

.REPT 0

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

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDLB-B-D
PRODUCT NAME: DL11-E ON LINE TESTS
DATE: 21-DECEMBER-1975
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: ROBERT WHITTON
REVISED BY: AL LOSCHAK JUN-75
M. McNALLY DEC-75 ;SUPPORT SOFTWARE SWITCH REGISTE

THIS MAINDEC OBSOLETE MAINDEC-11-DZDLBA

COPYRIGHT (C) DIGITAL EQUIPMENT CORPORATION
1972, 1975

THE MATERIAL IN THIS DOCUMENT IS FOR INFORMATION
PURPOSES ONLY AND IS SUBJECT TO CHANGE WITHOUT NOTICE.
DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY
FOR THE USE OF SOFTWARE ON EQUIPMENT WHICH IS NOT
SUPPLIED BY IT.
DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY
FOR ANY ERRORS WHICH MAY APPEAR IN THE DOCUMENT.

49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91

1. ABSTRACT

TWO SEPARATE DIAGNOSTIC PROGRAMS ARE PROVIDED FOR THE DL11-E (ASYNCHRONOUS LINE INTERFACE), MAINDEC-11-DZDLA (DL11-E OFF LINE TESTS) AND MAINDEC-11-DZDLB (DL11-E ON LINE TESTS). THE OFF LINE TESTS TEST ALL DL11-E LOGIC AND MAY BE USED TO INDIVIDUALLY TEST UP TO 31 DL11-E'S. THE OFF LINE TESTS DO NOT REQUIRE THE USE OF A MODEM, HOWEVER A SPECIAL JUMPER CONNECTOR IS REQUIRED. THE ON LINE TESTS ARE ESSENTIALLY DATA RELIABILITY TESTS REQUIRING THE USE OF MODEMS AND A SUITABLE TERMINAL DEVICE.

THREE STARTING ADDRESSES ARE PROVIDED. THEY ARE:

- 200 - NORMAL START
- 210 - REMAP DEVICES PRESENT AND RESTART
- 220 - MODIFY DEVICE ADDRESSES IF NON STANDARD INSTRUCTIONS TO DO THIS ARE TYPED OUT.

THIS DOCUMENT DESCRIBES THE ON LINE TESTS.

THE AVAILABLE TESTS ARE:

- PRG0 SINGLE CHARACTER LINE MODE DATA TEST
- PRG1 BINARY COUNT LINE MODE DATA TEST
- PRG2 MESSAGE TRANSMIT ONLY W/W/O PARITY
- PRG3 RECEIVE DATA TEST
- PRG4 MESSAGE TRANSMIT (SPIRAL) ONLY W/W/O PARITY.

2. REQUIREMENTS

2.1 EQUIPMENT

- A. PDP 11 SYSTEM
- B. DL11-E(S)
- C. SUITABLE TERMINAL DEVICE (ASR 33, 37, DATA POINT, ETC)
- D. MODEM TYPE 103 OR 202 OR EQUIVALENT

2.2 STORAGE

THIS PROGRAM USES 8K OF MEMORY

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

3. OPERATING PROCEDURE:

3.1 LOADING PROCEDURE

THE ABSOLUTE LOADER IS USED TO LOAD THE PROGRAM.

3.2 DL11-E PARAMETER SELECTION

THE SELECTABLE DL11-E PARAMETERS ARE:

BIT1=0	CHARACTER LENGTH
BIT1	CHAR. LENGTH
0	8
1	7
0	6
1	5

WHEN A TERMINAL IS INVOLVED DL11-E PARAMETERS SHOULD BE SET ACCORDING TO TERMINAL SPECIFICATIONS.

3.3 PDP-11 STANDARD OPERATING PARAMETERS

SW15	1	HALT ON ERROR
SW14	1	SCOPE LOOP (NOT USED)
SW13	1	INHIBIT ERROR PRINTOUT
SW12	1	INHIBIT TRACE TRAP (NOT USED)
SW11	1	INHIBIT ITERATION (NOT USED)

CONFIGURATION 2: THIS CONFIGURATION TRANSMITS DATA FROM BOTH THE CALLED TRANSMITTER AND THE TRANSMITTER CONNECTED TO THE LINE THAT WAS CALLING, I.E. IN ADDITION TO THE DATA TRANSMITTED AS IN CONFIGURATION 1 DATA IS ALSO TRANSMITTED IN THE REVERSE DIRECTION. TO INITIATE PROGRAM ACTION CALL THE DL11-E YOU WISH TO TRANSMIT ON FROM THE DL11-E YOU WISH TO RECEIVE/TRANSMIT ON. WHEN THE PHONE RINGS AT THE PD11 THE PROGRAM WILL REQUEST THE CONFIGURATION AND MODEM TYPE. TYPE BIT0-1 =10 AND BIT200. NOTE: *****DO NOT USE MODEM TYPE 202 (OR EQUIV) USING CONFIG #2***** THE PROGRAM WILL REQUEST THE LINE YOU CALLED FROM. TYPE THE NUMBER IN OCTAL FOLLOWED BY A CR. WHEN THE CARRIER IS HEARD IN THE HEADSET PRESS THE DATA BUTTON ON THE DATA SET. NOTE YOU HAVE APPROXIMATELY 10 SECONDS IN WHICH TO DO THIS. WHEN THE 'HANDSHAKING IS COMPLETED THE PROGRAM WILL REQUEST TWO SETS OF DL11-E PARAMETERS. THE CHARACTER LENGTH OF BOTH SETS MUST BE THE SAME AND THE SPEED OF THE SECOND SET MUST BE GREATER THAN THE SPEED OF THE FIRST. WHEN THE PARAMETERS HAVE BEEN LOADED THE PROGRAM WILL TYPE 'LINE CONNECTION MADE' AND BEGIN TO WAY DATA TRANSMISSION. WHEN 100. CHARACTERS HAVE BEEN RECEIVED AND CHECKED THE BELL WILL RING AT THE TTY, AND ANOTHER BLOCK OF 100. CHARACTERS WILL BE PROCESSED. NOTE, DL11-E#X REFERS TO THE 'CALLED' DL11-E, AND DL11-E#Y REFERS TO THE 'CALLING' DL11-E.

3.5

LINE NUMBERS

LINE NUMBER REFERS TO THE ADDRESSES TO WHICH THE DL11-E RESPONDS.

LINE 00	77561X	LINE 10	77571X	LINE 20	77601X	LINE 30	77611X
LINE 01	77562X	LINE 11	77572X	LINE 21	77602X	LINE 31	77612X
LINE 02	77563X	LINE 12	77573X	LINE 22	77603X	LINE 32	77613X
LINE 03	77564X	LINE 13	77574X	LINE 23	77604X	LINE 33	77614X
LINE 04	77565X	LINE 14	77575X	LINE 24	77605X	LINE 34	77615X
LINE 05	77656X	LINE 15	77576X	LINE 25	77606X	LINE 35	77616X
LINE 06	77567X	LINE 16	77577X	LINE 26	77607X	LINE 36	77617X
LINE 07	77570X	LINE 17	77600X	LINE 27	77610X		

215
 216
 217
 218
 219
 220
 221
 222
 223
 224
 225
 226
 227
 228
 229
 230
 231
 232
 233
 234
 235
 236
 237
 238
 239
 240
 241
 242
 243
 244
 245
 246
 247
 248
 249
 250
 251

4. USE PROCEDURE

THIS PROGRAM HAS BEEN MODIFIED TO RUN WITH OR WITHOUT
A CONSOLE PROCESSOR.
IF A CONSOLE MACHINE IS USED; THEN THE PROGRAM
LOOKS AT THE HARDWARE SWITCH REGISTER.
IF A CONSOLE-LESS MACHINE IS USED; THEN THE PROGRAM
AUTOMATICALLY LOOKS AT THE CONTENTS OF LOCATION
SOFTSR (176) AS A SWITCH REGISTER.

IT'S THE RESPONSIBILITY OF THE OPERATOR TO SET UP
THIS LOCATION PRIOR TO STARTING THE PROGRAM.

- 4.1 PRG0 SINGLE CHARACTER LINE MODE DATA TEST
 - A. LOAD ADDRESS = 000200 (RESTART LOAD ADDR. = 000200)
 - B. START = PROGRAM WILL REQUEST PROGRAM NUMBER
 - C. THE PROGRAM WILL NOW REQUEST THE DATA. TYPE THE DATA IN OCTAL FOLLOWED BY A CARRIAGE RETURN.
 - D. MAKE LINE CONNECTION. SEE SECT 3.4

4.2 PRG1 = BINARY COUNT LINE MODE DATA TEST

- A. LOAD ADDRESS = 000200
- B. START = PROGRAM WILL REQUEST PROGRAM NUMBER
- C. MAKE LINE CONNECTION SEE SECT 3.4

4.3 PRG2 = SPECIAL MESSAGE XMIT ONLY

- A. LOAD ADDRESS = 000200
- B. START = PROGRAM WILL REQUEST PROGRAM NUMBER
- C. DEPRESS START = THE PROGRAM WILL IDENTIFY ITSELF AND TYPE INSTRUCTIONS TO SELECT DESIRED DL11-E PARAMETERS

(SEE SECT 3.2)

D. TYPE IN PARAMETERS. IF IT IS DESIRED TO TRANSMIT DATA WITH PARITY SET BIT6. ALSO SET BITS TO TRANSMIT ODD PARITY AND CLEAR TO TRANSMIT EVEN PARITY.

BIT6	1/0	ENABLE/DISABLE PARITY
BIT5	1/0	TRANSMIT ODD/EVEN PARITY

E. WHEN 'MAKE LINE CONNECTION' IS TYPED CALL THE DL11-E

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
292
293
294
295
296
297
298
299
300
301
302
303
304
305

DZDLBB MACY11 27(657) 17-NOV-75 14:50 PAGE 8
DZDLBB.SRC

306
307
308
309

*OU WISH TO TRANSMIT ON FROM THE TERMINAL MODEM.
*WHEN THE 'HANDSHAKING' IS COMPLETED THE MESSAGE
*THE QUICK BROWN FOX JUMPED OVER THE LAZY DOGS BACK
*0123456789' WILL BE TRANSMITTED. TO TERMINATE, HANG UP.

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
349
350
351
352
353
354
355
356
357

4.4 PRG3 - RECEIVE TRANSMIT MESSAGE TEST

- A. LOAD ADDRESS = 000200
- B. START - PROGRAM WILL REQUEST PROGRAM NUMBER
- C. THE PROGRAM WILL IDENTIFY ITSELF AND TYPE INSTRUCTIONS TO SELECT DESIRED OPTIONS.
- D. SET IN OPTIONS AND PRESS CONTINUE.
- E. WHEN 'MAKE LINE CONNECTION' IS TYPED CALL THE DL11-E YOU WISH TO TRANSMIT ON. WHEN THE 'HANDSHAKING' IS COMPLETED THE DL11-E WILL TRANSMIT A CR LF TO THE TERMINAL DEVICE. AT THIS TIME YOU MAY BEGIN TO SEND DATA FROM THE DEVICE TO THE DL11-E WHERE IT WILL BE ECHOED BACK TO THE TERMINAL. TYPE ANY CHARACTER TO SIGNAL START OF MESSAGE. THEN TYPE MESSAGE AND THE SAME CHARACTER TO SIGNAL END OF MESSAGE. CONTROL C WILL CAUSE THE BUFFERS CONTENTS TO BE TRANSMITTED WHEN TYPED.
- F. IF NO ECHO IS DESIRED (ON A CHARACTER BASIS FOR EXAMPLE WHEN USING A TERMINAL THAT PRODUCES ITS OWN LOCAL COPY) SET BIT7 OF SWITCH REGISTER.

4.5 PRG4 - SPECIAL MESSAGE XMIT ONLY

- A. LOAD ADDRESS = 000200
- B. OPTIONS
 - 1. BITS 0-2 = 4
 - 2. BITS 3-6 = LINE NUMBER (SEE SECT 3.5)
- C. DEPRESS START - THE PROGRAM WILL IDENTIFY ITSELF AND TYPE INSTRUCTIONS TO SELECT DESIRED DL11-E PARAMETERS (SEE SECT 3.2)

D. SET IN PARAMETERS IF IT IS DESIRED TO TRANSMIT DATA WITH PARITY RAISE SR6. ALSO RAISE SR5 TO TRANSMIT ODD PARITY AND LOWER TO TRANSMIT EVEN PARITY.

BIT6	1/0	ENABLE/DISABLE PARITY
BIT5	1/0	TRANSMIT ODD/EVEN PARITY

E. WHEN 'MAKE LINE CONNECTION' IS TYPED CALL THE DL11-E YOU WISH TO TRANSMIT ON FROM THE TERMINAL MODEM. WHEN THE 'HANDSHAKING' IS COMPLETED A SPIRAL PATTERN WILL BE TRANSMITTED. TO TERMINATE, HANG UP.

358 PROGRAM DESCRIPTIONS
359
360
361 5.1 PRG0 - SINGLE CHARACTER LINE MODE DATA TEST
362 PRG0 TRANSMITS USER SPECIFIED DATA AND A CARRIAGE RETURN/LINE FEED
363 EVERY 72ND CHARACTER.
364
365 5.2 PRG1 - BINARY COUNT PATTERN LINE MODE DATA TEST
366 PRG1 TRANSMITS A BINARY COUNT PATTERN. THIS PROGRAM IS THE SAME
367 AS PRG0 EXCEPT FOR THE DATA TRANSMITTED.
368
369 5.3 PRG2 - SPECIAL MESSAGE TRANSMIT ONLY
370 PRG2 TRANSMITS THE MESSAGE
371 THE QUICK BROWN FOX JUMPED OVER THE LAZY DOGS BACK 0123456789.
372 NO DATA ERROR CHECKING IS PERFORMED BY THE PROGRAM.
373
374 5.4 PRG3 - RECEIVE/TRANSMIT MESSAGE TEST
375 PRG3 - RECEIVES DATA FROM A TERMINAL AND READS THE RECEIVED MESSAGE
376 BACK, AND TYPES THE MESSAGE ON THE PDP-11 TTY WHEN THE MESSAGE
377 IS TERMINATED. CHARACTERS MAY BE ECHOED BACK (IF REQUIRED) ON
378 A CHARACTER BASIS THEREBY CREATING LOCAL COPY AS THE MESSAGE
379 IS TYPED.
380 TRANSMISSION MAY BEGIN AT THE TERMINAL WHEN A CR/LF IS RECEIVED
381 AT THE TERMINAL. THIS PROGRAM IS RESTRICTED TO USE BY ONLY
382 FULL DUPLEX MODEMS.
383
384 5.5 PRG4 - SPECIAL MESSAGE TRANSMIT ONLY
385 PRG4 TRANSMITS A SPIRAL PATTERN.
386 NO DATA CHECKING IS PERFORMED BY THE PROGRAM.
387
388
389
390

6.0

ERRORS ARE TWO TYPES OF ERRORS WHICH ARE DETECTED BY THESE TESTS
THERE ARE TWO TYPES OF ERRORS WHICH ARE DETECTED BY THESE TESTS
LINE FAILURE, AND DATA ERRORS.
DATA ERRORS ARE DETECTED ONLY IN PRG 0 & 1 WHEN USING CONFIG-
URATIONS 1 OR 2. DATA ERRORS IN THE OTHER TESTS MAY BE DETECT-
ED BY VISUAL INSPECTION OF THE DATA AT THE TERMINAL.
LINE FAILURES ARE REPORTED BY TYPING AT THE PC, THE RECEIVER CON-
TROL STATUS REGISTER ADDRESS, AND ITS CONTENTS. SEE THE PRO-
GRAM LISTING FOR A DETAILED DESCRIPTION OF THE ERROR.
THE MOST FREQUENTLY ENCOUNTERED ERROR WILL PROBABLY BE THE
LOSS OF CARRIER. THIS ERROR WILL BE REPORTED IF AFTER A LINE
CONNECTION IS MADE THE CARRIER IS LOST, EITHER BY 'HANGING UP'
OR A 'GLITCH' ON THE LINE CAUSING THE CARRIER TO MOMENTARILY
DROP. IN EITHER INSTANCE THE PROGRAM DISCONNECTS THE DL11-E
FROM THE MODEM (BY CLEARING DATA TERMINAL READY) AND THE
LINE WILL HAVE TO BE RECONNECTED TO RESUME TESTING.
IF IT IS PHYSICALLY IMPOSSIBLE TO GET TO THE DATA BUTTON
WITHIN THE TIME ALLOTTED (APPROX. 10 SECONDS) TO MAKE THE
LINE CONNECTION, THIS TIME MAY BE INCREASED BY PUTTING A
LARGER NUMBER INTO THE DELAY. PATCH THE LARGER NUMBER
INTO THE ADDRESS FOLLOWING THE DELAY EMT (BETWEEN RINT8G AND
RINT8H). FOR EXAMPLE PATCHING IN 72860 WILL ALLOW APPROXIMATELY
30 SECONDS IN WHICH TO RESPOND.

DATA ERRORS ARE REPORTED BY TYPING THE PC, THE RECEIVER CONTROL
REGISTER ADDRESS OF THE LINE THAT FAILED, WHAT THE DATA SHOULD
HAVE BEEN, WHAT THE DATA WAS, AND THE CHARACTER NUMBER.

PC=XXXXXX 174010 DATA S/R 301 WAS 321 CHAR NO 23

THIS TYPEOUT INDICATES A DATA ERROR ON LINE 1
IF CONFIGURATION 2 IS SELECTED TWO ERROR TYPEOUTS MAY OCCUR
FOR A SINGLE ERROR DEPENDING ON WHERE THE ERROR OCCURED. CON-
FIGURATION 2 COMPARES THE DATA RECEIVED AT THE CALLED DL11-E
WITH THE DATA TRANSMITTED BY THE CALLED DL11-E, AND ALSO THE DATA
RECEIVED AT THE CALLING DL11-E (CALLER) WITH THE DATA TRANSMITTED
BY THE CALLED DL11-E.
IF FOR EXAMPLE A DATA ERROR OCCURED AT THE RECEIVER OF THE CALLING
DL11-E CAUSING IT TO TRANSMIT INCORRECT DATA TO THE CALLED DL11-E TWO
TYPEOUTS WILL OCCUR AS SHOWN BELOW:

PC=XXXXXX 174010 DATA S/R 301 WAS 321 CHAR NO 23
PC=XXXXXX 174000 DATA S/R 301 WAS 321 CHAR NO 23

THESE TYPEOUTS SHOW THAT THE RECEIVER ON LINE 0 WAS THE CAUSE
OF THE ERROR AND THE RECEIVER ON LINE 1 RECEIVED THE CORRECT
INCORRECT DATA.

.ENDR

391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
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 .TITLE DZDL88
442 .ENABLE A88,AMA
443 ;THIS TEST CHECKS THE DL11-E USING MODEMS
444 ;REFER ALSO TO TEST DZDLA (DL11-E OFF LINE TESTS)
445 ;STARTING PROCEDURE
446 ; LOAD ADDRESS 200
447 ;
448 ; STACK POINTER IS AT 1200
449 ; PRESS START
450 ;
451 ;AVAILABLE PROGRAMS
452 ; PRG0= SINGLE CHARACTER LINE MODE DATA TEST.
453 ; PRG1= SPECIAL BINARY COUNT LINE MODE DATA TEST.
454 ; PRG2= SPECIAL MESSAGE XMIT ONLY W/W/O PARITY
455 ; PRG3= RECEIVE DATA TEST
456 ; PRG4= SPIRAL PATTERN MESSAGE XMIT ONLY W/W/O PARITY
457 ; PRG5= DATA ECHO TEST (USES FACILITY AT MAYNARD)
458
459 ;STANDARD SR SWITCH OPTIONS (SWITCH SET TO A 1 )
460 ;ISR15= HALT ON ERROR.
461 ;ISR14= SCOPE (NOT USED)
462 ;ISR13= INHIBIT PRINTOUT
463 ;ISR12= INHIBIT TRACE (NOT USED)
464 ;ISR11= INHIBIT ITERATION (NOT USED)
465
466 000000 004520 ERTF ;UNASSIGNED TRAP
467 000002 000000 0
468 000004 004520 MACHERR: ERTF ;SP OVERFLOW, BUS ERROR TRAP
469 000006 000040 40
470 000010 004520 ERTF ;RESERVED INSTRUCTION TRAP
471 000012 000100 100
472 000014 004520 ERTF ;TRACE TRAP
473 000016 000140 140
474 000020 004626 MAPVEC ;TRAP TO MAP VECTOR
475 000022 000340 PRTY7
476 000024 002646 PFAL ;POWER FAIL TRAP
477 000026 000340 PRTY7
478 000030 002222 EMTINT ;EMT TRAP
479 000032 000340 PRTY7
480 000034 004520 ERTF
481 000036 000340 340
482 000040 000042 .+2
483 000042 000000 HALT
484 .REPT 119.
485 .+2
486 4 ;TRAP TO MAP MAKER
487 .ENDR
488
489 ;EQUATE STATEMENTS
490 PSW=17776
491 001200 STKPTR=1200
492 000000 OPEN=0
493 100000 MANUAL=BIT15
494 100000 BIT15=100000
  
```

```

495 040000 BIT14=40000
496 020000 BIT13=20000
497 010000 BIT12=10000
498 004000 BIT11=4000
499 002000 BIT10=2000
500 001000 BIT9=1000
501 000400 BIT8=400
502 000200 BIT7=200
503 000100 BIT6=100
504 000040 BIT5=40
505 000020 BIT4=20
506 000010 BIT3=10
507 000004 BIT2=4
508 000002 BIT1=2
509 000001 BIT0=1
510 005726 POPBP=5726 ;POP THE STACK. SAME AS TST (6)+
511 022626 POPSP=22626 ;POP STACK TWICE. SAME AS CMP (6)+,(6)+
512 000340 PRTY7=340 ;PRIORITY LEVEL DEFINITIONS
513 000300 PRTY6=300
514 000240 PRTY5=240
515 000200 PRTY4=200
516 000140 PRTY3=140
517 000100 PRTY2=100
518 000040 PRTY1=40
519 000000 PRTY0=0
520 104000 TYPE=EMT+0
521 104001 TYPES=EMT+1
522 104002 STALL=EMT+2
523 104003 ERROR=EMT+3
524 104004 DATCHK=EMT+4
525 104005 CHALT=EMT+5
526 104006 STRXV=EMT+6
527 104007 STTXV=EMT+7
528 104010 EMHLT=EMT+10
529 104011 SAVREG=EMT+11
530 104012 RSTREG=EMT+12
531 104013 ERROR1=EMT+13
532 104014 ERRTX=EMT+14
533 104015 ERRRN=EMT+15
534 104016 DELAY=EMT+16
535 000000 N=0
536 000000 A=0
537
538 .MACR CNVOA SRC,DST,COUNT
539 JSR X5,OADNV ;SD TO OCTAL TO ASCII CONVERT.
540 SRC ;SOURCE ADDR.
541 DST ;DESTINATION ADDR.
542 COUNT ;#OF DIGITS TO CONVERT.
543 .ENDM
544
545 .MACR ISR N
546 RISR'N' JSR MOV ;N',X5
547 JMP RISR ;GO TO COMMON INTERRUPT SERVICE ROUTINE
548
  
```



```

549 .ENDM
550
551 .MACR ISRT N
552 TISR*N'I MOV #'N',%0 ;PUT LINE # IN R0
553 JMP TISR ;GO TO COMMON INTERRUPT SERVICE
554
555 .ENDM
556
557 .MACR RRCV N,A
558 175610+N ;ADDRESS OF RECEIVER LINE # 'A'
559 .ENDM
560
561 .MACR RBUF N,A
562 175612+N ;ADDRESS OF RECEIVER BUFFER LINE # 'A'
563 .ENDM
564
565 .MACR TXMT N,A
566 175614+N ;ADDRESS OF TRANSMITTER CSR LINE # 'A'
567 .ENDM
568
569 .MACR TBUF N,A
570 175616+N ;ADDRESS OF TRANSMIT BUFFER LINE # 'A'
571 .ENDM
572
573
574 .=174
575 000174 177570 SRPTR: 177570
576 000176 000000 SOFTSR: 000000
577 000200 000200 .=200
578 000200 012737 001422 000300 MOV #START,STAD ;SET UP FOR NORMAL START
579 000206 000407 BR STCONT ;CONTINUE
580 000210 012737 001416 000300 MOV #REMAP,STAD ;SET FOR A REMAP START
581 000216 000403 BR STCONT
582 000220 012737 000204 000300 MOV #MODEV,STAD ;SET TO MODIFY ON START
583 000226 000226 STCONT:
584 000226 012706 001200 MOV #STKPTR,X6 ;SET BOTTOM OF STACK
585 000232 013746 000006 MOV 6,=(SP) ;SAVE CURRENT VECTOR
586 000236 013746 000004 MOV 4,=(SP)
587 000242 012737 000004 MOV #12,4 ;SET UP TIME OUT VECTOR
588 000250 005777 177720 TST #SRPTR ;TRY TO REFERENCE THE
589 ;HARDWARE SWITCH REGISTER
590 000254 000404 BR 28 ;BRANCH IF NO TIME OUT TRAP OCCURS
591 000256 012737 000176 000174 13: MOV #SOFTSR,SRPTR ;CHANGE THE SWITCH REGISTER
592 ;POINT TO A SOFTWARE SWITCH REGISTER
593 000264 022626 CMP (6)+,(6)+ ;RESTORE THE STACK
594 000266 012637 000004 23: MOV (6)+,4 ;RESTORE TIME OUT VECTOR
595 000272 012637 000006 MOV (6)+,6
596 000276 000137 JMP @PC)+ ;JUMP TO SELECTED START
597 000300 000000 STAD: 0
598 001200 001200 .=1200
599 001200 000000 RXCSR: 0 ;RECEIVER CSR UNDER TEST
600 001202 000000 RXBUF: 0 ;RECEIVER BUFFER UNDER TEST
601 001204 000000 TXCSR: 0 ;TRANSMITTER CSR UNDER TEST
602 001206 000000 TXBUF: 0 ;TRANSMITTER BUFFER UNDER TEST

```

```

603 001210 000000 RXVTR: OPEN ;RECEIVER VECTOR
604 001212 000200 RXLVL: PRTY4 ;RECEIVER PRIORITY LEVEL
605 001214 000000 TXVTR: OPEN ;TRANSMITTER VECTOR
606 001216 000200 TXLVL: PRTY4 ;TRANSMITTER PRIORITY LEVEL
607 001220 177560 TK3: 177560 ;LSR CSR
608 001222 177562 TKB: 177562 ;LSR BUFFER
609 001224 177564 TPS: 177564 ;LSR CSR
610 001226 177566 TPB: 177566 ;LSR BUFFER
611 001230 000060 TKVTR: 60 ;LSR INTERRUPT VECTOR
612 001232 000200 TKLVL: PRTY4 ;LSR PRIORITY LEVEL
613 001234 000064 TPVTR: 64 ;LSR INTERRUPT VECTOR
614 001236 000200 TPLVL: PRTY4 ;LSR PRIORITY LEVEL
615 001240 000000 PRGNUM: OPEN ;CONTAINS CURRENT PROGRAM#
616 001242 000000 PRGID: OPEN ;CONTAINS PROGRAM INDICATORS
617 001244 006424 PRGTAB: PRG0 ;PRG0 START ADDRESS
618 001246 006540 PRG1 ;PRG1 START ADDRESS
619 001250 006642 PRG2 ;PRG2 START ADDRESS
620 001252 006740 PRG3 ;PRG3 START ADDRESS
621 001254 007474 PRG4 ;PRG4 START ADDRESS
622 001256 007706 PRG5 ;PRG5 START ADDRESS
623 001260 002354 INCPRG
624 001262 002354 INCPRG
625 001264 003122 ENTTAB: TYP ;POINTER TO TYPEOUT ROUTINE
626 001266 003244 TYPB ;POINTER TO CHAINED MESSAGES ROUTINE
627 001270 000000 OPEN ;POINTER TO RANDOM STALL ROUTINE
628 001272 002512 ERR ;POINTER TO ERROR ROUTINE
629 001274 002412 DTCMK ;POINTER TO DATA CHECK ROUTINE
630 001276 002366 CHLT ;COMMON HALT
631 001300 003042 STRVRV ;POINTER TO ROUTINE TO SET RCVR VECTOR AND PRIORITY
632 001302 003072 STXMTV ;POINTER TO ROUTINE TO SET XMIT VECTOR AND PRIORITY
633 001304 002400 EHLT ;POINTER TO ERROR HALT ROUTINE
634 001306 002254 SAVRG ;POINTER TO SAVE REGISTERS ROUTINE
635 001310 002314 RSTRG ;POINTER TO RESTORE REGISTERS ROUTINE
636 001312 002534 ERR1 ;POINTER TO ERROR ROUTINE
637 001314 002710 TXERR ;POINTER TO XMIT ERROR ROUTINE
638 001316 002732 RXERR ;POINTER TO RCVR ERROR ROUTINE
639 001320 003466 DLY ;POINTER TO DELAY ROUTINE
640 001322 000000 PARBIT: OPEN
641 001324 000000 COUNT: OPEN
642 001326 000000 SAVE: OPEN
643 001330 000000 LINE: OPEN
644 001332 000000 CONFIG: OPEN
645 001334 000000 NUMBER: OPEN
646
647 001336 000000 OLDPB: 0
648 001340 000000 TOPC: 0
649 001342 000000 FROMPC: 0
650 001344 000000 FTITLE: 0
651 001346 000000 FNONE: 0
652 001350 000000 FMAP: 0
653 001352 000000 TEMP1: 0
654 001354 000000 LINENO: 0
655 001356 000000 RECDAT: OPEN
656 001360 000000 XMTDAT: OPEN

```

657	001362	000000		CARMSK:	OPFN		
658	001364	000000		CTRD:	OPFN		
659	001366	000000		TXCSR:	OPFN		
660	001370	000000		RXCSR:	OPFN		
661	001372	000000		TEMP:	OPFN		
662	001374	000000		SRT:	OPFN		
663	001376	000000		INBUFP:	OPFN		
664	001400	000000		BUFP:	OPFN		
665	001402	000000		CALLER:	OPFN		
666	001404	000000		CALLED:	OPFN		
667	001406	000000		OTBUFP:	OPFN		
668							
669	001410	000000		TBUFP:	OPFN		
670	001412	000000		MODEM:	OPFN		
671	001414	000000					
672							
673							
674	001416	005037	001344	REMAP:	CLR	FTITLE	
675	001422	012706	001200	START:	MOV	#STKPTR,X6	;/SET BOTTOM OF SP STACK.
676	001426	000005			RESET		
677	001430	005037	177776		CLR	PSW	
678	001434	012737	004520		MOV	#ERTP,MACHER	
679	001442	012737	000040		MOV	#40,MACHER+2	
680	001450	005737	001344		TST	FTITLE	;/TITLE PRINTED
681	001454	001145			BNE	START1	;/YES, SKIP THIS
682	001456	104000			TYPE		
683	001460	012012			HTITLE		
684	001462	005237	001344		INC	FTITLE	;/SET TITLE PRINTED FLAG
685	001466	005037	001346		CLR	FNONE	;/CLEAR NO DEVICE FLAG
686	001472	012737	177777		MOV	#-1,LINENO	
687	001500	012737	001550		MOV	#MAPNE,MACHER	;/SET UP FOR NO DEVICE ANSWER
688	001506	012737	000340		MOV	#PRTY7,MACHER+2	
689	001514	012704	010432		MOV	#RCSR,X4	;/SET UP DEVICE POINTER
690	001520	005237	001354	MAP:	INC	LINENO	
691	001524	020427	010530		CMP	X4,#RBUF	;/LAST DEVICE
692	001530	001477			BEO	MAPEND	;/YES
693	001532	005037	177776		CLR	PSW	
694	001536	005774	000000		TST	0(4)	;/TEST DEVICE
695	001542	000240			NOP		
696	001544	000240			NOP		
697	001546	000404			BR	MAPOK	
698	001550	062704	000002	MAPNE:	ADD	#2,X4	
699	001554	022626			POPSP2		
700	001556	000760			BR	MAPA	
701	001560	011437	001352	MAPOK:	MOV	(4),TEMP1	;/SAVE DEVICE FOR TYPING
702	001564	004537	003530		JSR	X5,QACNV	
703	001570	001352			TEMP1		
704	001572	012154			MADDR		
705	001574	000006			6		
706	001576	004537	003530		JSR	X5,QACNV	
707	001602	001354			LINENO		
708	001604	012146			NLINE		
709	001606	000002			2		
710	001610	011401			MOV	(4),X1	;/GET RXCSR DEVICE ADDRESS

711	001612	004737	004704		JSR	X7,FORMAD	
712	001616	052737	000001	001350	BIS	#BITS,PMAP	
713	001624	042777	000100	177352	BIC	#BITS,#TXCSR	
714	001632	052777	000100	177344	BIS	#BITS,#TXCSR	
715	001640	000240			NOP		
716	001642	012737	000340	177776	MOV	#PRTY7,PSW	
717	001650	005737	001210		TST	RXVTR	
718	001654	001406			BEO	MAPOKA	
719	001656	013701	001354		MOV	LINENO,X1	
720	001662	006301			ASL	X1	
721	001664	013761	001210	010334	MOV	RXVTR,VECTAB(1)	;/STORE VECTOR
722	001672	042777	000100	177304	MAPOKA:	BIC	#BITS,#TXCSR
723	001700	004537	003530		JSR	X5,QACNV	
724	001704	001210			RXVTR		
725	001706	012166			MTRAP		
726	001710	000004			4		
727	001712	104000			TYPE		
728	001714	012146			NLINE		
729	001716	005237	001346		INC	FNONE	
730	001722	062704	000002		ADD	#2,X4	
731	001726	000674			BR	MAPA	
732	001730	012737	004520	000004	MAPEND:	MOV	#ERTP,MACHER
733	001736	012737	000040	000006	MOV	#40,MACHER+2	
734	001744	005737	001346		TST	FNONE	
735	001750	001007			BNE	START1	
736	001752	104000			TYPE		
737	001754	012176			MAPERR:	MNONE	
738	001756	005037	001344		CLR	FTITLE	
739	001762	000000			HALT		
740	001764	000137	001422		JMP	START	
741	001770	005037	177776		START1:	CLR	PSW
742	001774	104000			TYPE		
743	001776	012213			MSWSEL		
744	002000	004737	003314		JSR	PC,ROOCT	;/GET INPUT
745	002004	012600			MOV	(SP)+,X0	;/(SR) TO R0
746	002006	042700	177770		BIC	#17770,X0	;/LIMIT (SR) TO BITS 2=0
747	002012	010037	001240		MOV	X0,PRNUM	;/SAVE PROGRAM #
748	002016	006300			ASL	X0	;/R0X2
749	002020	000170	001244		JMP	#RRTAB(0)	;/GO TO SELECTED PROGRAM.
750							
751	002024	012706	001200		MODEV:	MOV	#STKPTR,X6
752	002030	000005			RESET		
753	002032	005037	177776		CLR	PSW	
754	002036	104000			MODEV1:	TYPE	
755	002040	014435			MMOD1		
756	002042	004737	003314		JSR	PC,ROOCT	;/GET INPUT
757	002046	011600			MOV	(SP),X0	
758	002050	042700	177740		BIC	#17740,X0	
759	002054	006300			ASL	X0	
760	002056	022627	000036		CMP	(SP)+,#36	
761	002062	014403			BLOS	MODEV2	;/BRANCH IF > 36
762	002064	104000			TYPE		
763	002066	014363			MMODX		
764	002070	000762			BR	MODEV1	

```

765 002072 104000 MODEV2: TYPE
766 002074 014553 MHO02
767 002076 004737 003314 JSR PC,ROOCT ;GET INPUT
768 002102 032716 000001 BIT #BIT0,(SP)
769 002106 001403 BEQ MODEV3
770 002110 104000 TYPE
771 002112 014405 MHO00
772 002114 000766 BR MODEV2
773 002116 012601 MOV (SP)+,X1 ;SAVE DEV ADR
774 002120 010003 MOV X0,X3
775 002122 062703 010432 ADD #RCSR,X3
776 002126 010113 MOV X1,(3)
777 002130 062701 000002 ADD #2,X3 ;UPDATE DEV ADR
778 002134 010003 MOV X0,X3
779 002136 062703 010530 ADD #RBUF,X3
780 002142 010113 MOV X1,(3)
781 002144 062701 000002 ADD #2,X3 ;UPDATE DEV ADR
782 002150 010003 MOV X0,X3
783 002152 062703 010626 ADD #TCBR,X3
784 002156 010113 MOV X1,(3)
785 002160 062701 000002 ADD #2,X1 ;UPDATE DEV ADR
786 002164 010003 MOV X0,X3
787 002166 062703 010724 ADD #TBUF,X3
788 002172 010113 MOV X1,(3)
789 002174 104000 TYPE
790 002176 014621 MHO03
791 002200 004737 003314 JSR PC,ROOCT ;GET INPUT
792 002204 022627 77777 CMP (SP)+,#177777
793 002210 001712 BEQ MODEV1
794 002212 005037 001344 CLR FTITLE
795 002216 000137 001422 JMP START

;
;EMT TRAP INTERPRETER
796
797
798 002222 011646 EMTINT: MOV #%6,(6) ;GET SAVED PC.
799 002224 162716 SUR #2,%6 ;DECREMENT PC BY 2.
800 002230 017616 MOV #(%6),%6
801 002234 006116 EMTA1: RDL #%6 ;EMT ARG X 2.
802 002236 042716 BIC #177001,%6 ;REMOVE 7 MSB.
803 002242 062716 ADD #EMTAB,%6 ;FORM EMT RTN ADDR.
804 002246 017616 MOV #(%6),%6
805 002252 000136 JMP #(%6)+ ;GO TO EMT ROUTINE.

;
;SAVE REGS 0 TO 4 SUBROUTINE.
806
807
808 002254 012637 002310 SAVRG: MOV (6)+,SVRPC ;SAVE PC AND PSW.
809 002260 012637 002312 MOV (6)+,SVRPSW
810 002264 010446 MOV X4,=(6) ;SAVE REGS 0 = 4
811 002266 010346 MOV X3,=(6) ;IN STACK.
812 002270 010246 MOV X2,=(6)
813 002272 010146 MOV X1,=(6)
814 002274 010046 MOV X0,=(6)
815 002276 013746 002312 MOV SVRPSW,=(6) ;RESTORE PC AND PSW.
816 002302 013746 002310 MOV SVRPC,=(6)
817 002306 000002 RTI ;EXIT.
818 002310 000000 SVRPC: OPEN

```

```

819 002312 000000 SVRPSW: OPEN
820
821
822 002314 012637 002350 ;RESTORE REGS 0 TO 4 SUBROUTINE.
823 002320 012637 002352 RSTRG: MOV (6)+,RSTPC ;SAVE PC AND PSW.
824 002324 012600 MOV (6)+,RSTPSW
825 002326 012601 MOV (6)+,X0 ;RESTORE REGS 0 = 4
826 002330 012602 MOV (6)+,X1 ;FROM STACK.
827 002332 012603 MOV (6)+,X2
828 002334 012604 MOV (6)+,X3
829 002336 013746 002352 MOV RSTPSW,=(6) ;RESTORE PC AND PSW.
830 002342 013746 002350 MOV RSTPC,=(6)
831 002346 000002 RTI ;EXIT
832 002350 000000 RSTPC: OPEN
833 002352 000000 RSTPSW: OPEN
834 002354 104000 INCRG: TYPE ;TYPE INCORRECT PROGRAM SELECTED.
835 002356 013352 AINPRG
836 002360 000000 HALT
837 002362 000137 001422 JMP START
838
839 002366 011600 ;COMMON HALT ROUTINE
840 002370 162700 CHLT: MOV #%6,%0 ;DEVELOP ADDRESS OF CALLER.
841 002374 000000 SUB #2,%0
842 002376 000002 HALT ;HALT. ADDRESS OF CALL INSTRUCTION
843 ;IN DATA LIGHTS.
844
845 002400 005777 175570 ;CONDITIONAL ERROR HALT ROUTINE.
846 002404 100001 EHLT: TST #SRPTR ;CHECK FOR HALT ON ERROR.
847 002406 000000 BPL EHLTA ;BRANCH IF NO HALT DESIRED.
848 002410 000002 HALT ;HALT.
849 ;IN DATA LIGHTS.
850
851 002412 043737 001362 001360 ;DATA CHECK ROUTINE.
852 002420 123737 001366 001360 DTCHK: BIC CARMK,XMTDAT ;CLEAR UNTRANSMITTED BITS
853 002426 001430 BEQ DTCKA ;COMPARE TRANSMITTED AND RECEIVED
854 002430 CNVOA RECDAT,CWAS,3 ;CHARS. BRANCH IF SAME.
855 002442 CNVOA XMTDAT,C8B,3
856 002454 CNVOA RXCSR,CSRADD,6
857 002466 104013 ERROR1
858 002470 012357 CSRADD
859 002472 004537 004216 JSR 5,0DCNV ;CONVERT
860 002476 001364 CTRD ;CHAR #
861 002500 013343 CRNUM ;TO DECIMAL
862 002502 000004 4 ;4 BITS
863 002504 104013 ERROR1
864 002506 013300 CERDAT
865 002510 000002 DTCKA: RTI ;EXIT.
866
867 002512 012737 177777 002624 ;ERR: MOV #1,ERRB ;SET UP ONE MESSAGE CALL.
868 002520 012737 000240 002626 MOV #240,ERRB+2
869 002526 005037 002644 CLR ERRE
870 002532 000413 BR ERRA
871 002534 011637 002624 ERR1: MOV #%6,ERRB ;DEVELOP ADD'L MESSAGE ADDR.
872 002540 017737 000060 002624 MOV #ERRB,ERRB ;STORE AT ERRE.

```

```

873 002546 012737 177777 002626 MOV #=1,ERRB+2
874 002554 012737 000002 002604 MOV #2,ERRE
875 002562 032777 020000 175404 ERRA: RIT #BIT13,#SRPTR ;INHIBIT ERROR PRINT?
876 002570 001020 RNF ERRC ;BRANCH TO INHIBIT PRINT.
877 002572 011637 002642 MOV #2,ERRD ;DEVELOP CALLING ADDR.
878 002576 162737 000002 002642 SUB #2,ERRD
879 002604 CNVOA ERRD,APC.6 ;CONVERT CALL ADDR TO ASCII.
880 002616 104011 SAVREG
881 002620 104001 TYPE8
882 002622 012427 EMB ;TYPE:
883 002624 000000 ERRA: OPEN ;ERROR HEADER,
884 002626 177777 -1 ;ADDY'L ERROR MESSAGE IF ANY.
885 002630 104012 RSTREG
886 002632 104010 ERRC: EHALT ;GO ERR HALT IF DESIRED.
887 002634 063716 002644 ADD ERRE,%6
888 002640 000002 RTI ;EXIT.
889 002642 000000 ERRA: OPEN
890 002644 000000 ERRE: OPEN
891
892 ;POWER FAIL SERVICE
893 002646 012737 002656 000024 PFAIL: MOV #PWRUP,#24
894 002654 000000 HALT
895 002656 012737 002646 000024 PWRUP: MOV #PFAIL,#24
896 002664 000005 RESET
897 002666 012706 001200 MOV #STKPTR,%6
898 002672 104000 TYPE
899 002674 104325 MPWRP
900 002676 013700 001240 RESTR7: MOV PRGNUM,%0
901 002702 006300 ASL %0
902 002704 000170 001244 JMP #PROTAB(0)
903
904 002710 TXERR: CNVOA TXCSR,ATXWAS,6 ;CONVERT CONTENTS OF TXCSR TO ASCII.
905 002722 012737 012445 003026 MOV #ATXCSR,CRXTXB
906 002730 000410 BR CRXTX
907 002732 RXERR: CNVOA RXCSR,ARXWAS,6 ;CONVERT CONTENTS OF RXCSR TO ASCII.
908 002744 012737 012466 003026 MOV #ARXCSR,CRXTXB
909 002752 011637 003024 CRXTX: MOV %6,CRXTXA ;DEVELOP ADDR OF ADDY'L ERROR MESSAGE.
910 002756 017737 000042 003024 #CRXTXA,CRXTXA
911 002764 032777 020000 175202 BIT #BIT13,#SRPTR ;INHIBIT PRINT?
912 002772 001017 BNE ;BRANCH TO INHIBIT PRINT.
913 002774 011637 002642 MOV #2,ERRD ;DEVELOP CALLING ADDR.
914 003000 162737 000002 002642 SUB #2,ERRD
915 003006 CNVOA ERRD,APC.6 ;CONVERT CALLING ADDR TO ASCII.
916 003020 104001 TYPE8
917 003022 012427 EMB ;TYPE ERROR MESSAGE.
918 003024 000000 CRXTXA: OPEN ;ERR HEADER
919 003026 000000 CRXTXB: OPEN ;ADDY'L ERR MESSAGE
920 003030 177777 -1 ;TXCSR OR RXCSR CONTENTS.
921 003032 104010 CRXTXC: EHALT ;GO HALT IF DESIRED.
922 003034 062716 000002 ADD #2,%6
923 003040 000002 RTI ;EXIT.
924
925 ;ROUTINE TO SET RECEIVER INTERRUPT VECTOR AND PRIORITY
926 003042 017637 000000 003062 STRVRI: MOV #6,#STPRA+2 ;MOVE VECTOR ADDR TO STPRA+2

```

```

927 003050 062716 000002 ADD #2,%6 ;SET UP EXIT
928 003054 013701 001210 MOV RXVTR,%1
929 003060 012721 000000 STPRA: MOV #OPEN,(1)+ ;SET VECTOR ADDRESS
930 003064 013721 001212 MOV RXLVL,(1)+ ;SET PRIORITY
931 003070 000002 RTI ;EXIT
932
933 ;ROUTINE TO SET TRANSMITTER INTERRUPT VECTOR AND PRIORITY.
934 003072 017637 000000 003112 STXMYI: MOV #6,#STPPA+2 ;MOVE VECTOR ADDR TO STPPA+2
935 003100 062716 000002 ADD #2,%6 ;SET UP EXIT
936 003104 013701 001214 MOV TXVTR,%1
937 003110 012721 000000 STPPA: MOV #OPEN,(1)+ ;SET VECTOR ADDRESS.
938 003114 013721 001216 MOV TXLVL,(1)+ ;SET PRIORITY
939 003120 000002 RTI ;EXIT.
940
941 ;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER.
942 003122 010037 003312 TYP: MOV %0,SAVR0 ;SAVE R0
943 003126 011600 MOV #2,%6 ;SET ADDRESS THAT CONTAINS MESSAGE ADDRESS.
944 003130 062716 000002 ADD #2,%6 ;SET UP EXIT.
945 003134 011000 MOV #2,%6 ;ADDRESS OF MESSAGE TO R0.
946 003136 112037 003242 TYP: MOVB (0),TYPDAT ;GET CHARACTER
947 003142 122737 000100 003242 CMPB #100,TYPDAT ;CHECK FOR "0" CHARACTER
948 003150 001003 BNE TYPC ;BRANCH IF NOT "0".
949 003152 013700 003312 MOV SAVR0,%0 ;RESTORE R0
950 003156 000002 RTI ;TERMINATOR CHAR. DONE. EXIT.
951 003160 122737 000045 003242 TYP: CMPB #45,TYPDAT ;CHECK FOR "X".
952 003166 001412 BEQ TYPF ;BRANCH IF "X".
953 003170 004737 003176 JSR #7,TYPD ;TYPE CHAR IN TYPDAT
954 003174 000760 BR TYPB
955 003176 113777 003242 176022 TYPD: MOVB TYPDAT,%TPB ;OUTPUT CHARACTER TO PRINTER
956 003204 105777 176014 TSTB #TPB ;WAIT FOR DONE FLAG.
957 003210 100375 BPL #-4
958 003212 000207 RTS #7 ;EXIT
959 003214 112737 000015 003242 TYPF: MOVB #15,TYPDAT ;MOVE CARRIAGE RETURN CODE TO TYPDAT
960 003222 004737 003176 JSR #7,TYPD ;GO TYPE CHAR.
961 003226 112737 000012 003242 MOVB #12,TYPDAT ;MOVE LP CODE TO TYPDAT.
962 003234 004737 003176 JSR #7,TYPD ;GO TYPE CHAR.
963 003240 000736 BR TYPB
964 003242 000000 TYPDAT: OPEN
965
966 ;SUBROUTINE TO OUTPUT A SERIES OF ASCII MESSAGES ON TELETYPE PRINTER
967 003244 010037 003312 TYP: MOV %0,SAVR0
968 003250 011600 TYP: MOVB #2,%6 ;SET ADDRESS THAT CONTAINS MESSAGE ADDRESS
969 003252 062716 000002 ADD #2,%6 ;UPDATE TO NEXT MESSAGE ADDRESS
970 003256 011037 003306 MOV #2,TYPB ;ADDRESS OF MESSAGE TO TYPB
971 003262 022737 177777 003306 CMP #1,TYPB ;CHECK FOR TERMINATOR
972 003270 001003 BNE TYPB ;BRANCH IF NOT TERMINATOR.
973 003272 013700 003312 MOV SAVR0,%0 ;RESTORE R0
974 003276 000002 RTI ;TERMINATOR, EXIT
975 003300 013700 003312 TYP: MOV SAVR0,%0
976 003304 104000 TYPE ;CALL ON TYP SUB TO TYPE MESSAGE
977 003306 000000 TYP: OPEN ;ADDRESS OF MESSAGE GOES HERE
978 003310 000757 BR TYPB ;GO PROCESS NEXT MESSAGE
979 003312 000000 SAVR0: OPEN
980 ;SUBROUTINE TO READ OCTAL DATA FROM THE TELETYPE PRINTER

```

```

981 003314 011646          RDOCT: MOV      (SP),-(SP)      ;MAKE ROOM FOR DATA WORD
982 003316 010046          MOV      X0, -(SP)          ;SAVE R0
983 003320 010146          MOV      X1, -(SP)          ;SAVE R1
984 003322 005001          INDAT: CLR      X1          ;CLEAR DATA WORD
985 003324 005037 001324  RDOCT: CLR      COUNT        ;SET NO. OF DIGITS = 0
986 003330 105777 175664  RDDAT: TSTB     @TKS         ;TEST TTY READ STATUS
987 003334 100375          BPL      RDDAT            ;WAIT
988 003336 117746 175660  RDDAT: MOVB     @TKB, -(SP)    ;PUSH DIGIT ON STACK
989 003342 042716 000200  RDDAT: BIC      @BIT7, (SP)   ;
990 003346 105777 175652  ECDAT: TSTB     @TP8         ;TEST TTY PRINT STATUS
991 003352 100375          BPL      ECDAT            ;WAIT
992 003354 111677 175646  RDDAT: MOVB     (SP), @TPB    ;ECHO CHARACTER
993 003360 122716 000015  RDDAT: CMPB     @15, (SP)     ;IS IT A TERMINATOR?
994 003364 001432          BEQ      RETRN            ;BR IF YES
995 003366 122716 000177  RDDAT: CMPB     @177, (SP)    ;IS IT A RUBOUT?
996 003372 001423          BEQ      RREAD           ;BR IF YES
997 003374 122716 000060  RDDAT: CMPB     @60, (SP)     ;IS IT AN OCTAL DIGIT?
998 003400 003020          BEQ      RREAD           ;BR IF NO
999 003402 122716 000067  RDDAT: CMPB     @67, (SP)    ;TEST AGAIN
1000 003406 002413          BLT      RREAD           ;BR IF NO
1001 003410 005237 001324  RDDAT: INC      COUNT        ;INC NO. OF DIGITS
1002 003414 022737 000067 001324  RDDAT: CMP      @67, COUNT    ;MORE THAN SIX DIGITS?
1003 003422 003407          BLE      RREAD           ;BR IF YES
1004 003424 006301          ASL      X1              ;CLEAR LOWEST THREE BITS
1005 003426 006301          ASL      X1              ;OF DATA WORD
1006 003430 006301          ASL      X1              ;
1007 003432 162716 000060  RDDAT: SUB      @60, (SP)     ;CONVERT TO BINARY
1008 003436 062601          ADD      (SP)+, X1        ;ADD DIGIT TO DATA WORD
1009 003440 000733          BR       RDDAT           ;GET NEXT DIGIT
1010 003442 104000          RREAD: TYPE     DTERR      ;TELL USER ABOUT ILLEGAL CHARACTER
1011 003444 013232          TST      (SP)+           ;GET RID OF ILLEGAL CHARACTER
1012 003446 005726          BR       INDAT           ;START SUBROUTINE AGAIN
1013 003450 000724          MOV      X1, @10 (SP)    ;STORE DATA WORD ON STACK
1014 003452 010166 000010  RDDAT: TST      (SP)+           ;INC STACK POINTER
1015 003456 005726          MOV      (SP)+, X1       ;RESTORE R1
1016 003460 012601          MOV      (SP)+, X0       ;RESTORE R0
1017 003462 012600          RTS      PC              ;RETURN
1018 003464 000207          ;
1019          ;
1020          ;SUBROUTINE TO DELAY A SPECIFIED NUMBER OF MILLISECONDS.
1021 003466 011637 003526  DLY:  MOV      @X0, @DLCNT   ;GET DELAY COUNT ADDRESS.
1022 003472 062716 000002  DLY:  ADD      @2, @X0      ;SET UP EXIT ADDRESS
1023 003476 017746 000024  DLY:  MOV      @DLCNT, -(6) ;DELAY COUNT TO STACK
1024 003502 001407          BEQ      DLYC            ;
1025 003504 012746 000226  DLYA: MOV      @226, -(6)    ;1 MSEC COUNT TO STACK
1026 003510 005316  DLYB: DEC      @X6         ;DECREMENT 1 MSEC COUNT
1027 003512 001376  DLYB: BNE      DLYB         ;BRANCH IF NOT 0.
1028 003514 005726  DLYB: POPSP     ;ZERO, UNCOVER MSEC. COUNT.
1029 003516 005316  DLYB: DEC      @X6         ;DECREMENT IT
1030 003520 001371  DLYB: BNE      DLYA         ;BR IF NOT DONE DELAYING
1031 003522 005726  DLYC: POPSP     ;DONE
1032 003524 000002  DLYC: RTI          ;EXIT.
1033 003526 000000  DLYC: OPEN     ;CONTAINS MILLISECONDS COUNT ADDRESS.
1034          ;

```

```

1035          ;OCTAL TO ASCII CONVERT ROUTINE
1036 003530 104011  OACNV: SAVREG   ;SAVE REGS.
1037 003532 013500  OACNV: MOV      @5, X0      ;GET OCTAL VALUE.
1038 003534 012501  OACNV: MOV      (5)+, X1    ;GET DESTINATION ADDR.
1039 003536 012502  OACNV: MOV      (5)+, X2    ;GET CONVERT COUNT.
1040 003540 060201  OACNV: ADD      X2, X1      ;DEVELOP ADDR TO STORE 1ST CHAR.
1041 003542 010003  OACNV: MOV      X0, X3     ;
1042 003544 042703 177770  OACNV: BIC      @177770, X3  ;ISOLATE LEAST SIGNIFICANT DIGIT.
1043 003550 062703 000060  OACNV: ADD      @60, X3     ;CONVERT DIGIT TO ASCII.
1044 003554 110341  OACNV: MOVB     X3, -(1)    ;STORE ASCII CHARACTER.
1045 003556 042700  OACNV: BIC      @7, X0     ;
1046 003562 006000  OACNV: ROR      X0         ;
1047 003564 006000  OACNV: ROR      X0         ;
1048 003566 006000  OACNV: ROR      X0         ;
1049 003570 005302  OACNV: DEC      X2         ;DONE ALL DIGITS?
1050 003572 001363  OACNV: BNE      OACNV      ;BRANCH IF NOT DONE.
1051 003574 104012  OACNV: RSTREG   ;RESTORE REGS.
1052 003576 000205  OACNV: RTS      X5         ;DONE, EXIT.
1053          ;SUBROUTINE TO GENERATE PARITY ON DATA FOR 5,6,7,8 LEVEL CODE.
1054          ;PARITY BIT IS THE MSB OF THE CHARACTER PARITY CAN BE EITHER
1055          ;EVEN OR ODD
1056          ;GENERATES ODD/EVEN PARITY.

```

```

1057
1058 003600 032737 000200 001362 GENPAR: BIT #BIT7,CARMSK ;TEST LSR CHAR LENGTH
1059 003606 001411 REQ EIGHT ;CHAR IS 8
1060 003610 032737 000100 001362 BIT #BIT6,CARMSK ;TEST MSB CHAR LENGTH
1061 003616 001427 REQ SEVEN ;CHAR LENGTH IS 7
1062 003620 032737 000040 001362 BIT #BIT5,CARMSK
1063 003626 001412 REQ SIX
1064 003630 000433 BR FIVE
1065 003632 012737 000200 001322 EIGHT: MOV #BIT7,PARBIT ;PLACE PARITY BIT IN PROPER POSITION
1066 003640 012737 000007 001324 MOV #7,COUNT ;SET UP ROTATE COUNTER=7
1067 003646 042701 177600 BIC #177600,X1 ;MASK OFF UNUSED BITS
1068 003652 000433 BR ;GO AND GENERATE PARITY FOR 8
1069 003654 012737 000040 001322 SIX: MOV #BIT5,PARBIT ;PLACE PARITY BIT IN PROPER POSITION
1070 003662 012737 000005 001324 MOV #5,COUNT ;SET UP ROTATE COUNTER=5
1071 003670 042701 177740 BIC #177740,X1 ;MASK OFF UNUSED BITS
1072 003674 000422 BR ;GO AND GENERATE PARITY FOR
1073 003676 012737 000100 001322 SEVEN: MOV #BIT6,PARBIT ;PLACE PARITY BIT IN PROPER POSITION
1074 003704 012737 000006 001324 MOV #6,COUNT ;SET UP ROTATE COUNTER=6
1075 003712 042701 177700 BIC #177700,X1 ;MASK OFF UNUSED BITS
1076 003716 000411 BR ;GO AND GENERATE PARITY FOR 7
1077 003720 012737 000020 001322 FIVE: MOV #BIT4,PARBIT ;PLACE PARITY BIT IN PROPER POSITION
1078 003726 012737 000004 001324 MOV #4,COUNT ;SET UP ROTATE COUNTER=4
1079 003734 042701 177760 BIC #177760,X1 ;MASK OFF UNUSED BITS
1080 003740 000400 BR ;GO AND GENERATE PARITY FOR
1081 003742 010137 001326 DOIT: MOV X1,SAVE ;SAVE DATA
1082 003746 006001 AGAIN: ROR X1 ;ROTATE DATA
1083 003750 103415 BCS ADD1 ;IF CARRY SET ADD IN PARBIT
1084 003752 005337 001324 RTN: DEC COUNT ;DECREMENT COUNTER
1085 003756 001373 BNE AGAIN ;NOT DONE DO IT AGAIN
1086 003760 032737 000040 001374 BIT #BITS,8RT ;DONE EVEN OR ODD PARITY?
1087 003766 001403 BEQ DONE ;IF EVEN FINISHED
1088 003770 063737 001322 001326 ADD PARBIT,SAVE ;IF ODD ADD IN ANOTHER 1
1089 003776 013701 001326 DONE: MOV SAVE,X1 ;PLACE DATA + PARITY BACK IN R1
1090 004002 000207 RTS ;AND EXIT
1091 004004 063737 001322 001326 ADD1: ADD PARBIT,SAVE ;ADD PARBIT TO DATA
1092 004012 000757 BR RTN ;RETURN TO COUNTER
1093
1094 ;SUBROUTINE TO SELECT LINE AND LOAD VECTOR ASSIGNMENT
1095 004014 104000 LINSSEL: TYPE
1096 004016 014256 LLINE
1097 004020 004737 003314 JSR PC,RDOCT ;GET INPUT
1098 004024 012601 MOV #0P,X1 ;LOAD R1
1099 004026 042701 177407 BIC #177407,X1 ;MASK OFF ALL BUT LINE BITS
1100 004032 006201 ASR X1
1101 004034 004201 ASR X1
1102 004036 010137 001372 MOV X1,TEMP ;SAVE LINE #
1103 004042 012703 001200 MOV #RXCSR,X3 ;LOAD ADDRESS OF REGISTERS
1104 004046 012704 000004 MOV #4,X4 ;SET UP COUNTER
1105 004052 016102 010432 MOV RCSR(1),X2
1106 004056 010223 LINSB: MOV X2,(X3)+
1107 004064 062702 000002 ADD #2,X2
1108 004066 005304 DEC X4
1109 004068 001373 BNE LINSB
1110 004070 016101 010334 MOV VECTAB(1),X1 ;GET LINE VECTOR ADDRESS

```

```

1111 004074 010123 MOV X1,(X3)+ ;LOAD INTO PROG, RXVTR
1112 004076 022121 CMP (1)+,(1)+ ;ADD +4 TO RXVTR TO = TXVTR
1113 004100 005723 TST (3)+ ;POINT TO PROG TXVTR
1114 004102 010113 MOV X1,(X3) ;LOAD INTO PROG TXVTR
1115 004104 022737 000005 001240 CMP #5,PRGNUM ;RUNNING PROGRAM # 5
1116 004112 001001 BNE +4
1117 004114 000205 RTS ;RETURN TO PROG 5
1118 004116 000237 001372 ASR TEMP ;POSITION
1119 004122 CNVDA TEMP,TLINEX,2
1120 004134 104000 TYPE ;TYPE LINE # THAT
1121 004136 013665 ALINEX ;WAS CALLED
1122 004140 000205 RTS
1123
1124 ;SUBROUTINE TO LOAD BINARY COUNT PATTERN INTO OUTPUT BUFFER
1125 004142 105037 001334 INFIL: CLR# NUMBER ;INITIALIZE BINARY COUNT
1126 004146 012500 FILL: MOV (X)+,X0 ;GET ADDRESS
1127 004150 012537 001364 MOV (X)+,CYRD ;GET COUNT
1128 004154 113720 001334 FILL: MOV# NUMBER,(X)+ ;LOAD ADDRESS WITH BINARY COUNT
1129 004160 105237 001334 INCB NUMBER ;INC. BINARY COUNT
1130 004164 005337 001364 DEC CYRD ;DEC. COUNT
1131 004170 001371 BNE FILL
1132 004172 000205 RTS ;EXIT
1133
1134 ;SUBROUTINE TO MOVE A VARIABLE NUMBER OF BYTES.
1135 004174 104011 BMOVE: SAVREG ;SAVE REGS.
1136 004176 012501 MOV (X)+,X1 ;GET FROM ADDRESS
1137 004200 012502 MOV (X)+,X2 ;GET TO ADDRESS
1138 004202 012503 MOV (X)+,X3 ;GET COUNT
1139 004204 112122 BMOVE: MOV# (1)+,(2)+ ;MOVE BYTE
1140 004206 005303 DEC X3 ;DECREMENT COUNT
1141 004210 001375 BNE BMOVE ;BRANCH IF NOT DONE.
1142 004212 104012 RSTREG ;RESTORE REGS.
1143 004214 000205 RTS ;DONE EXIT
1144
1145 ;BINARY TO DECIMAL ASCII CONVERT SUBROUTINE.
1146 004216 104011 BDCNV: SAVREG ;SAVE REGS.
1147 004220 012700 004374 MOV #DECVAL,X0 ;SET UP ADDR TO STORE DECIMAL ASCII IN R0
1148 004224 013501 MOV #0(X)+,X1 ;BINARY VALUE TO R1.
1149 004226 012537 004304 MOV (X)+,BDCNVC ;DESTINATION ADDR TO BDCNVC.
1150 004232 012537 004306 MOV (X)+,BDCNVD ;COUNT TO BDCNVD.
1151 004236 012702 004362 MOV #ADTENP,X2 ;ADDR OF TEN POWER STRING TO R2.
1152 004242 012737 000005 004354 MOV #5,CNVCTR ;SET UP FOR 5 POWER CONVERSIONS.
1153 004250 012237 004360 BDCNVA: MOV (X)+,TENPWR ;MOVE POWER OF TEN VALUE TO TENPWR.
1154 004254 004737 004314 JSR X7,SUBTEN ;PERFORM CONVERSION
1155 004260 005337 004354 DEC CNVCTR ;DONE 5 CONVERSIONS?
1156 004264 001371 BNE BDCNVA ;BRANCH IF NOT YET 5.
1157 004266 163700 004306 SUB BDCNVD,X0 ;SET UP ADDR TO MOVE DECIMAL
1158 004272 010037 004302 MOV X0,BDCNVD ;DATA FROM.
1159 004276 004337 004174 JSR X5,BMOVE ;MOVE DECIMAL DATA TO DESTINATION.
1160 004302 000000 BDCNVA: OPEN ;SRC ADDR.
1161 004304 000000 BDCNVC: OPEN ;DEST ADDR.
1162 004306 000000 BDCNVD: OPEN ;COUNT.
1163 004310 104012 RSTREG ;RESTORE REGS.
1164 004312 000205 RTS ;YES, EXIT.

```

```
1165 004314 005037 004356 SUBTEN1 CLR DIGIT JCLEAR DIGIT
1166 004320 163701 004360 SUBTNA1 SUB TENPWR,X1 JSUBTRACT TEN POWER FROM BINARY VALUE.
1167 004324 103403 RCS SURTNB JBRANCH IF UNSUCCESSFUL SUBTRACTION.
1168 004326 005237 004356 INC DIGIT
1169 004332 000772 RR SUBTNA
1170 004334 063701 004360 SUBTNB1 ADD TENPWR,X1 JRESTORE SUBTRACTED VALUE.
1171 004340 062737 000060 004356 ADD #60,DIGIT JCONVERT (DIGIT) TO ASCII
1172 004346 113720 004356 MOVB DIGIT,(0)+ JMOVE ASCII CHAR TO DECIMAL FIELD.
1173 004352 000207 RTS X7 JEXIT.
1174 004354 000000 CNVCTR1 OPEN
1175 004356 000000 DIGIT1 OPEN
1176 004360 000000 TENPWR1 OPEN
1177 004362 023420 ADTENP1 10000.
1178 004364 001750 1000.
1179 004366 000144 100.
1180 004370 000012 10.
1181 004372 000001 1
1182 004374 000 040 040 DECVAL1 .BYTE 040,040,040,040,040,040
1183 004377 000 040 040
1184
1185 004402 104000 JSUBROUTINE TO SET CHARACTER LENGTH PARAMETER
1186 004404 013055 SETPAR1 TYPE JTYPE1 SELECT PARAMETERS.
1187 004406 004737 003314 JSR PC,RDOCT JGET INPUT
1188 004412 012637 001374 MOV (8P)+,SRT
1189 004416 CNVOA SRT,APARM,3
1190 004430 104000 TYPE
1191 004432 013355 PARMTB
1192 004434 012737 177400 001362 TBIT11 MOV #177400,CARMSK JSET CHARACTER MASK TO 8 BITS.
1193 004442 032737 000002 001374 BIT #BIT1,SRT JSEE IF SR BIT 1 IS SET.
1194 004450 001413 BEQ STPARA JBRANCH IF NOT SET.
1195 004452 012737 177700 001362 MOV #177700,CARMSK JCHANGE CHAR MASK TO 6 BITS.
1196 004460 032737 000001 001374 BIT #BIT0,SRT JSEE IF SR BIT0 IS SET.
1197 004466 001403 BEQ PAREX JBRANCH IF NOT SET.
1198 004470 012737 177740 001362 MOV #177740,CARMSK JCHANGE CHAR MASK TO 5 BITS.
1199 004476 000207 PAREX1 RTS JEXIT.
1200 004500 032737 000001 001374 STPARA1 BIT #BIT0,SRT JSEE IF SR BIT0 IS SET.
1201 004506 001773 BEQ STPARA-2 JBRANCH IF NOT SET.
1202 004510 012737 177600 001362 MOV #177600,CARMSK JCHANGE CHAR MASK TO 7 BITS.
1203 004516 000767 BR PAREX
1204
1205 JERROR TRAP HANDLER = TYPE TO AND FROM WHERE ERROR TRAP OCCURS
1206 004520 013737 177776 001336 ERTP1 MOV PSW,OLDP5 JSAVE OLDP5
1207 004526 012737 000340 177776 MOV #PRTY7,PSW
1208 004534 006237 001336 ASR OLDP5
1209 004540 006237 001336 ASR OLDP5
1210 004544 006237 001336 ASR OLDP5
1211 004550 042737 177740 001336 BIC #177740,OLDP5
1212 004556 013737 001336 001340 MOV OLDP5,TOPC
1213 004564 011637 001342 ERTPA1 MOV #%,FROMPC
1214 004570 004537 003530 JSR X5,OACNV
1215 004574 001340 TOPC
1216 004576 012314 MTO
1217 004600 000006 6
1218 004602 004537 003530 JSR X5,OACNV
```

```
1219 004606 001342 FROMPC
1220 004610 012346 MFROM
1221 004612 000006 6
1222 004614 104000 TYPE
1223 004616 012247 MTERR
1224 004620 000000 HALT
1225 004622 000137 001422 JMP START
1226
1227 JMAPVEC = MAP VECTOR VECTOR OR REPORT ERROR DEPENDING ON PHAP FLAG
1228 004626 011637 001340 MAPVEC1 MOV #%,TOPC
1229 004632 022626 POP0P2
1230 004634 011637 001342 MOV #%,FROMPC
1231 004640 162737 000004 001340 SUR #4,TOPC
1232 004646 005737 001350 TST PHAP
1233 004652 001746 BEQ ERTPA JNOT MAPPING, REPORT ERROR
1234 004654 013737 001340 001214 MOV TOPC,TVXVTR JSTORE VECTOR
1235 004662 162737 000004 001340 SUR #4,TOPC
1236 004670 013737 001340 001210 MOV TOPC,RXVTR
1237 004676 005037 001350 CLR PHAP
1238 004702 000002 RTI
1239
1240 JFORMAD = FORM DEVICE AT ADDRESS
1241 004704 010137 001200 FORMAD1 MOV X1,RXC5R
1242 004710 062701 000002 ADD #2,X1
1243 004714 010137 001202 MOV X1,RXBUF
1244 004720 062701 000002 ADD #2,X1
1245 004724 010137 001204 MOV X1,TXCSR
1246 004730 062701 000002 ADD #2,X1
1247 004734 010137 001206 MOV X1,TXBUF
1248 004740 000207 RTS X7
1249
1250 JSUBROUTINE TO MAKE LINE CONNECTION.
1251 004742 017737 174232 001370 LINCON1 MOV #RXC5R,RXC5RT
1252 004750 032737 020000 001370 BIT #BIT13,RXC5RT JYES, TO CLEAR TO SEND UP
1253 004756 001006 BNE LINEUP JYES CONNECTION IS MADE.
1254 004760 042777 000146 174212 LINCA1 BIC #146,#RXC5R JCLEAR IE BIT AND DTR, RG TO SND
1255 004766 005777 174210 TST #RXBUF JCLEAR DONE FLAG
1256 004772 104000 JTYPE
1257 004774 013251 MAKECON JMAKE LINE CONNECTION
1258 004776 017737 174176 001370 LINOB1 #RXC5R,RXC5RT
1259 005004 032737 040000 001370 BIT #BIT14,RXC5RT JDO YOU RING
1260 005012 001771 BEQ LINCB JGO WAIT FOR RING
1261 005014 052777 000006 174156 B18 #6,#RXC5R JSET DTR, RG TO SND
1262 005022 104016 DELAY JWAIT 10 SECONDS FOR
1263 005024 023420 10000. JCLEAR TO SEND
1264 005026 017737 174146 001370 MOV #RXC5R,RXC5RT
1265 005034 005777 174142 TST #RXBUF JCLEAR DONE
1266 005040 032737 020000 001370 BIT #BIT13,RXC5RT JIS CLEAR TO SEND UP?
1267 005046 001003 BNE LINCPC JYES, GO TO LINCPC
1268 005050 104015 ERRR1 JNO, PRINT ERROR MESSAGE
1269 005052 013377 LINCMM JCLEAR TO SEND NOT SET
1270 005054 000741 BR LINCPC JSTART OVER AGAIN
1271 005056 017737 174116 001370 LINCPC1 MOV #RXC5R,RXC5RT JCLEAR ALL FLAGS
1272 005064 005777 174112 TST #RXBUF JAND DONE
```

```

1273 005070 104000          TYPE          ITYPE MESSAGE
1274 005072 013427          LINMAD        I'CONNECTION IS MADE'
1275 005074 000205          LINEUP: RTS    IEXIT LINE CONNECTION ROUTINE WITH
1276                               I'SUBROUTINE TO OVERLAY <CRLF> IN DATA PATTERN (EVERY 72,ND CHAR)
1277 005076 012701 014710  OVRLAY: MOV     #OUTBUF,X1 IGET OUTBUF ADDRESS
1278 005102 012702 000016  OVRLYA: MOV     #14,,X2 IGET COUNTER
1279 005106 012711 105215  OVRLYA: MOV     #105215,(1) IINSERT CRLF
1280 005112 062701 000110  ADD         #72,,X1 IADD OFFSET
1281 005116 005302          DEC         X2 IDONE?
1282 005120 001372          RNE        OVRLYA
1283 005122 000207          RTS         7 IEXIT
1284
1285                               IRECEIVER INTERRUPTS COMMON HANDLER
1286 005124 000240          RISR: NOP
1287 005126 010037 001330  MOV         #0,LINE
1288 005132 006300          ASL        X0
1289 005134 016037 010432 001200  MOV     RCSR(0),RCSR IGET ADDRESS OF INTERRUPTING DL11=E'S RCSR
1290 005142 017737 174032 001370  MOV     @RCSR,RXCSR IGET CSR CONTENTS
1291 005150 100570          BMI        DCERR ICHECK INT
1292 005152 105737 001370  T8TB      RXCSR ITEST DONE
1293 005156 001002          BNE        RISRA
1294 005160 104003          ERROR      IFALSE INTERRUPT
1295 005162 000002          RTI        IEXIT
1296 005164 020037 001404  RISRA: CMP     #0,CALLED IDID CALLED LINE INTERRUPT?
1297 005170 001020          BNE        RISRB IBRANCH IF CALLER INTERRUPTED
1298 005172 005737 001412  MODEM     T8T ICHECK MODEM TYPE
1299 005176 001403          BEQ        RISRAA IBRANCH IF 103
1300 005200 005770 010530  T8T       @RBUF(0) IREAD CALLED LINES DATA
1301 005204 000002          RTI
1302 005206 117077 010530 174162  RISRAA: MOVB  @RBUF(0),@INBUFP ISTORE CHARACTER IN INPUT BUFFER
1303 005214 005237 001376          INC        INBUFP IINCREMENT POINTER
1304 005220 022737 017024 001376  CMP     #INBUFP+100,,INBUFP IHAVE 100, CHARACTERS BEEN RECEIVED?
1305 005226 001430          BEQ        RISRC IGO CHECK DATA IF YES
1306 005230 000002          RTI        IEXIT IF NO
1307 005232 117077 010530 174140  RISRB: MOVB  @RBUF(0),@BUFP ISTORE CHARACTER IN INTERMEDIATE DATA BUFFER
1308 005240 005237 001400          INC        BUFP IINCREMENT POINTER
1309 005244 022737 015066 001400  CMP     #BUFP+10,,BUFP IHAVE 10 CHARACTERS BEEN RECEIVED
1310 005252 002401          BLT        ,+4
1311 005254 000002          RTI        IEXIT
1312 005256 022737 000002 001332  CMP     #2,CONFIG IRUNNING CONFIGURATION 2?
1313 005264 001405          BEQ        RISRB
1314 005266 022737 015220 001400  CMP     #BUFP+100,,BUFP IHAVE 100, CHARACTERS BEEN RECEIVED?
1315 005274 001405          BEQ        RISRC IGO CHECK DATA IF YES, OTHERWISE
1316 005276 000002          RTI        IEXIT
1317 005300 052770 000100 010626  RISRB: BIS     #BIT6,@TCR(0) ISTART CALLERS TRANSMITTER
1318 005306 000002          RTI        IEXIT
1319
1320                               ICHECK DATA CONFIGURATION #1
1320 005310 000240          RISR: NOP
1321 005312 012737 000001 001364  MOV     #1,CTRD IINITIALIZE CHARACTER COUNT
1322 005320 012702 015054          MOV     @BUFP,X2 IPOINT R2 TO CALLERS RECEIVED DATA BUFFER
1323 005324 012703 014710          MOV     #OUTBUF,X3 IR3 = FIRST ADDRESS OF OUTPUT DATA BUFFER
1324 005330 010237 001400          MOV     X2,BUFP IRESTORE CALLERS RCVD DATA BUFFER PTR
1325 005334 022737 000001 001332  CMP     #1,CONFIG ICHECK CONFIGURATION
1326 005340 001015          BNE        RISRD

```

```

1327 005344 112337 001360  RISRCA: MOVB  (3)+,XMYDAT IGET TRANSMITTED CHARACTER
1328 005350 112237 001356  MOVB  (2)+,RECDAT IGET RECEIVED CHARACTER
1329 005354 104004          DATCHK
1330 005356 005237 001364          INC        CTRD IINCREMENT CHARACTER COUNT
1331 005362 022737 000101 001364  CMP     #101,CTRD IHAS ALL DATA BEEN CHECKED
1332 005370 001365          BNE        RISRCA
1333 005372 000137 005476          JMP        FINISH
1334
1335                               ICHECK DATA CONFIGURATION #2
1336 005376 000240          RISRD: NOP
1337 005400 012704 016660          MOV     #INBUF,X4 IPOINT R4 TO CALLED LINES RECEIVER
1338 005404 010437 001376          MOV     #4,INBUFP IDATA BUFFER & INIT, POINTER
1339 005410 012737 015054 001410  MOV     @BUFP,@BUFP
1340 005416 013701 001404          RISRDA: MOV     CALLED,X1
1341 005422 016137 010432 001200  MOV     RCSR(1),RCSR
1342 005430 112337 001360          MOVB  (3)+,XMYDAT
1343 005434 112237 001356          MOVB  (2)+,RECDAT ICOMPARE TRANSMITTED DATA WITH DATA
1344 005440 104004          DATCHK IRECEIVED BY CALLED LINE
1345 005442 013701 001402          MOV     CALLER,X1
1346 005446 016137 010432 001200  MOV     RCSR(1),RCSR
1347 005454 112437 001356          MOVB  (4)+,RECDAT ICOMPARE TRANSMITTED DATA WITH DATA
1348 005460 104004          DATCHK IRECEIVED BY CALLER
1349 005462 005237 001364          INC        CTRD
1350 005466 022737 000101 001364  CMP     #101,CTRD
1351 005474 001350          BNE        RISRDA
1352 005476 000240          FINISH: NOP
1353 005500 013701 001404          MOV     CALLED,X1
1354 005504 004537 004146          JSR     S,FILL
1355 005510 014710          OUTBUF
1356 005512 000144          100,
1357 005514 104000          TYPE
1358 005516 013350          ENDPAS
1359 005520 052771 000100 010626  BIS     #BIT6,@TCR(1)
1360 005526 000240          NOP
1361 005530 000002          RTI
1362
1363 005532 052770 100000 010530  IERROR SERVICE: ROUTINE
1364 005540 001402          DCERR: BIT   #BIT15,@RBUF(0) ITEST ERROR
1365 005544 104015          BEQ        RISRP
1366 005544 012337          ERRRX  CSRADD
1367 005546 012737 014710 001406  RISRP: MOV     #OUTBUF,@OBUFP ISET OUTPUT BUFFER POINTER
1368 005554 012737 016660 001376  MOV     #INBUF,INBUFP ISET INPUT BUFFER POINTER
1369 005562 012737 015054 001400  MOV     #BUFP,BUFP ISET INTERMEDIATE BUFFER POINTER
1370 005570 012737 015054 001410  MOV     #BUFP,@BUFP ISET POINTER FOR CONFIG #2 TRANSMITTER
1371 005576 032737 000000 001370  BIT     #BIT16,RXCSR ICHECK RING INDICATOR
1372 005604 001005          BNE        RISREX IBRANCH IF RING
1373 005606 000737 006234          JSR     7,DISCON IERROR SET = NO RING
1374 005612 104015          ERRRX  CSRADD
1375 005614 012337          RTI
1376 005616 000002          RISREX: CNVDA LINE,TLINE,2
1377 005620          TYPE
1378 005632 104000          ALTNE
1379 005634 013715          MOV     #0,CALLED
1380 005636 010037 001404          MOV     #0,CALLED

```



```

1381 005642 004737 003314 JSR PC,ROOCT ;GET INPUT
1382 005646 011637 001412 MOV (#P),MODEM ;GET MODEM TYPE
1383 005652 004737 177773 001412 RIC #177773,MODEM ;0=103,4=202
1384 005660 012637 001332 MOV (#P)+,CONFIG
1385 005664 004737 177774 001332 RIC #177774,CONFIG
1386 005672 001042 ANF RISRFC ;GO TO SERVICE FOR CONFIG 1 OR 2
1387 005674 004737 006250 JSR 7,CONN ;CONNECT LINE IF CONFIGURATION 0
1388 005700 104000 TYPE ;TYPE MESSAGE TO PRESS DATA
1389 005702 014215 BUTTON ;BUTTON ON DATA PHONE
1390 005704 104016 DELAY ;WAIT FOR CARRIER
1391 005706 023420 10000. ;10 SECONDS
1392 005710 005770 010530 TST #RBUF(0) ;READ BUFFER TO CLEAR DONE
1393 005714 032770 020000 010432 BIT #BIT13,#RCSR(0) ;TEST FOR CLEAR TO SEND
1394 005722 001000 BNE RISRFB
1395 005724 104003 ERROR ;ERROR! DID NOT RECEIVE CLEAR TO SEND
1396 ;WITHIN TIME ALLOTTED (10 SEC.)
1397 005726 004737 006234 JSR 7,DISCON ;DISCONNECT LINE
1398 005732 000002 RTI ;AND EXIT
1399 005734 016037 010626 001204 RISRFB: MOV TCSR(0),TXCSR ;GET CALLED LINES TXCSR ADDRESS
1400 005742 004737 004402 JSR 7,SETPAR ;LOAD USER PARAMETERS
1401 005746 104000 TYPE ;TYPE 'LINE CONNECTION
1402 005750 013427 LINMAD ;MADE'
1403 005752 CNVDA CONFIG,TCONFIG,2
1404 005764 104000 TYPE
1405 005766 013456 ACONFIG
1406 005770 052770 000100 010626 BIS #BIT6,#TCSR(0)
1407 005776 000002 RTI ;AND EXIT
1408
1409 ;
1410 ;HERE IF CONFIGURATION 1 OR 2
1411 006000 104000 RISRFC: TYPE ;TASK USER WHICH LINE HE IS
1412 006002 014065 WRU ;DIALING ON
1413 006004 004737 003314 JSR PC,ROOCT ;GET INPUT
1414 006010 012601 012601 MOV (#P)+,X1 ;GET LINE #
1415 006012 004701 177740 BIC #177740,X1 ;MASK UNUSED BITS
1416 006016 010137 001330 MOV X1,LINE
1417 006022 CNVDA LINE,URA,2
1418 006024 TYPE ;REPORT LINE # ON TTY
1419 006026 UR
1420 006028 ASL X1
1421 006030 MOV X1,CALLER ;SAVE CALLERS LINE #
1422 006032 004737 006250 JSR 7,CONN ;CONNECT CALLED LINE
1423 006034 052771 000002 010432 BIS #BIT1,#RCSR(1) ;SET DTR ON CALLERS LINE
1424 006036 104000 TYPE ;TYPE MESSAGE TO PRESS DATA
1425 006038 014215 BUTTON ;ON DATA PHONE
1426 006040 104016 DELAY ;WAIT 10 SECONDS FOR CLEAR TO SEND
1427 006042 023420 10000. ;SET AT CALLED LINE
1428 006044 027071 010530 010530 CMP #RBUF(0),#RBUF(1) ;READ BUFFERS
1429 006046 032770 020000 010432 BIT #BIT13,#RCSR(0) ;TEST FOR CLEAR TO SEND AT CALLED LINE
1430 006048 001007 BNE RISRFB
1431 006050 104003 ERROR ;ERROR! CLEAR TO SEND NOT SET AT CALLED LINE
1432 006110 004737 006234 RISRFB: JSR 7,DISCON ;DISCONNECT
1433 006114 004771 000006 010432 BIC #6,#RCSR(1) ;LINE
1434 006122 000002 RTI ;AND EXIT
1435 006124 022737 000002 001332 RISRFB: CMP #2,CONFIG

```

```

1435 006132 001414 BEO RISRFB
1436 006134 022771 010000 010432 CMP #BIT12,#RCSR(1) ;CHECK CARRIER AT CALLERS LINE
1437 006142 001003 BNE RISRFB
1438 006144 104003 ERROR ;ERROR! NO CARRIER AT CALLERS LINE
1439 006146 000137 006110 JMP RISRFB ;GO DISCONNECT LINES
1440 006152 016137 010432 001200 RISRFB: MOV RCSR(1),RXCSR
1441 006160 000137 005734 JMP RISRFB ;GO GET PARAMETERS AND ENABLE
1442 ;CALLED TRANSMITTER AND EXIT
1443
1444 ;
1445 006164 032771 020000 010432 ;HERE IF CONFIGURATION 2
1446 006172 001003 RISRFB: BIT #BIT13,#RCSR(1) ;TEST CALLERS CLEAR TO SEND
1447 006174 104003 BNE RISRFB
1448 006176 000137 006110 JMP RISRFB ;ERROR! NO CTS AT CALLERS LINE
1449 006202 016037 010432 001200 RISRFB: MOV RCSR(0),RXCSR ;GO DISCONNECT LINE AND EXIT
1450 006210 016137 010626 001204 MOV TCSR(1),TXCSR
1451 006216 004737 004402 JSR 7,SETPAR ;GO GET PARAMETERS FOR CALLERS
1452 ;TRANSMITTER AND CALLED RECEIVER
1453 006222 016137 010432 001200 MOV RCSR(1),RXCSR
1454 006230 000137 005734 JMP RISRFB
1455
1456 ;
1457 006234 004770 000006 010432 ;SUBROUTINE TO DISCONNECT LINE R0 HAS LINE #
1458 006242 005770 010432 DISCON: BIC #6,#RCSR(0)
1459 006246 000207 TST #RCSR(0)
1460 RTS
1461
1462 ;
1463 006250 052770 000006 010432 ;SUBROUTINE TO CONNECT LINE R0 HAS LINE #
1464 006256 000207 CONN: BIS #6,#RCSR(0) ;SET DTR, R0 TO BND
1465 RTS
1466
1467 ;
1468 006260 000240 ;TRANSMITTER INTERRUPT COMMON HANDLER
1469 006262 006300 TISR: NOP
1470 006264 105770 010626 ASL X0 ;R0 HAS LINE #
1471 006270 100000 TSTB #TCSR(0) ;CHECK FOR DONE
1472 006272 104003 BMT TISRA ;BRANCH IF DONE
1473 006274 000002 ERROR ;ERROR! FALSE INTERRUPT
1474 006276 000002 RTI ;EXIT
1475 006278 005737 001332 TISRAA: TST CONFIG ;THIS CONFIGURATION 0?
1476 006302 001420 TISRC: TST TISRC ;BRANCH IF YES
1477 006304 020037 001402 BEO X0,CALLER ;DID CALLER INTERRUPT
1478 006310 001015 CMP #X0,CALLER
1479 006312 117770 010724 BNE TISRC
1480 006320 005237 001410 MOV #TBUFFP,#TBUFFP(0) ;TRANSMIT
1481 006324 022737 015220 INC #TBUFFP ;STEP POINTER
1482 006332 001003 CMP #BUFF+1000,#TBUFFP
1483 006334 042770 000100 010626 BNE #+10
1484 006342 000002 BIC #BIT6,#TCSR(0)
1485 006344 117770 173036 010724 RTI
1486 006352 005237 001406 TISRC: MOV #0TBUFFP,#TBUFFP(0) ;TRANSMIT THE NEXT CHARACTER
1487 006354 005737 001332 INC #TBUFFP ;STEP POINTER TO NEXT CHAR.
1488 006356 005737 001332 CONF: TST CONFIG ;HAS CONFIGURATION 0 SELECTED
1489 006362 001010 BNE TISRB ;BRANCH IF CONFIG #1 OR #2
1490 006364 022737 016660 001406 CMP #OUTBUF+1000,#TBUFFP+1000 ;CHARS. BEEN SENT
1491 006372 001340 TISRAA: BNE ;EXIT IF NOT
1492 006374 012737 014710 001406 TISRB: MOV #OUTBUF,#TBUFFP ;RESET POINTER
1493 006402 000002 RTI ;AND EXIT

```

```

1489
1490 006404 022737 015054 001406 TISRR: CMP #OUTBUF+100,OTBUFF;HAVE 100. CHARS. BEEN SENT?
1491 006412 001330 RNF TISRAA JEXIT IF NOT
1492 006414 042770 000100 010626 R1C #BIT6,@TCR(0) JDISABLE TRANSMITTER INTERRUPT
1493 006422 000764 BR TISRRB JRESET POINTER AND EXIT
1494
1495
1496
1497
1498 006424 000240
1499 006426 104000
1500 006430 012507
1501 006432 104000
1502 006434 013143
1503 006436 004737 003314
1504 006442 112601
1505 006444 010137 014710
1506 006450 004537 004174
1507 006454 014710
1508 006456 014711
1509 006460 001747
1510 006462 004737 005076
1511 006466 004737 010274
1512 006472 004737 010226
1513 006476 004737 010162
1514 006502 012737 000340 177776
1515 006510 012702 000140
1516 006514 012701 010432
1517 006520 004537 010116
1518 006524 104000
1519 006526 013251
1520 006530 005037 177776
1521 006534 000001
1522 006536 000776
1523
1524
1525
1526 006540 104000
1527 006542 012553
1528 006544 012737 105215 014710
1529 006552 004537 004142
1530 006556 014712
1531 006560 001750
1532 006562 012737 000100 001334
1533 006570 004737 010274
1534 006574 004737 010226
1535 006600 004737 010162
1536 006604 012737 000340 177776
1537 006612 012702 000140
1538 006616 012701 010432
1539 006622 004537 010116
1540 006626 104000
1541 006630 013251
1542 006632 005037 177776

*****
PRG0 = SINGLE CHARACTER LINE MODE TEST.
*****
PRG0:  NOP          JBEGIN PRG0
      TYPE         JTYPE
      PBTIT        JTITLE
      PBTIT        JPROGRAM TITLE
      TYPE         JTYPE
      SELCAR       JTYPE
      JSR          PC,RDOCT JGET INPUT
      MOV          MOVVB J(3P)+,X1 JGET USER SPECIFIED DATA
      MOV          MOV J,X1,OUTBUF JAND
      JSR          S,BMOVE JLOAD
      OUTBUF       JINTO
      OUTBUF+1    JOUTPUT
      999.         JBUFFER
      JSR          J,OVRLAY JOVER LAY CR,LF'S IN DATA
      JSR          J,LDPRI JLOAD PRIORITY LEVEL IN VECTOR+2
      JSR          J,LDTVEC JLOAD TRANSMITTER VECTORS
      JSR          J,LDVVEC JLOAD RECEIVER VECTORS
      MOV          @PRTY7,PSW JSET PROCESSOR PRIORITY=7
      MOV          #140,X2 JSET IE
      MOV          #RCSR,X1 JBIT IN
      JSR          S,MOVIT JALL RECEIVERS
      TYPE         JTYPE
      MAKCON       J'MAKE LINE CONNECTION'
      CLR          PSW JSET PROCESSOR PRIORITY=0
      WAIT        JWAIT
      BR          PRG0A JHERE

*****
PRG0A:
      BR          PRG0A

*****
PRG1 = SPECIAL BINARY COUNT PATTERN LINE MODE TEST.
*****
PRG1:  TYPE         JTYPE PROGRAM TITLE.
      PBTIT        JTITLE
      MOV          #105215,OUTBUF JLOAD CRLF
      JSR          S,INFIL JLOAD OUTPUT
      OUTBUF+2    JWITH BINARY
      1000.       JCOUNT PATTERN
      MOV          #100,NUMBER
      JSR          J,LDPRI JLOAD PRIORITY LEVEL IN VECTOR +2
      JSR          J,LDTVEC JLOAD TRANSMITTER VECTORS
      JSR          J,LDVVEC JLOAD RECEIVER VECTORS
      MOV          @PRTY7,PSW JSET PROCESSOR PRIORITY=7
      MOV          #140,X2 JGET IE BIT
      MOV          #RCSR,X1 JGET FIRST CSR ADDRESS
      JSR          S,MOVIT JAND MOVE IT
      TYPE         JTYPE
      MAKCON       J'MAKE LINE CONNECTION'
      CLR          PSW JSET PROCESSOR PRIORITY=0
  
```

```

1543 006636 000001
1544 006640 000776
1545
1546
1547
1548
1549
1550
1551 006642 104000
1552 006644 012622
1553 006646 004537 004014
1554 006652 004737 004402
1555 006656 052777 000004 172314
1556 006664 004537 004742
1557 006670 012702 013504
1558 006674 112201
1559 006676 020127 000045
1560 006702 001772
1561 006704 032737 000100 001374
1562 006712 001402
1563 006714 004737 003600
1564 006720 004537 004742
1565 006724 010177 172256
1566 006730 105777 172250
1567 006734 100375
1568 006736 000756
1569
1570
1571
1572 006740 104000
1573 006742 012672
1574 006744 004537 004014
1575 006750 004737 004402
1576 006754 012706 001176
1577 006760 052777 000004 172212
1578 006766 004537 004742
1579 006772 104000
1580 006774 007036
1581 006776 104007
1582 007000 007270
1583 007002 005037 007470
1584 007006 013700 001234
1585 007012 012720 007422
1586 007016 013710 001236
1587 007022 012701 014710
1588 007026 052737 100000 007466
1589 007034 004737 007634
1590 007040 052777 000140 172132
1591 007046 005037 177776
1592 007052 000001
1593 007054 000776
1594 007056 017737 172116 001370
1595 007064 100461
1596 007066 105737 001370

PRG1C: WAIT          JWAIT
      BR          PRG1C JHERE

*****
PRG2=SPECIAL MESSAGE TRANSMIT ONLY THIS PROGRAM TRANSMITS
*****
JTHE MESSAGE 'A QUICK BROWN FOX JUMPED OVER THE LAZY DOGS
JBACK 1234567890.'
J
PRG2:  TYPE         JTYPE PROGRAM
      PBTIT        JTITLE
      JSR          S,LINSEL JGO SET PARAMETERS
      JSR          J,SETPAR JSET REQUEST TO SEND
      BIS          @BIT2,@RCSR JGO MAKE LINE CONNECTION
      PRG3A: JSR          S,LINCON JGET ADDRESS OF MESSAGE
      PRG3B: MOV          @PRGM,X2 JGET FIRST CHARACTER
      PRG2C: MOV          (2)+,X1 JTERMINATOR CHARACTER
      CMP          X1,#X JSEND MESSAGE
      BEQ          PRG2B JPARITY ENABLED
      RIT          @BIT0,SRT
      BEQ          ,+6
      JSR          J,GENPAR JGENERATE PARITY
      JSR          S,LINCON JCHECK LINE CONNECTION
      MOV          X1,@XBUF JLOAD BUFFER
      TSTB        @TXCSR JAND WAIT FOR CHARACTER
      BPL          ,+4 JTO BE TRANSMITTED
      BR          PRG2C JGET NEXT CHARACTER.

*****
PRG3=PROGRAM TO RECEIVE A MESSAGE.
*****
PRG3:  TYPE         JTYPE PROGRAM
      PBTIT        JTITLE
      JSR          S,LINSEL JGET PARAMETERS
      JSR          J,SETPAR JREPOSITION STACK POINTER
      MOV          @STKPTR+2,X6 JSET REQUEST TO SEND
      PRG3A: BIS          @BIT2,@RCSR JMAKE LINE CONNECTION
      JSR          S,LINCON JSET RECEIVER INTERRUPT
      STRXV       JTO THIS ADDRESS
      RINTV       JSET TRANSMITTER INTERRUPT
      STXV        JTO THIS ADDRESS
      TINTV       JTO THIS ADDRESS
      CLR          WORDS
      MOV          TPVTR,#0
      MOV          @TPINT,(0)+ JLOAD TELEPRINTER VECTOR
      MOV          TPLVL,(0) JAND PRIORITY
      MOV          @OUTBUF,X1 JGET BUF ADD
      BIS          @BIT15,TPLAG JSET BIT 15
      JSR          J,TCRLF JSEND CRLF
      CLR          #140,@RCSR JENABLE RECEIVER INTERRUPTS
      PSW

*****
PRG3B:
      BR          PRG3B

*****
PRG3C:
      BR          PRG3C

*****
PRG3D:
      BR          PRG3D

*****
PRG3E:
      BR          PRG3E

*****
PRG3F:
      BR          PRG3F

*****
PRG3G:
      BR          PRG3G

*****
PRG3H:
      BR          PRG3H

*****
PRG3I:
      BR          PRG3I

*****
PRG3J:
      BR          PRG3J

*****
PRG3K:
      BR          PRG3K

*****
PRG3L:
      BR          PRG3L

*****
PRG3M:
      BR          PRG3M

*****
PRG3N:
      BR          PRG3N

*****
PRG3O:
      BR          PRG3O

*****
PRG3P:
      BR          PRG3P

*****
PRG3Q:
      BR          PRG3Q

*****
PRG3R:
      BR          PRG3R

*****
PRG3S:
      BR          PRG3S

*****
PRG3T:
      BR          PRG3T

*****
PRG3U:
      BR          PRG3U

*****
PRG3V:
      BR          PRG3V

*****
PRG3W:
      BR          PRG3W

*****
PRG3X:
      BR          PRG3X

*****
PRG3Y:
      BR          PRG3Y

*****
PRG3Z:
      BR          PRG3Z
  
```

```
1597 007072 100064 RPL ERR3R
1598 007074 005237 007470 INC WORDS
1599 007100 017737 172076 007472 MOV @RXBUF,RXBUFT JGET DATA
1600 007106 113711 007472 MOV# RXBUFT,(1)
1601 007112 005737 007472 TST RXBUFT
1602 007116 100455 RMI ERR3C
1603 007120 105777 171050 TSTB @SRPTR JECHO OPTION SELECTED
1604 007124 100405 RMI RINT3A
1605 007126 105777 172052 TSTB @TXCSR
1606 007132 100375 BPL =-4
1607 007134 111177 172046 MOV# (1),@TXBUF JECHO CHARACTER
1608 007140 023727 007470 001604 RINT3A: CMP WORDS,#900. JEND OF BUFFER ALLOWED
1609 007146 001411 BEQ RINT3B JYES EXIT
1610 007150 005737 007466 TST TFLAG JIS THIS THE FIRST
1611 007154 100441 RMI RINT3E JCHARACTER BRANCH IF YES.
1612 007156 121103 CMPB (1),X3 JLAST CHARACTER RECEIVED
1613 007160 001404 BEQ RINT3B
1614 007162 122127 000203 CMPB (1)+,#203 JCONTROL C
1615 007166 001401 BEQ RINT3B
1616 007170 000002 RTI JEXIT
1617 007172 005037 007470 RINT3B: CLR WORDS
1618 007176 042777 000140 171774 BIC #140,@RXCSR JDISABLE RECEIVER
1619 007204 012701 014711 MOV #OUTBUF+1,X1 JINITIALIZE BUFFER POINTER
1620 007210 010102 MOV X1,X2
1621 007212 052777 000100 171764 BIS #BIT6,@TXCSR JENABLE TRANSMITTER
1622 007220 052777 000100 171776 RIS #BIT6,@TPS JENABLE TELEPRINTER
1623 007226 000002 RTI JEXIT
1624 007230 104015 ERR3A: ERRRX JTYPE ERROR MESSAGE
1625 007232 013605 LFAIL J
1626 007234 042777 000140 171736 BIC #140,@RXCSR JDISABLE RECEIVER
1627 007242 000604 BR PRG3A
1628 007244 104015 ERR3B: ERRRX JTYPE
1629 007246 012366 RINTM JERROR MESSAGE
1630 007250 000002 RTI JEXIT
1631 007252 104015 ERR3C: ERRRX
1632 007254 013603 ROVER
1633 007256 000002 RTI
1634 J
1635 007260 005037 007466 RINT3E: CLR TFLAG
1636 007264 112103 MOV# (1)+,X3
1637 007266 000002 RTI
1638 J
1639 007270 017737 171710 001366 TINT3: MOV @TXCSR,TXCST JGET TXCSR DATA
1640 007276 105737 001366 TSTB TXCST JTEST
1641 007302 100016 BPL TINT3B
1642 007304 112177 171676 MOV# (1)+,@TXBUF JTRANSMIT CHARACTER
1643 007310 005237 007470 INC WORDS
1644 007314 121103 CMPB (1),X3 JALL CHARACTERS TRANSMITTED
1645 007316 001431 BEQ TINT3C
1646 007320 023727 007470 001604 CMP WORDS,#900.
1647 007326 001425 BEQ TINT3C
1648 007330 121127 000203 CMPB (1),#203 J= CONTROL C
1649 007334 001422 BEQ TINT3C
1650 007336 000002 RTI JRETURN TO MAIN PROGRAM
```

```
1651 007340 017737 171600 001372 TINT3B: MOV @TPS,TEMP JSAVE TELEPRINTER STATUS
1652 007346 005077 171652 CLR @TPS JDISABLE INTERRUPT
1653 007352 105777 171646 TSTB @TPS JWAIT FOR
1654 007356 100375 BPL =-4 JTELEPRINTER TO FINISH
1655 007360 104014 ERRTX JTYPE
1656 007362 012407 TINTM JERROR MESSAGE
1657 007364 105777 171634 TSTB @TPS JWAIT FOR TELEPRINTER
1658 007370 100375 BPL =-4 JTO FINISH
1659 007372 013777 001372 171624 MOV TEMP,@TPS JRESTORE TELEPRINTER STATUS
1660 007400 000002 RTI JEXIT
1661 J
1662 007402 042777 000100 171574 TINT3C: BIC #BIT6,@TXCSR JDISABLE INTERRUPT
1663 007410 032777 000100 171606 BIT #BIT6,@TPS JIS TTY ACTIVE
1664 007416 001421 BEQ PRG3EX
1665 007420 000002 RTI
1666 J
1667 J
1668 007422 112277 171600 TPINT: MOV# (2)+,@TPB JTYPE CHARACTER
1669 007426 121203 CMPB (2),X3 JHAS THIS THE LAST CHAR.
1670 007430 001404 BEQ TPINTA
1671 007432 121227 000203 CMPB (2),#203 J= CONTROL C
1672 007436 001401 BEQ TPINTA
1673 007440 000002 RTI
1674 007442 042777 000100 171554 TPINTA: BIC #BIT6,@TPS JDISABLE INTERRUPT
1675 007450 032777 000100 171526 BIT #BIT6,@TXCSR JIS TRANSMITTER ACTIVE
1676 007456 001401 BEQ =+4
1677 007460 000002 RTI JEXIT
1678 007462 000137 006744 PRG3EX: JMP PRG3A
1679 007466 000000 TFLAG: 0
1680 007470 000000 WORDS: 0
1681 007472 000000 RXBUFT: 0
1682 J
1683 J*****
1684 JPRG4=SPECIAL MESSAGE TRANSMIT ONLY THIS PROGRAM TRANSMITS
1685 J*****
1686 JMESSAGE SPIRAL PATTERN
1687 J
1688 PRG4: TYPE JTYPE PROGRAM
1689 P4TIT JTITLE
1690 JSR 5,LINSEL JGO SET PARAMETERS
1691 JSR 7,SETPAR JSET REQUEST TO SEND
1692 BIS #BIT2,@RXCSR JINIT PAGE WIDTH
1693 MOV #78,COLMN JSET LINE START CHAR
1694 MOV #40,X3 JGO MAKE LINE CONNECTION
1695 PRG4A: JSR 5,LINCON
1696 JSR 7,TCRLF
1697 PRG4B: MOV X3,X2 JGET FIRST CHARACTER
1698 PRG4C: MOV# X2,X1 JGET CHARACTER
1699 CMPB X1,#136 JTERMINATOR CHARACTER
1700 BNE PRG4D JRESEND MESSAGE
1701 MOV #40,X2
1702 BR PRG4C
1703 PRG4D: BIT #BIT6,SRT JPARITY ENABLED
1704 007570 004737 JSR =+6 7,GENPAR JGENERATE PARITY
```

```

1705 007574 004537 000702 JSR 5,LINCON ;CHECK LINE CONNECTION
1706 007600 010177 171402 MOV #1,PTXBUF ;LOAD BUFFER
1707 007604 105777 171374 TSTB #TXCSR ;AND WAIT FOR CHARACTER
1708 007610 100375 BPL ,+4 ;TO BE TRANSMITTED
1709 007612 005202 INC #2 ;SET FOR NEXT CHAR
1710 007614 005337 007652 DEC COLMN ;ALL COLUMNS PRINTED?
1711 007620 001350 BNE PRG4C ;NO, GET NEXT CHAR
1712 007622 012737 000110 007652 MOV #72,COLMN ;RESET COLUMN COUNTER
1713 007630 004737 007654 JSR 7,TCRLF
1714 007634 005203 INC #3 ;UPDATE LINE START CHAR
1715 007636 120327 000136 CMPB #3,#136 ;LAST IN SET
1716 007642 001336 BNE PRG4B ;NO
1717 007644 012703 000040 MOV #40,#3 ;YES, RESET
1718 007650 000733 BR PRG4B ;GET NEXT CHARACTER.
1719 007652 000000 COLMN: 0
1720 ;
1721 ;
1722 ;
1723 007654 112777 000015 171324 TCRLF: MOVB #15,PTXBUF ;SEND CR,LF
1724 007662 105777 171316 TSTB #TXCSR
1725 007666 100375 BPL ,+4
1726 007670 112777 000012 171310 MOVB #12,PTXBUF
1727 007676 105777 171302 TSTB #TXCSR
1728 007702 100375 BPL ,+4
1729 007704 000207 RTS #7
1730 ;
1731 ;
1732 ;
1733 007706 104000 *****
1734 007710 012777 ;PROGRAM 5
1735 007712 004537 004014 PRG5: TYPE PSTIT
1736 007716 000005 JSR 5,LINSEL
1737 007720 004737 004402 RESET
1738 007724 005277 000006 171246 JSR 7,SETPAR ;SET DTR RQ TO SND
1739 007732 104000 BIS #6,PRXCSR ;TYPE MESSAGE TO MAKE
1740 007734 013251 PRG5A: TYPE MAKCON ;LINE CONNECTION
1741 007736 000000 HALT ;WAIT FOR USER TO MAKE LINE CONNECTION
1742 007740 005777 171236 TST #PRXBUF ;READ BUFFER
1743 007744 032777 020000 171226 BIT #BIT13,PRXCSR ;TEST FOR CLEAR TO SEND
1744 007752 001003 BNE PRG5B
1745 007754 104000 PRG5AA: TYPE ;TYPE ERROR MESSAGE
1746 007756 013377 LINCHM BR PRG5A ;AND TRY AGAIN
1747 007760 000764 PRG5B: TYPE
1748 007762 104000 LINMAD CLR ERRCNT
1749 007764 013427 PRG5BB: MOV #PRG2M,X2 ;GET BASE ADDRESS OF DATA TO BE TRANSMITTED
1750 007766 005037 010114 PRG5C: MOV (2)+,#1 ;GET A CHARACTER
1751 007772 012702 013504 PRG5D: CMP #1,#% ;WAS IT THE TERMINATOR?
1752 007776 112201 BEQ PRG5E
1753 010000 020127 000045 BIT #BIT6,BRT ;WAS PARITY OPTION SELECTED?
1754 010004 001400 BEQ PRG5F ;BRANCH IF NO PARITY DESIRED
1755 010006 032737 000100 001374 JSR 7,GENPAR ;GENERATE PARITY ON CHAR. IN R1
1756 010014 001402 BIT #BIT13,PRXCSR ;CHECK CLEAR TO SEND
1757 010016 004737 003600
1758 010022 032777 020000 171150

```

```

1759 010030 001751 BEQ PRG5AA ;TYPE ERROR MSG. IF NOT SET
1760 010032 010177 171150 MOV #1,PTXBUF ;TRANSMIT THE CHARACTER
1761 010036 005777 171140 TSTB #PRXBUF ;ANY ERROR FLAG?
1762 010042 100001 BPL ,+4 ;BRANCH IF NO ERROR FLAG
1763 010044 104003 ERROR ;ERROR! SOME ERROR FLAG IS SET
1764 010046 105777 171126 TSTB #PRXCSR ;WAIT FOR THE RECEIVER TO RECEIVE
1765 010052 100375 BPL ,+4 ;THE TRANSMITTED CHARACTER
1766 010054 117703 171122 MOVB #PRXBUF,#3 ;SAVE IT IN R3
1767 010060 043701 001362 BIC CARMASK,#1 ;CLEAR NON-TRANSMITTED BITS
1768 010064 120103 CMPB #1,#3 ;WAS RECEIVED & TRANSMITTED DATA THE SAME
1769 010066 001403 BEQ PRG5D
1770 010070 104003 ERROR ;ERROR! DATA ERROR
1771 010072 005237 010114 INC ERRCNT
1772 010076 105777 171102 PRG5D: TSTB #TXCSR ;WAIT FOR TRANSMITTER TO FINISH
1773 010102 100375 BPL ,+4
1774 010104 000734 BR PRG5C
1775 010106 104000 PRG5E: TYPE
1776 010110 013350 ENDPAS BR PRG5BB
1777 010112 000727
1778 010114 000000 ERRCNT: OPEN
1779 ;THIS ROUTINE MOVES THE CONTENTS OF R2 TO THE ADDRESS SPECIFIED
1780 ;BY R1
1781 010116 012737 000006 000004 MOVIT: MOV #6,4 ;SET UP FOR RETURN
1782 010124 012737 000002 000006 MOV #2,6
1783 010132 012700 000037 MOV #31,%0 ;GET COUNTER
1784 010136 010231 MOVIT: MOV #2,(1)+ ;MOVE THE DATA
1785 010140 005300 DEC #0 ;ALL DATA MOVED?
1786 010142 001375 BNE MOVITA ;NO, RETURN
1787 010144 012737 004520 000004 MOV #ERTP,MACHER
1788 010152 012737 000040 000006 MOV #40,MACHER+2
1789 010160 000205 RTS 5 ;RETURN
1790 ;
1791 ;
1792 ;
1793 ;
1794 ;
1795 ;
1796 ;
1797 ;
1798 ;
1799 ;
1800 ;
1801 ;
1802 ;
1803 ;
1804 ;
1805 010226 012701 011412 ;SUBROUTINE TO LOAD ALL VECTORS
1806 010232 012702 010334 LDVECB: MOV #RISR0,X1
1807 010236 012703 000010 MOV #VECTAB,X2
1808 010242 012704 000037 MOV #10,X3
1809 010246 032712 000001 MOV #31,%4
1810 010252 001003 LDVECB: BIT #BIT0,(2) ;DOES THIS VECTOR EXIST
1811 010254 011200 BNE LDVEC1 ;NO, SKIP LOADING
1812 010256 010160 000004 LDVEC1: ADD #3,X1 ;LOAD VECTOR
1813 010258 005722 TST (2)+
1814 010260 005304 DEC #4
1815 010262 001367 BNE LDVECB
1816 010264 000207 RTS 7
1817 ;
1818 ;
1819 ;
1820 ;
1821 ;
1822 ;
1823 ;
1824 ;
1825 ;
1826 ;
1827 ;
1828 ;
1829 ;
1830 ;
1831 ;
1832 ;
1833 ;
1834 ;
1835 ;
1836 ;
1837 ;
1838 ;
1839 ;
1840 ;
1841 ;
1842 ;
1843 ;
1844 ;
1845 ;
1846 ;
1847 ;
1848 ;
1849 ;
1850 ;
1851 ;
1852 ;
1853 ;
1854 ;
1855 ;
1856 ;
1857 ;
1858 ;
1859 ;
1860 ;
1861 ;
1862 ;
1863 ;
1864 ;
1865 ;
1866 ;
1867 ;
1868 ;
1869 ;
1870 ;
1871 ;
1872 ;
1873 ;
1874 ;
1875 ;
1876 ;
1877 ;
1878 ;
1879 ;
1880 ;
1881 ;
1882 ;
1883 ;
1884 ;
1885 ;
1886 ;
1887 ;
1888 ;
1889 ;
1890 ;
1891 ;
1892 ;
1893 ;
1894 ;
1895 ;
1896 ;
1897 ;
1898 ;
1899 ;
1900 ;
1901 ;
1902 ;
1903 ;
1904 ;
1905 ;
1906 ;
1907 ;
1908 ;
1909 ;
1910 ;
1911 ;
1912 ;
1913 ;
1914 ;
1915 ;
1916 ;
1917 ;
1918 ;
1919 ;
1920 ;
1921 ;
1922 ;
1923 ;
1924 ;
1925 ;
1926 ;
1927 ;
1928 ;
1929 ;
1930 ;
1931 ;
1932 ;
1933 ;
1934 ;
1935 ;
1936 ;
1937 ;
1938 ;
1939 ;
1940 ;
1941 ;
1942 ;
1943 ;
1944 ;
1945 ;
1946 ;
1947 ;
1948 ;
1949 ;
1950 ;
1951 ;
1952 ;
1953 ;
1954 ;
1955 ;
1956 ;
1957 ;
1958 ;
1959 ;
1960 ;
1961 ;
1962 ;
1963 ;
1964 ;
1965 ;
1966 ;
1967 ;
1968 ;
1969 ;
1970 ;
1971 ;
1972 ;
1973 ;
1974 ;
1975 ;
1976 ;
1977 ;
1978 ;
1979 ;
1980 ;
1981 ;
1982 ;
1983 ;
1984 ;
1985 ;
1986 ;
1987 ;
1988 ;
1989 ;
1990 ;
1991 ;
1992 ;
1993 ;
1994 ;
1995 ;
1996 ;
1997 ;
1998 ;
1999 ;
2000 ;

```

```

1813 010262 000301 LDVEC2: ADD X3,X1
1814 010264 005722 TST (2)+
1815 010266 005704 DEC X4
1816 010270 001366 RNE LOTVED
1817 010272 000207 RTS 7
1818
1819
ROUTINE TO LOAD PRIORITY LEVEL 7 IN VECTOR +2
1820 010274 012701 010334 LOPRI1: MOV #VECTOR,X1 ;GET BASE VECTOR
1821 010300 012702 000300 MOV #300,X2 ;GET LEVEL 7
1822 010304 012703 000037 MOV #31,X3 ;LOAD COUNTER
1823 010310 032711 000001 LDPRIA: BIT #BIT0,(1) ;DOES THIS VECTOR EXIST
1824 010314 001003 RNE LOPRIX ;NO SKIP LOADING
1825 010316 011104 MOV (1),X4 ;LOAD VECTOR +2
1826 010320 010264 000002 MOV X2,2(4)
1827 010324 005721 LDPRIX: TST (1)+ ;POINT TO NEXT VECTOR
1828 010326 005303 DEC X3 ;DECREMENT COUNTER
1829 010330 001367 RNE LDPRIA
1830 010332 000207 RTS 7
1831
;VECTOR ASSIGNMENT TABLE
1832 010334 000301 VECTAB: 301 ;LINE 0 VECTOR
1833 010336 000311 311 ;LINE 1 VECTOR
1834 010340 000321 321 ;LINE 2 VECTOR
1835 010342 000331 331 ;LINE 3 VECTOR
1836 010344 000341 341 ;LINE 4 VECTOR
1837 010346 000351 351 ;LINE 5 VECTOR
1838 010350 000361 361 ;LINE 6 VECTOR
1839 010352 000371 371 ;LINE 7 VECTOR
1840 010354 000401 401 ;LINE 10 VECTOR
1841 010356 000411 411 ;LINE 11 VECTOR
1842 010360 000421 421 ;LINE 12 VECTOR
1843 010362 000431 431 ;LINE 13 VECTOR
1844 010364 000441 441 ;LINE 14 VECTOR
1845 010366 000451 451 ;LINE 15 VECTOR
1846 010370 000461 461 ;LINE 16 VECTOR
1847 010372 000471 471 ;LINE 17 VECTOR
1848 010374 000501 501 ;LINE 20 VECTOR
1849 010376 000511 511 ;LINE 21 VECTOR
1850 010400 000521 521 ;LINE 22 VECTOR
1851 010402 000531 531 ;LINE 23 VECTOR
1852 010404 000541 541 ;LINE 24 VECTOR
1853 010406 000551 551 ;LINE 25 VECTOR
1854 010410 000561 561 ;LINE 26 VECTOR
1855 010412 000571 571 ;LINE 27 VECTOR
1856 010414 000601 601 ;LINE 30 VECTOR
1857 010416 000611 611 ;LINE 31 VECTOR
1858 010420 000621 621 ;LINE 32 VECTOR
1859 010422 000631 631 ;LINE 33 VECTOR
1860 010424 000641 641 ;LINE 34 VECTOR
1861 010426 000651 651 ;LINE 35 VECTOR
1862 010430 000661 661 ;LINE 36 VECTOR
1863
;DL11-E REGISTER ADDRESSES
1864 000000 N=0
1865 000000 A=0
1866 010432 RCSR: .REPT 31.
  
```

```

1867 RRCV \N,\A
1868 N=N+10
1869 A=A+1
1870 .ENDR
1871 000000 N=0
1872 000000 A=0
1873 010530 RBUF: .REPT 31.
1874 RBUFF \N,\A
1875 N=N+10
1876 A=A+1
1877 .ENDR
1878 000000 N=0
1879 000000 A=0
1880 010626 TCSR: .REPT 31.
1881 TXMT \N,\A
1882 N=N+10
1883 A=A+1
1884 .ENDR
1885 000000 N=0
1886 000000 A=0
1887 010724 TBUF: .REPT 31.
1888 TBUFF \N,\A
1889 N=N+10
1890 A=A+1
1891 .ENDR
1892 000000 N=0
1893 .REPT 31.
1894 ISR \N
1895 N=N+1
1896 .ENDR
1897 000000 N=0
1898 .REPT 32.
1899 ISRT \N
1900 N=N+1
1901 .ENDR
1902
;MESSAGES
1903 012012 042045 030514 026461 MTITLE: .ASCII *DL11-E ON LINE TEST - MAINDEC-11-DZDLB-0X*
1904 012020 020105 047117 046040
1905 012026 047111 020105 042524
1906 012034 052123 026440 046440
1907 012042 044501 042116 041505
1908 012050 030455 026461 055104
1909 012056 046104 026502 022502
1910 012064 046445 050101 047440 .ASCII *SNAP OF DEVICES PRESENT*
1911 012072 020106 042504 044526
1912 012100 042503 020123 051120
1913 012106 051505 047105 022524
1914 012114 046045 047111 020105 .ASCII *3LINE D=ADR TRAP ATX0*
1915 012122 042040 040455 051104
1916 012130 020040 020040 052040
1917 012136 040522 020120 052101
1918 012144 040045
1919 012146 020040 020040 020040 MLINE: .ASCII " "
1920 012154 020040 020040 020040 MDADR: .ASCII " "
  
```

1921	012162	020040	020040						
1922	012166	020040	020040	020040	MTTRAP:	.ASCII	'	X0'	
1923	012174	040345							
1924	012176	047045	047117	020105	MNONE:	.ASCII	'	XNONE FOUNDX0'	
1925	012204	047506	047125	022504					
1926	012212	100							
1927	012213	045	054524	042520	MSWSEL:	.ASCII	'	XTYPE IN PROGRAM NUMBER	0'
1928	012220	044440	020116	051120					
1929	012226	043517	040522	020115					
1930	012234	052516	041115	051105					
1931	012242	020040	020040	100					
1932	012247	045	051105	047522	MYERR:	.ASCII	'	XERROR = UNEXPECTED TRAP'	
1933	012254	020122	020055	047125					
1934	012262	054105	042520	052103					
1935	012270	042105	052040	040522					
1936	012276	120							
1937	012277	045	051124	050101		.ASCII	'	XTRAPPED TO	0'
1938	012304	042520	020104	047524					
1939	012312	020040							
1940	012314	020040	020040	020040	HTD:	.ASCII	'		
1941	012322	020040							
1942	012324	052045	040522	050120		.ASCII	'	XTRAPPED FROM PC	0'
1943	012332	042105	043040	047522					
1944	012340	020113	041520	020040	MFROM:	.ASCII	'		0'
1945	012346	020040	020040	020040					
1946	012354	020040	100						
1947	012357	040	020040	020040	CSRADD:	.ASCII	'		0'
1948	012364	040040							
1949	012366	043045	046101	042523	RINTM:	.ASCII	'	XFALSE INT. RCVR0'	
1950	012374	044440	052116	020056					
1951	012402	041522	051126	100					
1952	012407	045	040506	051514	TINTM:	.ASCII	'	XFALSE INT XMIT0'	
1953	012414	020105	047111	020124					
1954	012422	046530	052111	100					
1955	012427	045	041520	020075	EM0:	.ASCII	'	XPC=	0'
1956	012434	020040	020040	020040	APC:	.ASCII	'		0'
1957	012442	020040	100						
1958	012445	040	052040	041530	ATXCSR:	.ASCII	'	TXCSR =	0'
1959	012452	051123	036440	040					
1960	012457	040	020040	020040	ATXWAS:	.ASCII	'		0'
1961	012464	040040							
1962	012466	020040	054122	051503	ARXCSR:	.ASCII	'	RXC8R =	0'
1963	012474	020122	020075						
1964	012500	020040	020040	020040	ARXWAS:	.ASCII	'		0'
1965	012506	100							
1966	012507	045	050045	043522	P0TIT:	.ASCII	'	XXP00 = SINGLE CHAR LINE MODE TEST0'	
1967	012514	020060	020055	044523					
1968	012522	043516	042514	041440					
1969	012530	040510	020122	044514					
1970	012536	042516	046440	042117					
1971	012544	020105	042524	052123					
1972	012552	100							
1973	012553	045	050045	043522	P1TIT:	.ASCII	'	XXP01 = SPEC BIN COUNT LINE MODE TEST0'	
1974	012560	020061	020055	050123					

1975	012566	041505	041040	047111					
1976	012574	041440	052517	052116					
1977	012602	046040	047111	020105					
1978	012610	047515	042504	052040					
1979	012616	051505	040124						
1980	012622	022445	051120	031107	P2TIT:	.ASCII	'	XXP02 = SPECIAL MESSAGE LINE MODE TEST0'	
1981	012630	026440	051440	042520					
1982	012636	044503	046101	046440					
1983	012644	051505	040523	042507					
1984	012652	046040	047111	020105					
1985	012660	047515	042504	052040					
1986	012666	051505	040124						
1987	012672	022445	051120	031507	P3TIT:	.ASCII	'	XXP03 = RECEIVE MESSAGE TEST0'	
1988	012700	026440	051040	041505					
1989	012706	044505	042526	046440					
1990	012714	051505	040523	042507					
1991	012722	052040	051505	040124					
1992	012730	022445	051120	032107	P4TIT:	.ASCII	'	XXP04 = SPECIAL MESSAGE TEST (SPIRAL)0'	
1993	012736	026440	051440	042520					
1994	012744	044503	046101	046440					
1995	012752	051505	040523	042507					
1996	012760	052040	051505	020124					
1997	012766	051450	044520	040522					
1998	012774	024514	100						
1999	012777	045	051120	032507	P5TIT:	.ASCII	'	XXP05 = DATA ECHO TEST USING MAYNARD FACILITY0'	
2000	013004	026440	042040	052101					
2001	013012	020101	041505	047510					
2002	013020	052040	051505	020124					
2003	013026	051525	047111	020107					
2004	013034	040515	047131	051101					
2005	013042	020104	040506	044503					
2006	013050	044514	054524	100					
2007	013055	045	054524	042520	SELPAR:	.ASCII	'	XTYPE IN PARAMETERS AS FOLLOWS0'	
2008	013062	044440	020116	040520					
2009	013070	040522	042515	042524					
2010	013076	051522	040440	020123					
2011	013104	047506	046114	053517					
2012	013112	035123							
2013	013114	041045	052111	026461		.ASCII	'	XBIT1=0 = CHAR LENGTH0'	
2014	013122	020060	020075	044103					
2015	013130	051101	046040	047105					
2016	013136	052107	022610	100					
2017	013143	045	054524	042520	SELCAR:	.ASCII	'	XTYPE TEST CHAR CODE IN BIT7-BITS OF AN OCTAL BYTE	0'
2018	013150	052040	051505	020124					
2019	013156	044103	051101	041440					
2020	013164	042117	020105	047111					
2021	013172	041040	052111	026467					
2022	013200	044502	030124	047440					
2023	013206	020106	047101	047440					
2024	013214	052103	046101	041040					
2025	013222	052131	020105	020040					
2026	013230	040040							
2027	013232	044445	046114	043505	DTERR:	.ASCII	'	XILLEGAL DATA0'	
2028	013240	046101	042040	052101					


```
2137 014314 031440 033455 020040
2138 014322 020040 100
2139 014325 045 042522 047503 MPWRF: .ASCII 'XRECOVERED FROM POWER FAILURE'
2140 014332 042526 042522 020104
2141 014340 051106 046517 050040
2142 014346 053517 051105 043040
2143 014350 044501 052514 042522
2144 014362 100
2145 014363 045 046111 042514 MMODX: .ASCII 'XILLEGAL LINE NO.'
2146 014370 040507 020114 044514
2147 014376 042516 047040 027117
2148 014404 100
2149 014405 045 046111 042514 MMODD: .ASCII 'XILLEGAL DEVICE ADDRESS'
2150 014412 040507 020114 042504
2151 014420 044526 042503 040440
2152 014426 042104 042522 051523
2153 014434 100
2154 014435 045 054524 042520 MMODI: .ASCII 'XTYPE IN OCTAL I BIT 0-5 THE LINE NUMBER OF DEVICE ADDRESS TO BE'
2155 014442 044440 020116 041517
2156 014450 040524 020114 020072
2157 014456 044502 020124 026460
2158 014464 020065 044124 020105
2159 014472 044514 042516 047040
2160 014500 046525 042502 020122
2161 014506 043117 042040 053105
2162 014514 041511 020105 042101
2163 014522 051104 051505 020123
2164 014530 047524 041040 105
2165 014535 040 047515 044504 .ASCII ' MODIFIED '
2166 014542 044506 042105 020040
2167 014550 020040 100
2168 014553 045 054524 042520 MMOD2: .ASCII 'XTYPE IN NEW RXCSR DEVICE ADDRESS '
2169 014560 044440 020116 042516
2170 014566 020127 054122 051503
2171 014574 020122 042504 044526
2172 014602 042503 040440 042104
2173 014610 042522 051523 020040
2174 014616 020040 100
2175 014621 045 054524 042520 MMOD3: .ASCII 'XTYPE IN 177777 TO CHANGE ANOTHER'
2176 014626 044440 020116 033461
2177 014634 033467 033467 020040
2178 014642 047524 041040 040510
2179 014650 043516 020105 047101
2180 014656 052117 042510 122
2181 014663 040 042504 044526 .ASCII ' DEVICE ADDRESS '
2182 014670 042503 040440 042104
2183 014676 042522 051523 020040
2184 014704 020040 100
2185 014710 040
2186 014710 020000 OUTBUF: OPEN
2187 016660 016660 .OUTBUF+1000.
2188 016660 020000 INBUF: OPEN
2189 020630 020630 .INBUF+1000.
2190 015054 015054 BUFF=OUTBUF+100.
```

```
2191 020630 000001 DEND: .END
```


A	= 000037	ACONFI	013456	ADD1	000004	ADTENP	004362
AGAIN	003746	AINPRG	013352	ALTNF	013715	ALINEX	013665
APARM	013373	APC	012434	ARXCSR	012466	ARXWAS	012500
ATXCSR	012445	ATXWAS	012457	RDCNV	004216	RDCNVA	004250
RDCNVB	004302	RDCNVC	004304	RDCNVD	004306	RIT0	= 000001
RIT1	= 000002	RIT10	= 002000	RIT11	= 004000	RIT12	= 010000
RIT13	= 020000	RIT14	= 040000	RIT14	= 100000	RIT2	= 000004
RIT3	= 000010	RIT4	= 000020	RIT5	= 000040	RIT6	= 000100
RIT7	= 000200	RIT8	= 000400	RIT9	= 001000	RMOVA	004204
RMOVE	004174	RUFF	= 015054	RUFFP	001400	BUTTON	014215
CALLED	001404	CALLER	001402	CARMSK	001362	CERDAT	013300
CHALT	= 104005	CHLT	002366	CNVCTR	004354	COLMN	007652
CONFIG	001332	CONN	006250	COUNT	001324	CRNUM	013343
CRXTX	002752	CRXTXA	003024	CRXTYB	003026	CRXTXC	003032
CSB	013313	CSRADD	012357	CTRD	001364	CWAS	013325
DATCHK	= 104004	NCERR	005532	DECVL	004374	DELAY	= 104016
DEND	020630	DIGIT	004356	DIACON	006234	DLCNT	003526
DLY	003466	DLYA	003504	DLYB	003510	DLVC	003522
DOIT	003742	DONE	003776	DTCHK	002412	DTCHKA	002510
DTERR	013232	FCDAT	003346	EMALT	= 104010	EMLT	002400
EHLTA	002410	FIGHT	003632	EMTA	002234	EMTINT	002222
EMTTAB	001264	FM0	012427	ENDPA8	013350	ERR	002512
ERRA	002562	FRB	002624	ERRC	002632	ERRCNT	0101.4
ERRD	002642	FRRE	002644	ERROR	= 104003	ERROR1	= 104013
ERRRX	= 104015	FRRTX	= 104014	ERR1	002534	ERR3A	007230
ERR3B	007244	FRRC	007252	ERTP	004520	ERTPA	004570
FILL	004146	FILLA	004154	FINISH	008476	FIVE	003720
FMAP	001350	FNONE	001346	FORMAD	004704	FROMPC	001342
FTITLE	001344	GENPAR	003600	INBUF	016660	INBUF	001376
INCRPG	002354	INDAT	003322	INFIL	004142	LDLINE	014286
LDPR1	010274	LDPR1A	010310	LDPR1X	010324	LDVTEC	010226
LDVTEC	010246	LDVECB	010202	LDVECS	010162	LDVEC1	010214
LOVEC2	010262	LFAIL	013605	LINCA	004760	LINCB	004776
LINCP	005056	LINCMB	013377	LINCON	004742	LINE	001330
LINENO	001354	LINCUP	005074	LINMAD	013427	LINSA	004056
LINSEL	004014	MACHER	000004	MAKCON	013251	MANUAL	= 100000
MAPA	001520	MAPEND	001730	MAPERR	001752	MAPNE	001550
MAPOK	001560	MAPOKA	001672	MAPVEC	004626	MDADR	012194
MFR0M	012346	MLINE	012146	MMODD	014405	MMD0X	014363
MMOD1	014435	MMOD2	014553	MMOD3	014621	MNONE	012176
MODEM	001412	MODEV	002024	MODEV1	002036	MODEV2	002072
MODEV3	002116	MOVIT	010116	MOVITA	010136	MPWRF	014325
MSWSEL	012213	MTERR	012247	MTITLE	012012	MT0	012314
MTRAP	012166	N	= 000040	NUMBER	001334	OACNV	003530
OACNVA	003542	NL0PS	001336	OPEN	= 000000	OTBUPP	001406
OUTBUF	014710	OVRLAY	005076	OVRLYA	005106	PARBIT	001322
PAREX	004476	PARMTS	013355	PC	=X000007	PFAIL	002646
POPSP	= 005726	POPSP2	022626	PRGID	001242	PRGNUM	001240
PRGTAB	001244	PRG0	006424	PRG0A	006534	PRG1	006540
PRG1C	006636	PRG2	006642	PRG2A	006664	PRG2B	006670
PRG2C	006674	PRG2M	013504	PRG3	006740	PRG3A	006754
PRG3EX	007462	PRG4	007474	PRG4A	007530	PRG4B	007540
PRG4C	007542	PRG4D	007560	PRG5	007706	PRG5A	007732
PRG5AA	007754	PRG5B	007762	PRG5BB	007772	PRG5C	007776

PRG5D	010076	PRG5E	010106	PRTY0	= 000000	PRTY1	= 000040
PRTY2	= 000100	PRTY3	= 000140	PRTY4	= 000200	PRTY5	= 000240
PRTY6	= 000300	PRTY7	= 000340	PSW	= 177776	PWRUP	002656
POTIT	012507	PITIT	012553	P2TIT	012622	P3TIT	012672
P4TIT	012730	P5TIT	012777	RBUF	010530	RCSR	010432
RDDAT	003330	RDOCT	003314	RECDAT	001356	REMAP	001416
RESTRY	002676	RETRN	003452	RINTM	012366	RINT3	007056
RINT3A	007140	RINT3B	007172	RINT3E	007260	RISR	005124
RISRA	005164	RISRAA	005206	RISRB	005232	RISRBB	005300
RISRC	005310	RISRCA	005344	RISRD	005376	RISRDA	005416
RISREX	005620	RISRF	005546	RISRFB	005734	RISRFC	006000
RISRFD	006110	RISRFE	006152	RISRFF	006124	RISRFG	006164
RISRFH	006202	RISR0	011022	RISR1	011032	RISR10	011122
RISR11	011132	RISR12	011142	RISR13	011152	RISR14	011162
RISR15	011172	RISR16	011202	RISR17	011212	RISR2	011042
RISR20	011222	RISR21	011232	RISR22	011242	RISR23	011252
RISR24	011262	RISR25	011272	RISR26	011302	RISR27	011312
RISR3	011052	RISR30	011322	RISR31	011332	RISR32	011342
RISR33	011352	RISR34	011362	RISR35	011372	RISR36	011402
RISR4	011062	RISR5	011072	RISR6	011102	RISR7	011112
RDOVER	013623	RREAD	003442	RSTP	002330	RSTPSW	002332
RSTREG	= 104012	RSTRG	002314	RTN	003752	RXBUF	001202
RXBUFT	007472	RXC3R	001200	RXC3RT	001370	RXERR	002732
RXLVL	001212	RXVTR	001210	R0	=X000000	R1	=X000001
R2	=X000002	R3	=X000003	R4	=X000004	R5	=X000005
SAVE	001326	SAVREG	= 104011	SAVR0	002254	SAVR0	003312
SELCAR	013143	SELPAR	013055	SETPAR	004402	SEVEN	003676
SIX	003654	SOFTSR	000176	SP	=X000006	SRPTR	000174
SRT	001374	STAD	000300	STALL	= 104002	START	001422
START1	001770	STCONT	000226	STKPTR	= 001200	STPARA	004500
STPPA	003110	STPRA	003060	STRVTR	003042	STRXV	= 104006
STTXV	= 104007	STXMTV	003072	SURTEN	004314	SUBTNA	004320
SUBTNB	004334	SVRPC	002310	SVRPSW	002312	TBIT1	004434
TBUF	010724	TBUFFP	001410	TCONFI	013477	TCRLF	007654
YCSR	010626	TEMP	001372	TEMP1	001352	TENPWR	004360
YFLAG	007466	TINTM	012407	TINT3	007270	YINT3B	007340
TINT3C	007402	TISR	006260	TISRA	006276	TISRAA	006274
TISRB	006404	TISRBB	006374	TISRC	006344	TISR0	011412
TISR1	011422	TISR10	011512	TISR11	011522	TISR12	011532
TISR13	011542	TISR14	011552	TISR15	011562	TISR16	011572
TISR17	011602	TISR2	011432	TISR20	011612	TISR21	011622
TISR22	011632	TISR23	011642	TISR24	011652	TISR25	011662
TISR26	011672	TISR27	011702	TISR3	011442	TISR30	011712
TISR31	011722	TISR32	011732	TISR33	011742	TISR34	011752
TISR35	011762	TISR36	011772	TISR37	012002	TISR4	011452
TISR5	011462	TISR6	011472	TISR7	011502	YKR	001222
YKLV	001232	YK8	001220	YKVT	001230	TLINE	013725
YLINEX	013675	YK9	001220	YK10	001226	TPINT	007422
YPINTA	007442	YK11	001236	YK12	001224	YK13	001234
YXBUF	001206	YK13	001204	YK14	001226	YK15	002710
YXLV	001216	YK15	001214	YK16	001222	YK17	003136
YK17	003160	YK18	003176	YK19	003242	YK20	= 104000
YK20	= 104001	YK21	003214	YK22	003244	YK23	003300
YK23	003250	YK24	003306	YK25	014162	YK26	014212

DZDLBB MACY11 27(657) 17-NOV-75 14:50 PAGE 48
DZDLBB.SRC SYMBOL TABLE

VECTAB 010334 WORDS 007470 WRU 014065 XMTDAT 001360
= 020630

ERRORS DETECTED: 0

*DZDLBB,DZDLBB=DZDLBB.SRC/SOL
RUN-TIME: 10 17 0 SECONDS
CORE USED: 10K