005134 001411          BEQ      FRAMER        ;NO
1304 005136 005263 013674          INC      OR(R3)        ;KEEP TRACK OF OVERRUN ERRORS
1305 005142 022763 000005 013040      CMP      #5, ERACR(R3) ;HAS UNIT EXCEEDED ERROR LIMIT?
1306 005150 002534          BLT      RECEXT        ;YES, EXIT
1307 005152 112720 000001          MOVB   #01, (R0)+     ;YES, POST AS OVERRUN ERROR
1308 005156 000522          BR      RECERR
1309
1310 005160 032737 020000 013204      FRAMER: BIT      #20000, SAVCHR ;IS IT A FRAMING ERROR?
1311 005166 001411          BEQ      PARITY        ;NO
1312 005170 005263 013736          INC      FRM(R3)       ;KEEP TRACK OF FRAMING ERRORS
1313 005174 022763 000005 013040      CMP      #5, ERACR(R3) ;HAS UNIT EXCEEDED ERROR LIMIT?
1314 005202 002517          BLT      RECEXT        ;YES, EXIT
1315 005204 112720 000002          MOVB   #02, (R0)+     ;YES, POST AS FRAMING ERROR
1316 005210 000505          BR      RECERR
1317
1318 005212 005263 014000          PARITY: INC      PAR(R3) ;KEEP TRACK OF PARITY ERRORS
1319 005216 022763 000005 013040      CMP      #5, ERACR(R3) ;HAS UNIT EXCEEDED ERROR LIMIT?
1320 005224 002506          BLT      RECEXT        ;YES, EXIT
1321 005226 112720 000003          MOVB   #03, (R0)+     ;POST AS PARITY ERROR
1322 005232 000474          BR      RECERR
1323
1324 005234 005763 013354          RECVR2: TST      RECSWH(R3) ;CURRENTLY RECEIVING DATA?
1325 005240 001033          BNE      RECVR4        ;YES, SAVE CHAR.
1326 005242 105737 013204          TSTB   SAVCHR         ;NO, NULL CHAR.?
1327 005246 001475          BEQ      RECEXT
1328 005250 122737 000377 013204      CMPB   #377, SAVCHR   ;= TO START CODE CHAR.?
1329 005256 001411          BEQ      RECVR3        ;YES, SET UP TO RECEIVE DATA
1330 005260 005263 014042          INC      ST(R3)
1331 005264 022763 000005 013040      CMP      #5, ERACR(R3) ;HAS UNIT EXCEEDED ERROR LIMIT?

```



```

1332 005272 002463          BLT      RECEXT      ;YES, EXIT
1333 005274 112720 000004    MOVB     #04,(R0)+   ;NO, POST AS ILLEGAL START CODE
1334 005300 000451          BR       RECERR
1335 005302 005263 013354    RECVR3: INC     RECSWH(R3) ;SET RECEIVER SOFTWARE SW.
1336 005306 046337 012146 013164    BIC     UNITNO(R3),ACTIVE ;KEEP TRACK OF SYSTEM STATUS
1337 005314 012704 015032    MOV     #BUFFER,R4  ;SET UP BUFFER POINTER
1338 005320 060104          ADD     R1,R4       ;ADD OFFSET
1339 005322 005024          CLR     (R4)+
1340 005324 005024          CLR     (R4)+
1341 005326 000404          BR      RECR4A
1342 005330 122737 000377 013204    RECVR4: CMPB    #377,SAVCHR ;CHAR. = TO START CODE?
1343 005336 001761          BEQ     RECVR3      ;YES, RE-SET RECEIVER
1344 005340 113714 013204    RECR4A: MOVB   SAVCHR,(R4) ;SAVE CHARACTER IN BUFFER
1345 005344 005737 013336    TST     RMODE       ;RUNNING DATA REPORT MOLE?
1346 005350 001403          BEQ     RECVRS      ;NO
1347 005352 005237 013342    INC     ERRFLG      ;YES, FORCE DATA TYPEOUT
1348 005356 111420          MOVB   (R4),(R0)+   ;SAVE CHAR. IN ERROR BUFFER
1349 005360 122724 000014    RECVR5: CMPB    #EOP,(R4)+ ;CHAR. = END OF PARAGRAPH?
1350 005364 001026          BNE     RECEXT      ;NO, EXIT
1351 005366 005063 013354    CLR     RECSWH(R3)  ;YES, CLEAR RECEIVER SW.
1352 005372 005263 013526    INC     RECCTR(R3)  ;COUNT NO. OF BLOCKS RECEIVED
1353 005376 005263 013422    INC     TRNSWH(R3)  ;SET UP TO TRANSMIT
1354 005402 056337 012146 013164    BIS     UNITNO(R3),ACTIVE ;KEEP TRACK OF SYSTEM STATUS
1355 005410 012704 015032    MOV     #BUFFER,R4  ;SET UP BUFFER POINTER
1356 005414 060104          ADD     R1,R4       ;ADD OFFSET
1357 005416 111472 012214    MOVB   (R4),DXBUFD(R2) ;START UP TRANSMITTER
1358 005422 000407          BR      RECEXT
1359 005424 005263 013040    RECERR: INC     ERRCTR(R3) ;KEEP TRACK OF NO. OF ERRORS
1360 005430 062737 000002 013342    ADD     #2,ERRFLG
1361 005436 110310          MOVB   R3,(R0)
1362 005440 106220          ASRB   (R0)+
1363 005442 000207    RECEXT: RTS      PC      ;RETURN

```

```

1364
1365 ;*****
1366 ;SUBROUTINE ENTERED TO SERVICE ALL DL11 TRANSMITTER INTERRUPTS
1367 ;TRANSMITTER ERROR CODES ARE AS FOLLOWS:
1368 ;06 = ILLEGAL TRANSMITTER INTERRUPT
1369 ;*****
1370

```

```

1371 005444 005763 012630    TRNMIT: TST     INTRAN(R3) ;INITIALIZATION INTERRUPT?
1372 005450 001003          BNE     .+10        ;NO, SERVICE IT.
1373 005452 005263 012630    INC     INTRAN(R3)  ;YES, IGNORE 1ST TRANSFER INTERRUPT.
1374 005456 000434          BR      EXTRAN
1375 005460 005763 012672    TST     HLDLO(R3)
1376 005464 001031          BNE     EXTRAN
1377 005466 005763 012734    TST     SENDSW(R3) ;IN SEND MODE?
1378 005472 001026          BNE     EXTRAN      ;YES, IGNORE INTERRUPT
1379 005474 005763 012776    TST     BOOTSW(R3) ;BOOTING OVER PROGRAMS?
1380 005500 001023          BNE     EXTRAN      ;YES, IGNORE INTERRUPT
1381 005502 005763 013422    TST     TRNSWH(R3) ;TRANSMITTER SW. SET?
1382 005506 001003          BNE     TRANOK      ;YES, LEGAL INTERRUPT
1383 005510 112720 000006    MOVB   #06,(R0)+   ;POST AS ILLEGAL TRANSMITTER INTERRUPT
1384 005514 000743          BR      RECERR      ;EXIT VIA RECEIVER ERROR REPORTER
1385 005516 122724 000014    TRANOK: CMPB    #EOP,(R4)+ ;LAST TRANSFER MADE?
1386 005522 001010          BNE     NXTCHR      ;NO, TRANSMIT NEXT CHAR.
1387 005524 005063 013422    CLR     TRNSWH(R3) ;YES, CLR SW.

```

```

1388 005530 005263 013570
1389 005534 046337 012146 013164
1390 005542 000402
1391 005544 111472 012214
1392 005550 000207
1393
1394
1395
1396
1397
1398
1399
1400 005552 005077 173422
1401 005556 010237 013162
1402 005562 017602 000000
1403 005566 062716 000002
1404 005572 010146
1405 005574 005037 013340
1406 005600 112201
1407 005602 105701
1408 005604 001444
1409 005606 122701 000012
1410 005612 001421
1411 005614 122701 000377
1412 005620 001431
1413 005622 122701 000014
1414 005626 001433
1415 005630 122701 000045
1416 005634 001410
1417 005636 004737 005042
1418 005642 005237 013340
1419 005646 022737 000100 013340
1420 005654 003351
1421 005656 005037 013340
1422 005662 012701 000015
1423 005666 004737 005042
1424 005672 012701 000012
1425 005676 004737 005042
1426 005702 000736
1427
1428 005704 012701 000336
1429 005710 004737 005042
1430 005714 000760
1431
1432 005716 013702 013162
1433 005722 012601
1434 005724 000002

```

```

INC XFERCT(R3) ;COUNT NO. OF BLOCKS TRANSFERED
BIC UNITNO(R3),ACTIVE
BR EXTRAN ;EXIT
NXTCHR: MOVB (R4),@XBUFO(R2) ;TRANSMIT NEXT CHAR.
EXTRAN: RTS PC

```

```

:*****
:MESSAGE PRINT ROUTINE, ENTERED VIA EMT DISPATCH HANDLER.
:ROUTINE PICKS UP CONTENTS OF THE 'PC' AND USES THIS AS
:THE ADDRESS OF MESSAGE TO BE TYPED.
:*****

```

```

TYPMES: CLR @PSW
MOV R2,TYPSV2 ;SAVE R2
MOV @($P),R2 ;GET THE MESSAGE ADDRESS FROM START
ADD #2,($P) ;SET UP STACK TO EXIT
MOV R1,-($P) ;SAVE R1
CLR PRTCNT
TYPERA: MOVB (R2)+,R1 ;PICK UP CHAR.
TSTB R1 ;TEST FOR NULL CHARACTER
BEQ TYPEXT ;IF SO, EXIT
CMPB #12,R1 ;TEST FOR LINE FEED
BEQ TYPECL ;TEST FOR START CODE
CMPB #377,R1 ;TEST FOR 'END OF PARAGRAPH'
BEQ TYPERB
CMPB #EOP,R1 ;TEST FOR 'END OF PARAGRAPH'
BEQ TYPEXT
CMPB #45,R1 ;TEST FOR '%'
BEQ TYPECL ;IF = TYPE 'CR-LF'
JSR PC,TYPEIT ;OUTPUT CHAR.
INC PRTCNT
CMP #64,PRTCNT ;LINE FULL?
BGT TYPERA ;NO, CHECK NEXT CHAR.
TYPECL: CLR PRTCNT
MOV #15,R1
JSR PC,TYPEIT ;TYPE 'CR'
MOV #12,R1
JSR PC,TYPEIT ;TYPE 'LF'
BR TYPERA
TYPERB: MOV #336,R1
JSR PC,TYPEIT ;PRINT '+'
BR TYPECL ;TYPE 'CR/LF'
TYPEXT: MOV TYPSV2,R2 ;RESTORE R2
MOV ($P)+,R1 ;RESTORE R1
RTI ;RETURN

```

```

1435
1436
1437
1438
1439
1440
1441 005726 005077 173246
1442 005732 010137 006052
1443 005736 010237 006054
1444 005742 017601 000000
1445 005746 062716 000002
1446 005752 012737 000006 013332
1447 005760 012737 000376 006056
1448 005766 000401
1449 005770 006111
1450 005772 006111
1451 005774 006111
1452 005776 111102
1453 006000 143702 006056
1454 006004 052702 000260
1455 006010 132777 000200 173170
1456 006016 100374
1457 006020 110277 173164
1458 006024 012737 000370 006056
1459 006032 005337 013332
1460 006036 001354
1461 006040 013701 006052
1462 006044 013702 006054
1463 006050 000002
1464
1465 006052 000000
1466 006054 000000
1467 006056 000376
1468
1469 006060 012604
1470 006062 012603
1471 006064 012602
1472 006066 012601
1473 006070 000002

```

```

;*****
;SUBROUTINE TO TYPEOUT A '6' DIGIT OCTAL NO. THE 'PC' CONTAINS
;THE ADDRESS OF 'WORD' TO BE TYPED
;*****
OCTPRT: CLR      @PSW
        MOV      R1,OCTSV1      ;SAVE R1
        MOV      R2,OCTSV2      ;SAVE R2
        MOV      @($P),R1      ;THE ADDRESS OF WORD TO BE TYPED
        ADD      #2,($P)        ;SET UP STACK TO EXIT
        MOV      #6,KSTOR3
        MOV      #376,MASK      ;MASK FOR FIRST BIT
        BR       .+4
MOVEIT: ROL      (R1)
        ROL      (R1)
        ROL      (R1)
        MOVB     (R1),R2
        BICB     MASK,R2
        BIS      #260,R2
        BITB     #200,@TPS
        BPL     .-6
        MOVB     R2,@TPB        ;PRINT CHAR.
        MOV      #370,MASK      ;MASK FOR NEXT '5' DIGITS
        DEC      KSTOR3
        BNE     MOVEIT
        MOV      OCTSV1,R1      ;RESTORE R1
        MOV      OCTSV2,R2
        RTI
OCTSV1: 0
OCTSV2: 0
MASK:   376
RSTORE: MOV      (SP)+,R4      ;RESTORE WORKING REG.'S FROM STACK
        MOV      (SP)+,R3
        MOV      (SP)+,R2
        MOV      (SP)+,R1
        RTI
;RETURN

```

```

1474 ;*****
1475 ;SUBROUTINE TO CONVERT 'N' 'BCD' WORDS SEPERATED VIA COMMA'S TO OCTAL
1476 ;*****
1477
1478 006072 012704 014770 BCDBIN: MOV #TTYBUF,R4 ;PICK UP BUFFER POINTER
1479 006076 012737 013106 013104 MOV #BCDBUF,BCDPTR ;SET UP BUFFER POINTER
1480 006104 005037 013102 CLR BCDCTR ;COUNT NO. OF ENTRIES IN BUFFER
1481 006110 005001 BCDPNO: CLR R1
1482 006112 105714 BCD&N1: TSTB (R4) ;END OF DATA?
1483 006114 001431 BEQ BCDEND ;YES, EXIT
1484 006116 112403 MOVB (R4)+,R3 ;SAVE IT
1485 006120 022703 000054 CMP #54,R3 ;CHAR. = TO ','?
1486 006124 001003 BNE BCD&N2 ;NO
1487 006126 004737 006160 JSR PC,BCDBN3 ;YES, SAVE CURRENT WORD
1488 006132 000766 BR BCD&N0 ;CONVERT NEXT WORD
1489 006134 042703 000360 BCD&N2: BIC #360,R3 ;STIPE NO. TO BCD
1490 006140 010102 MOV R1,R2 ;SAVE CURRENT TOTAL
1491 006142 006301 ASL R1 ;NX2
1492 006144 006301 ASL R1 ;NX4
1493 006146 006301 ASL R1 ;NX8
1494 006150 060201 ADD R2,R1 ;NX9
1495 006152 060201 ADD R2,R1 ;NX10
1496 006154 060301 ADD R3,R1 ;N+NEW NO.
1497 006156 000755 BR BCD&N1
1498
1499 006160 010177 004720 BCD&N3: MOV R1,BCDPTR ;SAVE WORD
1500 006164 062737 000002 013104 ADD #2,BCDPTR ;UPDATE POINTER
1501 006172 005237 013102 INC BCDCTR ;COUNT NO. OF ENTRIES IN BUFFER
1502 006176 000207 RTS PC ;RETURN
1503
1504 006200 004737 006160 BCDEND: JSR PC,BCDBN3 ;SAVE WORD
1505 006204 012737 013106 013104 MOV #BCDBUF,BCDPTR
1506 006212 000207 RTS PC ;EXIT

```

```

1507
1508
1509
1510
1511 006214 005077 172760
1512 006220 010237 006054
1513 006224 017602 000000
1514 006230 011202
1515 006232 042702 160000
1516 006236 062716 000002
1517 006242 010146
1518 006244 010446
1519 006246 012704 006324
1520 006252 012701 177777
1521 006256 005201
1522 006260 161402
1523 006262 100375
1524 006264 062402
1525 006266 052701 000260
1526 006272 004737 005042
1527 006276 005714
1528 006300 001364
1529 006302 012701 000240
1530 006306 004737 005042
1531 006312 013702 006054
1532 006316 012604
1533 006320 012601
1534 006322 000002
1535
1536 006324 001750
1537 006326 000144
1538 006330 000012
1539 006332 000001
1540 006334 000000

```

```

;*****
;SUBROUTINE TO PRINT DECIMAL VALUE
;*****
XBINDEC:CLR      QPSW
                MOV     R2,OCTSV2      ;SAVE R2
                MOV     Q(SP),R2      ;PICK UP ADDRESS OF VALUE
                MOV     (R2),R2      ;MOVE VALUE TO R2
                BIC     #160000,R2    ;CAN ONLY PRINT A '4' DIGIT NO.
                ADD     #2,(SP)      ;SET UP STACK TO EXIT
                MOV     R1,-(SP)
                MOV     R4,-(SP)
                MOV     #DECPTR,R4
TYPT1:  MOV     #-1,R1
TYPT2:  INC     R1
                SUB     (R4),R2
                BPL     TYPT2
                ADD     (R4)+,R2
DEC1:   BIS     #260,R1
                JSR     PC,TYPEIT
                TST     (R4)
                BNE     TYPT1
                MOV     #240,R1
                JSR     PC,TYPEIT
                MOV     OCTSV2,R2
                MOV     (SP)+,R4
                MOV     (SP)+,R1
                RTI

                ;DONE?
                ;NO
                ;YES, TYPE SPACE

                ;RESTORE R2

DECPTR: 1000.
        100.
        10.
        1.
        0

```

```

1541
1542
1543
1544
1545 006236 010046
1546 006340 010146
1547 006342 010246
1548 006344 010346
1549 006346 010446
1550 006350 010546
1551 006352 013746 000024
1552 006356 010637 013324
1553 006362 012737 006372 000024
1554 006370 000000
1555 006372 012777 000340 172600
1556 006400 005001
1557 006402 005201
1558 006404 001376
1559 006406 013706 013324
1560 006412 012637 000024
1561 006416 012605
1562 006420 012604
1563 006422 012603
1564 006424 012602
1565 006426 012601
1566 006430 012600
1567 006432 104000
1568 006434 011320
1569 006436 000137 002006
1570
1571
1572
1573
1574
1575
1576
1577 006442 011646
1578 006444 162716 000002
1579 006450 017616 000000
1580 006454 005716
1581 006456 001001
1582 006460 000000
1583 006462 006316
1584 006464 042716 177001
1585 006470 062716 006502
1586 006474 017616 000000
1587 006500 000136
1588
1589
1590
1591 006502 005552
1592 006504 005726
1593 006506 006214

```

```

;*****
;POWER FAIL HANDLER
;*****
PWRFAL: MOV    RO, -(SP)
        MOV    R1, -(SP)
        MOV    R2, -(SP)
        MOV    R3, -(SP)
        MOV    R4, -(SP)
        MOV    R5, -(SP)
        MOV    24, -(SP)
        MOV    SP, TEMP
        MOV    #PWRUP, @#24
        HALT
PWRUP:  MOV    #340, @PSW
        CLR    R1
        INC    R1
        BNE   -2
        MOV    TEMP, SP
        MOV    (SP)+, @#24
        MOV    (SP)+, R5
        MOV    (SP)+, R4
        MOV    (SP)+, R3
        MOV    (SP)+, R2
        MOV    (SP)+, R1
        MOV    (SP)+, R0
        PRINT
        MES4
        JMP    LDVECT

;*****
;EMT DISPATCH SERVICE ROUTINE
;ARGUMENT OF EMT IS EXTRACTED AND USED AS OFFSET TO OBTAIN POINTER
;TO THE SELECTED SUBROUTINE.
;*****
EMTSRV: MOV    (SP), -(SP)
        SUB    #2, (SP)
        MOV    @((SP), (SP))
        TST   (SP)
        BNE   EMTOK
        HALT
EMTOK:  ASL    (SP)
        BIC   #177001, (SP)
        ADD   #EMTTAB, (SP)
        MOV   @((SP), (SP))
        JMP   @((SP)+

;EMT DISPATCH TABLE
EMTTAB: TYPMES
        OCTPRT
        XBINDEC
;GET PC FOR TO RETURN
;PC OF EMT
;GET EMT
;IS EMT VALID?
;INVALID EMT
;MULTIPLY EMT ARG BY '2'
;CLEAR UNWANTED BITS
;POINTER TO SUBROUTINE ADDRESS
;SUBROUTINE ADDRESS
;GO TO SUBROUTINE
;SUBROUTINE TO PRINT ASCII MESSAGES.
;SUBROUTINE TO PRINT A '6' DIGIT OCTAL NO.
;SUBROUTINE TO CONVERT OCTAL TO BINARY & PRNT IT

```

M03

```

1594
1595
1596 006510 010146
1597 006512 010246
1598 006514 010346
1599 006516 010446
1600 006520 012701 000000
1601 006524 012702 000000
1602 006530 012703 000000
1603 006534 016304 013464
1604 006540 004737 005064
1605 006544 010463 013464
1606 006550 000137 006060
1607
1608
1609
1610 006554 010146
1611 006556 010246
1612 006560 010346
1613 006562 010446
1614 006564 012701 000764
1615 006570 012702 000020
1616 006574 012703 000002
1617 006600 016304 013464
1618 006604 004737 005064
1619 006610 010463 013464
1620 006614 000137 006060
1621
1622
1623
1624 006620 010146
1625 006622 010246
1626 006624 010346
1627 006626 010446
1628 006630 012701 001750
1629 006634 012702 000040
1630 006640 012703 000004
1631 006644 016304 013464
1632 006650 004737 005064
1633 006654 010463 013464
1634 006660 000137 006060
1635
1636
1637 006664 010146
1638 006666 010246
1639 006670 010346
1640 006672 010446
1641 006674 012701 002734
1642 006700 012702 000060
1643 006704 012703 000006
1644 006710 016304 013464
1645 006714 004737 005064
1646 006720 010463 013464
1647 006724 000137 006060
1648
1649

```

```

;SUBROUTINE ENTERED TO SERVICE LINE 0 RECEIVER INTERRUPTS.
RECVO0: MOV     R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
        MOV     R2,-(SP)
        MOV     R3,-(SP)
        MOV     R4,-(SP)
        MOV     #0,R1       ;BUFFER OFFSET IS '0'
        MOV     #0,R2       ;REGISTER OFFSET IS '0'
        MOV     #0,R3       ;ADDRESS OFFSET IS '0'
        MOV     SAVR4(R3),R4 ;SETUP BUFFER POINTER
        JSR     PC,RECVER   ;SERVICE RECEIVER INTERRUPT
        MOV     R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP     RSTORE     ;RESTORE WORKING REG.'S FROM STACK

;SUBROUTINE ENTERED TO SERVICE LINE 1 RECEIVER INTERRUPTS
RECVO1: MOV     R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
        MOV     R2,-(SP)
        MOV     R3,-(SP)
        MOV     R4,-(SP)
        MOV     #500.,R1    ;BUFFER OFFSET
        MOV     #16.,R2    ;REGISTER OFFSET
        MOV     #2.,R3     ;ADDRESS OFFSET
        MOV     SAVR4(R3),R4 ;SETUP BUFFER POINTER
        JSR     PC,RECVER   ;SERVICE RECEIVER INTERRUPT
        MOV     R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP     RSTORE     ;RESTORE WORKING REG.'S

;SUBROUTINE ENTERED TO SERVICE LINE 2 RECEIVER INTERRUPTS
RECVO2: MOV     R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
        MOV     R2,-(SP)
        MOV     R3,-(SP)
        MOV     R4,-(SP)
        MOV     #1000.,R1   ;BUFFER OFFSET
        MOV     #32.,R2    ;REGISTER OFFSET
        MOV     #4.,R3     ;ADDRESS OFFSET
        MOV     SAVR4(R3),R4 ;SETUP BUFFER POINTER
        JSR     PC,RECVER   ;SERVICE RECEIVER INTERRUPT
        MOV     R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP     RSTORE     ;RESTORE WORKING REG.'S

;SUBROUTINE ENTERED TO SERVICE LINE 3 RECEIVER INTERRUPTS
RECVO3: MOV     R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
        MOV     R2,-(SP)
        MOV     R3,-(SP)
        MOV     R4,-(SP)
        MOV     #1500.,R1  ;BUFFER OFFSET
        MOV     #48.,R2    ;REGISTER OFFSET
        MOV     #6.,R3     ;ADDRESS OFFSET
        MOV     SAVR4(R3),R4 ;SETUP BUFFER POINTER
        JSR     PC,RECVER   ;SERVICE RECEIVER INTERRUPT
        MOV     R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP     RSTORE     ;RESTORE WORKING REG.'S

;SUBROUTINE ENTERED TO SERVICE LINE 4 RECEIVER INTERRUPTS

```

```
1650 006730 010146
1651 006732 010246
1652 006734 010346
1653 006736 010446
1654 006740 012701 003720
1655 006744 012702 000100
1656 006750 012703 000010
1657 006754 016304 013464
1658 006760 004737 005064
1659 006764 010463 013464
1660 006770 000137 006060
```

```
RECVO4: MOV R1,-(SP) ;SAVE WORKING REG.'S ON THE STACK
        MOV R2,-(SP)
        MOV R3,-(SP)
        MOV R4,-(SP)
        MOV #2000,R1 ;BUFFER OFFSET
        MOV #64,R2 ;REGISTER OFFSET
        MOV #8,R3 ;ADDRESS OFFSET
        MOV SAVR4(R3),R4 ;SETUP BUFFER POINTER
        JSR PC,RECVER ;SERVICE RECEIVER INTERRUPT
        MOV R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP RSTORE ;RESTORE WORKING REG.'S
```

;SUBROUTINE ENTERED TO SERVICE LINE 5 RECEIVER INTERRUPTS

```
1661
1662
1663
1664 006774 010146
1665 006776 010246
1666 007000 010346
1667 007002 010446
1668 007004 012701 004704
1669 007010 012702 000120
1670 007014 012703 000012
1671 007020 016304 013464
1672 007024 004737 005064
1673 007030 010463 013464
1674 007034 000137 006060
```

```
RECVO5: MOV R1,-(SP) ;SAVE WORKING REG.'S ON THE STACK
        MOV R2,-(SP)
        MOV R3,-(SP)
        MOV R4,-(SP)
        MOV #2500,R1 ;BUFFER OFFSET
        MOV #80,R2 ;REGISTER OFFSET
        MOV #10,R3 ;ADDRESS OFFSET
        MOV SAVR4(R3),R4 ;SETUP BUFFER POINTER
        JSR PC,RECVER ;SERVICE RECEIVER INTERRUPT
        MOV R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP RSTORE ;RESTORE WORKING REG.'S
```

;SUBROUTINE ENTERED TO SERVICE LINE 6 RECEIVER INTERRUPTS

```
1675
1676
1677
1678 007040 010146
1679 007042 010246
1680 007044 010346
1681 007046 010446
1682 007050 012701 005670
1683 007054 012702 000140
1684 007060 012703 000014
1685 007064 016304 013464
1686 007070 004737 005064
1687 007074 010463 013464
1688 007100 000137 006060
```

```
RECVO6: MOV R1,-(SP) ;SAVE WORKING REG.'S ON THE STACK
        MOV R2,-(SP)
        MOV R3,-(SP)
        MOV R4,-(SP)
        MOV #3000,R1 ;BUFFER OFFSET
        MOV #96,R2 ;REGISTER OFFSET
        MOV #12,R3 ;ADDRESS OFFSET
        MOV SAVR4(R3),R4 ;SETUP BUFFER POINTER
        JSR PC,RECVER ;SERVICE RECEIVER INTERRUPT
        MOV R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP RSTORE ;RESTORE WORKING REG.'S
```

;SUBROUTINE ENTERED TO SERVICE LINE 7 RECEIVER INTERRUPTS

```
1689
1690
1691 007104 010146
1692 007106 010246
1693 007110 010346
1694 007112 010446
1695 007114 012701 006654
1696 007120 012702 000160
1697 007124 012703 000016
1698 007130 016304 013464
1699 007134 004737 005064
1700 007140 010463 013464
1701 007144 000137 006060
```

```
RECVO7: MOV R1,-(SP) ;SAVE WORKING REG.'S ON THE STACK
        MOV R2,-(SP)
        MOV R3,-(SP)
        MOV R4,-(SP)
        MOV #3500,R1 ;BUFFER OFFSET
        MOV #112,R2 ;REGISTER OFFSET
        MOV #14,R3 ;ADDRESS OFFSET
        MOV SAVR4(R3),R4 ;SETUP BUFFER POINTER
        JSR PC,RECVER ;SERVICE RECEIVER INTERRUPT
        MOV R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP RSTORE ;RESTORE WORKING REG.'S
```

;SUBROUTINE ENTERED TO SERVICE LINE 8 RECEIVER INTERRUPTS

```
1702
1703
1704 007150 010146
1705 007152 010246
```

```
RECVO8: MOV R1,-(SP) ;SAVE WORKING REG.'S ON THE STACK
        MOV R2,-(SP)
```



```

1706 007154 010346      MOV      R3,-(SP)
1707 007156 010446      MOV      R4,-(SP)
1708 007160 012701 007640      MOV      #4000,R1      ;BUFFER OFFSET
1709 007164 012702 000200      MOV      #128,R2      ;REGISTER OFFSET
1710 007170 012703 000020      MOV      #16,R3      ;ADDRESS OFFSET
1711 007174 016304 013464      MOV      SAVR4(R3),R4 ;SETUP BUFFER POINTER
1712 007200 004737 005064      JSR      PC,RECVER    ;SERVICE RECEIVER INTERRUPT
1713 007204 010463 013464      MOV      R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1714 007210 000137 006060      JMP      RSTORE      ;RESTORE WORKING REG.'S

:SUBROUTINE ENTERED TO SERVICE LINE 9 RECEIVER INTERRUPTS
1716
1718 007214 010146      RECV09: MOV      R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
1719 007216 010246      MOV      R2,-(SP)
1720 007220 010346      MOV      R3,-(SP)
1721 007222 010446      MOV      R4,-(SP)
1722 007224 012701 010624      MOV      #4500,R1      ;BUFFER OFFSET
1723 007230 012702 000220      MOV      #144,R2      ;REGISTER OFFSET
1724 007234 012703 000022      MOV      #18,R3      ;ADDRESS OFFSET
1725 007240 016304 013464      MOV      SAVR4(R3),R4 ;SETUP BUFFER POINTER
1726 007244 004737 005064      JSR      PC,RECVER    ;SERVICE RECEIVER INTERRUPT
1727 007250 010463 013464      MOV      R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1728 007254 000137 006060      JMP      RSTORE      ;RESTORE WORKING REG.'S

:SUBROUTINE ENTERED TO SERVICE LINE 10 RECEIVER INTERRUPTS
1729
1731 007260 010146      RECV10: MOV      R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
1732 007262 010246      MOV      R2,-(SP)
1733 007264 010346      MOV      R3,-(SP)
1734 007266 010446      MOV      R4,-(SP)
1735 007270 012701 011610      MOV      #5000,R1      ;BUFFER OFFSET
1736 007274 012702 000240      MOV      #160,R2      ;REGISTER OFFSET
1737 007300 012703 000024      MOV      #20,R3      ;ADDRESS OFFSET
1738 007304 016304 013464      MOV      SAVR4(R3),R4 ;SETUP BUFFER POINTER
1739 007310 004737 005064      JSR      PC,RECVER    ;SERVICE RECEIVER INTERRUPT
1740 007314 010463 013464      MOV      R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1741 007320 000137 006060      JMP      RSTORE      ;RESTORE WORKING REG.'S

:SUBROUTINE ENTERED TO SERVICE LINE 11 RECEIVER INTERRUPTS
1742
1744
1745 007324 010146      RECV11: MOV      R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
1746 007326 010246      MOV      R2,-(SP)
1747 007330 010346      MOV      R3,-(SP)
1748 007332 010446      MOV      R4,-(SP)
1749 007334 012701 012574      MOV      #5500,R1      ;BUFFER OFFSET
1750 007340 012702 000260      MOV      #176,R2      ;REGISTER OFFSET
1751 007344 012703 000026      MOV      #22,R3      ;ADDRESS OFFSET
1752 007350 016304 013464      MOV      SAVR4(R3),R4 ;SETUP BUFFER POINTER
1753 007354 004737 005064      JSR      PC,RECVER    ;SERVICE RECEIVER INTERRUPT
1754 007360 010463 013464      MOV      R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1755 007364 000137 006060      JMP      RSTORE      ;RESTORE WORKING REG.'S

:SUBROUTINE ENTERED TO SERVICE LINE 12 RECEIVER INTERRUPTS
1756
1757
1758 007370 010146      RECV12: MOV      R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
1759 007372 010246      MOV      R2,-(SP)
1760 007374 010346      MOV      R3,-(SP)
1761 007376 010446      MOV      R4,-(SP)

```

```

1762 007400 012701 013560      MOV      #6000, R1      ;BUFFER OFFSET
1763 007404 012702 000300      MOV      #192, R2      ;REGISTER OFFSET
1764 007410 012703 000030      MOV      #24, R3       ;ADDRESS OFFSET
1765 007414 016304 013464      MOV      SAVR4(R3), R4 ;SETUP BUFFER POINTER
1766 007420 004737 005064      JSR      PC, RECVER    ;SERVICE RECEIVER INTERRUPT
1767 007424 010463 013464      MOV      R4, SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1768 007430 000137 006060      JMP      RSTORE        ;RESTORE WORKING REG.'S
1769
1770      ;SUBROUTINE ENTERED TO SERVICE LINE 13 RECEIVER INTERRUPTS
1771
1772 007434 010146      RECV13: MOV      R1, -(SP)      ;SAVE WORKING REG.'S ON THE STACK
1773 007436 010246      MOV      R2, -(SP)
1774 007440 010346      MOV      R3, -(SP)
1775 007442 010446      MOV      R4, -(SP)
1776 007444 012701 014544      MOV      #6500, R1     ;BUFFER OFFSET
1777 007450 012702 000320      MOV      #208, R2      ;REGISTER OFFSET
1778 007454 012703 000032      MOV      #26, R3       ;ADDRESS OFFSET
1779 007460 016304 013464      MOV      SAVR4(R3), R4 ;SETUP BUFFER POINTER
1780 007464 004737 005064      JSR      PC, RECVER    ;SERVICE RECEIVER INTERRUPT
1781 007470 010463 013464      MOV      R4, SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1782 007474 000137 006060      JMP      RSTORE        ;RESTORE WORKING REG.'S
1783
1784      ;SUBROUTINE ENTERED TO SERVICE LINE 14 RECEIVER INTERRUPTS
1785
1786 007500 010146      RECV14: MOV      R1, -(SP)      ;SAVE WORKING REG.'S ON THE STACK
1787 007502 010246      MOV      R2, -(SP)
1788 007504 010346      MOV      R3, -(SP)
1789 007506 010446      MOV      R4, -(SP)
1790 007510 012701 015530      MOV      #7000, R1     ;BUFFER OFFSET
1791 007514 012702 000340      MOV      #224, R2      ;REGISTER OFFSET
1792 007520 012703 000034      MOV      #28, R3       ;ADDRESS OFFSET
1793 007524 016304 013464      MOV      SAVR4(R3), R4 ;SETUP BUFFER POINTER
1794 007530 004737 005064      JSR      PC, RECVER    ;SERVICE RECEIVER INTERRUPT
1795 007534 010463 013464      MOV      R4, SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1796 007540 000137 006060      JMP      RSTORE        ;RESTORE WORKING REG.'S
1797 007544 000002      RTI                    ;EXIT
1798
1799      ;SUBROUTINE ENTERED TO SERVICE LINE 15 RECEIVER INTERRUPTS
1800
1801 007546 010146      RECV15: MOV      R1, -(SP)      ;SAVE WORKING REG.'S ON THE STACK
1802 007550 010246      MOV      R2, -(SP)
1803 007552 010346      MOV      R3, -(SP)
1804 007554 010446      MOV      R4, -(SP)
1805 007556 012701 016514      MOV      #7500, R1     ;BUFFER OFFSET
1806 007562 012702 000360      MOV      #240, R2      ;REGISTER OFFSET
1807 007566 012703 000036      MOV      #30, R3       ;ADDRESS OFFSET
1808 007572 016304 013464      MOV      SAVR4(R3), R4 ;SETUP BUFFER POINTER
1809 007576 004737 005064      JSR      PC, RECVER    ;SERVICE RECEIVER INTERRUPT
1810 007602 010463 013464      MOV      R4, SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1811 007606 000137 006060      JMP      RSTORE        ;RESTORE WORKING REG.'S
1812
1813      ;SUBROUTINE ENTERED TO SERVICE LINE 0 TRANSMITTER
1814
1815 007612 010146      XFER00: MOV      R1, -(SP)      ;SAVE WORKING REG.'S ON THE STACK
1816 007614 010246      MOV      R2, -(SP)
1817 007616 010346      MOV      R3, -(SP)
1818 007620 010446      MOV      R4, -(SP)
1819 007622 012701 000000      MOV      #0, R1        ;BUFFER OFFSET IS '0'
1820 007626 012702 000000      MOV      #0, R2        ;REGISTER OFFSET IS '0'

```

```

1818 007632 012703 000000      MOV      #0,R3          ;ADDRESS OFFSET IS '0'
1819 007636 016304 013464      MOV      SAVR4(R3),R4
1820 007642 004737 005444      JSR      PC,TRNMIT
1821 007646 010463 013464      MOV      R4,SAVR4(R3)  ;SAVE BUFFER POINTER ON RETURN
1822 007652 000137 006060      JMP      RSTORE        ;RESTORE WORKING REG.'S
1823
1824      ;SUBROUTINE ENTERED TO SERVICE LINE 1 TRANSMITTER
1825
1826 007656 010146      XFER01: MOV      R1,-(SP)  ;SAVE WORKING REG.'S ON THE STACK
1827 007660 010246      MOV      R2,-(SP)
1828 007662 010346      MOV      R3,-(SP)
1829 007664 010446      MOV      R4,-(SP)
1830 007666 012701 000764      MOV      #500,R1      ;BUFFER OFFSET
1831 007672 012702 000020      MOV      #16,R2      ;REGISTER OFFSET
1832 007676 012703 000002      MOV      #2,R3       ;ADDRESS OFFSET
1833 007702 016304 013464      MOV      SAVR4(R3),R4
1834 007706 004737 005444      JSR      PC,TRNMIT
1835 007712 010463 013464      MOV      R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1836 007716 000137 006060      JMP      RSTORE        ;RESTORE WORKING REG.'S
1837
1838      ;SUBROUTINE ENTERED TO SERVICE LINE 2 TRANSMITTER
1839
1839 007722 010146      XFER02: MOV      R1,-(SP)  ;SAVE WORKING REG.'S ON THE STACK
1840 007724 010246      MOV      R2,-(SP)
1841 007726 010346      MOV      R3,-(SP)
1842 007730 010446      MOV      R4,-(SP)
1843 007732 012701 001750      MOV      #1000,R1    ;BUFFER OFFSET
1844 007736 012702 000040      MOV      #32,R2     ;REGISTER OFFSET
1845 007742 012703 000004      MOV      #4,R3      ;ADDRESS OFFSET
1846 007746 016304 013464      MOV      SAVR4(R3),R4
1847 007752 004737 005444      JSR      PC,TRNMIT
1848 007756 010463 013464      MOV      R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1849 007762 000137 006060      JMP      RSTORE        ;RESTORE WORKING REG.'S
1850
1851      ;SUBROUTINE ENTERED TO SERVICE LINE 3 TRANSMITTER
1852
1853 007766 010146      XFER03: MOV      R1,-(SP)  ;SAVE WORKING REG.'S ON THE STACK
1854 007770 010246      MOV      R2,-(SP)
1855 007772 010346      MOV      R3,-(SP)
1856 007774 010446      MOV      R4,-(SP)
1857 007776 012701 002734      MOV      #1500,R1   ;BUFFER OFFSET
1858 010002 012702 000060      MOV      #48,R2    ;REGISTER OFFSET
1859 010006 012703 000006      MOV      #6,R3     ;ADDRESS OFFSET
1860 010012 016304 013464      MOV      SAVR4(R3),R4
1861 010016 004737 005444      JSR      PC,TRNMIT
1862 010022 010463 013464      MOV      R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
1863 010026 000137 006060      JMP      RSTORE        ;RESTORE WORKING REG.'S
1864
1865      ;SUBROUTINE ENTERED TO SERVICE LINE 4 TRANSMITTER
1866
1866 010032 010146      XFER04: MOV      R1,-(SP)  ;SAVE WORKING REG.'S ON THE STACK
1867 010034 010246      MOV      R2,-(SP)
1868 010036 010346      MOV      R3,-(SP)
1869 010040 010446      MOV      R4,-(SP)
1870 010042 012701 003720      MOV      #2000,R1   ;BUFFER OFFSET
1871 010046 012702 000100      MOV      #64,R2    ;REGISTER OFFSET
1872 010052 012703 000010      MOV      #8,R3     ;ADDRESS OFFSET
1873 010056 016304 013464      MOV      SAVR4(R3),R4

```

```

1874 010062 004737 005444      JSR    PC,TRNMIT
1875 010066 010463 013464      MOV    R4,SAVR4(R3)      ;SAVE BUFFER POINTER ON RETURN
1876 010072 000137 006060      JMP    RSTORE           ;RESTORE WORKING REG.'S
1877
1878                               ;SUBROUTINE ENTERED TO SERVICE LINE 5 TRANSMITTER
1879
1880 010076 010146                XFER05: MOV    R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
1881 010100 010246                MOV    R2,-(SP)
1882 010102 010346                MOV    R3,-(SP)
1883 010104 010446                MOV    R4,-(SP)
1884 010106 012701 004704      MOV    #2500.,R1         ;BUFFER OFFSET
1885 010112 012702 000120      MOV    #80.,R2           ;REGISTER OFFSET
1886 010116 012703 000012      MOV    #10.,R3          ;ADDRESS OFFSET
1887 010122 016304 013464      MOV    SAVR4(R3),R4
1888 010126 004737 005444      JSR    PC,TRNMIT
1889 010132 010463 013464      MOV    R4,SAVR4(R3)     ;SAVE BUFFER POINTER ON RETURN
1890 010136 000137 006060      JMP    RSTORE           ;RESTORE WORKING REG.'S
1891
1892                               ;SUBROUTINE ENTERED TO SERVICE LINE 6 TRANSMITTER
1893 010142 010146                XFER06: MOV    R1,-(SP)     ;SAVE WORKING REG.'S ON THE STACK
1894 010144 010246                MOV    R2,-(SP)
1895 010146 010346                MOV    R3,-(SP)
1896 010150 010446                MOV    R4,-(SP)
1897 010152 012701 005670      MOV    #3000.,R1        ;BUFFER OFFSET
1898 010156 012702 000140      MOV    #96.,R2          ;REGISTER OFFSET
1899 010162 012703 000014      MOV    #12.,R3         ;ADDRESS OFFSET
1900 010166 016304 013464      MOV    SAVR4(R3),R4
1901 010172 004737 005444      JSR    PC,TRNMIT
1902 010176 010463 013464      MOV    R4,SAVR4(R3)     ;SAVE BUFFER POINTER ON RETURN
1903 010202 000137 006060      JMP    RSTORE           ;RESTORE WORKING REG.'S
1904
1905                               ;SUBROUTINE ENTERED TO SERVICE LINE 7 TRANSMITTER
1906
1907 010206 010146                XFER07: MOV    R1,-(SP)     ;SAVE WORKING REG.'S ON THE STACK
1908 010210 010246                MOV    R2,-(SP)
1909 010212 010346                MOV    R3,-(SP)
1910 010214 010446                MOV    R4,-(SP)
1911 010216 012701 006654      MOV    #3500.,R1        ;BUFFER OFFSET
1912 010222 012702 000160      MOV    #112.,R2         ;REGISTER OFFSET
1913 010226 012703 000016      MOV    #14.,R3         ;ADDRESS OFFSET
1914 010232 016304 013464      MOV    SAVR4(R3),R4
1915 010236 004737 005444      JSR    PC,TRNMIT
1916 010242 010463 013464      MOV    R4,SAVR4(R3)     ;SAVE BUFFER POINTER ON RETURN
1917 010246 000137 006060      JMP    RSTORE           ;RESTORE WORKING REG.'S
1918                               ;SUBROUTINE ENTERED TO SERVICE LINE 8 TRANSMITTER
1919
1920 010252 010146                XFER08: MOV    R1,-(SP)     ;SAVE WORKING REG.'S ON THE STACK
1921 010254 010246                MOV    R2,-(SP)
1922 010256 010346                MOV    R3,-(SP)
1923 010260 010446                MOV    R4,-(SP)
1924 010262 012701 007640      MOV    #4000.,R1        ;BUFFER OFFSET
1925 010266 012702 000200      MOV    #128.,R2         ;REGISTER OFFSET
1926 010272 012703 000020      MOV    #16.,R3         ;ADDRESS OFFSET
1927 010276 016304 013464      MOV    SAVR4(R3),R4
1928 010302 004737 005444      JSR    PC,TRNMIT
1929 010306 010463 013464      MOV    R4,SAVR4(R3)     ;SAVE BUFFER POINTER ON RETURN

```

```

1930 010312 000137 006060          JMP      RSTORE          ;RESTORE WORKING REG.'S
1931
1932          ;SUBROUTINE ENTERED TO SERVICE LINE 9 TRANSMITTER
1933
1934 010316 010146          XFER09: MOV      R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
1935 010320 010246          MOV      R2,-(SP)
1936 010322 010346          MOV      R3,-(SP)
1937 010324 010446          MOV      R4,-(SP)
1938 010326 012701 010624          MOV      #4500.,R1          ;BUFFER OFFSET
1939 010332 012702 000220          MOV      #144.,R2          ;REGISTER OFFSET
1940 010336 012703 000022          MOV      #18.,R3           ;ADDRESS OFFSET
1941 010342 016304 013464          MOV      SAVR4(R3),R4
1942 010346 004737 005444          JSR      PC,TRNMIT
1943 010352 010463 013464          MOV      R4,SAVR4(R3)      ;SAVE BUFFER POINTER ON RETURN
1944 010356 000137 006060          JMP      RSTORE          ;RESTORE WORKING REG.'S
1945          ;SUBROUTINE ENTERED TO SERVICE LINE 10 TRANSMITTER
1946
1947 010362 010146          XFER10: MOV      R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
1948 010364 010246          MOV      R2,-(SP)
1949 010366 010346          MOV      R3,-(SP)
1950 010370 010446          MOV      R4,-(SP)
1951 010372 012701 011610          MOV      #5000.,R1          ;BUFFER OFFSET
1952 010376 012702 000240          MOV      #160.,R2          ;REGISTER OFFSET
1953 010402 012703 000024          MOV      #20.,R3           ;ADDRESS OFFSET
1954 010406 016304 013464          MOV      SAVR4(R3),R4
1955 010412 004737 005444          JSR      PC,TRNMIT
1956 010416 010463 013464          MOV      R4,SAVR4(R3)      ;SAVE BUFFER POINTER ON RETURN
1957 010422 000137 006060          JMP      RSTORE          ;RESTORE WORKING REG.'S
1958
1959          ;SUBROUTINE ENTERED TO SERVICE LINE 11 TRANSMITTER
1960
1961 010426 010146          XFER11: MOV      R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
1962 010430 010246          MOV      R2,-(SP)
1963 010432 010346          MOV      R3,-(SP)
1964 010434 010446          MOV      R4,-(SP)
1965 010436 012701 012574          MOV      #5500.,R1          ;BUFFER OFFSET
1966 010442 012702 000260          MOV      #176.,R2          ;REGISTER OFFSET
1967 010446 012703 000026          MOV      #22.,R3           ;ADDRESS OFFSET
1968 010452 016304 013464          MOV      SAVR4(R3),R4
1969 010456 004737 005444          JSR      PC,TRNMIT
1970 010462 010463 013464          MOV      R4,SAVR4(R3)      ;SAVE BUFFER POINTER ON RETURN
1971 010466 000137 006060          JMP      RSTORE          ;RESTORE WORKING REG.'S
1972          ;SUBROUTINE ENTERED TO SERVICE LINE 12 TRANSMITTER
1973
1974 010472 010146          XFER12: MOV      R1,-(SP)      ;SAVE WORKING REG.'S ON THE STACK
1975 010474 010246          MOV      R2,-(SP)
1976 010476 010346          MOV      R3,-(SP)
1977 010500 010446          MOV      R4,-(SP)
1978 010502 012701 013560          MOV      #6000.,R1          ;BUFFER OFFSET
1979 010506 012702 000300          MOV      #192.,R2          ;REGISTER OFFSET
1980 010512 012703 000030          MOV      #24.,R3           ;ADDRESS OFFSET
1981 010516 016304 013464          MOV      SAVR4(R3),R4
1982 010522 004737 005444          JSR      PC,TRNMIT
1983 010526 010463 013464          MOV      R4,SAVR4(R3)      ;SAVE BUFFER POINTER ON RETURN
1984 010532 000137 006060          JMP      RSTORE          ;RESTORE WORKING REG.'S
1985

```

```

1986
1987
1988 010536 010146
1989 010540 010246
1990 010542 010346
1991 010544 010446
1992 010546 012701 014544
1993 010552 012702 000320
1994 010556 012703 000032
1995 010562 016304 013464
1996 010566 004737 005444
1997 010572 010463 013464
1998 010576 000137 006060
1999
2000
2001 010602 010146
2002 010604 010246
2003 010606 010346
2004 010610 010446
2005 010612 012701 015530
2006 010616 012702 000340
2007 010622 012703 000034
2008 010626 016304 013464
2009 010632 004737 005444
2010 010636 010463 013464
2011 010642 000137 006060
2012
2013
2014
2015 010646 010146
2016 010650 010246
2017 010652 010346
2018 010654 010446
2019 010656 012701 016514
2020 010660 012702 000360
2021 010666 012703 000036
2022 010672 016304 013464
2023 010676 004737 005444
2024 010702 010463 013464
2025 010706 000137 006060
    
```

;SUBROUTINE ENTERED TO SERVICE LINE 13 TRANSMITTER

```

XFER13: MOV R1,-(SP) ;SAVE WORKING REG.'S ON THE STACK
        MOV R2,-(SP)
        MOV R3,-(SP)
        MOV R4,-(SP)
        MOV #6500,R1 ;BUFFER OFFSET
        MOV #208,R2 ;REGISTER OFFSET
        MOV #26,R3 ;ADDRESS OFFSET
        MOV SAVR4(R3),R4
        JSR PC,TRNMIT
        MOV R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP RSTORE ;RESTORE WORKING REG.'S
    
```

;SUBROUTINE ENTERED TO SERVICE LINE 14 TRANSMITTER

```

XFER14: MOV R1,-(SP) ;SAVE WORKING REG.'S ON THE STACK
        MOV R2,-(SP)
        MOV R3,-(SP)
        MOV R4,-(SP)
        MOV #7000,R1 ;BUFFER OFFSET
        MOV #224,R2 ;REGISTER OFFSET
        MOV #28,R3 ;ADDRESS OFFSET
        MOV SAVR4(R3),R4
        JSR PC,TRNMIT
        MOV R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP RSTORE ;RESTORE WORKING REG.'S
    
```

;SUBROUTINE ENTERED TO SERVICE LINE 15 TRANSMITTER

```

XFER15: MOV R1,-(SP) ;SAVE WORKING REG.'S ON THE STACK
        MOV R2,-(SP)
        MOV R3,-(SP)
        MOV R4,-(SP)
        MOV #7500,R1 ;BUFFER OFFSET
        MOV #240,R2 ;REGISTER OFFSET
        MOV #30,R3 ;ADDRESS OFFSET
        MOV SAVR4(R3),R4
        JSR PC,TRNMIT
        MOV R4,SAVR4(R3) ;SAVE BUFFER POINTER ON RETURN
        JMP RSTORE ;RESTORE WORKING REG.'S
    
```

```

2026      :OVERLAY VECTOR AREA
2027 010712 012701 000210  OVRLAY: MOV    #210,%1      ;GET DL11-E VECTOR BASE ADDRESS
2028 010716 012702 000212      MOV    #212,%2
2029 010722 012703 000004      MOV    #4,%3
2030 010726 010221      OVRLYA: MOV    %2,(1)+      ;LOAD VECTOR WITH IOT ERROR TRAP
2031 010730 010321      MOV    %3,(1)+
2032 010732 062702 000004      ADD    #4,%2
2033 010736 020127 001000      CMP    %1,#1000      ;ALL VECTORS BEEN LOADED
2034 010742 001401      BEQ    OVRLYB
2035 010744 000770      BR     OVRLYA
2036 010746 000207  OVRLYB: RTS    7          ;EXIT
2037
2038      ;MAPVEC - MAP VECTOR OR REPORT ERROR DEPENDING ON FMAP FLAG
2039
2040 010750 011637 013346  MAPVEC: MOV    (SP),TOPC
2041 010754 022626      POP2SP
2042 010756 011637 013350      MOV    (SP),FROMPC
2043 010762 005737 013352      TST    FMAP          ;MAPPING?
2044 010766 001427      BEQ    ERTRAP        ;NO REPORT ERROR
2045 010770 162737 000002 013346  SUB    #2,TOPC        ;SETUP TO LOAD TRAN'S ADDRESSES
2046 010776 013711 013346      MOV    TOPC,(R1)     ;STORE BR ADDRESS
2047 011002 162737 000002 013346  SUB    #2,TOPC
2048 011010 013741 013346      MOV    TOPC,-(R1)    ;STORE VECTOR
2049 011014 162737 000002 013346  SUB    #2,TOPC        ;SET UP TO LOAD RECEIVER ADDRESSES
2050 011022 013741 013346      MOV    TOPC,-(R1)    ;STORE BR ADDRESS
2051 011026 162737 000002 013346  SUB    #2,TOPC
2052 011034 013741 013346      MOV    TOPC,-(R1)
2053 011040 005037 013352      CLR    FMAP
2054 011044 000002      RTI
2055
2056      :ERROR TRAP HANDLER, ENTERED ON ILLEGAL TRAPS
2057
2058 011046 104000  ERTRAP: PRINT
2059 011050 011361      MESS
2060 011052 162737 000004 013346  SUB    #4,TOPC        ;TEXT 'ILLEGAL TRAP TO'
2061 011060 104001      PRTOCT
2062 011062 013346      TOPC                ; TYPE 'PC' TRAPPED TOO
2063 011064 104000      PRINT
2064 011066 011436      MESS                ;TEXT 'FROM'
2065 011070 162737 000002 013350  SUB    #2,FROMPC
2066 011076 104001      PRTOCT
2067 011100 013350      FROMPC              ;TYPE WHERE IT TRAPPED FROM
2068 011102 000000      HALT                ;WAIT FOR FIX
2069 011104 000137 001314  JMP    START          ;RE-START TEST
2070

```

2071	011110	000			.BYTE	
2072					;MESSAGES	
2073						
2074	011111	045	050045	050104	TITLE: .ASCII	;:%PDP-11/VT20 HOST DIAGNOSTIC PROGRAM%;
2075	011116	030455	027451	052126		
2076	011124	030062	044040	051517		
2077	011132	020124	044504	043501		
2078	011140	047516	052123	041511		
2079	011146	050040	047522	051107		
2080	011154	046501	045			
2081	011157	115	044501	042116	.ASCII	;MAINDEC-11-DZVTE-B-PB;
2082	011164	041505	030455	026461		
2083	011172	055104	052126	026505		
2084	011200	026502	041120			
2085	011204	020040	030440	027460	.ASCIZ	; 10/17/74.%;
2086	011212	033461	033457	027064		
2087	011220	000045				
2088						
2089	011222	047045	020117	046104	MES1: .ASCIZ	;%NO DL11 ADDRESSES PRESENT?;
2090	011230	030461	040440	042104		
2091	011236	042522	051523	051505		
2092	011244	050040	042522	042523		
2093	011252	052116	037477	000		
2094						
2095	011257	045	047516	051040	MES2: .ASCIZ	;%NO RESPONSE FROM DEVICE ;
2096	011264	051505	047520	051516		
2097	011272	020105	051106	046517		
2098	011300	042040	053105	041511		
2099	011306	020105	000040			
2100						
2101	011312	044514	042516	000040	MES3: .ASCIZ	;LINE ;
2102	011320	022445	042522	047503	MES4: .ASCIZ	;:%RECOVERED FROM POWER FAILURE.%;
2103	011326	042526	042522	020104		
2104	011334	051106	046517	050040		
2105	011342	053517	051105	043040		
2106	011350	044501	052514	042522		
2107	011356	022456	000			
2108						
2109	011361	045	046111	042514	MES5: .ASCIZ	;%ILLEGAL TRAP TO ;
2110	011366	040507	020114	051124		
2111	011374	050101	052040	020117		
2112	011402	000				
2113						
2114						
2115	011403	045	044506	051522	MFIAD: .ASCIZ	;%FIRST DL11 LINE ADDRESS? ;
2116	011410	020124	046104	030461		
2117	011416	046040	047111	020105		
2118	011424	042101	051104	051505		
2119	011432	037523	000040			
2120						
2121	011436	043040	047522	020115	MES6: .ASCIZ	; FROM ;
2122	011444	000				
2123						
2124	011445	045	044514	052123	MES7: .ASCIZ	;%LIST OF ACTIVE DL11 RECEIVERS;
2125	011452	047440	020106	041501		
2126	011460	044524	042526	042040		

2127	011466	030514	020061	042522	
2128	011474	042503	053111	051105	
2129	011502	000123			
2130					
2131	011504	046045	047111	020105	MES8: .ASCIZ ;%LINE IN OUT OR. PAR. FRAM REC. TRAN ST. HELD;
2132	011512	047111	020040	047440	
2133	011520	052125	020040	051117	
2134	011526	020056	050040	051101	
2135	011534	020056	051106	046501	
2136	011542	051040	041505	020056	
2137	011550	051124	047101	051440	
2138	011556	027124	020040	042510	
2139	011564	042114	000		
2140					
2141					
2142	011567	040	044504	043501	MES9: .ASCIZ ; DIAG. MODE ENABLED.;
2143	011574	020056	047515	042504	
2144	011602	042440	040516	046102	
2145	011610	042105	000056		
2146	011614	051045	040505	042504	MES11: .ASCIZ ;%READER DEVICE ADDRESS? ;
2147	011622	020122	042504	044526	
2148	011630	042503	040440	042104	
2149	011636	042522	051523	020077	
2150	011644	000			
2151	011645	126	041505	047524	MES13: .ASCIZ ;VECTOR ADDRESS? ;
2152	011652	020122	042101	051104	
2153	011660	051505	037523	000040	
2154	011666	044440	046114	043505	CODE00: .ASCIZ ; ILLEGAL RECVR. INTERRUPT.%.;
2155	011674	046101	051040	041505	
2156	011702	051126	020056	047111	
2157	011710	042524	051122	050125	
2158	011716	027124	027045	000	
2159					
2160	011723	040	053117	051105	CODE01: .ASCIZ ; OVERRUN ERROR.%.;
2161	011730	052522	020116	051105	
2162	011736	047522	027122	027045	
2163	011744	000			
2164					
2165	011745	040	051106	046501	CODE02: .ASCIZ ; FRAMING ERROR.%.;
2166	011752	047111	020107	051105	
2167	011760	047522	027122	027045	
2168	011766	000			
2169					
2170	011767	040	040520	044522	CODE03: .ASCIZ ; PARITY ERROR.%.;
2171	011774	054524	042440	051122	
2172	012002	051117	022456	000056	
2173					
2174	012010	044440	046114	043505	CODE04: .ASCIZ ; ILLEGAL START CODE.%.;
2175	012016	046101	051440	040524	
2176	012024	052122	041440	042117	
2177	012032	027105	027045	000	
2178					
2179	012037	040	046111	042514	CODE05: .ASCIZ ; ILLEGAL READER INTERRUPT.%.;
2180	012044	040507	020114	042522	
2181	012052	042101	051105	044440	
2182	012060	052116	051105	052522	

```

2183 012066 052120 022456 000056
2184
2185 012074 044440 046114 043505
2186 012102 046101 052040 040522
2187 012110 051516 020056 047111
2188 012116 042524 051122 050125
2189 012124 027124 027045 000
2190
2191 012131 040 000
2192 012133 045 000
2193 012135 045 000056
2194 012140 000134
2195
2196
2197
2198 012142 000000
2199 012144 000000
2200 012146 000001
2201 012150 000002
2202 012152 000004
2203 012154 000010
2204 012156 000020
2205 012160 000040
2206 012162 000100
2207 012164 000200
2208 012166 000400
2209 012170 001000
2210 012172 002000
2211 012174 004000
2212 012176 010000
2213 012200 020000
2214 012202 040000
2215 012204 100000
2216
2217
2218
2219 012206 000000
2220 012210 000000
2221 012212 000000
2222 012214 000000
2223 012216 000000
2224 012220 000000
2225 012222 000000
2226 012224 000000
2227 012626
2228
2229
2230
2231
2232 012626 000000
2233 012630 000000
2234 012672
2235 012672 000000
2236 012734
2237 012734 000000
2238 012776

```

CODE06: .ASCIZ ; ILLEGAL TRANS. INTERRUPT.%;

```

SPACE: .ASCIZ ; ;
CRLF: .ASCIZ ;%;
DOT: .ASCIZ ;%.;
SLASH: .ASCIZ ;\;
.EVEN

```

```

MONFLG: 0
MEMSIZ: 0
UNITNO: 1
          2
          4
          10
          20
          40
          100
          200
          400
          1000
          2000
          4000
          10000
          20000
          40000
          100000

```

;DL11 ACTIVE DEVICE ADDRESSES LOADED BY MAPPER

```

RCSRO: 0
RBUFO: 0
XCSRO: 0
XBUFO: 0
RVTRO: 0
RLVLO: 0
XVTRO: 0
XLVLO: 0
.=.+256.

```

;SAVE FOR POSSIBLE DL11 ADDRESSES

;ADDRESSES AND CONSTANTS

```

HERE: 0
INTRAN: 0
HLLO: 0
SENDSW: 0
.=.+40
.=.+40
.=.+40

```

;SOFTWARE SW. FOR LEGAL TRANSMITTER INTERRUPTS

2239	012776	000000	BOOTSW: 0	;SET WHEN BOOTING PROGRAM
2240		013040	. = .+40	
2241	013040	000000	ERRCTR: 0	
2242		013102	. = .+40	
2243	013102	000000	BCDCTR: 0	
2244	013104	000000	BCDPTR: 0	
2245	013106	000000	BCDBUF: 0	
2246		013150	. = .+40	
2247	013150	000000	RCHAR: 0	
2248	013152	000000	RSTAT: 0	
2249	013154	000000	SYSSWH: 0	
2250	013156	000000	TTYPTR: 0	
2251	013160	000000	CNTR: 0	
2252	013162	000000	TYPV2: 0	
2253	013164	000000	ACTIVE: 0	
2254	013166	000000	RUBSWH: 0	
2255	013170	000000	SCHAR: 0	
2256	013172	000000	LINNO: 0	
2257	013174	000000	ADDOFF: 0	
2258	013176	000000	REGOFF: 0	
2259	013200	000000	BUFOFF: 0	
2260	013202	000000	CONSFL: 0	
2261	013204	000000	SAVCHR: 0	
2262	013206	000000	DEVADR: 0	
2263	013210	000000	PRTFLG: 0	
2264	013212	000000	BOOTP1: 0	
2265	013214	000000	BOOTP2: 0	
2266	013216	000000	BOOTAD: 0	
2267	013220	000000	BOOTOF: 0	;CONTAINS THE BOOT OFFSET NO.
2268		013262	. = .+40	
2269	013262	000000	BOOTLN: 0	;CONTAINS THE BOOT ADDR. LINE NO.
2270		013324	. = .+40	
2271	013324	000000	TEMP: 0	;TEMPORARY STORAGE
2272	013326	000000	KSTOR1: 0	;PERMANENT STORAGE
2273	013330	000000	KSTOR2: 0	;PERMANENT STORAGE
2274	013332	000000	KSTOR3: 0	
2275	013334	000000	KSTOR4: 0	
2276	013336	000000	RMODE: 0	
2277	013340	000000	PRTCNT: 0	
2278	013342	000000	ERRFLG: 0	;SOFTWARE SW.
2279	013344	000000	PRTSWH: 0	;SOFTWARE SW.
2280	013346	000000	TOPC: 0	
2281	013350	000000	FROMPC: 0	
2282	013352	000000	FMAP: 0	;SOFTWARE SW. SET IF MAPPING
2283	013354	000000	RECSWH: 0	;RECEIVER SOFTWARE SW. SET=RECEIVING
2284		013416	. = .+40	;ALLOCATE FOR '16' RECEIVERS
2285			;READER DEVICE ADDRESSES	
2286				
2287	013416	000000	RCSR: 0	
2288	013420	000000	RDBR: 0	
2289				
2290	013422	000000	TRNSWH: 0	;TRANSMITTER SOFTWARE SW, SET=TRANSMITTING
2291		013464	. = .+40	
2292	013464	000000	SAVR4: 0	;BUFFER AREA TO SAVE CONTENTS OF 'R4'
2293		013526	. = .+40	;ALLOCATE '1' LOCATION FOR EACH UNIT.
2294	013526	000000	RECNR: 0	

2295		013570
2296	013570	000000
2297		013632
2298	013632	000000
2299		013674
2300	013674	000000
2301		013736
2302	013736	000000
2303		014000
2304	014000	000000
2305		014042
2306	014042	000000
2307		014104
2308	014104	000000
2309		014146
2310	014146	000000
2311		014770
2312	014770	000000
2313		015032
2314	015032	000000
2315		000001

XFERCT:	0	.=. +40
REC:	0	.=. +40
OR:	0	.=. +40
FRM:	0	.=. +40
PAR:	0	.=. +40
ST:	0	.=. +40
TRN:	0	.=. +40
ERRBUF:	0	.=. +400.
TTYBUF:	0	.=. +40
BUFFER:	0	.END

;ERROR BUFFER STORAGE AREA.

;DATA BUFFER STORAGE AREA.

TAGD	003476	1014	1019#											
TEMP	013324	664*	665	667*	701*	702	704*	1126*	1127	1148*	1149	1150*	1151	1152*
		1153	1154*	1155	1156*	1157	1158*	1159	1160*	1161	1162*	1163	1164*	1165
		1552*	1559	2271#										
TITLE	011111	606	2074#											
TKB	001204	535#	831											
TKS	001202	534#	589*	747*	1106*									
TOPC	013346	2040*	2045*	2046	2047*	2048	2049*	2050	2051*	2052	2060*	2062	2280#	
TPB	001210	537#	809*	816*	1274*	1457*								
TPS	001206	536#	805	1272	1455									
TRANOK	005516	1382	1385#											
TRN	014104	1160	2308#											
TRNMIT	005444	1371#	1820	1834	1847	1861	1874	1888	1901	1915	1928	1942	1955	1969
		1982	1996	2009	2023									
TRNSWH	013422	948*	1292	1353*	1381	1397*	2290#							
TSTERR	002240	757#	764	781	788	791	821							
TSTVA	001766	676	683#											
TTYBUF	014770	1104*	1105	1168	1478	2312#								
TTYPTR	013156	1043*	1044*	1045*	1057*	1058	1059*	1105*	2250#					
TYPECL	005656	1410	1416	1421#	1430									
TYPEIT	005042	836	843	846	879	989	1035	1060	1083	1271#	1273	1417	1423	1425
		1429	1526	1530										
TYPERA	005600	1406#	1420	1426										
TYPERB	005704	1412	1428#											
TYPEXT	005716	1408	1414	1432#										
TYPMES	005552	1400#	1591											
TYPSV2	013162	1401*	1432	2252#										
TYPT1	006252	1520#	1528											
TYPT2	006256	1521#	1523											
UNITNO	012146	949	1336	1354	1389	2200#								
XBINDE	006214	1511#	1593											
XBUFO	012214	950*	1000*	1249*	1357*	1391*	2222#							
XCSRO	012212	662	698	2221#										
XFERCT	013570	1150	1388*	2296#										
XFER00	007612	638	1812#											
XFER01	007656	1826#												
XFER02	007722	1839#												
XFER03	007766	1853#												
XFER04	010032	1866#												
XFER05	010076	1880#												
XFER06	010142	1893#												
XFER07	010206	1907#												
XFER08	010252	1920#												
XFER09	010316	1934#												
XFER10	010362	1947#												
XFER11	010426	1961#												
XFER12	010472	1974#												
XFER13	010536	1988#												
XFER14	010602	2001#												
XFER15	010646	2015#												
XLVLO	012224	663	2226#											
XVTR0	012222	2225#												
.	= 015034	513#	518	519#	526#	529#	532#	594	602	622	806	948	853	859
		1372	1448	1456	1558	2227#	2234#	2236#	2238#	2240#	2242#	2246#	2268#	2270#
		2284#	2291#	2293#	2295#	2297#	2299#	2301#	2303#	2305#	2307#	2309#	2311#	2313#

H05

ADD	624	654	684	685	711	712	713	714	715	925	926	1005	1006	1007	1017
	1019	1020	1050	1077	1085	1086	1087	1094	1167	1195	1196	1210	1242	1250	1257
	1338	1356	1360	1403	1445	1494	1495	1496	1500	1516	1524	1585	2032		
ASL	786	1047	1048	1049	1491	1492	1493	1583							
ASR	1063	1064	1065	1145											
ASRB	1245	1362													
BEG	666	703	788	815	819	839	851	862	864	866	868	876	899	909	945
	994	997	1014	1030	1032	1037	1079	1130	1199	1208	1293	1303	1311	1327	1329
	1343	1346	1408	1410	1412	1414	1416	1483	2034	2044					
BGE	841	1081													
BGT	1420														
BIC	670	683	785	832	1046	1062	1336	1389	1489	1515	1584				
BICB	1453														
BIS	649	651	653	669	671	709	710	747	845	949	1106	1354	1454	1525	
BIT	645	1129	1302	1310											
BITB	1455														
BLE	626	952	969	984	1041	1173									
BLOS	759														
BLT	1039	1296	1306	1314	1320	1332									
BMI	773	1248													
BNE	602	604	610	628	633	646	676	720	733	744	761	775	808	835	848
	853	856	859	870	873	882	928	999	1009	1022	1054	1089	1110	1169	1171
	1203	1241	1252	1259	1325	1350	1372	1376	1378	1380	1382	1386	1460	1486	1528
	1558	1591													
BPL	779	806	1273	1301	1456	1523									
BR	594	622	636	655	686	716	764	781	791	812	821	837	880	901	902
	911	915	942	954	972	973	986	987	1084	1175	1201	1246	1254	1298	1308
	1316	1322	1334	1341	1358	1374	1384	1390	1426	1430	1448	1488	1497	2035	
CLR	599	600	614	616	668	718	731	742	749	780	810	885	914	921	922
	923	946	947	950	963	964	990	991	992	1011	1015	1034	1073	1074	1075
	1095	1096	1102	1103	1104	1107	1120	1122	1131	1140	1216	1218	1234	1255	1256
	1339	1340	1351	1387	1400	1405	1421	1441	1480	1481	1511	1556	2053		
CLRB	1045	1059	1297												
CMP	601	625	627	719	732	743	758	787	818	834	838	840	927	1008	1021
	1029	1080	1109	1295	1305	1313	1319	1331	1419	1485	2033				
CMPB	807	814	847	850	852	858	861	863	865	867	869	872	875	996	1036
	1038	1040	1328	1342	1349	1385	1409	1411	1413	1415					
COM	908														
DEC	667	704	772	778	877	898	951	968	983	1057	1088	1170	1172	1198	1251
	1258	1459													
EMT	542														
HALT	790	967	1554	1582	2068										
INC	605	620	817	857	897	948	980	982	1044	1056	1128	1166	1197	1294	1304
	1312	1318	1330	1335	1347	1352	1353	1359	1373	1388	1418	1501	1521	1557	
JMP	530	682	849	854	860	871	874	883	933	1010	1023	1042	1051	1066	1132
	1176	1219	1569	1587	1606	1620	1634	1647	1660	1674	1688	1701	1714	1728	1741
	1755	1768	1782	1795	1809	1822	1836	1849	1863	1876	1890	1903	1917	1930	1944
	1957	1971	1984	1998	2011	2025	2069								
JSR	590	608	836	843	846	879	896	900	924	940	941	953	965	970	979
	985	989	1035	1060	1072	1083	1111	1141	1174	1189	1200	1205	1213	1417	1423
	1425	1429	1487	1504	1526	1530	1604	1618	1632	1645	1658	1672	1686	1699	1712
	1726	1739	1753	1766	1780	1793	1807	1820	1834	1847	1861	1874	1888	1901	1915
	1928	1942	1955	1969	1982	1996	2009	2023							
MOV	587	588	591	592	596	597	598	599	611	612	613	615	621	629	630
	631	643	644	647	648	650	652	662	663	664	674	679	694	695	696
	697	698	699	701	705	706	707	708	717	729	730	740	741	745	746

J05

PDP-11/VT20 HOST DIAGNOSTIC PROGRAM MACY11 27(732) 08-SEP-76 09:00 PAGE 63
DZVTEB.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

*DZVTEB, DZVTEB, SEQ/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DZVTEB.P11
RUN-TIME: 7 15 3 SECONDS
RUN-TIME RATIO: 55/27=2.0
CORE USED: 9K (18 PAGES)

