

DL11

C/D OFFLINE TEST

MD-11-DZDLA-F

EP-DZDLA-F-DL-A

OCT 1976

COPYRIGHT ©1976

digital

FICHE 1 OF 1

Made In U.S.A.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

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.

BEFORE STARTING ANY OF THE SELECTABLE PROGRAMS MAKE SURE THAT THE TTY IS IN REMOTE MODE; AND THAT THE PROGRAM SELECTED IS A LEGAL PROGRAM, IE. NO.0-4, OTHERWISE AN ERROR MESSAGE WILL OCCUR. TERMINATE ALL INPUTS WITH A CARRIAGE RETURN.

A MAP OF DEVICES PRESENT WILL BE TYPED AT RUN TIME. THIS MAP WILL NOT BE TYPED OUT AGAIN UNLESS THE PROGRAM IS RESTARTED AT LOCATION 200. A RESTART FROM THIS LOCATION WILL CAUSE THE MAP OF DEVICES TO BE TYPED OUT AGAIN AND THEN A NORMAL START WILL OCCUR.

4.1 PRGO INPUT/OUTPUT LOGIC TESTS

- A. LOAD ADDRESS = 000200 (RESTART LOAD ADDR. = 000204)
TYPE PROGRAM NUMBER = 0.
THE DIAGNOSTIC WILL IDENTIFY THE PROGRAM YOU SELECTED.
DISCONNECT THE DL11-E FROM THE MODEM AND INSERT THE JUMPER CONNECTOR IN THE MODEM END OF THE CABLE, AND PRESS CONTINUE.
NOTE, IF THE CABLE IS LEFT CONNECTED TO THE MODEM THE FOLLOWING TESTS WILL FAIL:
AT22, AT23, AT25, AT30, AT32, AT56
- B. THE PROGRAM WILL TYPE OUT INSTRUCTIONS TO SET IN THE DESIRED SR OPTIONS. PRESS CONTINUE WHEN THE OPTIONS ARE IN THE SR.
THE AVAILABLE OPTIONS ARE:
SR 0-6 ROUTINE TO BE RUN (IF ENABLED BY SR9)
SR7 DISABLE STALL MODE
SR9 LOOP SELECTED ROUTINE
SR10 HALT AT END OF CURRENT TEST
SR11 INHIBIT ITERATION
SR12 SELECT LINE NUMBER AND LOCK ON IT
SR13 INHIBIT PRINTOUT
SR14 SCOPE
SR15 HALT ON ERROR.
- C. THE PROGRAM WILL NOW REQUEST THE LINE # (IF SR12=1) YOU WISH TO TEST. TYPE THE LINE # AS REQUESTED, FOLLOWED BY A CARRIAGE RETURN. LINE NUMBER REFERS TO THE ADDRESSES TO WHICH THE DL11-E RESPONDS.

H01

MAIN. MACY11 27(732) 25-FEB-76 10:49 PAGE 7
DZDLAF.P11

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
358
359
360
361
362
363

5.4 PRG3-SINGLE CHARACTER MAINT MODE DATA TEST

PRG3 TRANSMITS, RECEIVES AND CHECKS RECEIVED DATA USING USER SPECIFIED DL11-E PARAMETERS, AND DATA.

5.4.1 ERROR PRINTOUTS

SELF EXPLANATORY ERROR PRINTOUTS ARE PROVIDED.

5.5 PRG4-SPECIAL BINARY COUNT MAINT MODE DATA TEST

PRG4 IS THE SAME AS PRG0 ROUTINE 54 EXCEPT THAT THE USER SPECIFIES DL11-E RUNNING PARAMETERS.

5.5.1 ERROR PRINTOUTS

SELF EXPLANATORY PRINTOUTS ARE PROVIDED.

6.0 POWER FAIL

A POWER FAIL ROUTINE IS INCLUDED IN THE PROGRAM. WHEN THE POWER FAILS THE PROGRAM WILL AUTOMATICALLY RESTART USING THE PRESENT SR OPTIONS AND THE LINE PREVIOUSLY SELECTED. NOTE: THE POWER MAY FAIL WHEN THE PROGRAM IS EXECUTING A 'RESET' INSTRUCTION. IN THIS CASE OPERATOR INTERVENTION IS NEEDED TO PRESS CONTINUE. AN ERROR TYPEOUT RESULTS AND WILL TYPE THE PROGRAM #, THE ROUTINE THAT WAS RUNNING AT THE TIME THE POWER FAILED (PROGRAM 0 ONLY), AND THE PC OF THE POWER FAIL ERROR CALL.

RECOVERED FROM POWER FAILURE.

P(PRG#) T(ROUTINE #) PC = (ADDRESS OF ERROR CALL)

.ENDR

364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419

.ENABLE ABS,AMA

:DL11-E,C/D DIAGNOSTIC PROGRAM (OFF LINE TESTS)

:PRG0- INPUT-OUTPUT LOGIC TESTS
:PRG1- TRANSMITTER SCOPE LOOP
:PRG2- RECEIVER SCOPE LOOP
:PRG3- SINGLE CHARACTER MAINTENANCE MODE DATA TEST
:PRG4- SPECIAL BINARY COUNT MAINTENANCE MODE DATA TEST

:STANDARD SR SWITCH OPTIONS (SWITCH SET TO A 1)

:SR15- HALT ON ERROR
:SR14- SCOPE.
:SR13- INHIBIT PRINTOUT
:SR12- SELECT LINE NUMBER AND LOCK ON IT
:SR11- INHIBIT ITERATION.
:SR10- HALT AT END CURRENT TEST, TEST NO. IN DATA LIGHTS
:SR9- SELECT ROUTINE.
:SR7- DISABLE STALL MODE AND RUN FULL SPEED.
:SR6 THROUGH SR0 - NUMBER OF ROUTINE TO BE SELECTED.

:STANDARD CONFIGURATION
:CHARACTER LENGTH 8
:STOP CODE 2

.=0
ERTP ;UNASSIGNED TRAP
0
MACHER: ERTP ;SP OVERFLOW, BUS ERROR TRAP
40
ERTP ;RESERVED INSTRUCTION TRAP
100
ERTP ;TRACE TRAP
140
MAPVEC ;TRAP TO MAP VECTOR
PRTY7
PFAIL ;POWER FAIL TRAP
PRTY7
EMTINT ;EMT TRAP
PRTY7
ERTP
340
.+2
HALT
. =46
LOGIC
.REPT 117.
.+2 ;TRAP TO TRAP REPORTER
4
.ENDR

;EQUATE STATEMENTS
PSW=177776
SPBOT=1176
NOP=240

000000
005706
000000
005706
000040
005706
000100
005706
000140
006014
000340
006234
000340
002766
000340
005706
000340
000042
000000
000046
005312

177776
001176
000240

420	000000	OPEN=0
421	100000	MANUAL=BIT15
422	100000	BIT15=100000
423	040000	BIT14=40000
424	020000	BIT13=20000
425	010000	BIT12=10000
426	004000	BIT11=4000
427	002000	BIT10=2000
428	001000	BIT9=1000
429	000400	BIT8=400
430	000200	BIT7=200
431	000100	BIT6=100
432	000040	BIT5=40
433	000020	BIT4=20
434	000010	BIT3=10
435	000004	BIT2=4
436	000002	BIT1=2
437	000001	BIT0=1
438	005726	POPSP=5726
439	022626	POPSP2=022626
440	000340	PRTY7=340
441	000300	PRTY6=300
442	000240	PRTY5=240
443	000200	PRTY4=200
444	000140	PRTY3=140
445	000100	PRTY2=100
446	000040	PRTY1=40
447	000000	PRTY0=0
448	104000	TYPE=EMT+0
449	104001	TYPES=EMT+1
450	104002	STALL=EMT+2
451	104003	ERROR=EMT+3
452	104004	DATCHK=EMT+4
453	104005	CHALT=EMT+5
454	104006	STRXV=EMT+6
455	104007	STTXV=EMT+7
456	104010	EHALT=EMT+10
457	104011	SRESET=EMT+11
458	104012	SCOPE=EMT+12
459	104013	SAVREG=EMT+13
460	104014	RSTREG=EMT+14
461	104015	ERROR1=EMT+15
462	104016	DELAY=EMT+16
463	104017	TIMERX=EMT+17
464	104020	TIMETX=EMT+20
465	177777	ATLAST=-1
466	100000	CD=100000
467		
468		.LIST ME
469	000174	.=174
470	000174 177570	SRPTR: 177570
471	000176 000000	SOFTSR: 000000
472	000200	.=200
473	000200 000137 001640	JMP @#STARTZ
474	000204	.=204
475	000204 000137 006414	JMP @#RESTART

```

;POP THE STACK. SAME AS TST (6)+
;POP STACK TWICE. SAME AS CMP (6)+,(6)+
;PRIORITY LEVEL DEFINITIONS

```

```

;FLAG FOR C/D TESTS

```

```

;GO TO START OF PROGRAM.

```

K01

.MAIN. MACY11 27(732) 25-FEB-76 10:49 PAGE 10
DZDLAF.P11

476 001200 . =1200

477
478
479
480
481
482
483 001200 175610
484 001202 175620
485 001204 175630
486 001206 175640
487 001210 175650
488 001212 175660
489 001214 175670
490 001216 175700
491 001220 175710
492 001222 175720
493 001224 175730
494 001226 175740
495 001230 175750
496 001232 175760
497 001234 175770
498 001236 176000
499 001240 176010
500 001242 176020
501 001244 176030
502 001246 176040
503 001250 176050
504 001252 176060
505 001254 176070
506 001256 176100
507 001260 176110
508 001262 176120
509 001264 176130
510 001266 176140
511 001270 176150
512 001272 176160
513 001274 176170
514 001276 177777
515 001300 177777

DEVICE ADDRESS LIST
LSB BIT0 IS SET TO A 1 BY MAPPER IF DEVICE NOT FOUND
TO TEST THAT LINE NOT FOUND CLEAR BIT0 IN THAT DEVICE ADDRESS
IN THIS TABLE AFTER MAPPING DONE

RXCRO: 175610	;LINE 0 DEVICE ADDRESS (RXCSR)
RXCR1: 175620	;LINE 1 DEVICE ADDRESS (RXCSR)
RXCR2: 175630	;LINE 2 DEVICE ADDRESS (RXCSR)
RXCR3: 175640	;LINE 3 DEVICE ADDRESS (RXCSR)
RXCR4: 175650	;LINE 4 DEVICE ADDRESS (RXCSR)
RXCR5: 175660	;LINE 5 DEVICE ADDRESS (RXCSR)
RXCR6: 175670	;LINE 6 DEVICE ADDRESS (RXCSR)
RXCR7: 175700	;LINE 7 DEVICE ADDRESS (RXCSR)
RXCR10: 175710	;LINE 10 DEVICE ADDRESS (RXCSR)
RXCR11: 175720	;LINE 11 DEVICE ADDRESS (RXCSR)
RXCR12: 175730	;LINE 12 DEVICE ADDRESS (RXCSR)
RXCR13: 175740	;LINE 13 DEVICE ADDRESS (RXCSR)
RXCR14: 175750	;LINE 14 DEVICE ADDRESS (RXCSR)
RXCR15: 175760	;LINE 15 DEVICE ADDRESS (RXCSR)
RXCR16: 175770	;LINE 16 DEVICE ADDRESS (RXCSR)
RXCR17: 176000	;LINE 17 DEVICE ADDRESS (RXCSR)
RXCR20: 176010	;LINE 20 DEVICE ADDRESS (RXCSR)
RXCR21: 176020	;LINE 21 DEVICE ADDRESS (RXCSR)
RXCR22: 176030	;LINE 22 DEVICE ADDRESS (RXCSR)
RXCR23: 176040	;LINE 23 DEVICE ADDRESS (RXCSR)
RXCR24: 176050	;LINE 24 DEVICE ADDRESS (RXCSR)
RXCR25: 176060	;LINE 25 DEVICE ADDRESS (RXCSR)
RXCR26: 176070	;LINE 26 DEVICE ADDRESS (RXCSR)
RXCR27: 176100	;LINE 27 DEVICE ADDRESS (RXCSR)
RXCR30: 176110	;LINE 30 DEVICE ADDRESS (RXCSR)
RXCR31: 176120	;LINE 31 DEVICE ADDRESS (RXCSR)
RXCR32: 176130	;LINE 32 DEVICE ADDRESS (RXCSR)
RXCR33: 176140	;LINE 33 DEVICE ADDRESS (RXCSR)
RXCR34: 176150	;LINE 34 DEVICE ADDRESS (RXCSR)
RXCR35: 176160	;LINE 35 DEVICE ADDRESS (RXCSR)
RXCR36: 176170	;LINE 36 DEVICE ADDRESS (RXCSR)
XORADD: 177777	;LINE 37 SPECIAL ADDRESS FOR XOR
RXEND: 177777	;LINE XX DEVICE ADDRESS (RXCSR)

CHARACTER LENGTH, PRIORITY, C/D MASK
INITIALLY SET FOR DL11-E, PRIORITY=4, CHARACTER LENGTH=8
BIT 15 SET TO A 1 = THAT LINE HAS DL11-C OR DL11-D
EX: 140377 = DL11C OR DL11D, PRIORITY = 4, CHARACTER LENGTH = 8
BITS 12-14 = PRIORITY LEVEL THAT LINE
BITS 0-7 = CHARACTER MASK EX. 377=8, 177=7, 77=6, 37=5

524 001302 040377
525 001304 040377
526 001306 040377
527 001310 040377
528 001312 040377
529 001314 040377
530 001316 040377
531 001320 040377

CMAS0: 040377	;LINE 0 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS1: 040377	;LINE 1 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS2: 040377	;LINE 2 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS3: 040377	;LINE 3 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS4: 040377	;LINE 4 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS5: 040377	;LINE 5 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS6: 040377	;LINE 6 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS7: 040377	;LINE 7 CHARACTER MASK, PRIORITY, C/D FLAG

532 001322 040377
533 001324 040377
534 001326 040377
535 001330 040377
536 001332 040377
537 001334 040377
538 001336 040377
539 001340 040377
540 001342 040377
541 001344 040377
542 001346 040377
543 001350 040377
544 001352 040377
545 001354 040377
546 001356 040377
547 001360 040377
548 001362 040377
549 001364 040377
550 001366 040377
551 001370 040377
552 001372 040377
553 001374 040377
554 001376 040377
555 001400 040377
556
557 001402 000000
558 001404 000000
559 001406 177740
560
561 001410 000000
562 001412 000000
563 001414 000000
564 001416 000000
565 001420 000000
566 001422 000000
567 001424 000000
568 001426 000000
569
570
571 001430 177560
572 001432 177562
573 001434 177564
574 001436 177566
575 001440 000060
576 001442 000200
577 001444 000064
578 001446 000200
579 001450 000000
580 001452 000000
581 001454 000000
582 001456 000000
583 001460 000000
584 001462 000000
585 001464 000000
586 001466 000000
587 001470 000000

CMAS10: 040377
CMAS11: 040377
CMAS12: 040377
CMAS13: 040377
CMAS14: 040377
CMAS15: 040377
CMAS16: 040377
CMAS17: 040377
CMAS20: 040377
CMAS21: 040377
CMAS22: 040377
CMAS23: 040377
CMAS24: 040377
CMAS25: 040377
CMAS26: 040377
CMAS27: 040377
CMAS30: 040377
CMAS31: 040377
CMAS32: 040377
CMAS33: 040377
CMAS34: 040377
CMAS35: 040377
CMAS36: 040377
CMAS37: 040377
;UMASK: 0
;RMSK: 0
;STLMSK: 177740
;RXCSR: 0
;RXBUF: 0
;TXCSR: 0
;TXBUF: 0
;RXVTR: 0
;RXLVL: 0
;TXVTR: 0
;TXLVL: 0
;*****
;TKS: 177560
;TKB: 177562
;TPS: 177564
;TPB: 177566
;TKVTR: 60
;TKLVL: PRTY4
;TPVTR: 64
;TPLVL: PRTY4
;PRGNUM: OPEN
;KSTART: OPEN
;CURTST: OPEN
;RTNNO: OPEN
;TNNO: 0
;NXTST: OPEN
;ICTR: OPEN
;SCOPTR: OPEN
;OLDPS: 0

;LINE 10 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 11 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 12 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 13 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 14 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 15 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 16 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 17 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 20 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 21 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 22 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 23 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 24 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 25 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 26 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 27 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 30 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 31 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 32 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 33 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 34 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 35 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 36 CHARACTER MASK, PRIORITY, C/D FLAG
;LINE 37 SPECIAL ADDRESS FOR XOR
;MASK FOR DEVICE UT
;MASK FOR CHAR LENGTH FOR DEVICE UT
;MASK FOR MAX RANDOM STALL
;RECEIVER UNDER TEST
;RECEIVER BUFFER UNDER TEST
;TRANSMITTER CSR UNDER TEST
;TRANSMITTER BUFFER UNDER TEST
;RECEIVER VECTOR UNDER TEST
;RECEIVER PRIORITY LEVEL UT
;TRANSMITTER VECTOR UNDER TEST
;TRANSMITTER PRIORITY LEVEL UT
;LSR CSR
;LSR BUFFER
;LSP CSR
;LSP BUFFER
;LSR INTERRUPT VECTOR
;LSR PRIORITY LEVEL
;LSP INTERRUPT VECTOR
;LSP PRIORITY LEVEL
;CONTAINS CURRENT PROGRAM#
;CURRENT PROGRAM START ADDRESS.
;CONTAINS ADDR OF CURRENT TEST.
;CONTAINS CURRENT TEST #.
;CONTAINES EDITED TNUM
;CONTAINS ADDR OF NEXT TEST.
;CONTAINS CURRENT ITERATION COUNT
;CONTAINS CURRENT SCOPE POINTER.
;PS SAVED FROM TRAP TO EMT ROUTINE


```

644 001662 005777 176306          TST      @SRPTR          ; TRY TO REFERENCE THE
645                                     ; HARDWARE SWITCH REGISTER
646 001666 000404          BR       2$              ; BRANCH IF NO TIME OUT TRAP OCCURS
647 001670 012737 000176 000174 1$:  MOV     #SOFTSR,SRPTR  ; CHANGE THE SWITCH REGISTER POINTER
648                                     ; TO POINT TO A SOFTWARE SWITCH REGISTER
649 001676 022626          CMP     (6)+,(6)+      ; RESTORE THE STACK
650 001700 012637 000004          MOV     (6)+,4         ; RESTORE TIME OUT VECTOR
651 001704 012637 000006          MOV     (6)+,6
652 001710 005037 001626          CLR     @#FTITLE
653 001714 013746 000004          MOV     @#4,-(%6)
654 001720 012737 002020 000004    MOV     #XORA,@#4
655 001726 005737 177060          TST     @#177060
656 001732 012637 000004          MOV     (%6)+,@#4
657 001736 012737 174000 001276    MOV     #174000,@#XORADD
658 001744 012737 177777 002016    MOV     #-1,@#XORFLG
659 001752 104000          TYPE
660 001754 001762          MESS1
661 001756 000137 002044          JMP     @#START
662 001762 005015 047531 020125  MESS1: .ASCII <15><12>'YOU ARE ON AN XOR TESTER@'
663 001770 051101 020105 047117
664 001776 040440 020116 047530
665 002004 020122 042524 052123
666 002012 051105          100
667 002016 002016          .EVEN
668 002016 000000          XORFLG: .WORD 0
669
670 002020 022626          XORA:  CMP     (%6)+,(%6)+
671 002022 012637 000004          MOV     (%6)+,@#4
672 002026 012737 177777 001276    MOV     #-1,@#XORADD
673 002034 005037 002016          CLR     @#XORFLG
674 002040 000137 002044          JMP     @#START
675
676
677          .MACR  TSTAA  AX,B,C,D,E
678          ;*****
679          AT'E':  C          ;TEST NUMBER
680                AT'D'      ;ADDRESS OF NEXT TEST
681                B          ;ITERATION COUNT
682                'AX'A      ;SCOPE ENTRY POINT
683                X=X+1
684          ;*****
685          .ENDM
686          .MACR  TSTA  B,AX,Z
687          TSTAA  AX,B,\X+1+Z,\X+2,\X+1
688          .ENDM
689
690 002044 012706 001176          START: MOV     #SPBOT,%6      ;SET BOTTOM OF SP STACK.
691 002050 012737 006234 000024    MOV     #PFAIL,24
692 002056 005037 001612          CLR     FOUNDV
693 002062 005037 001472          CLR     FMAP
694 002066 004737 003300          JSR     %7,CLRCO      ;CLEAR DEVICE UT PARAMETERS
695 002072 004737 003516          JSR     %7,OVRLAY    ;OVERLAY TRAP AREA
696 002076 005737 001626          TST     FTITLE       ;TITLE PRINTED AND MAP MADE
697 002102 001054          BNE     STARTI       ;YES, SKIP OVER THIS
698 002104 104000          TYPE
699 002106 015034          MTIT

```

700	002110	005237	001626		INC	FTITLE	
701	002114	005037	001630		CLR	FNONE	: CLEAR DEVICE PRESENT FLAG
702	002120	012737	002160	000004	MOV	#MAPNE, MACHER	: SET UP NO DEVICE PRESENT RETURN
703	002126	012704	001200		MOV	#RXCR0, %4	: SET UP DEVICE POINTER
704	002132	021437	001300		MAPA: CMP	(%4), #RXEND	: LAST DEVICE
705	002136	001430			BEG	MAPEND	: YES, EXIT
706	002140	042714	000001		BIC	#BIT0, (4)	: CLEAR ODD ADDRESS
707	002144	005037	177776		CLR	PSW	
708	002150	005774	000000		TST	(4)	: TEST DEVICE
709	002154	000240			NOP		
710	002156	000404			BR	MAPOK	
711	002160	052724	000001		MAPNE: BIS	#BIT0, (4)+	: NOT LIVING
712	002164	022626			POPSP2		
713	002166	000761			BR	MAPA	
714	002170	012437	001620		MAPOK: MOV	(4)+, TEMP1	: SAVE DEVICE ADDRESS FOR TYPING
715	002174	004537	004474		JSR	%5, OACNV	
716	002200	001620			TEMP1		
717	002202	015144			MDEVAD		
718	002204	000006			S		
719	002206	104000			TYPE		
720	002210	015144			MDEVAD		
721	002212	005237	001630		INC	FNONE	: SET HAVE DEVICE
722	002216	000745			BR	MAPA	
723	002220	012737	005706	000004	MAPEND: MOV	#ERTP, MACHER	: RESET TRAPS
724	002226	005737	001630		TST	FNONE	: ANY DEVICES PRESENT
725	002232	001424			BEG	MAPERR	: NO, ERROR
726	002234	012701	001200		START1: MOV	#RXCR0, %1	
727	002240	032711	000001		START2: BIT	#BIT0, (1)	: IS DEVICE LIVING
728	002244	001013			BNE	START3	: NO, CHECK FOR END
729	002246	010137	001614		MOV	%1, LINENO	: CALCULATE LINE NUMBER UNDER TEST
730	002252	162737	001200	001614	SUB	#RXCR0, LINENO	
731	002260	006237	001614		ASR	LINENO	
732	002264	011101			MOV	(1), %1	: YES, LOAD AND EXIT
733	002266	004737	006072		JSR	%7, FORMAD	
734	002272	000420			BR	START4	
735	002274	005721			START3: TST	(1)+	
736	002276	020127	001300		CMP	%1, #RXEND	: END OF TABLE
737	002302	001356			BNE	START2	: NO, LOOP
738	002304	104000			MAPERR: TYPE		
739	002306	015220			MNONE		
740	002310	005737	000042		TST	#42	: MONITOR LOAD
741	002314	001402			BEG	+6	: NO, CONTINUE
742	002316	000137	005302		JMP	PRGXTL	: YES, EXIT
743	002322	005037	001626		CLR	FTITLE	
744	002326	000000			HALT		
745	002330	000137	002044		JMP	START	
746	002334	012737	000001	001636	START4: MOV	#1, PASCNT	
747	002342	005037	177776		CLR	PSW	
748	002346	005037	001456		CLR	RTNNO	
749	002352	104000			TYPE		: CALL FOR PROGRAM NUMBER.
750	002354	015164			PGMSG		
751	002356	004737	003554		JSR	PC, RDOCT	: READ IN PROGRAM NUMBER.
752	002362	012600			MOV	(SP)+, %0	: INPUT DATA TO RO
753	002364	042700	177770		BIC	#177770, %0	: LIMIT (SR) TO BITS 3-0
754	002370	010037	001450		MOV	%0, PRGNUM	: SAVE PROGRAM #
755	002374	006300			ASL	%0	

756	002376	000170	001474		JMP	2PRGTAB(0)		:GO TO SELECTED PROGRAM.
757								
758	002402	013737	001452	001462	GETRDY:	MOV	KSTART,NXTST	:ADDR OF 1ST ROUTINE TO NXTST
759	002410	012737	005706	000004	GTRDYX:	MOV	#ERTP,MACHER	:RESET MACHER TRAP.
760	002416	012737	000040	000006		MOV	#40,MACHER+2	
761	002424	005037	001472			CLR	FMAP	
762	002430	012706	001176			MOV	#SPBOT,%6	:SET BOTTOM OF STACK.
763	002434	104011				SRESET		:ISSUE RESET.
764	002436	005037	177776			CLR	PSW	
765	002442	004737	002734		GTRDYA:	JSR	%7,FORWD	:ROLL FORWARD TO "NEXT" ROUTINE.
766	002446	032777	001000	175520		BIT	#BIT9,2SRPTR	:CHECK SELECT ROUTINE SWITCH
767	002454	001011				BNE	GTRDYC	:BRANCH IF SELECT ROUTINE SWITCH IS SET.
768	002456	005737	001402			TST	UMASK	:C/D DEVICE
769	002462	100003				BPL	GTRDA1	:NO, CONTINUE
770	002464	005737	001456			TST	RTNNO	:THIS A C/D TEST
771	002470	100364				BPL	GTRDYA	:NO, DO NEXT TEST
772	002472	000177	176756		GTRDA1:	JMP	2CURTST	:GO RUN CURRENT ROUTINE.
773	002476	000464				BR	CHNB	:NO GO. MANUAL RTN BYPASSED.
774	002500	017700	175470		GTRDYC:	MOV	2SRPTR,%0	: (SR) TO RD
775	002504	042700	177600			BIC	#177600,%0	:MASK UNDESIRED BITS
776	002510	123700	001456			CMPB	RTNNO,%0	:COMPARE RTNNO TO (RD)
777	002514	001002				BNE	GTRDYD	:BRANCH IF ROUTINE NOT FOUND YET.
778	002516	000177	176732			JMP	2CURTST	:GO RUN ROUTINE.
779	002522	022737	177777	001462	GTRDYD:	CMP	#-1,NXTST	:NO, CHECK FOR LAST ROUTINE.
780	002530	001344				BNE	GTRDYA	:BRANCH IF NOT LAST ROUTINE.
781	002532	004737	005100			JSR	%7,INCRTN	:YES, INCORRECT ROUTINE SELECTED.
782	002536	000721				BR	GETRDY	:START OVER.
783								
784	002540	032777	040000	175426	CHAINN:	BIT	#BIT14,2SRPTR	:CHECK FOR SCOPE OPTION.
785	002546	001403				BEQ	CHNA	:BRANCH IF SCOPE SW NOT SET.
786	002550	013716	001466		CHNAB:	MOV	SCOPTR,%6	:SET UP TO RETURN TO ROUTINE.
787	002554	000002				RTI		:RETURN TO ROUTINE.
788	002556	005737	002016		CHNA:	TST	2XORFLG	
789	002562	100011				BPL	1\$	
790	002564	013746	000004			MOV	2#4,-(%6)	
791	002570	012737	002676	000004		MOV	#XOR,2#4	
792	002576	005737	177060			TST	2#177060	:TEST FOR XOR
793	002602	012637	000004			MOV	(%6)+,2#4	
794	002606	032777	004000	175360	1\$:	BIT	#BIT11,2SRPTR	:TEST INHIBIT ITERATION SWITCH
795	002614	001003				BNE	CHNAA	:BRANCH IF INHIBIT ITERATION SW SET.
796	002616	005337	001464			DEC	ICTR	:DECREMENT ITERATION COUNT.
797	002622	001352				BNE	CHNAB	:BRANCH IF COUNT NOT 0.
798	002624	022626			CHNAA:	POPSP2		:POP STACK TWICE
799	002626	032777	002000	175340		BIT	#BIT10,2SRPTR	
800	002634	001405				BEQ	CHNB	
801	002636	013700	001456			MOV	RTNNO,%0	
802	002642	042700	100000			BIC	#BIT15,%6	
803	002646	000000				HALT		
804	002650	032777	001000	175316	CHNB:	BIT	#BIT9,2SRPTR	:CHECK SELECT ROUTINE SWITCH
805	002656	001251				BNE	GETRDY	:BRANCH IF SELECT RTN SW SET
806	002660	022737	177777	001462		CMP	#-1,NXTST	:LAST TEST?
807	002666	001250				BNE	GTRDYX	:BRANCH IF NOT LAST TEST.
808	002670	004737	005122			JSR	%7,PRGEND	:PROGRAM END.
809	002674	000642				BR	GETRDY	
810								
811	002676	022626			XOR:	CMP	(%6)+,(%6)+	


```

012 002700 012637 000004      MOV      (%5)+,2#4
013 002704 000721      BR       CHNAB
014
015      ;INIT FOR C/D - WITHOUT JUMPER RESET STARTS ASSEMBLING CHARACTER SETTING DONE
016      ;SET MAINT. DELAY, CLEAR RX DONE
017 002706 005737 001402      CDINIT: TST      UMASK      ;C-D DEVICE
018 002712 100007      9PL     CDINX      ;NO EXIT
019 002714 052777 000004 176472      BIS     #BIT2,2TXCSR ;SET MAINT BIT
020 002722 104016      DELAY   1500.      ;WAIT 1.5 SEC
021 002724 002734
022 002726 005777 176460      TST     2RXBUF     ;CLEAR RX DONE
023 002732 000207      CDINX: RTS      %7
024
025
026 002734 013705 001462      FORWD: MOV      NXTST,%5      ;ADDR OF NEXT ROUTINE TO R5.
027 002740 012537 001456      MOV     (5)+,RTNNO      ;GET NEXT ROUTINE NUMBER.
028 002744 012537 001462      MOV     (5)+,NXTST      ;GET ADDR OF NEXT "NEXT" ROUTINE.
029 002750 012537 001464      MOV     (5)+,ICTR      ;GET ITERATION COUNT.
030 002754 012537 001466      MOV     (5)+,SCOPTR     ;GET SCOPE LOOP ENTRY POINTER.
031 002760 010537 001454      MOV     %5,CURTST      ;ADDR OF NOW CURRENT TEST TO CURTST.
032 002764 000207      RTS      %7      ;EXIT FORWD SUBROUTINE.
033
034 002766 011646      EMTINT: MOV     2%6,-(6)      ;GET SAVED PC.
035 002770 162716 000002      SUB     #2,2%6      ;DECREMENT PC BY 2.
036 002774 017616 000000      MOV     2(6),2%6
037 003000 006316      EMTA:  ASL     2%6      ;EMT ARG X 2.
038 003002 042716 177001      BIC     #177001,2%6      ;REMOVE 7 MSB.
039 003006 062716 001514      ADD     #EMTTAB,2%6      ;FORM EMT RTN ADDR.
040 003012 017616 000000      MOV     2(6),2%6
041 003016 000136      JMP     2(6)+      ;GO TO EMT ROUTINE.
042
043      ;SAVE REGS 0 TO 4 SUBROUTINE.
044
045 003020 012637 003054      SAVRG: MOV     (6)+,SVRPC      ;SAVE PC AND PSW.
046 003024 012637 003056      MOV     (6)+,SVRPSW
047 003030 010446      MOV     %4,-(6)      ;SAVE REGS 0 - 4
048 003032 010346      MOV     %3,-(6)      ;IN STACK.
049 003034 010246      MOV     %2,-(6)
050 003036 010146      MOV     %1,-(6)
051 003040 010046      MOV     %0,-(6)
052 003042 013746 003056      MOV     SVRPSW,-(6)      ;RESTORE PC AND PSW.
053 003046 013746 003054      MOV     SVRPC,-(6)
054 003052 000002      RTI      ;EXIT.
055 003054 000000
056 003056 000000
057
058      ;RESTORE REGS 0 TO 4 SUBROUTINE.
059
060 003060 012637 003114      RSTRG: MOV     (6)+,RSTPC      ;SAVE PC AND PSW.
061 003064 012637 003116      MOV     (6)+,RSTPSW
062 003070 012600      MOV     (6)+,%0      ;RESTORE REGS 0 - 4
063 003072 012601      MOV     (6)+,%1      ;FROM STACK.
064 003074 012602      MOV     (6)+,%2
065 003076 012603      MOV     (6)+,%3
066 003100 012604      MOV     (6)+,%4
067 003102 013746 003116      MOV     RSTPSW,-(6)      ;RESTORE PC AND PSW.

```

```

868 003106 013746 003114      MOV      RSTPC,-(6)
869 003112 000002      RTI          ;EXIT
870 003114 000000      RSTPC: OPEN
871 003116 000000      RSTPSW: OPEN

:ROUTINE TO SET RECEIVER INTERRUPT VECTOR AND PRIORITY
874 003120 004737 006270      STLSRV: JSR  %7,TSTVEC
875 003124 017637 000000 003144      MOV      2(6),STPRA+2 ;MOVE VECTOR ADDR TO STPRA+2
876 003132 062716 000002      ADD      #2,2%6      ;SET UP EXIT
877 003136 013701 001420      MOV      RXVTR,%1
878 003142 012721 000000      STPRA:  MOV  #OPEN,(1)+ ;SET VECTOR ADDRESS
879 003146 013721 001422      MOV      RXLVL,(1)+  ;SET PRIORITY
880 003152 000002      RTI          ;EXIT

:ROUTINE TO SET TRANSMITTER INTERRUPT VECTOR AND PRIORITY.
883 003154 004737 006270      STLSPV: JSR  %7,TSTVEC
884 003160 017637 000000 003200      MOV      2(6),STPPA+2 ;MOVE VECTOR ADDR TO STPPA+2
885 003166 062716 000002      ADD      #2,2%6      ;SET UP EXIT
886 003172 013701 001424      MOV      TXVTR,%1
887 003176 012721 000000      STPPA:  MOV  #OPEN,(1)+ ;SET VECTOR ADDRESS.
888 003202 013721 001426      MOV      TXLVL,(1)+  ;SET PRIORITY
889 003206 000002      RTI          ;EXIT.

:ROUTINE TO ISSUE RESET.
892 003210 012700 052525      SRSETT: MOV  #52525,%0 ;DATA TO R0.
893 003214 005100      COM      %0      ;COMPLEMENT (R0).
894 003216 010037 003212      MOV      %0,SRSETT+2 ; (R0) TO SRSETT+2.
895 003222 000005      RESET    ;ISSUE RESET. (R0) IS
896 003224 000002      RTI          ;DISPLAYED. EXIT.

:RANDOM NUMBER GENERATOR. ROUTINE EXITS WITH NUMBER IN REGISTER 0.
899 003226 013700 003274      RNGEN:  MOV  RP1,%0
900 003232 006100      ROL      %0
901 003234 006100      ROL      %0
902 003236 063700 003276      ADD      RP2,%0
903 003242 010037 003274      MOV      %0,RP1
904 003246 006100      ROL      %0
905 003250 006100      ROL      %0
906 003252 063700 003276      ADD      RP2,%0
907 003256 006100      ROL      %0
908 003260 006100      ROL      %0
909 003262 010037 003276      MOV      %0,RP2
910 003266 013700 003274      MOV      RP1,%0
911 003272 000207      RTS      %7      ;EXIT. NUMBER IN R0
912 003274 001233      RP1:     1233
913 003276 007622      RP2:     7622

:CLRCD - CLEAR CURRENT DEVICE PARAMETERS
916 003300 005037 001416      CLRCD:  CLR  TXBUF
917 003304 005037 001414      CLR      TXCSR
918 003310 005037 001410      CLR      RXCSR
919 003314 005037 001412      CLR      RXBUF
920 003320 005037 001420      CLR      RXVTR
921 003324 005037 001424      CLR      TXVTR
922 003330 005037 001422      CLR      RXLVL
923 003334 005037 001426      CLR      TXLVL
  
```

```

924 003340 000207          RTS      %7
925
926          ;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER.
927          TYP:  MOV      %0,%0          ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS.
928 003342 011600          ADD      #2,%6          ;SET UP EXIT.
929 003344 062716 000002          MOV      %0,%0          ;ADDRESS OF MESSAGE TO RO.
930 003350 011000          TYPA:  MOVVB   (%0),TYPDAT ;GET CHARACTER
931 003352 112037 003462          CMPB    #100,TYPDAT      ;CHECK FOR"@"CHARACTER
932 003355 122737 000100 003462          SNE     TYPC            ;BRANCH IF NOT"@".
933 003364 001001          RTI     ;TERMINATOR CHAR. DONE. EXIT.
934 003366 000002          TYPC:  CMPB    #45,TYPDAT ;CHECK FOR"%".
935 003370 122737 000045 003462          BEQ     TYPF            ;BRANCH IF"%".
936 003376 001416          CMPB    #43,TYPDAT      ;NOT"%".CHECK FOR"#".
937 003400 122737 000043 003462          BEQ     TYPG            ;BRANCH IF"#".
938 003406 001417          JSR     %7,TYPD         ;TYPE CHAR IN TYPDAT
939 003410 004737 003416          BR      TYPA
940 003414 000756          TYPD:  MOVVB   TYPDAT,%TPB ;OUTPUT CHARACTER TO PRINTER
941 003416 113777 003462 176012          TSTB   %TPS            ;WAIT FOR DONE FLAG.
942 003424 105777 176004          BPL     -4
943 003430 100375          RTS     %7
944 003432 000207          TYPF:  MOVVB   #15,TYPDAT ;MOVE CARRIAGE RETURN CODE TO TYPDAT
945 003434 112737 000015 003462          JSR     %7,TYPD         ;GO TYPE CHAR.
946 003442 004737 003416          TYPG:  MOVVB   #12,TYPDAT ;MOVE LF CODE TO TYPDAT.
947 003446 112737 000012 003462          JSR     %7,TYPD         ;GO TYPE CHAR.
948 003454 004737 003416          BR      TYPA
949 003460 000734          TYPDAT: OPEN
950 003462 000000
951          ;SUBROUTINE TO OUTPUT A SERIES OF ASCII MESSAGES ON TELETYPE PRINTER
952          TYP:  MOV      %0,%0          ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
953 003464 011600          ADD      #2,%6          ;UPDATE TO NEXT MESSAGE ADDRESS
954 003466 062716 000002          MOV      %0,%0          ;ADDRESS OF MESSAGE TO TYP
955 003472 011037 003512          CMP     #-1,TYP        ;CHECK FOR TERMINATOR
956 003476 022737 177777 003512          BNE     TYP          ;BRANCH IF NOT TERMINATOR.
957 003504 001001          RTI     ;TERMINATOR. EXIT
958 003506 000002          TYP:  TYPE     ;CALL ON TYP SUB TO TYPE MESSAGE
959 003510 104000          TYP:  OPEN     ;ADDRESS OF MESSAGE GOES HERE
960 003512 000000          BR      TYP          ;GO PROCESS NEXT MESSAGE
961 003514 000763
962          ;OVERLAY VECTOR AREA
963          OVRLAY: MOV     #300,%1          ;GET DL11-E VECTOR BASE ADDRESS
964 003516 012701 000300          MOV     #302,%2
965 003522 012702 000302          MOV     #4,%3
966 003526 012703 000004          OVRLYA: MOV    %2,(1)+      ;LOAD VECTOR WITH IOT ERROR TRAP
967 003532 010221          MOV     %3,(1)+
968 003534 010321          ADD     #4,%2
969 003536 062702 000004          CMP     %1,#1000          ;ALL VECTORS BEEN LOADED
970 003542 020127 001000          BEQ     OVRLYB
971 003546 001401          BR      OVRLYA
972 003550 000770          OVRLYB: RTS     7          ;EXIT
973 003552 000207
974          ;SUBROUTINE TO READ OCTAL DATA FROM THE TELETYPE PRINTER
975          RDOCT: MOV     (SP),-(SP)        ;MAKE ROOM FOR DATA WORD
976 003554 011646          MOV     %0,-(SP)        ;SAVE RO
977 003556 010046          MOV     %1,-(SP)        ;SAVE R1
978 003560 010146          INDAT: CLR    %1          ;CLEAR DATA WORD
979 003562 005001          CLR    COUNT            ;SET NO. OF DIGITS = 0
980 003564 005037 001624

```

```

980 003570 105777 175634 RDDAT: TSTB 3TKS ;TEST TTY READ STATUS
981 003574 100375 BPL RDDAT ;WAIT
982 003576 117746 175630 MOVB 3TKB, -(SP) ;PUSH DIGIT ON STACK
983 003602 042716 000200 BIC #BIT7, (SP)
984 003606 105777 175622 ECDAT: TSTB 3TPS ;TEST TTY PRINT STATUS
985 003612 100375 BPL ECDAT ;WAIT
986 003614 111677 175616 MOVB (SP), 3TPB ;ECHO CHARACTER
987 003620 122716 000015 CMPB #15, (SP) ;IS IT A TERMINATOR?
988 003624 001432 BEQ RETRN ;BR IF YES
989 003626 122716 000177 CMPB #177, (SP) ;IS IT A RUBOUT?
990 003632 001423 BEQ RREAD ;BR IF YES
991 003634 122716 000060 CMPB #60, (SP) ;IS IT AN OCTAL DIGIT?
992 003640 003020 SGT RREAD ;BR IF NO
993 003642 122716 000067 CMPB #67, (SP) ;TEST AGAIN
994 003646 002415 BLT RREAD ;BR IF NO
995 003650 005237 001624 INC COUNT ;INC NO. OF DIGITS
996 003654 022737 000067 001624 CMP #67, COUNT ;MORE THAN SIX DIGITS?
997 003662 003407 BLE RREAD ;BR IF YES
998 003664 006301 ASL %1 ;CLEAR LOWEST THREE BITS
999 003666 006301 ASL %1 ;OF DATA WORD
1000 003670 006301 ASL %1
1001 003672 162716 000060 SUB #60, (SP) ;CONVERT TO BINARY
1002 003676 062601 ADD (SP)+, %1 ;ADD DIGIT TO DATA WORD
1003 003700 000733 BR RDDAT ;GET NEXT DIGIT
1004 003702 104000 RREAD: TYPE ;TELL USER ABOUT ILLEGAL CHARACTER
1005 003704 017262 DTERR
1006 003706 005726 TST (SP)+ ;GET RID OF ILLEGAL CHARACTER
1007 003710 000724 BR INDAT ;START SUBROUTINE AGAIN
1008 003712 010166 000010 RETRN: MOV %1, 10(SP) ;STORE DATA WORD ON STACK
1009 003716 005726 TST (SP)+ ;INC STACK POINTER
1010 003720 012601 MOV (SP)+, %1 ;RESTORE R1
1011 003722 012600 MOV (SP)+, %0 ;RESTORE R0
1012 003724 000207 RTS PC ;RETURN
1013
1014 ;SUBROUTINE TO DELAY A SPECIFIED NUMBER OF MILLISECONDS
1015 003726 011637 003772 DLY: MOV 3%6, DLCNT ;GET DELAY COUNT ADDRESS.
1016 003732 062716 000002 ADD #2, 3%6 ;SET UP EXIT ADDRESS
1017 003736 017746 000030 MOV 3DLCNT, -(6) ;DELAY COUNT TO STACK
1018 003742 001411 BEQ DLYC
1019 003744 005037 177776 CLR PSW ;SET PRIORITY 0
1020 003750 012746 000226 DLYA: MOV #226, -(6) ;1 MSEC COUNT TO STACK
1021 003754 005316 DLYB: DEC 3%6 ;DECREMENT 1 MSEC COUNT
1022 003756 001376 BNE DLYB ;BRANCH IF NOT 0.
1023 003760 005726 POPSP ;ZERO. UNCOVER MSECS. COUNT.
1024 003762 005316 DEC 3%6 ;DECREMENT IT
1025 003764 001371 BNE DLYA ;BR IF NOT DONE DELAYING
1026 003766 005726 DLYC: POPSP ;DONE
1027 003770 000002 RTI ;EXIT.
1028 003772 000000 DLCNT: OPEN ;CONTAINS MILLISECONDS COUNT ADDRESS.
1029
1030 ;SUBROUTINE TO STALL A RANDOM NUMBER OF MILLISECONDS. MAXIMUM STALL
1031 ;DETERMINED BY CONTENTS OF LOC STLMSK.
1032 003774 004737 003226 STAL: JSR %7, RNGEN ;GO GET RANDOM NUMBER.
1033 004000 043700 001406 BIC STLMSK, %0 ;# IN R0. APPLY STALL MASK.
1034 004004 001404 BEQ STALB ;BRANCH IF RESULT IS 0.
1035 004006 010037 004014 MOV %0, STALA

```

```

1036 004012 104016          DELAY          ;DELAY
1037 004014 000000          STALA: OPEN          ;DELAY COUNT
1038 004016 000002          STALB: RTI           ;DONE. EXIT.
1039
1040          ;SUBROUTINE TO GENERATE RANDOM CHARACTER COUNT
1041 004020 004737 003226  GRCNT: JSR    %7,RNGEN ;GET RANDOM NUMBER
1042 004024 043700 004040          BIC    RCMSK,%0      ;APPLY MASK
1043 004030 001773          BEQ    GRCNT         ;TRY AGAIN IF RESULT 0
1044 004032 010037 004042          MOV    %0,RNCNT    ;COUNT TO RNCNT
1045 004036 000207          RTS    %7          ;EXIT.
1046 004040 000000          RCMSK: OPEN        ;RANDOM CHARACTER MASK.
1047 004042 000000          RNCNT: OPEN       ;RANDOM CHARACTER COUNT.
1048
1049          ;SUBROUTINE TO SKIP ON FLAG AND TIME OUT IF SKIP FAILS
1050 004044 013737 001410 004112 TMRX: MOV    RXCSR,SIOT ;SET UP RXCSR ADDRESS
1051 004052 000403          BR    TIME1
1052 004054 013737 001414 004112 TMTX: MOV    TXCSR,SIOT ;SET UP TXCSR ADDRESS
1053 004062 005037 004110          TIME1: CLR   TIMER
1054 004066 005237 004110          TIME2: INC   TIMER
1055 004072 001405          BEQ    TIMEX       ;BRANCH IF COUNTER OVERFLOW
1056 004074 105777 000012          TSTB  %SIOT
1057 004100 100372          BPL    TIME2
1058 004102 062716 000002          ADD   #2,%6       ;SET UP EXIT RETURN
1059 004106 000002          TIMEX: RTI
1060 004110 000000          TIMER: 0
1061 004112 000000          SIOT: 0
1062
1063          ;SUBROUTINE TO SELECT LINE
1064 004114 032777 010000 174052 LINSEL: BIT  #BIT12,%SRPTR ;BRANCH IF SET
1065 004122 001003          BNE   LINSLX
1066 004124 005037 001612          CLR   FOUNDV
1067 004130 000205          RTS   5
1068 004132 004737 003516  LINSLX: JSR  %7,OVRLAY
1069 004136 004737 003300          JSR  %7,CLRCD
1070 004142 104000          TYPE
1071 004144 016666          LDLINE
1072 004146 004737 003554          JSR  PC,RDOCT
1073 004152 012637 001616          MOV  (SP)+,TEMP
1074 004156 042737 177740 001616  BIC  #177740,TEMP
  
```

1075	004164	013737	001616	001614	MOV	TEMP.LINENO	;SAVE FOR TYPING
1076	004172	006337	001616		ASL	TEMP	

```

1077 004176 013701 001616      MOV      TEMP,%1
1078 004202 016101 001200      MOV      RXCR0(1),%1      ;GET RXCSR DEVICE ADDRESS
1079 004206 032701 000001      BIT      #BIT0,%1        ;IS DEVICE THERE
1080 004212 001403          BEQ      LINB            ;YES
1081 004214 104000          LINA:    TYPE           ;NO, REPORT
1082 004216 017217          MNOLIN
1083 004220 000744          BR       LINS LX
1084 004222 004737 006072      JSR      %7,FORMAD
1085 004226 005037 177776      CLR      PSW
1086 004232 052737 000001 001472      BIS      #BIT0,FMAP      ;SET MAPPING FLAG
1087 004240 042777 000100 175146      BIC      #BIT6,@TXCSR
1088 004246 052777 000100 175140      BIS      #BIT6,@TXCSR
1089 004254 000240          NOP
1090 004256 000240          NOP
1091 004260 005737 001420      TST      RXVTR
1092 004264 001753          BEQ      LINA
1093 004266 042777 000100 175120      BIC      #BIT6,@TXCSR
1094 004274 012737 000340 177776      MOV      #PTY7,PSW
1095 004302 004537 004474      JSR      5,OACNV        ;TYPE LINE #
1096 004306 001614          LINENO
1097 004310 016725          SELINE
1098 004312 000002          2
1099 004314 104000          TYPE
1100 004316 016714          ALINE
1101 004320 000205          RTS      5
1102
1103          ;SUBROUTINE TO INITIALIZE BINARY COUNT PATTERNS
1104 004322 012737 177777 004344      INBIN:  MOV      #-1,RIND      ;SET ALL VARIABLES
1105 004330 004537 004562          JSR      %5,BMOVE        ;TO MINUS 1.
1106 004334 004344          RIND
1107 004336 004345          RIND+1
1108 004340 000013          11.
1109 004342 000207          RTS      %7            ;EXIT
1110 004344 000000          RIND:  OPEN
1111 004346 000000          PTO:   OPEN
1112 004350 000000          PT1:  OPEN
1113 004352 000000          PIND: OPEN
1114 004354 000000          PTO:  OPEN
1115 004356 000000          PTIP: OPEN
1116
1117          ;SPECIAL BINARY COUNT PATTERN SUBROUTINE. EXITS WITH BIN CHAR IN RO
1118 004360 013737 004346 004350      GTBIN:  MOV      PTO,PT1      ;PREVIOUS BIN CHAR TO PT1
1119 004366 005137 004350          COM      PT1
1120 004372 005137 004344          COM      RIND
1121 004376 001002          BNE      .+6
1122 004400 005237 004350          INC      PT1
1123 004404 042737 177400 004350      BIC      #177400,PT1     ;MASK TO 8 BITS
1124 004412 013737 004350 004346      MOV      PT1,PTO        ;SAVE BIN CHAR IN PTO
1125 004420 013700 004350          MOV      PT1,%0        ;BIN CHAR TO RO.
1126 004424 000207          RTS      %7            ;EXIT.
1127 004426 013737 004354 004356      GTBINP: MOV      PTO,PTIP      ;PREVIOUS BIN CHAR TO PTIP
1128 004434 005137 004356          COM      PTIP
1129 004440 005137 004352          COM      PIND
1130 004444 001002          BNE      .+6
1131 004446 005237 004356          INC      PTIP
1132 004452 042737 177400 004356      BIC      #177400,PTIP   ;MASK TO 8 BITS.

```

```

1133 004460 013737 004356 004354      MOV      PT1P,PTOP      ;SAVE BIN CHAR IN PTOP.
1134 004466 013701 004356          MOV      PT1P,%1       ;BIN CHAR TO R1.
1135 004472 000207          RTS      %7            ;EXIT.
1136
1137          ;OCTAL TO ASCII CONVERT ROUTINE
1138          OACNV: SAVREG
1139 004474 104013          MOV      @ (5)+,OACNVX ;GET OCTAL VALUE.
1139 004476 013537 004560          MOV      (5)+,%1       ;GET DESTINATION ADDR.
1140 004502 012501          MOV      (5)+,%2       ;GET CONVERT COUNT.
1141 004504 012502          ADD      %2,%1         ;DEVELOP ADDR TO STORE 1ST CHAR.
1142 004506 060201          OACNVA: MOV      OACNVX,%3
1143 004510 013703 004560          BIC      #177770,%3    ;ISOLATE LEAST SIGNIFICANT DIGIT.
1144 004514 042703 177770          ADD      #60,%3        ;CONVERT DIGIT TO ASCII.
1145 004520 062703 000060          MOV      %3,-(1)      ;STORE ASCII CHARACTER.
1146 004524 110341          BIC      #7,OACNVX
1147 004526 042737 000007 004560          ROR      OACNVX
1148 004534 006037 004560          ROR      OACNVX
1149 004540 006037 004560          ROR      OACNVX
1150 004544 006037 004560          DEC      %2            ;DONE ALL DIGITS?
1151 004550 005302          BNE      OACNVA        ;BRANCH IF NOT DONE.
1152 004552 001356          RSTREG
1153 004554 104014          RTS      %5            ;DONE. EXIT.
1154 004556 000205          OACNVX: OPEN
1155 004560 000000
1156
1157          ;SUBROUTINE TO MOVE A VARIABLE NUMBER OF BYTES.
1158          BMOVE: SAVREG
1158 004562 104013          MOV      (5)+,%1       ;SAVE REGS.
1159 004564 012501          MOV      (5)+,%2       ;GET "FROM" ADDRESS
1160 004566 012502          MOV      (5)+,%3       ;GET "TO" ADDRESS
1161 004570 012503          MOV      (1)+,(2)+    ;GET COUNT
1162 004572 112122          BMOVA: MOV      %3      ;MOVE BYTE
1163 004574 005303          DEC      %3            ;DECREMENT COUNT
1164 004576 001375          BNE      BMOVA        ;BRANCH IF NOT DONE.
1165 004600 104014          RSTREG
1166 004602 000205          RTS      %5            ;RESTORE REGS.
1167          ;DONE EXIT
1168          ;BINARY TO DECIMAL ASCII CONVERT SUBROUTINE.
1169 004604 104013          BDCNV: SAVREG
1170 004606 012700 004762          MOV      #DECVAL,%0    ;SET UP ADDR TO STORE DECIMAL ASCII IN RD
1171 004612 013501          MOV      @ (5)+,%1     ;BINARY VALUE TO R1.
1172 004614 012537 004672          MOV      (5)+,BDCNVC   ;GET DEST ADDR
1173 004620 012537 004674          MOV      (5)+,BDCNVD   ;GET CHAR COUNT
1174 004624 012702 004750          MOV      #ADTENP,%2    ;ADDR OF TEN POWER STRING TO R2.
1175 004630 012737 000005 004742          MOV      #5,CNVCTR     ;SET UP FOR 5 POWER CONVERSIONS.
1176 004636 012237 004746          BDCNVA: MOV      (2)+,TENPWR ;MOVE POWER OF TEN VALUE TO TENPWR.
1177 004642 004737 004702          JSR      %7,SUBTEN     ;PERFORM CONVERSION
1178 004646 005337 004742          DEC      CNVCTR        ;DONE 5 CONVERSIONS?
1179 004652 001371          BNE      BDCNVA        ;BRANCH IF NOT YET 5.
1180 004654 163700 004674          SUB      BDCNVD,%0
1181 004660 010037 004670          MOV      %0,BDCNVB
1182 004664 004537 004562          JSR      %5,BMOVE
1183 004670 000000          BDCNVB: 0
1184 004672 000000          BDCNVC: 0
1185 004674 000000          BDCNVD: 0
1186 004676 104014          RSTREG
1187 004700 000205          RTS      %5            ;YES. EXIT.
1188 004702 005037 004744          SUBTEN: CLR      DIGIT ;CLEAR DIGIT

```


1189	004706	163701	004746			SUBTNA: SUB	TENPWR,%1	;SUBTRACT TEN POWER FROM BINARY VALUE.
1190	004712	103403				BCS	SUBTNB	;BRANCH IF UNSUCCESSFUL SUBTRACTION.
1191	004714	005237	004744			INC	DIGIT	
1192	004720	000772				BR	SUBTNA	
1193	004722	063701	004746			SUBTNB: ADD	TENPWR,%1	;RESTORE SUBTRACTED VALUE.
1194	004726	062737	000060	004744		ADD	#60,DIGIT	;CONVERT (DIGIT) TO ASCII
1195	004734	113720	004744			MOVB	DIGIT,(0)+	;MOVE ASCII CHAR TO DECVAL FIELD.
1196	004740	000207				RTS	%7	;EXIT.
1197	004742	000000				CNVCTR: OPEN		
1198	004744	000000				DIGIT: OPEN		
1199	004746	000000				TENPWR: OPEN		
1200	004750	023420				ADTENP: 10000.		
1201	004752	001750					1000.	
1202	004754	000144					100.	
1203	004756	000012					10.	
1204	004760	000001					1	
1205	004762	040	040	040		DECVAL: .BYTE	040,040,040,040,040,040	
1206	004765	040	040	040				
1207	004770	042777	000002	174412		DATTST: BIC	#BIT1,@RXCSR	;CLEAR DATA TERM. READY
1208	004776	052777	000004	174410		BIS	#BIT2,@TXCSR	;SET MAINTENANCE BIT
1209	005004	012737	000144	001564		MOV	#100,@CTRO	;GET CHARACTER COUNT
1210	005012	105777	174376			DATAA: TSTB	@TXCSR	;WAIT FOR
1211	005016	100375				BPL	-4	;READY FLAG
1212	005020	004737	004426			JSR	?GTBINP	;GET CHARACTER
1213	005024	110137	001560			MOVB	%1,CRBUFA	;MOVE CHARACTER
1214	005030	004737	005374			JSR	?MASKIT	;MASK OFF NON TRANSMITTED BITS
1215	005034	110177	174356			MOVB	%1,@TXBUF	;TRANSMIT CHARACTER
1216	005040	105777	174344			TSTB	@RXCSR	;WAIT FOR
1217	005044	100375				BPL	-4	;DONE FLAG
1218	005046	117737	174340	001556		MOVB	@RXBUF,CRBUF	;GET RECEIVED CHARACTER
1219	005054	104004				DATCHK		;CHK DATA
1220	005056	005337	001564			DEC	CTRO	;DECREMENT CHARACTER COUNT
1221	005062	001353				BNE	DATAA	
1222	005064	005726				TST	(6)+	;POP STACK
1223	005066	104012				SCOPE		
1224						SETSRS: TYPE		;TYPE SELECT OPTION MESSAGE.
1225	005070	104000				ASETSRS		
1226	005072	016126				HALT		;COMMON HALT.
1227	005074	000000				RTS	%7	;EXIT.
1228	005076	000207				INCRTN: TYPE		;TYPE INCORRECT ROUTINE SELECTED.
1229	005100	104000				AINCRN		
1230	005102	016225				HALT		;COMMON HALT.
1231	005104	000000				RTS	%7	;EXIT.
1232	005106	000207				INCRPG: TYPE		
1233	005110	104000				AINCPG		
1234	005112	016346				HALT		
1235	005114	000000				JMP	START	
1236	005116	000137	002044			CLR	FOUNDV	
1237	005122	005037	001612			PRGEND: BIT	#BIT13,@SRPTR	;INHIBIT PRINT SET?
1238	005126	032777	020000	173040		BNE	PRGEXT	;BR IF SET
1239	005134	001026				JSR	%5,BDCNV	
1240	005136	004537	004604			PASCNT		
1241	005142	001636				APCNT		
1242	005144	016416				6		
1243	005146	000006				JSR	%5,OACNV	;CONVERT LINE NUMBER
1244	005150	004537	004474					

1245	005154	001614				LINENO		
1246	005156	016436				ACLIN		
1247	005160	000002				2		
1248	005162	004537	004474			JSR	%5, OACNV	; CONVERT RXCSR
1249	005166	001410				RXCSR		
1250	005170	016452				APRXC		
1251	005172	000006				6		
1252	005174	004537	004474			JSR	%5, OACNV	; CONVERT VECTOR
1253	005200	001420				RXVTR		
1254	005202	016473				APVEC		
1255	005204	000004				4		
1256	005206	104000				TYPE		; TYPE PROGRAM END.
1257	005210	016401				APGEN		
1258	005212	032777	010000	172754	PRGEXT:	BIT	#BIT12, JSRPTR	; LOCK ON LINE
1259	005220	001403				BEQ	PRGXT1	; BR IF NOT SET
1260	005222	005237	001636			INC	PASCNT	
1261	005226	000425				BR	PRGXTL	
1262	005230	013737	001614	001616	PRGXT1:	MOV	LINENO, TEMP	; GET LINENO
1263	005236	006337	001616			ASL	TEMP	
1264	005242	062737	000002	001616	PRGEC:	ADD	#2, TEMP	; UPDATE LINE NUMBER
1265	005250	013701	001616		PRGEA:	MOV	TEMP, %1	
1266	005254	016101	001200			MOV	RXCRO(1), %1	; GET RXCSR DEVICE ADDRESS
1267	005260	022701	177777			CMP	#177777, %1	; LAST ONE
1268	005264	001023				BNE	PRGEB	; NO, CONTINUE
1269	005266	005237	001636			INC	PASCNT	
1270	005272	005037	001614			CLR	LINENO	
1271	005276	005037	001616			CLR	TEMP	
1272	005302	013705	000042		PRGXTL:	MOV	#42, %5	
1273	005306	001405				BEQ	CONT	
1274	005310	000005				RESET		
1275	005312	004715			LOGIC:	JSR	7, (5)	
1276	005314	000240				NOP		
1277	005316	000240				NOP		
1278	005320	000240				NOP		
1279	005322	032777	010000	172644	CONT:	BIT	#BIT12, JSRPTR	; LOCK ON LINE
1280	005330	001747				BEQ	PRGEA	; BRANCH IF NOT SET
1281	005332	000207				RTS	7	
1282	005334	032701	000001		PRGEB:	BIT	#BIT0, %1	; DEVICE THERE
1283	005340	001340				BNE	PRGEC	; NO
1284	005342	006237	001616			ASR	TEMP	
1285	005346	013737	001616	001614		MOV	TEMP, LINENO	
1286	005354	004737	006072			JSR	%7, FORMAD	
1287	005360	000207				RTS	%7	; EXIT.
1288								
1289								
1290	005362	005777	172606					
1291	005366	100001						
1292	005370	000000						
1293	005372	000002						
1294								
1295								
1296	005374	013737	001402	001404				
1297	005402	042737	177000	001404				
1298	005410	005137	001404					
1299	005414	043737	001404	001560				
1300	005422	000207						

; CONDITIONAL ERROR HALT ROUTINE.

```

EHLT: TST JSRPTR ; CHECK FOR HALT ON ERROR.
      BPL EHLTA ; BRANCH IF NO HALT DESIRED.
      HALT ; HALT.
EHLTA: RTI ; IN DATA LIGHTS.

```

; MASKIT - MASK DATA ACCORDING TO LINE NUMBER

```

MASKIT: MOV UMASK, RMASK ; GET MASK
        BIC #177000, RMASK ; REMOVE C/D FLAG+PRIORITY
        COM RMASK
        BIC RMASK, CRBUFA ; MASK DESIRED BITS
        RTS 7

```

```

1301
1302      ; DATA CHECK ROUTINE, TEST ERROR BITS
1303 005424 017737 173762 001562 DTCHK:  MOV  3,RXBUF,CRBUF  ;DID ANY ERROR BITS SET
1304 005432 032737 170000 001562      BIT  #170000,CRBUF
1305 005440 001004      BNE  DTCHKX      ;YES, TYPE ERROR
1306 005442 023737 001556 001560      CMP  CRBUF,CRBUFA ;COMPARE EXPECTED AND RECEIVED
1307 005450 001421      BEQ  DTCHKA      ;CHARS. BRANCH IF SAME.
1308 005452 004537 004474 DTCHKX: JSR  %5,OACNV  ;GO TO OCTAL TO ASCII CONVERT.
1309 005456 001556      CRBUF ;SOURCE ADDR.
1310 005460 016103      AWAS  ;DESTINATION ADDR.
1311 005462 000003      3      ;#OF DIGITS TO CONVERT.
1312 005464 004537 004474      JSR  %5,OACNV  ;GO TO OCTAL TO ASCII CONVERT.
1313 005470 001560      CRBUFA ;SOURCE ADDR.
1314 005472 016072      AASB  ;DESTINATION ADDR.
1315 005474 000003      3      ;#OF DIGITS TO CONVERT.
1316 005476 004537 004474      JSR  %5,OACNV
1317 005502 001562      CRBUFB
1318 005504 016116      ARXBUF
1319 005506 000006      6
1320 005510 104015      ERROR1
1321 005512 016060      ERDAT
1322 005514 000002      DTCHKA: RTI
1323
1324      ; ERROR HANDLER
1325 005516 012737 177777 005666 ERR:  MOV  #-1,ERRB  ;SET UP ONE MESSAGE CALL.
1326 005524 012737 000240 005670      MOV  #240,ERRB+2
1327 005532 005037 005704      CLR  ERRE
1328 005536 000413      BR   ERRA
1329 005540 011637 005666      ERR1: MOV  3%6,ERRB  ;DEVELOP ADDT'L MESSAGE ADDR.
1330 005544 017737 000116 005666      MOV  3ERRB,ERRB ;STORE AT ERRB.
1331 005552 012737 177777 005670      MOV  #-1,ERRB+2
1332 005560 012737 000002 005704      MOV  #2,ERRE
1333 005566 032777 020000 172400 ERRRA: BIT  #BIT13,3SRPTR ;INHIBIT ERROR PRINT?
1334 005574 001036      BNE  ERRC      ;BRANCH TO INHIBIT PRINT.
1335 005576 011637 005702      MOV  3%6,ERRD  ;DEVELOP CALLING ADDR.
1336 005602 162737 000002 005702      SUB  #2,ERRD
1337 005610 013737 001456 001460      MOV  RTNNO,TNNO
1338 005616 042737 100000 001460      BIC  #BIT15,TNNO
1339 005624 004537 004474      JSR  %5,OACNV  ;GO TO OCTAL TO ASCII CONVERT.
1340 005630 005702      ERRO ;SOURCE ADDR.
1341 005632 015247      APC  ;DESTINATION ADDR.
1342 005634 000006      6      ;#OF DIGITS TO CONVERT.
1343 005636 004537 004474      JSR  %5,OACNV  ;GO TO OCTAL TO ASCII CONVERT.
1344 005642 001410      RXCSR ;SOURCE ADDR.
1345 005644 015266      MRXNUM ;DESTINATION ADDR.
1346 005646 000006      6      ;#OF DIGITS TO CONVERT.
1347 005650 004537 004474      JSR  %5,OACNV  ;GO TO OCTAL TO ASCII CONVERT.
1348 005654 001456      RTNNO ;SOURCE ADDR.
1349 005656 015237      ATNUMB ;DESTINATION ADDR.
1350 005660 000003      3      ;#OF DIGITS TO CONVERT.
1351 005662 104001      TYPES ;TYPE:
1352 005664 015235      EMO ;ERROR HEADER,
1353 005666 000000      ERRB: OPEN ;ADDT'L ERROR MESSAGE IF ANY.
1354 005670 177777      -1
1355 005672 104010      ERRC: EHALT ;GO ERR HALT IF DESIRED.
1356 005674 063716 005704      ADD  ERRE,3%6

```

```

1357 005700 000002 RTI ;EXIT.
1358 005702 000000 ERRD: OPEN
1359 005704 000000 ERRE: OPEN
1360
1361 ;ERROR TRAP HANDLER - TYPE TO AND FROM WHERE ERROR TRAP OCCURRED
1362 005706 013737 177776 001470 ERTF: MOV PSW,OLDPS ;SAVE OLD STATUS
1363 005714 012737 000340 177776 MOV #PRTY7,PSW
1364 005722 006237 001470 ASR OLDPS
1365 005726 006237 001470 ASR OLDPS
1366 005732 006237 001470 ASR OLDPS
1367 005736 042737 177740 001470 BIC #177740,OLDPS
1368 005744 013737 001470 001632 MOV OLDPS,TOPC
1369 005752 011637 001634 MOV %R6,FROMPC ;GET FROM PC
1370 005756 004537 004474 ERTPA: JSR %R5,ORCNV
1371 005762 001632 TOPC
1372 005764 017154 MTO
1373 005766 000006 6
1374 005770 004537 004474 JSR %R5,ORCNV
1375 005774 001634 FROMPC
1376 005776 017206 MFROM
1377 006000 000006 6
1378 006002 104000 TYPE
1379 006004 017107 MTERR.
1380 006006 000000 HALT
1381 006010 000137 002044 JMP START
1382
1383 ;MAPVEC - MAP VECTOR OR REPORT ERROR DEPENDING ON FMAP FLAG
1384 006014 011637 001632 MAPVEC: MOV %R6,TOPC
1385 006020 022626 POPSP2
1386 006022 011637 001634 MOV %R6,FROMPC
1387 006026 162737 000004 001632 SUB #4,TOPC
1388 006034 005737 001472 TST FMAP
1389 006040 001746 BEQ ERTPA ;NOT MAPPING, REPORT ERROR
1390 006042 013737 001632 001424 MOV TOPC,TVTR ;STORE VECTOR
1391 006050 162737 000004 001632 SUB #4,TOPC
1392 006056 013737 001632 001420 MOV TOPC,RXVTR
1393 006064 005037 001472 CLR FMAP
1394 006070 000002 RTI
1395
1396 ;FORMAD-FORM DEVICE AT ADDRESSES
1397 006072 010137 001410 FORMAD: MOV %R1,RXCSR
1398 006076 062701 000002 ADD #2,%R1
1399 006102 010137 001412 MOV %R1,RXBUF
1400 006106 062701 000002 ADD #2,%R1
1401 006112 010137 001414 MOV %R1,TXCSR
1402 006116 062701 000002 ADD #2,%R1
1403 006122 010137 001416 MOV %R1,TXBUF
1404 006126 013737 001614 001616 MOV LINENO,TEMP ;GET PRIORITY
1405 006134 006337 001616 ASL TEMP
1406 006140 062737 001302 001616 ADD #CMASO,TEMP
1407 006146 017737 173444 001620 MOV %TEMP,TEMP1
1408 006154 013737 001620 001402 MOV TEMP1,UMASK
1409 006162 000337 001620 SWAB TEMP1
1410 006166 006337 001620 ASL TEMP1
1411 006172 042737 177437 001620 BIC #177437,TEMP1
1412 006200 013737 001620 001422 MOV TEMP1,RXLVL

```

```

1413 006206 013737 001620 001426      MOV      TEMP1, TXLVL
1414 006214 000207                      RTS      %7
1415
1416                      ;DOTHIS - SELECTABLE TEST DECISION MAKER
1417
1418 006216 032777 001000 171750 DOTHIS: BIT      #BIT9, JSRPTR      ;IS SELECT TEST SWITCH SET
1419 006224 001002                      BNE     GOBACK      ;RETURN TO TEST IF SW SET
1420 006226 000137 002410                      JMP     GTRDYX      ;GO TO NEXT TEST
1421 006232 000207                      GOBACK: RTS      %7
1422
1423 006234 012737 006244 000024 PFAIL:  MOV      #PWRUP, 24
1424 006242 000000                      HALT
1425 006244 012737 006234 000024 PWRUP:  MOV      #PFAIL, 24
1426 006252 000005                      RESET
1427 006254 012706 001176                      MOV      #SPBOT, %6
1428 006260 104000                      TYPE
1429 006262 017403                      MPWRF
1430 006264 104003                      ERROR
1431 006266 000452                      BR      RESTART
1432
1433                      ;DECIDE IF VECTOR TO BE MAPPED AND MAP
1434 006270 022737 000000 001612 TSTVEC: CMP      #0, FOUNOV      ;NEED VECTOR MAPPING
1435 006276 001045                      BNE     TSTVEX      ;NO, EXIT
1436 006300 004737 003516                      JSR     %7, OVRLAY
1437 006304 005037 001420                      CLR     RXVTR
1438 006310 005037 177776                      CLR     PSW
1439 006314 052737 000001 001472 BIS      #BIT0, FMAP      ;SET MAPPING FLAG
1440 006322 042777 000100 173054 BIC      #BIT6, JTXCSR      ;CAUSE INTERRUPT
1441 006330 052777 000100 173056 BIS      #BIT6, JTXCSR
1442 006336 000240                      NOP
1443 006340 000240                      NOP
1444 006342 005737 001420 TST      RXVTR      ;DID TRAP OCCUR?
1445 006346 001011                      BNE     TSTVA      ;YES, OK
1446 006350 032777 020000 171616 BIT      #BIT13, JSRPTR
1447 006356 001344                      BNE     TSTVEC
1448 006360 104000                      TYPE
1449 006362 017265                      INTER
1450 006364 104003                      ERROR
1451 006366 000137 006270                      JMP     TSTVEC
1452 006372 042777 000100 173014 TSTVA:  BIC      #BIT6, JTXCSR
1453 006400 012737 000340 177776 MOV      #PRTY7, PSW      ;RAISE PRIORITY, RETURN
1454 006406 005237 001612 INC      FOUNOV
1455 006412 000207 TSTVEX: RTS      %7
1456
1457                      ;RESTART ROUTINE
1458 006414 013700 001450 RESTART: MOV     PRGNUM, %0
1459 006420 006300                      ASL     %0
1460 006422 000170 006426 JMP      @RSTART(0)      ;GO RESTART SELECTED PROGRAM
1461
1462 006426 006500 RSTART: PRG0A      ;PROGRAM 0 RESTART ADDRESS
1463 006430 014546 PRG1A      ;PROGRAM 1 RESTART ADDRESS
1464 006432 014614 PRG2A      ;PROGRAM 2 RESTART ADDRESS
1465 006434 014706 PRG3A      ;PROGRAM 3 RESTART ADDRESS
1466 006436 014736 PRG4A      ;PROGRAM 4 RESTART ADDRESS
1467 006440 005110 INCRPG
1468 006442 005110 INCRPG

```

```

1469 006444 005110          INCRPG
1470
1471          ;
1472          ;PRGO - INPUT-OUTPUT LOGIC TESTS
1473
1474 006446 012737 006504 001452 PRGO:  MOV    #ATO,KSTART
1475 006454 005737 000042          TST    J#42          ;MONITOR LOAD
1476 006460 001005          BNE    PRGOB          ;YES, START TEST
1477 006462 104000          TYPE           ;TYPE TITLE AND INSTRUCTIONS
1478 006464 015277          POTIT
1479 006466 000000          HALT
1480 006470 004737 005070          JSR    7,SETSR
1481 006474 004537 004114          PRGOB: JSR    5,LINSEL      ;GO GET LINE # FROM USER
1482 006500 000137 002402          PRGOA: JMP    GETRDY      ;GET STARTED.
1483          X=-1
1484 006504          TSTA   1000.,AA,CD
1485 006504          TSTAA  AA,1000.,\X+1+CD,\X+2,\X+1
1486
1487 006504 100000          ATO:   100000          ;TEST NUMBER *
1488 006506 006536          AT1   ;ADDRESS OF NEXT TEST *
1489 006510 001750          1000. ;ITERATION COUNT *
1490 006512 006514          AAA   ;SCOPE ENTRY POINT *
1491          X=X+1
1492
1493          ;*****
1494          ;TEST ABILITY TO REFERENCE RECEIVER CSR WITHOUT TRAPPING
1495 006514 012737 006530 000004 AAA:   MOV    #AAE,MACHER ;SET UP MACHINE ERROR TRAP.
1496 006522 005077 172662          CLR    RXCSCR          ;REFERENCE RXCSR
1497 006526 104012          AAB:   SCOPE           ;OK IF NO TRAP. SCOPE
1498 006530 022626          AAE:   POPSP2
1499 006532 104003          ERROR          ;TRAPPED WHEN REFERENCING RXCSR.
1500 006534 000774          BR     AAB
1501          TSTA   1000.,AB,CD
1502 006536          TSTAA  AB,1000.,\X+1+CD,\X+2,\X+1
1503
1504          ;*****
1505 006536 100001          AT1:   100001          ;TEST NUMBER *
1506 006540 006576          AT2   ;ADDRESS OF NEXT TEST *
1507 006542 001750          1000. ;ITERATION COUNT *
1508 006544 006546          ABA   ;SCOPE ENTRY POINT *
1509          X=X+1
1510
1511          ;*****
1512          ;TEST ABILITY TO REFERENCE RECEIVER BUFFER WITHOUT TRAPPING
1513 006546 012737 006570 000004 ABA:   MOV    #ABE,MACHER ;SET UP MACHINE ERROR TRAP.
1514 006554 005737 002016          TST    J#XORFLG
1515 006560 100402          BMI    ABB
1516 006562 005777 172624          TST    RRBXBUF        ;REFERENCE RXBUF
1517 006566 104012          ABB:   SCOPE           ;OK IF NO TRAP SCOPE
1518 006570 022626          ABE:   POPSP2
1519 006572 104003          ERROR          ;TRAPPED WHEN REFERENCING RXBUF
1520 006574 000774          BR     ABB
1521          TSTA   1000.,AC,CD
1522 006576          TSTAA  AC,1000.,\X+1+CD,\X+2,\X+1
1523
1524          ;*****
1525 006576 100002          AT2:   100002          ;TEST NUMBER *
1526 006600 006630          AT3   ;ADDRESS OF NEXT TEST *
1527 006602 001750          1000. ;ITERATION COUNT *
1528 006604 006606          ACA   ;SCOPE ENTRY POINT *

```

```

1525      000002      X=X+1
1526      *****
1527      :TEST ABILITY TO REFERENCE TRANSMITTER CSR WITHOUT TRAPPING.
1528      006606 012737 006622 000004 ACA:  MOV  #ACE,MACHER ;SET UP MACHINE ERROR TRAP.
1529      006614 005777 172574      TST  @TXCSR ;REFERENCE TXCSR
1530      006620 104012      ACB:  SCOPE ;SCOPE
1531      006622 022626      ACE:  POPSP2
1532      006624 104003      ERROR ;TRAPPED WHEN REFERENCING TXCSR
1533      006626 000774      BR    ACB
1534      006630      TSTA 1000.,AD,CD
1535      006630      TSTAA AD,1000.,\X+1+CD,\X+2,\X+1
1536      *****
1537      006630 100003      AT3: 100003 ;TEST NUMBER
1538      006632 006662      AT4 ;ADDRESS OF NEXT TEST
1539      006634 001750      1000. ;ITERATION COUNT
1540      006636 006640      ADA ;SCOPE ENTRY POINT
1541      000003      X=X+1
1542      *****
1543      :TEST ABILITY TO REFERENCE TRANSMITTER BUFFER WITHOUT TRAPPING
1544      006640 012737 006654 000004 ADA:  MOV  #ADE,MACHER ;SET UP MACHINE ERROR TRAP.
1545      006646 005777 172544      TST  @TXBUF ;REFERENCE TX BUF.
1546      006652 104012      ADB:  SCOPE ;SCOPE
1547      006654 022626      ADE:  POPSP2
1548      006656 104003      ERROR ;TRAPPED WHEN REFERENCING TXBUF
1549      006660 000774      BR    ADB
1550      006662      TSTA 10.,AE,CD
1551      006662      TSTAA AE,10.,\X+1+CD,\X+2,\X+1
1552      *****
1553      006662 100004      AT4: 100004 ;TEST NUMBER
1554      006664 006762      AT5 ;ADDRESS OF NEXT TEST
1555      006666 000012      10. ;ITERATION COUNT
1556      006670 006672      AEA ;SCOPE ENTRY POINT
1557      000004      X=X+1
1558      *****
1559      :TEST THAT TXCSR BIT 0 (BREAK) CAN BE SET AND CLEARED
1560      :AND THAT RESET CLEARS IT
1561      006672 032777 000001 172514 AEA:  BIT  #BIT0,@TXCSR ;SEE IF BIT IS CLEAR
1562      006700 001402      BEQ  AEB ;BR IF CLEAR
1563      006702 104003      ERROR ;RESET DID NOT CLEAR IT
1564      006704 000421      BR    AED
1565      006706 052777 000001 172500 AEB:  BIS  #BIT0,@TXCSR ;SET TXCSR BIT 0
1566      006714 032777 000001 172472      BIT  #BIT0,@TXCSR ;DID IT SET
1567      006722 001002      BNE  AEC ;YES, GO ON
1568      006724 104003      ERROR ;TXCSR BIT0 FAILED TO SET
1569      006726 000410      BR    AED
1570      006730 042777 000001 172456 AEC:  BIC  #BIT0,@TXCSR ;CLEAR TXCSR BIT 0
1571      006736 032777 000001 172450      BIT  #BIT0,@TXCSR ;DID IT CLEAR
1572      006744 001401      BEQ  AED
1573      006746 104003      ERROR ;TXCSR BIT 0 DID NOT CLEAR
1574      006750 052777 000001 172436 AED:  BIS  #BIT0,@TXCSR ;ISSUE RESET TO CLEAR
1575      006756 104011      SRESET
1576      006760 104012      SCOPE
1577      006762      TSTA 10.,AG,CD
1578      006762      TSTAA AG,10.,\X+1+CD,\X+2,\X+1
1579      *****
1580      006762 100005      AT5: 100005 ;TEST NUMBER

```

F03

```

1581 006764 007062          AT6          ;ADDRESS OF NEXT TEST          *
1582 006766 000012          10.          ;ITERATION COUNT              *
1583 006770 006772          AGA          ;SCOPE ENTRY POINT          *
1584          000005          X=X+1          ;
1585          ;*****
1586          ;TEST THAT TXCSR BIT2 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
1587 006772 032777 000004 172414 AGA:  BIT      #BIT2,@TXCSR ;SEE IF TXCSR BIT2 IS CLEAR.
1588 007000 001402          BEQ      AGB          ;BRANCH IF BIT IS CLEAR.
1589 007002 104003          ERROR          ;RESET DID NOT CLEAR TXCSR BIT2
1590 007004 000421          BR       AGD          ;
1591 007006 052777 000004 172400 AGB:  BIS      #BIT2,@TXCSR ;SET TXCSR BIT2.
1592 007014 032777 000004 172372 AGB:  BIT      #BIT2,@TXCSR ;SEE IF BIT IS SET.
1593 007022 001002          BNE      AGC          ;BRANCH IF BIT IS SET.
1594 007024 104003          ERROR          ;TXCSR BIT2 FAILED TO SET.
1595 007026 000410          BR       AGD          ;
1596 007030 042777 000004 172356 AGC:  BIC      #BIT2,@TXCSR ;CLEAR TXCSR BIT2
1597 007036 032777 000004 172350 AGC:  BIT      #BIT2,@TXCSR ;SEE IF BIT IS CLEAR.
1598 007044 001401          BEQ      AGD          ;
1599 007046 104003          ERROR          ;TXCSR BIT2 FAILED TO CLEAR.
1600 007050 052777 000004 172336 AGD:  BIS      #BIT2,@TXCSR ;SET TXCSR BIT2.
1601 007056 104011          SRESET          ;ISSUE RESET TO CLEAR BIT.
1602 007060 104012          SCOPE          ;SCOPE
1603 007062          TSTA     10. AJ,CD
1604 007062          TSTAA   AJ,10.,\X+1+CD,\X+2,\X+1
1605          ;*****
1606 007062 100006          AT6:  100006          ;TEST NUMBER                    *
1607 007064 007170          AT7          ;ADDRESS OF NEXT TEST          *
1608 007066 000012          10.          ;ITERATION COUNT              *
1609 007070 007072          AJA          ;SCOPE ENTRY POINT          *
1610          000006          X=X+1          ;
1611          ;*****
1612          ;TEST THAT TXCSR BIT6 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
1613 007072 012737 000340 177776 AJA:  MOV      #PRTY7,PSW ;SET PRIORITY 7.
1614 007100 032777 000100 172306 AJA:  BIT      #BIT6,@TXCSR ;SEE IF TXCSR BIT6 IS CLEAR.
1615 007106 001402          BEQ      AJB          ;BRANCH IF BIT IS CLEAR.
1616 007110 104003          ERROR          ;RESET DID NOT CLEAR TXCSR BIT6
1617 007112 000421          BR       AJD          ;
1618 007114 052777 000100 172272 AJB:  BIS      #BIT6,@TXCSR ;SET TXCSR BIT6.
1619 007122 032777 000100 172264 AJB:  BIT      #BIT6,@TXCSR ;SEE IF BIT IS SET.
1620 007130 001002          BNE      AJC          ;BRANCH IF BIT IS SET.
1621 007132 104003          ERROR          ;TXCSR BIT6 FAILED TO SET.
1622 007134 000410          BR       AJD          ;
1623 007136 042777 000100 172250 AJC:  BIC      #BIT6,@TXCSR ;CLEAR TXCSR BIT6
1624 007144 032777 000100 172242 AJC:  BIT      #BIT6,@TXCSR ;SEE IF BIT IS CLEAR.
1625 007152 001401          BEQ      AJD          ;
1626 007154 104003          ERROR          ;TXCSR BIT6 FAILED TO CLEAR.
1627 007156 052777 000100 172230 AJD:  BIS      #BIT6,@TXCSR ;SET TXCSR BIT6.
1628 007164 104011          SRESET          ;ISSUE RESET TO CLEAR BIT.
1629 007166 104012          SCOPE          ;SCOPE
1630 007170          TSTA     100. AK,CD
1631 007170          TSTAA   AK,100.,\X+1+CD,\X+2,\X+1
1632          ;*****
1633 007170 100007          AT7:  100007          ;TEST NUMBER                    *
1634 007172 007214          AT10         ;ADDRESS OF NEXT TEST          *
1635 007174 000144          100.         ;ITERATION COUNT              *
1636 007176 007200          AKA          ;SCOPE ENTRY POINT          *

```



```

1637          000007          X=X+1
1638          ;*****
1639          ;TEST THAT TXCSR BIT 7 (READY BIT) IS SET UPON ENTERING ROUTINE AND
1640          ;THAT IT CAN BE READ RELIABLY.
1641 007200 105777 172210 AKA: TSTB  TXCSR          ;SEE IF TXCSR BIT 7 IS SET.
1642 007204 100402          BMI      AKB          ;BRANCH IF SET.
1643 007206 104003          ERROR          ;TXCSR BIT 7 NOT SET.
1644 007210 104011          SRESET          ;ISSUE RESET TO CLEAR BIT IF ERROR
1645 007212 104012          AKB: SCOPE          ;SCOPE
1646 007214          TSTA  100.,AL,0
1647 007214          TSTAA AL,100.,\X+1+0,\X+2,\X+1
1648          ;*****
1649 007214 000010 AT10: 10          ;TEST NUMBER
1650 007216 007276          AT11          ;ADDRESS OF NEXT TEST
1651 007220 000144          100.          ;ITERATION COUNT
1652 007222 007224          ALA          ;SCOPE ENTRY POINT
1653          000010          X=X+1
1654          ;*****
1655          ;TEST THAT RXCSR BIT 1 CAN BE SET + CLEARED
1656 007224 042777 000002 172156 ALA: BIC  #BIT1,DRXCSR
1657 007232 052777 000002 172150          BIS  #BIT1,DRXCSR          ;SET RXCSR BIT1
1658 007240 032777 000002 172142          BIT  #BIT1,DRXCSR          ;SEE IF BIT IS SET
1659 007246 001002          BNE  ALY          ;BRANCH IF SET
1660 007250 104003          ERROR          ;RXCSR BIT 1 FAILED TO SET
1661 007252 000410          BR   ALZ
1662 007254 042777 000002 172126 ALY: BIC  #BIT1,DRXCSR          ;CLEAR RXCSR BIT 1
1663 007262 032777 000002 172120          BIT  #BIT1,DRXCSR          ;SEE IF BIT IS CLEAR
1664 007270 001401          BEQ  ALZ
1665 007272 104003          ERROR          ;RXCSR BIT 1 FAILED TO CLEAR
1666 007274 104012          ALZ: SCOPE          ;SCOPE
1667 007276          TSTA  10.,AP,0
1668 007276          TSTAA AP,10.,\X+1+0,\X+2,\X+1
1669          ;*****
1670 007276 000011 AT11: 11          ;TEST NUMBER
1671 007300 007376          AT12          ;ADDRESS OF NEXT TEST
1672 007302 000012          10.          ;ITERATION COUNT
1673 007304 007306          APA          ;SCOPE ENTRY POINT
1674          000011          X=X+1
1675          ;*****
1676          ;TEST THAT RXCSR BIT2 IS CLEAR AND CAN BE READ RELIABLY.
1677 007306 032777 000004 172074 APA: BIT  #BIT2,DRXCSR          ;SEE IF RXCSR BIT2 IS CLEAR.
1678 007314 001402          BEQ  APB          ;BRANCH IF BIT IS CLEAR.
1679 007316 104003          ERROR          ;RXCSR BIT2 IS NOT CLEAR.
1680 007320 000421          BR   APD
1681 007322 052777 000004 172060 APB: BIS  #BIT2,DRXCSR          ;SET RXCSR BIT2
1682 007330 032777 000004 172052          BIT  #BIT2,DRXCSR          ;SEE IF BIT IS SET
1683 007336 001002          BNE  APCX          ;BRANCH IF SET
1684 007340 104003          ERROR          ;RXCSR BIT2 FAILED TO SET
1685 007342 000410          BR   APD
1686 007344 042777 000004 172036 APCX: BIC  #BIT2,DRXCSR          ;CLEAR RXCSR BIT2
1687 007352 032777 000004 172030          BIT  #BIT2,DRXCSR          ;SEE IF BIT IS CLEAR
1688 007360 001401          BEQ  APD
1689 007362 104003          ERROR          ;RXCSR BIT2 FAILED TO CLEAR
1690 007364 052777 000004 172016 APD: BIS  #BIT2,DRXCSR          ;SET BIT
1691 007372 104011          SRESET          ;ISSUE RESET TO CLEAR BIT
1692 007374 104012          SCOPE

```

```

1693 007376          TSTA  10.,AQ,0
1694 007376          TSTAA AQ,10.,\X+1+0,\X+2,\X+1
1695 ;*****
1696 007376 000012    AT12:  12          ;TEST NUMBER *
1697 007400 007476          AT13          ;ADDRESS OF NEXT TEST *
1698 007402 000012          10.          ;ITERATION COUNT *
1699 007404 007406          AQA          ;SCOPE ENTRY POINT *
1700          000012          X=X+1          ; *
1701 ;*****
1702 ;TEST THAT RXCSR BIT3 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
1703 007406 032777 000010 171774 AQA:  BIT  #BIT3,QRXCSR ;SEE IF RXCSR BIT3 IS CLEAR.
1704 007414 001402          BEQ  AQB          ;BRANCH IF BIT IS CLEAR.
1705 007416 104003          ERROR          ;RESET DID NOT CLEAR RXCSR BIT3
1706 007420 000421          BR  AQB          ;
1707 007422 052777 000010 171760 AQB:  BIS  #BIT3,QRXCSR ;SET RXCSR BIT3.
1708 007430 032777 000010 171752          BIT  #BIT3,QRXCSR ;SEE IF BIT IS SET.
1709 007436 001002          BNE  AQC          ;BRANCH IF BIT IS SET.
1710 007440 104003          ERROR          ;RXCSR BIT3 FAILED TO SET..
1711 007442 000410          BR  AQC          ;
1712 007444 042777 000010 171736 AQC:  BIC  #BIT3,QRXCSR ;CLEAR RXCSR BIT3
1713 007452 032777 000010 171730          BIT  #BIT3,QRXCSR ;SEE IF BIT IS CLEAR.
1714 007460 001401          BEQ  AQB          ;
1715 007462 104003          ERROR          ;RXCSR BIT3 FAILED TO CLEAR.
1716 007464 052777 000010 171716 AQB:  BIS  #BIT3,QRXCSR ;SET RXCSR BIT3.
1717 007472 104011          SRESET          ;ISSUE RESET TO CLEAR BIT.
1718 007474 104012          SCOPE          ;SCOPE
1719 007476          TSTA  10.,AR,0
1720 007476          TSTAA AR,10.,\X+1+0,\X+2,\X+1
1721 ;*****
1722 007476 000013    AT13:  13          ;TEST NUMBER *
1723 007500 007604          AT14          ;ADDRESS OF NEXT TEST *
1724 007502 000012          10.          ;ITERATION COUNT *
1725 007504 007506          ARA          ;SCOPE ENTRY POINT *
1726          000013          X=X+1          ; *
1727 ;*****
1728 ;TEST THAT RXCSR BITS CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
1729 007506 012737 000340 177776 ARA:  MOV  #PRTY7,PSW ;PRTY7 TO INHIBIT ANY INT
1730 007514 032777 000040 171666          BIT  #BITS,QRXCSR ;SEE IF RXCSR BITS IS CLEAR.
1731 007522 001402          BEQ  ARB          ;BRANCH IF BIT IS CLEAR.
1732 007524 104003          ERROR          ;RESET DID NOT CLEAR RXCSR BITS
1733 007526 000421          BR  ARB          ;
1734 007530 052777 000040 171652 ARB:  BIS  #BITS,QRXCSR ;SET RXCSR BITS.
1735 007536 032777 000040 171644          BIT  #BITS,QRXCSR ;SEE IF BIT IS SET.
1736 007544 001002          BNE  ARC          ;BRANCH IF BIT IS SET.
1737 007546 104003          ERROR          ;RXCSR BITS FAILED TO SET.
1738 007550 000410          BR  ARC          ;
1739 007552 042777 000040 171630 ARC:  BIC  #BITS,QRXCSR ;CLEAR RXCSR BITS
1740 007560 032777 000040 171622          BIT  #BITS,QRXCSR ;SEE IF BIT IS CLEAR.
1741 007566 001401          BEQ  ARD          ;
1742 007570 104003          ERROR          ;RXCSR BIT4 FAILED TO CLEAR.
1743 007572 052777 000040 171610 ARD:  BIS  #BITS,QRXCSR ;SET RXCSR BITS.
1744 007600 104011          SRESET          ;ISSUE RESET TO CLEAR BIT.
1745 007602 104012          SCOPE          ;SCOPE
1746 007604          TSTA  10.,AS,CD
1747 007604          TSTAA AS,10.,\X+1+CD,\X+2,\X+1
1748 ;*****

```

```

1749 007604 100014 AT14: 100014 ;TEST NUMBER *
1750 007606 007712 AT15 ;ADDRESS OF NEXT TEST *
1751 007610 000012 10. ;ITERATION COUNT *
1752 007612 007614 ASA ;SCOPE ENTRY POINT *
1753 000014 X=X+1 *
1754 ;*****
1755 ;TEST THAT RXCSR BIT6 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
1756 007614 012737 000340 177776 ASA: MOV #PTY7,PSW ;SET PRIORITY 7.
1757 007622 032777 000100 171560 BIT #BIT6,@RXCSR ;SEE IF RXCSR BIT6 IS CLEAR.
1758 007630 001402 BEQ ASB ;BRANCH IF BIT IS CLEAR.
1759 007632 104003 ERROR ;RESET DID NOT CLEAR RXCSR BIT6
1760 007634 000421 BR ASD
1761 007636 052777 000100 171544 ASB: BIS #BIT6,@RXCSR ;SET RXCSR BIT6.
1762 007644 032777 000100 171536 BIT #BIT6,@RXCSR ;SEE IF BIT IS SET.
1763 007652 001002 BNE ASC ;BRANCH IF BIT IS SET.
1764 007654 104003 ERROR ;RXCSR BIT6 FAILED TO SET.
1765 007656 000410 BR ASD
1766 007660 042777 000100 171522 ASC: BIC #BIT6,@RXCSR ;CLEAR RXCSR BIT6
1767 007666 032777 000100 171514 BIT #BIT6,@RXCSR ;SEE IF BIT IS CLEAR.
1768 007674 001401 BEQ ASD
1769 007676 104003 ERROR ;RXCSR BIT6 FAILED TO CLEAR.
1770 007700 052777 000100 171502 ASD: BIS #BIT6,@RXCSR ;SET RXCSR BIT6.
1771 007706 104011 SRESET ;ISSUE RESET TO CLEAR BIT.
1772 007710 104012 SCOPE ;SCOPE
1773 007712 TSTA 100.,AT,0
1774 007712 TSTAA AT,100.,\X+1+0,\X+2,\X+1
1775 ;*****
1776 007712 000015 AT15: 15 ;TEST NUMBER *
1777 007714 007740 AT16 ;ADDRESS OF NEXT TEST *
1778 007716 000144 100. ;ITERATION COUNT *
1779 007720 007722 ATA ;SCOPE ENTRY POINT *
1780 000015 X=X+1 *
1781 ;*****
1782 ;TEST THAT RXCSR BIT7 IS CLEAR AND CAN BE READ RELIABLY.
1783 007722 032777 000200 171460 ATA: BIT #BIT7,@RXCSR ;SEE IF RXCSR BIT7 IS CLEAR.
1784 007730 001402 BEQ ATB ;BRANCH IF BIT IS CLEAR.
1785 007732 104003 ERROR ;RXCSR BIT7 IS NOT CLEAR.
1786 007734 104011 SRESET ;RESET IF ERROR
1787 007736 104012 ATB: SCOPE ;SCOPE
1788 007740 TSTA 100.,AX,0
1789 007740 TSTAA AX,100.,\X+1+0,\X+2,\X+1
1790 ;*****
1791 007740 000016 AT16: 16 ;TEST NUMBER *
1792 007742 007766 AT17 ;ADDRESS OF NEXT TEST *
1793 007744 000144 100. ;ITERATION COUNT *
1794 007746 007750 AXA ;SCOPE ENTRY POINT *
1795 000016 X=X+1 *
1796 ;*****
1797 ;TEST THAT RXCSR BIT10 IS CLEAR AND CAN BE READ RELIABLY.
1798 007750 032777 002000 171432 AXA: BIT #BIT10,@RXCSR ;SEE IF RXCSR BIT10 IS CLEAR.
1799 007756 001402 BEQ AXB ;BRANCH IF BIT IS CLEAR.
1800 007760 104003 ERROR ;RXCSR BIT10 IS NOT CLEAR.
1801 007762 104011 SRESET ;RESET BIT IF ERROR
1802 007764 104012 AXB: SCOPE ;SCOPE
1803 007766 TSTA 100.,AY,CD
1804 007766 TSTAA AY,100.,\X+1+CD,\X+2,\X+1

```

```

1805 ;*****
1806 007766 100017 AT17: 100017 ;TEST NUMBER *
1807 007770 010014 AT20 ;ADDRESS OF NEXT TEST *
1808 007772 000144 100. ;ITERATION COUNT *
1809 007774 007776 AYA ;SCOPE ENTRY POINT *
1810 000017 X=X+1 ;
1811 ;*****
1812 ;TEST THAT RXCSR BIT11 IS CLEAR AND CAN BE READ RELIABLY.
1813 007776 032777 004000 171404 AYA: BIT #BIT11,ARXCSR ;SEE IF RXCSR BIT11 IS CLEAR.
1814 010004 001402 BEQ AYB ;BRANCH IF BIT IS CLEAR.
1815 010006 104003 ERROR ;RXCSR BIT11 IS NOT CLEAR.
1816 010010 104011 SRESET ;RESET BIT IF ERROR
1817 010012 104012 AYB: SCOPE ;SCOPE
1818 010014 TSTA 100.,AZ,CD
1819 010014 TSTAA AZ,100.,\X+1+CD,\X+2,\X+1
1820 ;*****
1821 010014 100020 AT20: 100020 ;TEST NUMBER *
1822 010016 010042 AT21 ;ADDRESS OF NEXT TEST *
1823 010020 000144 100. ;ITERATION COUNT *
1824 010022 010024 AZA ;SCOPE ENTRY POINT *
1825 000020 X=X+1 ;
1826 ;*****
1827 ;TEST THAT RXCSR BIT14 IS CLEAR AND CAN BE READ RELIABLY.
1828 010024 032777 040000 171356 AZA: BIT #BIT14,ARXCSR ;SEE IF RXCSR BIT14 IS CLEAR.
1829 010032 001402 BEQ AZB ;BRANCH IF BIT IS CLEAR.
1830 010034 104003 ERROR ;RXCSR BIT14 IS NOT CLEAR.
1831 010036 104011 SRESET ;RESET BIT IF ERROR
1832 010040 104012 AZB: SCOPE ;SCOPE
1833 010042 TSTA 100.,AAA,CD
1834 010042 TSTAA AAA,100.,\X+1+CD,\X+2,\X+1
1835 ;*****
1836 010042 100021 AT21: 100021 ;TEST NUMBER *
1837 010044 010070 AT22 ;ADDRESS OF NEXT TEST *
1838 010046 000144 100. ;ITERATION COUNT *
1839 010050 010052 AAAA ;SCOPE ENTRY POINT *
1840 000021 X=X+1 ;
1841 ;*****
1842 ;TEST THAT RXCSR BIT15 IS CLEAR AND CAN BE READ RELIABLY.
1843 010052 032777 100000 171330 AAAA: BIT #BIT15,ARXCSR ;SEE IF RXCSR BIT15 IS CLEAR.
1844 010060 001402 BEQ AAAB ;BRANCH IF BIT IS CLEAR.
1845 010062 104003 ERROR ;RXCSR BIT15 IS NOT CLEAR.
1846 010064 104011 SRESET ;RESET BIT IF ERROR
1847 010066 104012 AAAB: SCOPE ;SCOPE
1848 ;
1849 ;ALL PREVIOUS TESTS MUST HAVE BEEN RUN SUCCESSFULLY PRIOR
1850 ;TO RUNNING THE FOLLOWING TESTS. ALSO, THE JUMPER CONNECTOR
1851 ;MUST BE INSERTED IN THE DL11-E CABLE IN PLACE OF THE MODEM. COMMENTS
1852 ;REFER TO OPERATION WITH JUMPER INSERTED.
1853 ;
1854 010070 TSTA 100.,AFB,0
1855 010070 TSTAA AFB,100.,\X+1+0,\X+2,\X+1
1856 ;*****
1857 010070 000022 AT22: 22 ;TEST NUMBER *
1858 010072 010154 AT23 ;ADDRESS OF NEXT TEST *
1859 010074 000144 100. ;ITERATION COUNT *
1860 010076 010100 AFBA ;SCOPE ENTRY POINT *

```

```

1861          000022          X=X+1
1862          ;*****
1863          ;TEST THAT CARRIER DETECT SETS AND CLEARS WHEN DATA TERMINAL
1864          ;READY SETS AND CLEARS.
1865 010100 052777 000002 171302 AFBA: BIS #BIT1,DRXCSR ;SET DATA TERMINAL READY
1866 010106 004737 011762          JSR %7,TIME ;DELAY
1867 010112 032777 010000 171270 BIT #BIT12,DRXCSR ;TEST CARRIER DETECT
1868 010120 001002          BNE AFBB ;SHOULD BE SET
1869 010122 104003          ERROR ;WASN'T
1870 010124 000412          BR AFBC
1871 010126 042777 000002 171254 AFBB: BIC #BIT1,DRXCSR ;CLEAR DATA TERMINAL READY
1872 010134 004737 011762          JSR %7,TIME ;DELAY
1873 010140 032777 010000 171242 BIT #BIT12,DRXCSR ;TEST CARRIER DETECT
1874 010146 001401          BEQ AFBC
1875 010150 104003          ERROR ;WAS SET, ERROR
1876 010152 104012          AFBC: SCOPE
1877 010154          TSTA 100.,AGB,0
1878 010154          TSTAA AGB,100.,\X+1+0,\X+2,\X+1
1879          ;*****
1880 010154 000023          AT23: 23 ;TEST NUMBER
1881 010156 010326          AT24 ;ADDRESS OF NEXT TEST
1882 010160 000144          100. ;ITERATION COUNT
1883 010162 010164          AGBA ;SCOPE ENTRY POINT
1884          X=X+1
1885          ;*****
1886          ;TEST THAT MODEM INTERRUPT (BIT 15) SETS WHEN CARRIER DETECT
1887          ;CHANGES STATE, AND IS CLEARED WHEN RXCSR IS READ.
1888 010164 042777 000002 171216 AGBA: BIC #BIT1,DRXCSR ;CLEAR DATA TERMINAL READY
1889 010172 004737 011762          JSR %7,TIME ;DELAY
1890 010176 017737 171206 001606 MOV DRXCSR,RXCSRT ;READ RXCSR
1891 010204 032777 100000 171176 BIT #BIT15,DRXCSR ;TEST MODEM INTERRUPT
1892 010212 001402          BEQ AGBB ;WAS CLEAR GO TO AGBB
1893 010214 104003          ERROR ;WASN'T CLEAR
1894 010216 000442          BR AGBE ;GO TO SCOPE
1895 010220 052777 000002 171162 AGBB: BIS #BIT1,DRXCSR ;SETTING DATA TERMINAL READY
1896          ;CAUSES CARRIER DETECT TO SET.
1897          ;WHICH CAUSES MODEM INTERRUPT TO SET
1898          ;DELAY
1899 010226 004737 011762          JSR %7,TIME
1900 010232 017737 171152 001606 MOV DRXCSR,RXCSRT ;MOVE RXCSR TO TEMPORARY LOCATION
1901 010240 032737 100000 001606 BIT #BIT15,RXCSRT ;TEST MODEM INTERRUPT
1902 010246 001002          BNE AGBC ;SHOULD BE SET GO TO AGBC
1903 010250 104003          ERROR ;WAS CLEAR
1904 010252 000424          BR AGBE ;GO TO SCOPE
1905 010254 032777 100000 171126 AGBC: BIT #BIT15,DRXCSR ;MODEM INTERRUPT BIT SHOULD
1906          ;HAVE BEEN CLEARED
1907          ;IT WAS GO TO AGBD
1908          ;IT WASN'T
1909 010262 001402          BEQ AGBD ;GO TO SCOPE
1910 010264 104003          ERROR ;CLEARING DATA TERMINAL READY
1911 010266 000416          BR AGBE ;CAUSES CARRIER DETECT TO CLEAR
1912 010270 042777 000002 171112 AGBD: BIC #BIT1,DRXCSR ;BUT MODEM INTERRUPT WILL SET
1913          ;DELAY
1914          ;MOV RXCSR TO TEMPORARY LOCATION
1915          ;TEST MODEM INTERRUPT
1916 010276 004737 011762          JSR %7,TIME
1917 010302 017737 171102 001606 MOV DRXCSR,RXCSRT
1918 010310 032737 100000 001606 BIT #BIT15,RXCSRT
1919 010316 001002          BNE AGBE ;SHOULD BE SET
1920 010320 104003          ERROR ;IT WASN'T

```

```

1917 010322 000400
1918 010324 104012
1919 010326
1920 010326
1921
1922 010326 000024
1923 010330 010440
1924 010332 000144
1925 010334 010336
1926 010334 000024
1927
1928
1929
1930 010336 042777 000002 171044
1931 010344 004737 011762
1932 010350 032777 020000 171032
1933 010356 001402
1934 010360 104003
1935 010362 000425
1936 010364 052777 000002 171016
1937 010372 004737 011762
1938 010376 032777 020000 171004
1939 010404 001002
1940 010406 104003
1941 010410 000412
1942 010412 042777 000002 170770
1943 010420 004737 011762
1944 010424 032777 020000 170756
1945 010432 001401
1946 010434 104003
1947 010436 104012
1948 010440
1949 010440
1950
1951 010440 000025
1952 010442 010534
1953 010444 000144
1954 010446 010450
1955 010446 000025
1956
1957
1958
1959 010450 042777 000004 170732
1960 010456 004737 011762
1961 010462 052777 000004 170720
1962 010470 004737 011762
1963 010474 032777 040000 170706
1964 010502 001001
1965 010504 104003
1966 010506 042777 000004 170674
1967 010514 004737 011762
1968 010520 032777 040000 170662
1969 010526 001401
1970 010530 104003
1971 010532 104012
1972 010534

AGBE: BR AGBE
SCOPE ;SCOPE
TSTA 100.,AJB,0
TSTAA AJB,100.,\X+1+0,\X+2,\X+1
;*****
AT24: 24 ;TEST NUMBER *
AT25 ;ADDRESS OF NEXT TEST *
100. ;ITERATION COUNT *
AJBA ;SCOPE ENTRY POINT *
X=X+1
;*****
;TEST THAT CLEAR TO SEND (BIT13) SETS/CLEARs WHEN DATA TERMINAL
;READY SETS/CLEARs.
AJBA: BIC #BIT1,ARXCSR ;CLEAR DATA TERMINAL READY
JSR %7,TIME ;DELAY
BIT #BIT13,ARXCSR ;TEST CLEAR TO SEND
BEQ AJBB ;CLEAR TO SEND SHOULD BE CLEAR
ERROR
BR AJBD
AJBB: BIS #BIT1,ARXCSR ;SET DATA TERMINAL READY
JSR %7,TIME ;DELAY
BIT #BIT13,ARXCSR ;TEST CLEAR TO SEND
BNE AJBC ;BRANCH IF SET
ERROR ;CLEAR TO SEND SHOULD BE SET
BR AJBD
AJBC: BIC #BIT1,ARXCSR ;CLEAR DATA TERMINAL READY
JSR %7,TIME ;DELAY
BIT #BIT13,ARXCSR ;TEST CLEAR TO SEND
BEQ AJBD ;CLEAR TO SEND SHOULD BE CLEAR
ERROR
AJBD: SCOPE ;SCOPE
TSTA 100.,AKB,0
TSTAA AKB,100.,\X+1+0,\X+2,\X+1
;*****
AT25: 25 ;TEST NUMBER *
AT26 ;ADDRESS OF NEXT TEST *
100. ;ITERATION COUNT *
AKBA ;SCOPE ENTRY POINT *
X=X+1
;*****
;TEST THAT RING (BIT 14 RXCSR) SETS WHEN REQUEST TO
;SEND SETS AND CLEARs AND RESET CLEARs RING
AKBA: BIC #BIT2,ARXCSR ;CLEAR REQUEST TO SEND
JSR %7,TIME ;DELAY
BIS #BIT2,ARXCSR ;SET REQUEST TO SEND
JSR %7,TIME ;DELAY
BIT #BIT14,ARXCSR ;TEST RING
BNE AKBC ;RING SHOULD BE SET
ERROR
AKBC: BIC #BIT2,ARXCSR ;CLEAR REQUEST TO SEND
JSR %7,TIME ;DELAY
BIT #BIT14,ARXCSR ;TEST RING
BEQ .+4 ;SHOULD BE CLEAR
ERROR
SCOPE ;SCOPE
TSTA 100.,AOB,0

```

M03

```

1973 010534          TSTAA  AOB,100.,\X+1+0,\X+2,\X+1
1974          ;*****
1975 010534 000026  AT26:  26          ;TEST NUMBER *
1976 010536 010646          AT27          ;ADDRESS OF NEXT TEST *
1977 010540 000144          100.          ;ITERATION COUNT *
1978 010542 010544          AOB A          ;SCOPE ENTRY POINT *
1979          X=X+1          ;*****
1980          ;TEST THAT MODEM INTERRUPT (BIT 15 RXCSR) SETS WHEN RING SETS.
1981          ;*****
1982 010544 042777 000004 170636 AOB A:  BIC  #BIT2,ARXCSR ;CLEAR REQUEST TO SEND
1983 010552 004737 011762          JSR  %7,TIME ;DELAY
1984 010556 032777 100000 170624          BIT  #BIT15,ARXCSR ;TEST MODEM INTERRUPT BIT
1985 010564 001402          BEQ  AOB B
1986 010566 104003          ERROR
1987 010570 000425          BR   AOB D
1988 010572 052777 000004 170610 AOB B:  BIS  #BIT2,ARXCSR ;SET REQUEST TO SEND
1989 010600 004737 011762          JSR  %7,TIME ;DELAY
1990 010604 032777 100000 170576          BIT  #BIT15,ARXCSR ;TEST MODEM INTERRUPT BIT
1991 010612 001002          BNE  AOB C
1992 010614 104003          ERROR
1993 010616 000412          BR   AOB D
1994 010620 042777 000004 170562 AOB C:  BIC  #BIT2,ARXCSR ;CLEAR REQUEST TO SEND
1995 010626 004737 011762          JSR  %7,TIME ;DELAY
1996 010632 032777 100000 170550          BIT  #BIT15,ARXCSR ;TEST MODEM INTERRUPT BIT
1997 010640 001401          BEQ  AOB D
1998 010642 104003          ERROR
1999 010644 104012          AOB D:  SCOPE          ;SCOPE
2000 010646          TSTA  100.,ALB,0
2001 010646          TSTAA ALB,100.,\X+1+0,\X+2,\X+1
2002          ;*****
2003 010646 000027  AT27:  27          ;TEST NUMBER *
2004 010650 010760          AT30          ;ADDRESS OF NEXT TEST *
2005 010652 000144          100.          ;ITERATION COUNT *
2006 010654 010656          ALBA          ;SCOPE ENTRY POINT *
2007          X=X+1          ;*****
2008          ;TEST THAT SUPERVISORY RECEIVE DATA (BIT 10 RXCSR) SETS/CLEAR
2009          ;WHEN SUPERVISORY XMIT DATA SETS/CLEAR.
2010          ;*****
2011 010656 042777 000010 170524 ALBA:  BIC  #BIT3,ARXCSR ;CLEAR SUPERVISOR XMIT DATA
2012 010664 004737 011762          JSR  %7,TIME ;DELAY
2013 010670 032777 002000 170512          BIT  #BIT10,ARXCSR ;TEST SUPERVISORY RECEIVE DATA.
2014 010676 001402          BEQ  ALB B
2015 010700 104003          ERROR          ;SHOULD HAVE BEEN CLEAR
2016 010702 000425          BR   ALB D
2017 010704 052777 000010 170476 ALBB:  BIS  #BIT3,ARXCSR ;SET SUPERVISORY XMIT DATA
2018 010712 004737 011762          JSR  %7,TIME ;DELAY
2019 010716 032777 002000 170464          BIT  #BIT10,ARXCSR ;TEST SUPERVISORY RECEIVE DATA
2020 010724 001002          BNE  ALB C
2021 010726 104003          ERROR          ;SHOULD HAVE BEEN SET
2022 010730 000412          BR   ALB D
2023 010732 042777 000010 170450 ALBC:  BIC  #BIT3,ARXCSR ;CLEAR SUPERVISORY XMIT DATA
2024 010740 004737 011762          JSR  %7,TIME ;DELAY
2025 010744 032777 002000 170436          BIT  #BIT10,ARXCSR ;TEST SUPERVISORY RECEIVE DATA
2026 010752 001401          BEQ  ALB D
2027 010754 104003          ERROR          ;SHOULD HAVE BEEN CLEAR
2028 010756 104012          ALB D:  SCOPE          ;SCOPE
  
```

```

2029 010760          TSTA 100.,AMB,0
2030 010760          TSTAA AMB,100.,\X+1+0,\X+2,\X+1
2031                ;*****
2032 010760 000030  AT30: 30 ;TEST NUMBER *
2033 010762 011120          AT31 ;ADDRESS OF NEXT TEST *
2034 010764 000144          100. ;ITERATION COUNT *
2035 010766 010770          AMBA ;SCOPE ENTRY POINT *
2036                X=X+1 ;
2037                ;*****
2038                ;TEST THAT SUP REC DATA TRANSISTIONS SET MODEM INTERRUPT
2039 010770 042777 000010 170412 AMBA: BIC #BIT3,@RXCSR ;CLEAR SUP REC
2040 010776 004737 011762          JSR %7,TIME ;DELAY
2041 011002 052777 000010 170400          BIS #BIT3,@RXCSR ;SET SUP REC
2042 011010 004737 011762          JSR %7,TIME ;DELAY
2043 011014 032777 100000 170366          BIT #BIT15,@RXCSR ;TEST MODEM INTERRUPT
2044 011022 001002          BNE AMBB ;MODEM INTERRUPT SHOULD BE SET
2045 011024 104003          ERROR
2046 011026 000433          BR AMBE
2047 011030 032777 100000 170352 AMBB: BIT #BIT15,@RXCSR ;MODEM INTERRUPT SHOULD BE
2048 011036 001402          BEQ AMBC ;CLEARED BY PREVIOUS READ
2049 011040 104003          ERROR
2050 011042 000425          BR AMBE
2051 011044 042777 000010 170336 AMBC: BIC #BIT3,@RXCSR ;1-0 TRANS OF SUP REC DATA
2052 011052 004737 011762          JSR %7,TIME ;DELAY
2053 011056 032777 100000 170324          BIT #BIT15,@RXCSR ;TEST MODEM INTERRUPT
2054 011064 001002          BNE AMBD ;SHOULD BE SET
2055 011066 104003          ERROR
2056 011070 000412          BR AMBE
2057 011072 052777 000010 170310 AMBD: BIS #BIT3,@RXCSR ;0-1 TRANS OF SUP REC DATA
2058 011100 004737 011762          JSR %7,TIME ;DELAY
2059 011104 032777 100000 170276          BIT #BIT15,@RXCSR ;TEST MODEM INTERRUPT
2060 011112 001001          BNE AMBE ;SHOULD BE SET
2061 011114 104003          ERROR
2062 011116 104012          AMBE: SCOPE
2063 011120          TSTA 10.,ABA,CD
2064 011120          TSTAA ABA,10.,\X+1+CD,\X+2,\X+1
2065                ;*****
2066 011120 100031  AT31: 100031 ;TEST NUMBER *
2067 011122 011224          AT32 ;ADDRESS OF NEXT TEST *
2068 011124 000012          10. ;ITERATION COUNT *
2069 011126 011130          ABAA ;SCOPE ENTRY POINT *
2070                X=X+1 ;
2071                ;*****
2072                ;TEST THAT RESET CLEARS ALL TXCSR BITS, AND SETS BIT 7 (READY)
2073 011130 012737 000340 177776 ABAA: MOV #PRTY7,PSW ;SET PRIORITY 7.
2074 011136 012777 177777 170250          MOV #-1,@TXCSR ;SET ALL POSSIBLE BITS IN TXCSR
2075 011144 104011          SRESET ;ISSUE RESET TO CLEAR BITS
2076 011146 022777 000200 170240          CMP #BIT7,@TXCSR ;SEE IF ONLY BIT 7 IS SET.
2077 011154 001422          BEQ ABAB ;BRANCH IF ONLY BIT 7 IS SET
2078 011156 017737 170232 001604          MOV @TXCSR,TXCST ;SAVE CONTENTS OF TXCSR
2079 011164 012737 000200 001616          MOV #BIT7,TEMP ;MOVE EXPECTED TXCSR TO TEMP.
2080 011172 004537 004474          JSR %5,OACNV ;GO TO OCTAL TO ASCII CONVERT.
2081 011176 001616          TEMP ;SOURCE ADDR.
2082 011200 015451          ATXSB ;DESTINATION ADDR.
2083 011202 000006          6 ;#OF DIGITS TO CONVERT.
2084 011204 004537 004474          JSR %5,OACNV ;GO TO OCTAL TO ASCII CONVERT.

```



```

01141 011410 104020          TIMETX          :TIME OUT TX DONE
01142 011412 104003          ERROR           :ERROR, DONE NOT SETTING
01143 011414 005077 167776    CLR            JTXBUF      :LOAD TX BUF
01144 011420 105777 167770    TSTB          JTXCSR      :TEST TXCSR BIT 7 (READY BIT)
01145 011424 100002          BPL            ADAB        :BRANCH IF BIT NOT SET.
01146 011426 104003          ERROR           :ERROR. LOADING TXBUF FAILED TO CLEAR READY.
01147 011430 000407          BR              ADAC
01148 011432 104020          ADAB: TIMETX          :WAIT FOR DONE
01149 011434 104003          ERROR           :DONE NEVER SET
01150 011436 032777 000200 167750 BIT            #BIT7,JTXCSR
01151 011444 001001          BNE            .+4
01152 011446 104003          ERROR           :READY DID NOT SET
01153 011450 104011          ADAC: SRESET
01154 011452 104012          SCOPE          :SCOPE.
01155 011454          TSTA          1,AIA,CD
01156 011454          TSTAA         AIA,1.,\X+1+CD,\X+2,\X+1
:*****
AT34: 100034          :TEST NUMBER *
01158 011456 011776          AT35          :ADDRESS OF NEXT TEST *
01159 011460 000001          1            :ITERATION COUNT *
01160 011462 011464          AIAA          :SCOPE ENTRY POINT *
01161 011462 000034          X=X+1
:*****
:TEST THAT TRANSMIT SPEEDS ARE ARRANGED IN ASCENDING ORDER BY CHECKING THAT TIME
:TO READY BIT (TXCSR BIT 7) DECREASES AS A HIGHER SPEED IS SELECTED.
AIAA: JSR            %7,DOTHIS :TEST IF THIS TEST SELECTED
01164 011464 004737 006216    TYPES
01165 011470 104001          MSETTX
01166 011472 016773          MSETC
01167 011474 017021          MS0
01168 011476 017303          -1
01169 011500 177777          HALT
01170 011502 000000          JSR            %7,AIAS :OUTPUT CHAR AND TIME.
01171 011504 004737 011712    MOV            AIAST,CTR0 :MOVE ELAPSED TIME TO CTR0.
01172 011510 013737 011774 001564 TYPE
01173 011516 104000          MS1
01174 011520 017313          HALT
01175 011522 000000          JSR            %7,AIAS :OUTPUT CHAR AND TIME.
01176 011524 004737 011712    MOV            AIAST,CTR1 :MOVE ELAPSED TIME TO CTR1.
01177 011530 013737 011774 001566 TYPE
01178 011536 104000          MS2
01179 011540 017323          HALT
01180 011542 000000          JSR            %7,AIAS :OUTPUT CHAR AND TIME.
01181 011544 004737 011712    MOV            AIAST,CTR2 :MOVE ELAPSED TIME TO CTR2.
01182 011550 013737 011774 001570 TYPE
01183 011556 104000          MS3
01184 011560 017333          HALT
01185 011562 000000          JSR            %7,AIAS :OUTPUT CHAR AND TIME.
01186 011564 004737 011712    MOV            AIAST,CTR3 :MOVE ELAPSED TIME TO CTR3.
01187 011570 013737 011774 001572 TYPE
01188 011576 104000          MS4
01189 011600 017343          HALT
01190 011602 000000          JSR            %7,AIAS :OUTPUT CHAR AND TIME
01191 011604 004737 011712    MOV            AIAST,CTR4 :MOVE ELAPSED TIME TO CTR4
01192 011610 013737 011774 001574 TYPE
01193 011616 104000          MS5
01194 011620 017353

```

```

01197 011622 000000 HALT
01198 011624 004737 011712 JSR %7, AIAS ;OUTPUT CHAR AND TIME
01199 011630 013737 011774 001576 MOV AIAST, CTR5 ;MOVE ELAPSED TIME TO CTR5
02000 011636 104000 TYPE
02001 011640 017353 MS6
02002 011642 000000 HALT
02003 011644 004737 011712 JSR %7, AIAS ;OUTPUT CHAR AND TIME
02004 011650 013737 011774 001600 MOV AIAST, CTR6 ;MOVE ELAPSED TIME TO CTR6
02005 011656 104000 TYPE
02006 011660 017373 MS7
02007 011662 000000 HALT
02008 011664 004737 011712 JSR %7, AIAS ;OUTPUT CHAR AND TIME
02009 011670 013737 011774 001602 MOV AIAST, CTR7 ;MOVE ELAPSED TIME TO CTR7
02010 011676 004737 014424 JSR %7, CMPT ;CHECK THAT CTR0 THROUGH CTR7 CONTAIN
02011 011702 000402 BR AIAF ;DESCENDING VALUES
02012 011704 104015 ERROR! ;TRANSMIT SPEEDS NOT ARRANGED IN
02013 011706 015535 ETXTIM ;ASCENDING ORDER.
02014 011710 104012 RIAF: SCOPE
02015
02016 011712 005037 011774 AIAS: CLR AIAST ;CLEAR ELAPSED TIME COUNTER.
02017 011716 105777 167472 TSTB @TXCSR ;WAIT FOR TX READY.
02018 011722 100375 BPL -4
02019 011724 005077 167466 CLR @TXBUF
02020 011730 105777 167460 TSTB @TXCSR
02021 011734 100375 BPL -4
02022 011736 005077 167454 CLR @TXBUF ;LOAD TXBUF.
02023 011742 004737 011762 AIASA: JSR %7, TIME ;WAIT 75 US
02024 011746 005237 011774 INC AIAST ;INCREMENT ELAPSED TIME COUNTER.
02025 011752 105777 167436 TSTB @TXCSR ;READY SET?
02026 011756 100371 BPL AIASA ;BRANCH IF READY NOT SET.
02027 011760 000207 RTS ;EXIT.
02028
02029 011762 012700 000017 TIME: MOV #15., %0
02030 011766 005300 TIM1: DEC %0
02031 011770 001376 BNE TIM1
02032 011772 000207 RTS %7
02033 011774 000000 AIAST: OPEN
02034 011776 TSTA 10., ALA, 0
02035 011776 TSTAA ALA, 10., \X+1+0, \X+2, \X+1
02036
02037 011776 000035 AT35: 35 ;TEST NUMBER *
02038 012000 012052 AT36 ;ADDRESS OF NEXT TEST *
02039 012002 000012 10. ;ITERATION COUNT *
02040 012004 012006 ALAA ;SCOPE ENTRY POINT *
02041 000035 X=X+1
02042
02043 ;*****
02044 ;TEST THAT OUTPUTTING A CHARACTER WITH THE MAINTENANCE BIT SET (TXCSR BIT 2)
02045 ;RESULTS IN DONE BIT SETTING (RXCSR BIT 7) NO LATER THAN 500 MSECS, AND
02046 ;THAT RESET INSTRUCTION CLEARS THE DONE BIT
02047 012006 052777 000004 167400 ALAA: BIS #BIT2, @TXCSR ;SET MAINTENANCE BIT
02048 012014 005077 167376 CLR @TXBUF ;LOAD TXBUF
02049 012020 104016 DELAY ;WAIT 500 MSECS.
02050 012022 000764 500.
02051 012024 105777 167360 TSTB @RXCSR ;SEE IF DONE BIT IS SET
02052 012030 100402 BMI ALAB ;BRANCH IF DONE BIT IS SET

```

```

2253 012032 104003          ERROR          ;DONE BIT FAILED TO SET
2254 012034 000405          BR          ALAC
2255 012036 104011          ALAB: SRESET          ;ISSUE RESET TO CLEAR DONE BIT
2256 012040 105777 167344  TSTB          2RXCSR          ;SEE IF DONE BIT IS CLEARED
2257 012044 100001          BPL          ALAC          ;BRANCH IF DONE BIT IS CLEARED
2258 012046 104003          ERROR          ;RESET FAILED TO CLEAR DONE BIT
2259 012050 104012          ALAC: SCOPE          ;SCOPE
2260 012052          TSTA          100.,AMA,CD
2261 012052          TSTAA          AMA,100.,\X+1+CD,\X+2,\X+1
2262          ;*****
2263 012052 100036          AT36: 100036          ;TEST NUMBER *
2264 012054 012116          AT37          ;ADDRESS OF NEXT TEST *
2265 012056 000144          100.          ;ITERATION COUNT *
2266 012060 012062          AMAA          ;SCOPE ENTRY POINT *
2267          X=X+1
2268          ;*****
2269          ;TEST THAT DONE BIT (RXCSR BIT 7) IS CLEARED BY READING RXBUF.
2270          ;DONE SET BY OUTPUTTING CHARACTER WITH MAINTENANCE BIT SET (TXCSR BIT 2)
2271 012062 052777 000004 167324 AMAA: BIS          #BIT2,2TXCSR          ;SET MAINTENANCE BIT (TXCSR BIT 2)
2272 012070 005077 167322          CLR          2TXBUF          ;LOAD TXBUF
2273 012074 104017          TIMERX          ;WAIT FOR DONE BIT TO SET.
2274 012076 104003          ERROR
2275 012100 005777 167306          TST          2RXBUF          ;READ RXBUF TO CLEAR DONE BIT
2276 012104 105777 167300          TSTB          2RXCSR          ;SEE IF DONE BIT IS CLEAR
2277 012110 100001          BPL          AMAC          ;BRANCH IF DONE BIT IS CLEAR
2278 012112 104003          ERROR          ;READING RXBUF FAILED TO CLEAR DONE BIT
2279 012114 104012          AMAC: SCOPE          ;SCOPE
2280          TSTA          100.,AOA,CD
2281 012116          TSTAA          AOA,100.,\X+1+CD,\X+2,\X+1
2282          ;*****
2283 012116 100037          AT37: 100037          ;TEST NUMBER *
2284 012120 012226          AT40          ;ADDRESS OF NEXT TEST *
2285 012122 000144          100.          ;ITERATION COUNT *
2286 012124 012132          AOAA          ;SCOPE ENTRY POINT *
2287          X=X+1
2288          ;*****
2289          ;TEST THAT RECEIVER ACTIVE SETS WHEN CHAR STARTS AND
2290          ;CLEARS WHEN RECEIVER DONE SETS
2291 012126 004737 002706          JSR          %7,CDINIT          ;INIT IF C-D DEVICE
2292 012132 052777 000004 167254 AOAA: BIS          #BIT2,2TXCSR          ;SET MAINT
2293 012140 005077 167252          CLR          2TXBUF          ;TRANSMIT CHAR
2294 012144 005037 001616          CLR          TEMP          ;CLEAR BUSY INDICATOR
2295 012150 032777 004000 167232 AOAB: BIT          #BIT11,2RXCSR          ;IS RECEIVER ACTIVE SET
2296 012156 001402          BEQ          AOAB1          ;BRANCH IF CLEAR
2297 012160 005237 001616          INC          TEMP          ;YES, REMEMBER THAT
2298 012164 105777 167220          AOAB1: TSTB          2RXCSR          ;SEE IF DONE SET
2299 012170 100367          BPL          AOAB
2300 012172 023727 001616 000000          CMP          TEMP,#0          ;DID RECEIVER ACTIVE SET
2301 012200 001002          BNE          AOAC
2302 012202 104003          ERROR          ;RECEIVER ACTIVE NEVER SET
2303 012204 000405          BR          AOAD
2304 012206 032777 004000 167174 AOAC: BIT          #BIT11,2RXCSR          ;DID DONE CLEAR ACTIVE
2305 012214 001401          BEQ          AOAD
2306 012216 104003          ERROR          ;NO, RECEIVER ACTIVE DID NOT CLEAR
2307 012220 005777 167166          AOAD: TST          2RXBUF          ;CLEAR RX DONE
2308 012224 104012          SCOPE

```

```

2309 012226 TSTA 1, AQA, 0
2310 012226 TSTAA AQA, 1., \X+1+0, \X+2, \X+1
2311
2312 012226 000040 *****
2313 012230 012530 AT40: 40 ;TEST NUMBER *
2314 012232 000001 AT41 ;ADDRESS OF NEXT TEST *
2315 012234 012236 1 ;ITERATION COUNT *
2316 000040 AQA ;SCOPE ENTRY POINT *
2317 X=X+1
2318 *****
2319 ;TEST THAT RECEIVE SPEEDS ARE ARRANGED IN ASCENDING ORDER BY CHECKING THAT TIME
2320 ;ELAPSED TO DONE BIT SETTING (RXCSR BIT 7) DECREASES AS A HIGHER SPEED
2321 ;THIS IS NOT DONE IN MAINTENANCE MODE TX AND RX
2322 ;POTS MUST BE STEPPED TOGETHER
2323 ;IS SELECTED.
2324 012236 004737 006216 AQA: JSR %7, DOTHIS ;CHECK IF THIS TEST TO BE DONE
2325 012242 104001 TYPES
2326 012244 016745 MSETRX
2327 012246 017021 MSETC
2328 012250 017303 MS0
2329 012252 177777 -1
2330 012254 000000 HALT
2331 012256 004737 012464 JSR %7, AQAS ;OUTPUT CHARACTER AND TIME DONE BIT
2332 012262 013737 012526 001564 MOV AQAST, CTR0 ;MOVE ELAPSED TIME TO CTR0
2333 012270 104000 TYPE
2334 012272 017313 MS1
2335 012274 000000 HALT
2336 012276 004737 012464 JSR %7, AQAS ;OUTPUT CHARACTER AND TIME DONE BIT
2337 012302 013737 012526 001566 MOV AQAST, CTR1 ;MOVE ELAPSED TIME TO CTR1
2338 012310 104000 TYPE
2339 012312 017323 MS2
2340 012314 000000 HALT
2341 012316 004737 012464 JSR %7, AQAS ;OUTPUT CHARACTER AND TIME DONE BIT.
2342 012322 013737 012526 001570 MOV AQAST, CTR2 ;MOVE ELAPSED TIME TO CTR2.
2343 012330 104000 TYPE
2344 012332 017333 MS3
2345 012334 000000 HALT
2346 012336 004737 012464 JSR %7, AQAS ;OUTPUT CHARACTER AND TIME DONE BIT
2347 012342 013737 012526 001572 MOV AQAST, CTR3 ;MOVE ELAPSED TIME TO CTR3.
2348 012350 104000 TYPE
2349 012352 017343 MS4
2350 012354 000000 HALT
2351 012356 004737 012464 JSR %7, AQAS
2352 012362 013737 012526 001574 MOV AQAST, CTR4
2353 012370 104000 TYPE
2354 012372 017353 MS5
2355 012374 000000 HALT
2356 012376 004737 012464 JSR %7, AQAS
2357 012402 013737 012526 001576 MOV AQAST, CTR5
2358 012410 104000 TYPE
2359 012412 017363 MS6
2360 012414 000000 HALT
2361 012416 004737 012464 JSR %7, AQAS
012422 013737 012526 001600 MOV AQAST, CTR6

```

```

2362 012430 104000 TYPE
2363 012432 017373 MS7
2364 012434 000000 HALT
2365 012436 004737 012464 JSR %7, AQAS
2366 012442 013737 012526 001602 MOV AQAST, CTR7
2367 012450 004737 014424 JSR %7, CMPT ;CHECK THAT CTR0 THROUGH CTR3 CONTAIN
2368 012454 000402 BR AQAB ;DESCENDING VALUES.
2369 012456 104015 ERROR1 ;RECEIVE SPEEDS NOT ARRANGED IN
2370 012460 015577 ERXTIM ;ASCENDING ORDER.
2371 012462 104012 AQAB: SCOPE ;SCOPE
2372
2373 012464 005037 012526 AQAS: CLR AQAST ;CLEAR ELAPSED TIME COUNTER AQAST
2374 012470 105777 166720 TSTB @TXCSR ;WAIT FOR TX READY.
2375 012474 100375 BPL -4
2376 012476 005777 166710 TST @RXBUF ;CLEAR DONE BIT IF SET
2377 012502 005077 166710 CLR @TXBUF ;LOAD TXBUF
2378 012506 004737 011762 AQASA: JSR %7, TIME
2379 012512 005237 012526 INC AQAST ;INCREMENT ELAPSED TIME COUNTER
2380 012516 105777 166666 TSTB @RXCSR ;DONE SET?
2381 012522 100371 BPL AQASA ;BRANCH IF DONE NOT SET
2382 012524 000207 RTS %7 ;EXIT
2383 012526 000000 AQAST: OPEN ;ELAPSED TIME COUNTER
2384 012530 TSTA 10., ARA, CD
2385 012530 TSTAA ARA, 10., \X+1+CD, \X+2, \X+1
2386
2387 012530 100041 AT41: 100041 ;TEST NUMBER *
2388 012532 012674 AT42 ;ADDRESS OF NEXT TEST *
2389 012534 000012 10. ;ITERATION COUNT *
2390 012536 012540 ARAA ;SCOPE ENTRY POINT *
2391 000041 X=X+1 ; *
2392
2393 ;*****
2394 ;TEST CORRECT OPERATION OF DATA OVERRUN BIT (RXBUF BIT 14)
2395 ARAA: JSR %7, ARAA ;OUTPUT CHARACTER AND WAIT 500 MSECS
2396 012544 004737 012654 JSR %7, ARAA ;OUTPUT CHARACTER AND WAIT 500 MSECS
2397 012550 017737 166636 001610 MOV @RXBUF, RXBUF ;SAVE RXBUF CONTENTS + CLEAR DONE
2398 012556 032737 040000 001610 BIT #BIT14, RXBUF ;SEE IF DATA OVERRUN BIT WAS SET
2399 012564 001002 BNE .+6 ;BRANCH IF BIT WAS SET
2400 012566 104003 ERROR
2401 012570 104012 SCOPE
2402 012572 005737 001610 TST RXBUF ;SEE THAT ERROR BIT WAS SET (RXBUF BIT 15)
2403 012576 100402 BMI .+6
2404 012600 104003 ERROR ;ERROR BIT FAILED TO SET WHEN OVERRUN SET
2405 012602 104012 SCOPE
2406 012604 032777 040000 166600 BIT #BIT14, @RXBUF ;SEE THAT DATA OVERRUN WAS NOT
2407 012612 001002 BNE .+6 ;CLEARED WHEN RXBUF WAS READ
2408 012614 104003 ERROR ;BRANCH IF SET
2409 012616 104012 SCOPE ;READING RXBUF CLEARED DATA OVERRUN
2410 012620 004737 012654 JSR %7, ARAA
2411 012624 032777 100000 166560 BIT #BIT15, @RXBUF ;OUTPUT CHAR +WAIT 500MS
2412 012632 001402 BEQ .+6 ;TEST THAT ERROR CLEARED
2413 012634 104003 ERROR
2414 012636 104012 SCOPE
2415 012640 032777 040000 166544 BIT #BIT14, @RXBUF ;TEST THAT OVERRUN CLEARED
2416 012646 001401 BEQ .+4
2417 012650 104003 ERROR

```

```

2418 012652 104012          SCOPE          ;SCOPE
2419 012654 052777 000004 166532 ARAS:  BIS      #BIT2,ATXCSR ;SET MAINTENANCE BIT
2420 012662 005077 166530          CLR      ATXBUF  ;LOAD TXBUF
2421 012666 104016          DELAY          ;DELAY 500 MSECS
2422 012670 000764          500.
2423 012672 000207          RTS      %7      ;EXIT
2424 012674          TSTA     10.,ATA,CD
2425 012674          TSTAA   ATA,10.,\X+1+CD,\X+2,\X+1
2426
2427 012674 100042          AT42:  100042 ;TEST NUMBER *
2428 012676 012762          AT43          ;ADDRESS OF NEXT TEST *
2429 012700 000012          10.          ;ITERATION COUNT *
2430 012702 012714          ATAA          ;SCOPE ENTRY POINT *
2431          X=X+1 ; *
2432
2433          ;*****
2434          ;TEST THAT TRANSMITTER IS ABLE TO INTERRUPT. IF THE INTERRUPT IS SERVICED,
2435          ;IT WILL HAVE OCCURRED AT THE CORRECT VECTOR.
2436 012704 004737 003516          JSR      7,OVRLAY ;GO TO OVER LAY ROUTINE
2437 012710 104007          STTXV          ;SET TX INTERRUPT SERVICE
2438 012712 012750          ATAC          ;TO ATAC
2439 012714 042777 000100 166472 ATAA:  BIC      #BIT6,ATXCSR ;DISABLE TX INTERRUPT
2440 012722 005037 177776          CLR      PSW    ;SET PROCESSOR PRIORITY TO 0
2441 012726 052777 000100 166460          BIS      #BIT6,ATXCSR ;ENABLE TX INTERRUPT
2442 012734 000240          NOP
2443 012736 104003          ERROR          ;READY DID NOT CAUSE AN INTERRUPT
2444 012740 042777 000100 166446          BIC      #BIT6,ATXCSR
2445 012750 042777 000100 166436 ATAB:  SCOPE          ;SCOPE
2446 012756 022626          ATAC:  BIC      #BIT6,ATXCSR ;HERE IF INT. DISABLE TX INT
2447 012760 000772          POPSP2
2448          BR      ATAB
2449 012762          TSTA     1000.,AUA,CD
2450          TSTAA   AUA,1000.,\X+1+CD,\X+2,\X+1
2451
2451 012762 100043          AT43:  100043 ;TEST NUMBER *
2452 012764 013040          AT44          ;ADDRESS OF NEXT TEST *
2453 012766 001750          1000.        ;ITERATION COUNT *
2454 012770 012776          AUAA          ;SCOPE ENTRY POINT *
2455          X=X+1 ; *
2456
2457          ;*****
2458          ;TEST THAT READY DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR IS
2459          ;AT THE SAME PRIORITY AS THE TRANSMITTER INTERRUPT REQUEST LEVEL
2460 012772 104007          STTXV          ;SET TX INTERRUPT SERVICE TO
2461 012774 013032          AUAC
2462 012776 013737 001426 177776 AUAA:  MOV      TXLVL,PSW ;SET PROCESSOR PRIORITY SAME AS TX PRIORITY
2463 013004 042777 000100 166402          BIC      #BIT6,ATXCSR
2464 013012 052777 000100 166374          BIS      #BIT6,ATXCSR ;ENABLE TX INTERRUPTS
2465 013020 000240          NOP
2466 013022 042777 000100 166364 AUAB:  BIC      #BIT6,ATXCSR ;OK IF NO INTERRUPT OCCURS. DISABLE INTERRUPTS
2467 013030 104012          SCOPE          ;SCOPE
2468 013032 022626          AUAC:  POPSP2          ;HERE IF INTERRUPT OCCURS. POP STOCK TWICE
2469 013034 104003          ERROR          ;TX INTERRUPTED WITH PROCESSOR AT SAME
2470 013036 000771          BR      AUAB    ;PRIORITY AS THE TRANSMITTER
2471 013040          TSTA     10.,AVA,CD
2472          TSTAA   AVA,10.,\X+1+CD,\X+2,\X+1
2473
2473 013040 100044          AT44:  100044 ;TEST NUMBER *

```

```

2474 013042 013124 AT45 ;ADDRESS OF NEXT TEST *
2475 013044 000012 10. ;ITERATION COUNT *
2476 013046 013054 AVAA ;SCOPE ENTRY POINT *
2477 000044 X=X+1 ; *
2478 ;***** *
2479 ;TEST THAT TRANSMITTER INTERRUPTS WHEN PROCESSOR IS AT PRIORITY ONE LEVEL *
2480 ;LOWER THAN THE TRANSMITTER INTERRUPT PRIORITY. *
2481 013050 104007 STTXV ;SET TX INTERRUPT SERVICE TO AVAB
2482 013052 013112 AVAB
2483 013054 042777 000100 166332 AVAA: BIC #BIT6,ATXCSR ;DISABLE TX INTERRUPTS
2484 013062 013737 001426 177776 MOV TXLVL,PSW ;SET PROCESSOR PRIORITY TO ONE LEVEL
2485 013070 162737 000040 177776 SUB #40,PSW ;LOWER THAN TX PRIORITY
2486 013076 052777 000100 166310 BIS #BIT6,ATXCSR ;ENABLE TX INTERRUPTS
2487 013104 000240 NOP
2488 013106 104003 ERROR ;TX FAILED TO INTERRUPT
2489 013110 000401 BR AVAC
2490 013112 022626 AVAB: POPSP2 ;HERE IF INTERRUPT OCCURS. POP STOCK TWICE
2491 013114 042777 000100 166272 AVAC: BIC #BIT6,ATXCSR ;DISABLE TX INTERRUPTS
2492 013122 104012 SCOPE ;SCOPE
2493 013124 TSTA 100.,AWA,CD
2494 013124 TSTAA AWA,100.,\X+1+CD,\X+2,\X+1
2495 ;***** *
2496 013124 100045 AT45: 100045 ;TEST NUMBER *
2497 013126 013222 AT46 ;ADDRESS OF NEXT TEST *
2498 013130 000144 100. ;ITERATION COUNT *
2499 013132 013134 AVAA ;SCOPE ENTRY POINT *
2500 000045 X=X+1 ; *
2501 ;***** *
2502 ;TEST THAT TRANSMITTER DOES NOT REINTERRUPT AFTER THE INITIAL INTERRUPT HAS *
2503 ;OCCURRED AND HAS BEEN SERVICED. *
2504 013134 104007 AWAA: STTXV ;SET TX INTERRUPT SERVICE TO AWAC
2505 013136 013174 AWAC
2506 013140 042777 000100 166246 BIC #BIT6,ATXCSR ;DISABLE TX INTERRUPTS
2507 013146 005037 177776 CLR PSW ;SET PROCESSOR PRIORITY TO 0
2508 013152 052777 000100 166234 BIS #BIT6,ATXCSR ;ENABLE TX INTERRUPTS
2509 013160 000240 NOP
2510 013162 104003 ERROR ;TRANSMITTER FAILED TO INTERRUPT
2511 013164 042777 000100 166222 AWAB: BIC #BIT6,ATXCSR ;DISABLE TX INTERRUPTS
2512 013172 104012 SCOPE ;SCOPE
2513 013174 012777 013214 166222 AWAC: MOV #AWAE,ATXVTR ;HERE IF INTERRUPT OCCURS. CHANGE EXIT
2514 013202 012716 013210 MOV #AWAD,AT%6 ;POINTER TO AWAD AND EXIT INTERRUPT
2515 013206 000002 RTI
2516 013210 000240 AWAD: NOP ;OK IF NO INTERRUPT REOCCURS.
2517 013212 000764 BR AWAB
2518 013214 022626 AWAE: POPSP2 ;HERE IF INTERRUPT REOCCURS
2519 013216 104003 ERROR ;TX REINTERRUPTED AFTER RTI
2520 013220 000761 BR AWAB
2521 013222 TSTA 10.,AXA,CD
2522 013222 TSTAA AXA,10.,\X+1+CD,\X+2,\X+1
2523 ;***** *
2524 013222 100046 AT46: 100046 ;TEST NUMBER *
2525 013224 013306 AT47 ;ADDRESS OF NEXT TEST *
2526 013226 000012 10. ;ITERATION COUNT *
2527 013230 013246 AXAA ;SCOPE ENTRY POINT *
2528 000046 X=X+1 ; *
2529 ;***** *

```



```

2530          :TEST THAT RECEIVER DONE BIT IS ABLE TO INTERRUPT. IF THE INTERRUPT IS
2531          :SERVICED IT WILL HAVE OCCURRED AT THE CORRECT VECTOR.
2532 013232 004737 003516          JSR      7,OVRLAY          ;GO TO OVERLAY ROUTINE
2533 013236 104006          STRXV          ;SET RX INTERRUPT SERVICE TO AXAB
2534 013240 013274          AXAB
2535 013242 004737 014404          JSR      %7,STRXD          ;SET RX DONE BIT
2536 013246 042777 000100 166134 AXAA: BIC      #BIT6,DRXCSR          ;DISABLE RX INTERRUPTS
2537 013254 005037 177776          CLR      PSW          ;SET PROCESSOR PRIORITY TO 0
2538 013260 052777 000100 166122          BIS      #BIT6,DRXCSR          ;ENABLE RX INTERRUPTS
2539 013266 000240          NOP
2540 013270 104003          ERROR          ;RX FAILED TO INTERRUPT
2541 013272 000401          BR      AXAC
2542 013274 022626          POPSP2          ;HERE IF INTERRUPT OCCURS
2543 013276 042777 000100 166104 AXAB: BIC      #BIT6,DRXCSR          ;DISABLE INT EN
2544 013304 104012          AXAC: SCOPE          ;SCOPE
2545 013306          TSTA      10.,AXI,0
2546 013306          TSTAA     AXI,10.,\X+1+0,\X+2,\X+1
2547          :*****
2548 013306 000047          AT47: 47          ;TEST NUMBER *
2549 013310 013370          AT50          ;ADDRESS OF NEXT TEST *
2550 013312 000012          10.          ;ITERATION COUNT *
2551 013314 013326          AX1A          ;SCOPE ENTRY POINT *
2552          000047          X=X+1          ;
2553          :*****
2554          :TEST THAT MODEM INTERRUPT BIT IS ABLE TO INTERRUPT. IF THE INTERRUPT IS
2555          :SERVICED IT WILL HAVE OCCURRED AT THE CORRECT VECTOR.
2556 013316 004737 003516          JSR      7,OVRLAY          ;GO TO OVERLAY ROUTINE
2557 013322 104006          STRXV          ;SET RX INTERRUPT SERVICE TO AXAB
2558 013324 013356          AX1B
2559 013326 042777 000044 166054 AX1A: BIC      #44,DRXCSR          ;DISABLE MODEM INTERRUPTS
2560 013334 005037 177776          CLR      PSW          ;SET PROCESSOR PRIORITY TO 0
2561 013340 052777 000044 166042          BIS      #44,DRXCSR          ;ENABLE MODEM INTERRUPTS,RQ TO SND
2562 013346 104016          DELAY          ;
2563 013350 000005          5.
2564 013352 104003          ERROR          ;MODEM FAILED TO INTERRUPT
2565 013354 000401          BR      AX1C
2566 013356 022626          POPSP2          ;HERE IF INTERRUPT OCCURS
2567 013360 042777 000040 166022 AX1B: BIC      #BIT5,DRXCSR          ;DISABLE INT EN
2568 013366 104012          AX1C: SCOPE
2569 013370          TSTA      1000.,AYA,CD
2570 013370          TSTAA     AYA,1000.,\X+1+CD,\X+2,\X+1
2571          :*****
2572 013370 100050          AT50: 100050          ;TEST NUMBER *
2573 013372 013452          AT51          ;ADDRESS OF NEXT TEST *
2574 013374 001750          1000.          ;ITERATION COUNT *
2575 013376 013410          AYAA          ;SCOPE ENTRY POINT *
2576          000050          X=X+1          ;
2577          :*****
2578          :TEST THAT RECEIVER DONE BIT DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR
2579          :IS AT THE SAME PRIORITY LEVEL AS THE RECEIVER INTERRUPT REQUEST LEVEL
2580 013400 104006          STRXV          ;SET RX INTERRUPT SERVICE TO AYAC
2581 013402 013444          AYAC
2582 013404 004737 014404          JSR      %7,STRXD          ;SET RX DONE BIT
2583 013410 042777 000100 165772 AYAA: BIC      #BIT6,DRXCSR          ;DISABLE RX INTERRUPTS
2584 013416 013737 001422 177776          MOV      RXLVL,PSW          ;SET PROCESSOR PRIORITY SAME AS RECEIVER'S
2585 013424 052777 000100 165756          BIS      #BIT6,DRXCSR          ;ENABLE RX INTERRUPTS

```

```

2586 013432 000240      NOP
2587 013434 042777 000100 165746 AYAB: BIC      #BIT6,ARXCSR ;OK IF NO INTERRUPT. DISABLE RX INTERRUPTS
2588 013442 104012      SCOPE ;SCOPE
2589 013444 022626      AYAC: POPSP2 ;HERE IF INTERRUPT OCCURS. POP STOCK TWICE
2590 013446 104003      ERROR ;RX INTERRUPTED WITH PROCESOR AT SAME
2591 013450 000771      BR      AYAB ;PRIORITY AS THE RECEIVER
2592 013452      TSTA    10.,AZA,CD
2593 013452      TSTAA  AZA,10.,\X+1+CD,\X+2,\X+1
2594 ;*****
2595 013452 100051      AT51: 100051 ;TEST NUMBER *
2596 013454 013542      AT5?  ;ADDRESS OF NEXT TEST *
2597 013456 000012      10. ;ITERATION COUNT *
2598 013460 013472      AZAA ;SCOPE ENTRY POINT *
2599 ;X=X+1 *
2600 ;*****
2601 ;TEST THAT RECEIVER DONE BIT CAUSES INTERRUPT WHEN PROCESSOR IS AT PRIORITY
2602 ;ONE LEVEL LOWER THAN THE RECEIVER'S INTERRUPT REQUEST LEVEL.
2603 013462 104006      STRXV ;SET RX INTERRUPT TO AZAB
2604 013464 013530      AZAB
2605 013466 004737 014404      JSR     %7,STRXD ;SET RX DONE BIT
2606 013472 042777 000100 165710 AZAA: BIC     #BIT6,ARXCSR ;DISABLE RX INTERRUPTS
2607 013500 013737 001422 177776      MOV     RXLVL,PSW ;SET PROCESSOR PRIORITY ONE LEVEL
2608 013506 162737 000040 177776      SUB     #40,PSW ;LOWER THAN RECEIVER'S PRIORITY
2609 013514 052777 000100 165666      BIS     #BIT6,ARXCSR ;ENABLE RX INTERRUPTS
2610 013522 000240      NOP
2611 013524 104003      ERROR ;RX FAILED TO INTERRUPT WITH PROCESSOR AT
2612 013526 000401      BR      AZAC ;PRIORITY ONE LEVEL LOWER THAN RECEIVER'S
2613 013530 022626      AYAB: POPSP2 ;HERE IF INTERRUPT OCCURS
2614 013532 042777 000100 165650 AZAC: BIC     #BIT6,ARXCSR ;DISABLE RX INTERRUPTS
2615 013540 104012      SCOPE ;SCOPE
2616 013542      TSTA    100.,AAB,CD
2617 013542      TSTAA  AAB,100.,\X+1+CD,\X+2,\X+1
2618 ;*****
2619 013542 100052      AT52: 100052 ;TEST NUMBER *
2620 013544 013640      AT53  ;ADDRESS OF NEXT TEST *
2621 013546 000144      100. ;ITERATION COUNT *
2622 013550 013556      AABA ;SCOPE ENTRY POINT *
2623 ;X=X+1 *
2624 ;*****
2625 ;TEST THAT RECEIVER DOES NOT INTERRUPT AFTER THE INITIAL INTERRUPT HAS
2626 ;OCCURED AND DONE BIT HAS NOT BEEN CLEARED
2627 013552 004737 014404      JSR     %7,STRXD ;SET RX DONE BIT
2628 013556 104006      AABA: STRXV ;SET RX INTERRUPT SERVICE TO AABC
2629 013560 013612      AABC
2630 013562 042777 000100 165620      BIC     #BIT6,ARXCSR ;DISABLE RX INTERRUPTS
2631 013570 052777 000100 165612      BIS     #BIT6,ARXCSR ;ENABLE RX INTERRUPTS
2632 013576 000240      NOP
2633 013600 104003      ERROR ;RX FAILED TO INTERRUPT
2634 013602 042777 000100 165600 AABB: BIC     #BIT6,ARXCSR ;DISABLE RX INTERRUPTS
2635 013610 104012      SCOPE ;SCOPE
2636 013612 012777 013632 165600 AABC: MOV     #AABE,ARXVTR ;HERE IF INTERRUPT OCCURS. CHANGE SERVICE TO
2637 013620 012716 013626      MOV     #AABD,%6 ;AABE, SET EXIT POINTER TO AABD
2638 013624 000002      RTI ;EXIT INTERRUPT SERVICE
2639 013626 000240      AABD: NOP ;OK IF NO INTERRUPT REOCCURS
2640 013630 000764      BR      AABB
2641 013632 022626      AABE: POPSP2 ;HERE IF INTERRUPT REOCCURS

```

```

2642 013634 104003          ERROR                      ;RX REINTERRUPTED AFTER RTI
2643 013636 000761          BR                      ABBB
2644 013640                TSTA          100. ABB,CD
2645 013640                TSTAA         ABB,100.,\X+1+CD,\X+2,\X+1
2646                ;*****
2647 013640 100053          AT53: 100053                ;TEST NUMBER *
2648 013642 013700          AT54                ;ADDRESS OF NEXT TEST *
2649 013644 000144          100.                ;ITERATION COUNT *
2650 013646 013650          ABBA                ;SCOPE ENTRY POINT *
2651                X=X+1
2652                ;*****
2653                ;TEST THAT READING RXCSR DOES NOT CLEAR DONE BIT (RXCSR BIT 7 )
2654 013650 004737 014404          ABBA: JSR          %7,STRXD                ;SET RX DONE BIT
2655 013654 017737 165530 001606          MOV          @RXCSR,RXCST                ;SAVE CONTENT OF RXCSR
2656 013662 105777 165522          TSTB         @RXCSR                ;SEE IF DONE BIT IS CLEAR
2657 013666 100401          BMI          ABBB                ;BRANCH IF DONE BIT IS NOT CLEAR
2658 013670 104003          ERROR
2659 013672 005777 165514          ABBB: TST          @RXBUF                ;CLEAR DONE BIT IF SET
2660 013676 104012          SCOPE
2661 013700                TSTA          100. ACB,CD
2662 013700                TSTAA         ACB,100.,\X+1+CD,\X+2,\X+1
2663                ;*****
2664 013700 100054          AT54: 100054                ;TEST NUMBER *
2665 013702 013764          AT55                ;ADDRESS OF NEXT TEST *
2666 013704 000144          100.                ;ITERATION COUNT *
2667 013706 013714          ACBA                ;SCOPE ENTRY POINT *
2668                X=X+1
2669                ;*****
2670                ;TEST THAT DONE CAN CAUSE INT WITH ERROR SET
2671 013710 104006          STRXV                ;SET RX INTERRUPT SERVICE TO ACBB.
2672 013712 013752          ACBB
2673 013714 004737 014404          ACBA: JSR          %7,STRXD                ;SET RX DONE BIT
2674 013720 004737 014404          JSR          %7,STRXD                ;SET RX DATA OFLOW
2675 013724 042777 000100 165456          BIC          #BIT6,@RXCSR                ;DISABLE RX INTERRUPTS
2676 013732 005037 177776          CLR          PSW                ;SET PROCESSOR PRIORITY TO 0
2677 013736 052777 000100 165444          BIS          #BIT6,@RXCSR                ;ENABLE RX INTERRUPTS
2678 013744 000240          NOP
2679 013746 104003          ERROR                      ;RX DONE FAILED TO CAUSE INTERRUPT
2680 013750 000401          BR                      ACBC
2681 013752 022626          ACBB: POPSP2                ;HERE IF INTERRUPT OCCURS. POP STOCK TWICE
2682 013754 042777 000100 165426          ACBC: BIC          #BIT6,@RXCSR
2683 013762 104012          SCOPE
2684 013764                TSTA          3. ,ADD,CD
2685 013764                TSTAA         ADD,3.,\X+1+CD,\X+2,\X+1
2686                ;*****
2687 013764 100055          AT55: 100055                ;TEST NUMBER *
2688 013766 014006          AT56                ;ADDRESS OF NEXT TEST *
2689 013770 000003          3.                ;ITERATION COUNT *
2690 013772 014000          ADDA                ;SCOPE ENTRY POINT *
2691                X=X+1
2692                ;*****
2693                ;DATA TEST USING NORMAL CONFIGURATION
2694 013774 004737 002706          ADDA: JSR          %7,CDINIT                ;INIT IF C-D DEVICE
2695 014000 004537 004770          JSR          5, @ATTST
2696 014004 104012          SCOPE
2697 014006                TSTA          3. ,APB,0

```

```

2698 014006          TSTAA  APB,3.,\X+1+0,\X+2,\X+1
2699                ;*****
2700 014006 000056  AT56:  56          ;TEST NUMBER *
2701 014010 014100          AT57          ;ADDRESS OF NEXT TEST *
2702 014012 000003          3.          ;ITERATION COUNT *
2703 014014 014022          APBA          ;SCOPE ENTRY POINT *
2704          000056          X=X+1          ;
2705                ;*****
2706                ;DATA TEST USING JUMPER CONNECTOR.
2707                ;USES SPECIAL BINARY COUNT PATTERN FOR DATA. NO INTERRUPT.
2708 014016 004737 004322  JSR      7,INBIN          ;INITIALIZE BINARY COUNT PATTERN
2709 014022 012737 001750 001564 APBA:  MOV      #1000.,CTRD          ;SET CHARACTER COUNT TO 1000
2710 014030 104020          APBB:  TIMETX          ;TIME OUT TX DONE
2711 014032 104003          ERROR          ;ERROR DONE NOT SETTING
2712 014034 004737 004426  JSR      7,GTBINP          ;GET BINARY CHARACTER
2713 014040 110137 001560  MOVB    %1,CRBUFA          ;SAVE CHAR IN CRBUFA AND
2714 014044 004737 005374  JSR      7,MASKIT          ;MASK OFF NON TRANSMITTED BITS
2715 014050 110177 165342  MOVB    %1,RTXBUF          ;LOAD CHAR.
2716 014054 104017          TIMERX          ;TIME OUT RX DONE
2717 014056 104003          ERROR          ;ERROR DONE NOT SETTING
2718 014060 117737 165326 001556  MOVB    @RXBUF,CRBUF          ;LOAD RECEIVED DATA INTO CRBUF
2719 014066 104004          DATCHK          ;CHECK DATA
2720 014070 005337 001564  DEC      CTRD          ;TESTED 1000 CHARACTERS
2721 014074 001355          BNE      APBB          ;BRANCH IF NOT
2722 014076 104012          SCOPE          ;YES. SCOPE
2723 014100          TSTA   3.,EXT,0
2724 014100          TSTAA  EXT,3.,\X+1+0,\X+2,\X+1
2725                ;*****
2726 014100 000057  AT57:  57          ;TEST NUMBER *
2727 014102 014164          AT60          ;ADDRESS OF NEXT TEST *
2728 014104 000003          3.          ;ITERATION COUNT *
2729 014106 014110          EXTA          ;SCOPE ENTRY POINT *
2730          000057          X=X+1          ;
2731                ;*****
2732                ;TEST THAT RDR BUSY TURNS OFF RDR ENABLE
2733                ;WHEN RUN ON AN XOR TESTER
2734                EXTA:  RESET          ;RESET
2735 014110 000005          INC      @RXCSR          ;SET RDR ENABLE, SEE IF RDE IS TURNED OFF BY RDR BUSY
2736 014112 005277 165272          MOV      #-10,3$+2
2737 014116 012737 177770 014154  2$:  INC      3$+2          ;WAIT LOOP FOR XOR TESTER
2738 014124 005237 014154          BNE      2$
2739 014130 001375          CLR      @TXBUF          ;SHIP OUT CHAR.
2740 014132 005077 165260          MOV      #-50000,3$+2
2741 014136 012737 130000 014154  5$:  TSTB    @RXCSR          ;TEST COMPLETE
2742 014144 105777 165240          BMI      6$
2743 014150 100404          INC      #-10          ;ALLOW TIME FOR RDR DONE TO SET
2744 014152 005227 177770          BNE      5$
2745 014156 001372          ERROR          ;FAILURE OF RDR DONE TO SET
2746 014160 104003          SCOPE
2747 014162 104012          TSTA   10.,EX,0
2748 014164          TSTAA  EX,10.,\X+1+0,\X+2,\X+1
2749 014164                ;*****
2750                ;*****
2751 014164 000060  AT60:  60          ;TEST NUMBER *
2752 014166 014234          AT61          ;ADDRESS OF NEXT TEST *
2753 014170 000012          10.          ;ITERATION COUNT *

```

```

2754 014172 014174 EXA ;SCOPE ENTRY POINT *
2755 000060 X=X+1 ; *
2756 ;***** *
2757 ;TEST THAT WHEN RDR ENABLE IS SET THAT THE RXCSR DONE *
2758 ;BIT IS CLEARED *
2759 014174 000005 EXA: RESET *
2760 014176 004737 014404 JSR PC,STRXD ;SET RCVR DONE *
2761 014202 005277 165202 INC @RXCSR ;SET ENABLE *
2762 014206 105777 165176 TSTB @RXCSR ;DONE SHOULD CLEAR *
2763 014212 100001 BPL 1$ *
2764 014214 104003 ERROR ;DONE NOT CLEAR *
2765 014216 012737 177770 014226 1$: MOV #-10,3$+2 *
2766 014224 005227 177770 3$: INC #-10 ;WAIT 100MIC. SEC. FOR XOR *
2767 014230 001375 BNE 3$ *
2768 014232 104012 SCOPE *
2769 014234 TSTA 3.,EXA,0 *
2770 014234 TSTAA EXA,3.,\X+1+0,\X+2,\X+1 *
2771 ;***** *
2772 014234 000061 AT61: 61 ;TEST NUMBER *
2773 014236 014270 AT62 ;ADDRESS OF NEXT TEST *
2774 014240 000003 3. ;ITERATION COUNT *
2775 014242 014244 EXAA ;SCOPE ENTRY POINT *
2776 000061 X=X+1 ; *
2777 ;***** *
2778 014244 005737 002016 EXAA: TST XORFLG ;CHECKING JUMPER CONNECTIONS FOR XOR, RCVR *
2779 014250 100006 BPL 3$ *
2780 014252 012777 177777 165130 MOV #-1,@RXCSR *
2781 014260 005777 165124 TST @RXCSR *
2782 014264 000005 RESET *
2783 014266 104012 3$: SCOPE *
2784 014270 TSTA 3.,EXB,0 *
2785 014270 TSTAA EXB,3.,\X+1+0,\X+2,\X+1 *
2786 ;***** *
2787 014270 000062 AT62: 62 ;TEST NUMBER *
2788 014272 014324 AT63 ;ADDRESS OF NEXT TEST *
2789 014274 000003 3. ;ITERATION COUNT *
2790 014276 014300 EXBA ;SCOPE ENTRY POINT *
2791 000062 X=X+1 ; *
2792 ;***** *
2793 014300 005737 002016 EXBA: TST XORFLG ;SAME AS ABOVE BUT FOR XMTR *
2794 014304 100006 BPL 4$ *
2795 014306 012777 177677 165100 MOV #177677,@TXCSR *
2796 014314 005777 165074 TST @TXCSR *
2797 014320 000005 RESET *
2798 014322 104012 4$: SCOPE *
2799 014324 TSTAA AQB,10.,\X+1+CD, LAST,\X+1 *
2800 ;***** *
2801 014324 100063 AT63: 100063 ;TEST NUMBER *
2802 014326 177777 ATLAST ;ADDRESS OF NEXT TEST *
2803 014330 000012 10. ;ITERATION COUNT *
2804 014332 014334 AQBA ;SCOPE ENTRY POINT *
2805 000063 X=X+1 ; *
2806 ;***** *
2807 ;TEST THAT WHEN TXCSR BIT 0 IS SET THAT THE OUTPUT DATA LINE *
2808 ;IS PULLED TO A SPACE. *
2809 014334 004737 002706 AQBA: JSR %7,CDINIT ;INIT IF C-D DEVICE *

```

00000000	014040	052777	000004	165046
00000000	014040	052777	000001	165040
00000000	014040	052777	000052	165034
00000000	014040	104017		
00000000	014040	104003	165020	000000
00000000	014040	104001		
00000000	014040	104003		
00000000	014040	104011		
00000000	014040	104012		

```

BIS #BIT2, @TXCSR ;SET MAINTENANCE BIT IN TXCSR
BIS #BIT0, @TXCSR ;SET BREAK BIT
MOV #252, @TXBUF ;LOAD BUFFER
TIMERX ;TIME OUT RX DONE
ERROR ;ERROR DONE NOT SETTING
CMPB @RXBUF, #0 ;CHARACTER RECEIVED SHOULD BE 0
BFG .+4
ERROR ;CHARACTER OTHER THAN 0
SRESET ;ISSUE RESET
SCOPE

```


2844
2845
2846
2847
2848
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2880
2881

014522 104000
014524 015641
014526 004537 004132
014532 104000
014534 015736
014536 004737 003554
014542 012637 001622
014546 113737 001622 014554
014554 113777 001622 164634
014562 104016
014564 000000
014566 000767

014570 104000
014572 015701
014574 004537 004132
014600 104000
014602 015736
014604 004737 003554
014610 012637 001622
014614 052777 000004 164572
014622 113737 001622 014640
014630 113777 001622 164560
014636 104016
014640 000000
014642 017700 164544
014646 000005
014650 000005
014652 000005
014654 000005
014656 000005
014660 000755

004132

003554

001622

001622

001622

000000

000767

004132

003554

001622

000004

001622

001622

017700

000005

000005

000005

000005

000005

000755

014554
164634

164572
014640
164560

164544

:PRG1 - TRANSMITTER SCOPE LOOP

PRG1: TYPE
P1TIT
JSR S,LINSLX
TYPE
SELCAD
JSR PC,RDOCT
MOV (SP)+,TEMP2
PRG1A: MOVB TEMP2,PRG1B
MOVB TEMP2+1,@TXBUF
DELAY
PRG1B: OPEN
BR PRG1A

:TYPE PROGRAM TITLE.
:GO GET LINE # FROM USER
:TYPE SELECT CHAR AND DELAY.
:READ IN DATA.
:STORE DATA.
:DELAY COUNT TO PRG1B.
:LOAD TXBUF.
:DELAY # OF MSECS. SET AT SR.
:REPEAT.

:PRG2 - RECEIVER SCOPE LOOP.

PRG2: TYPE
P2TIT
JSR S,LINSLX
TYPE
SELCAD
JSR PC,RDOCT
MOV (SP)+,TEMP2
PRG2A: BIS #BIT2,@TXCSR
MOVB TEMP2,PRG2B
MOVB TEMP2+1,@TXBUF
DELAY
PRG2B: OPEN
MOV @RXBUF,R0
RESET
RESET
RESET
RESET
RESET
BR PRG2A

:TYPE PROGRAM TITLE.
:GO GET LINE # FROM USER
:TYPE SELECT CHAR AND DELAY.
:READ IN DATA.
:STORE DATA.
:SET MAINTENANCE BIT.
:DELAY COUNT TO PRG2B.
:LOAD TXBUF.
:DELAY # OF MSECS. SET IN SR.
:RXBUF CONTENTS TO R0.
:DISPLAY CONTENTS OF RXBUF (IN R0).
:BY ISSUING 5 RESET INSTRUCTIONS


```

2882
2883
2884
2885
2886 014662 104000
2887 014664 016504
2888 014666 004537 004132
2889 014672 104000
2890 014674 016627
2891 014676 004737 003554
2892 014702 012637 001622
2893 014706 113737 001623 001560 PRG3A:
2894 014714 004737 014754
2895 014720 000772
2896
2897
2898
2899 014722 104000
2900 014724 016554
2901 014726 004537 004132
2902 014732 004737 004322
2903 014736 004737 004426 PRG4A:
2904 014742 110137 001560
2905 014746 004737 014754
2906 014752 000771
2907
2908 014754 032777 000200 163212
2909 014762 001001
2910 014764 104002
2911 014766 104020
2912 014770 104003
2913 014772 052777 000004 164414
2914 015000 005777 164406
2915 015004 013777 001560 164404
2916 015012 004737 005374
2917 015016 104017
2918 015020 104003
2919 015022 017737 164364 001556
2920 015030 104004
2921 015032 000207

:*****
:PRG3 - SINGLE CHARACTER MAINTENANCE MODE DATA TEST.
:*****
PRG3: TYPE ;TYPE PROGRAM TITLE.
P3TIT
JSR 5,LINSLX ;GO GET LINE # FROM USER
TYPE ;TYPE: SELECT CHARACTER.
SELCAR
JSR PC,RDOCT ;GET TEST CHAR AND DELAY FROM USER.
MOV (SP)+,TEMP2 ;STORE TEST CHAR AND DELAY.
PRG3A: MOVB TEMP2+1,CRBUFA ;MOVE DATA CHAR TO CRBUFA.
JSR %7,MOUTIN ;GO OUTPUT, RECEIVE, AND CHECK DATA.
BR PRG3A

:*****
:PRG4 - SPECIAL BINARY COUNT MAINTENANCE MODE DATA TEST.
:*****
PRG4: TYPE ;TYPE PROGRAM TITLE.
P4TIT
JSR 5,LINSLX ;GO GET LINE # FROM USER
JSR %7,INBIN ;INITIALIZE BINARY COUNT.
PRG4A: JSR %7,GTBINP ;GET BINARY CHARACTER.
MOVB %1,CRBUFA ;SAVE AT CRBUFA.
JSR %7,MOUTIN ;GO OUTPUT, RECEIVE, AND CHECK DATA.
BR PRG4A ;REPEAT.
;SUBROUTINE TO OUTPUT, RECEIVE, AND CHECK DATA WITH MAINTENANCE BIT SET.
MOUTIN: BIT #BIT7,JSRPTR ;SEE IF BIT 7 IS SET.
BNE .+4 ;BRANCH IF SET.
STALL ;SET. DO A RANDOM STALL.
TIMETX ;TIME OUT TX DONE
ERROR ;ERROR DONE NOT SETTING
BIS #BIT2,ITXCSR ;SET MAINTENANCE BIT.
TST @RXBUF ;CLR RX DONE
MOV CRBUFA,@TXBUF ;LOAD TXBUF.
JSR 7,MASKIT ;MASK OFF NON TRANSMITTED BITS
TIMERX ;TIME OUT RX DONE
ERROR ;ERROR DONE NOT SETTING
MOV @RXBUF,CRBUF ;MOVE CHAR IN RX BUFFER TO CRBUF.
DATCHK ;COMPARE EXPECTED AND RECEIVED DATA
RTS %7 ;EXIT.

```

0923
0924
0925
0926
0927
0928
0929
0930
0931
0932
0933
0934
0935
0936
0937
0938
0939
0940
0941
0942
0943
0944
0945
0946
0947
0948
0949
0950
0951
0952
0953
0954
0955
0956
0957
0958
0959
0960
0961
0962
0963
0964
0965
0966
0967
0968
0969
0970
0971
0972
0973
0974
0975
0976
0977

015034 042045 030514 026461
015042 026105 027503 020104
015050 043117 020106 044514
015056 042516 052040 051505
015064 020124 020055 040515
015072 047111 042504 026503
015100 030461 042055 042132
015106 040514 042455 045
015113 045 040515 020120
015120 043117 042040 053105
015126 041511 051505 050040
015134 042522 042523 052116
015142 040045
015144 020040 020040 020040
015152 020040 020040 020040
015160 020040 040045
015164 052045 050131 020105
015172 047111 050040 047522
015200 051107 046501 047040
015206 046525 042502 020122
015214 020040 040040
015220 047045 047117 020105
015226 047506 047125 022504
015234 100
015235 045 124
015237 040 020040 050040
015244 036503 040
015247 040 020040 020040
015254 020040 051040 041530
015262 051123 020075
015266 020040 020040 020040
015274 020040 100
015277 045 050045 043522
015304 020060 020055 047111
015312 052520 026524 052517
015320 050124 052125 046040
015326 043517 041511 052040
015334 051505 051524 020056
015342 042045 051511 047503
015350 047116 041505 020124
015356 046104 030461 042455
015364 043040 047522 020115
015372 047515 042504 115
015377 040 047101 020104
015404 047503 047116 041505
015412 020124 052512 050115
015420 051105 052040 020117
015426 040503 046102 027105
015434 040045
015436 054124 051503 020122
015444 027523 035102 040
015451 040 020040 020040
015456 020040 053440 051501

;ASCII MESSAGES

MTIT: .ASCII '%DL11-E,C/D OFF LINE TEST - MAINDEC-11-DZDLA-E%'

.ASCII '%MAP OF DEVICES PRESENT%'

MDEVAD: .ASCII ' %'

PGMSG: .ASCII '%TYPE IN PROGRAM NUMBER %'

MNONE: .ASCII '%NONE FOUND%'

EMD: .ASCII '%T'
ATNUMB: .ASCII '% PC= '

APC: .ASCII '% RXCSR= '

MRXNUM: .ASCII '% %'

POTIT: .ASCII '%PRGO - INPUT-OUTPUT LOGIC TESTS. '

.ASCII '%DISCONNECT DL11-E FROM MODEM'

.ASCII '% AND CONNECT JUMPER TO CABLE.%'

ATXCSR: .ASCII '%TXCSR S/B: '

ATXSB: .ASCII '% WAS: '

G05

.MAIN. MACY11 27(732) 25-FEB-76 10:49 PAGE 58
DZDLAF.P11

2978	015464	020072					
2979	015466	020040	020040	020040	ATXWAS:	.ASCII	' 3'
2980	015474	100					
2981	015475	122	041530	051123	ARXCSR:	.ASCII	'RXCSR S/B: '
2982	015502	051440	041057	020072			
2983	015510	020040	020040	020040	ARXSB:	.ASCII	' WAS: '
2984	015516	020040	040527	035123			
2985	015524	040					
2986	015525	040	020040	020040	ARXWAS:	.ASCII	' 3'
2987	015532	020040	100				
2988	015535	124	020130	050123	ETXTIM:	.ASCII	'TX SPEEDS NOT IN ASCENDING ORDER.3'
2989	015542	042505	051504	047040			
2990	015550	052117	044440	020116			
2991	015556	051501	042503	042116			
2992	015564	047111	020107	051117			
2993	015572	042504	027122	100			
2994	015577	122	020130	050123	ERXTIM:	.ASCII	'RX SPEEDS NOT IN ASCENDING ORDER.3'
2995	015604	042505	051504	047040			
2996	015612	052117	044440	020116			
2997	015620	051501	042503	042116			
2998	015626	047111	020107	051117			
2999	015634	042504	027122	100			
3000	015641	045	050045	043522	PITIT:	.ASCII	'%XPRG1 - TRANSMITTER SCOPE LOOP3'
3001	015646	020061	020055	051124			
3002	015654	047101	046523	052111			
3003	015662	042524	020122	041523			
3004	015670	050117	020105	047514			
3005	015676	050117	100				
3006	015701	045	050045	043522	P2TIT:	.ASCII	'%XPRG2 - RECEIVER SCOPE LOOP3'
3007	015706	020062	020055	042522			
3008	015714	042503	053111	051105			
3009	015722	051440	047503	042520			
3010	015730	046040	047517	040120			
3011	015736	052045	050131	020105	SELCAD:	.ASCII	'%TYPE TEST.CHAR. CODE IN BITS 15-8,TYPE DELAY TIME IN BITS 7-0'
3012	015744	042524	052123	041440			
3013	015752	040510	027122	041440			
3014	015760	042117	020105	047111			
3015	015766	041040	052111	020123			
3016	015774	032461	034055	052054			
3017	016002	050131	020105	042504			
3018	016010	040514	020131	044524			
3019	016016	042515	044440	020116			
3020	016024	044502	051524	033440			
3021	016032	030055					
3022	016034	047445	020106	047101		.ASCII	'%OF AN OCTAL WORD.%3'
3023	016042	047440	052103	046101			
3024	016050	053440	051117	027104			
3025	016056	040045					
3026	016060	040504	040524	051440	ERDAT:	.ASCII	'DATA S/B: '
3027	016066	041057	020072				
3028	016072	020040	020040	040527	AASB:	.ASCII	' WAS: '
3029	016100	035123	040				
3030	016103	040	020040		AWAS:	.ASCII	'
3031	016106	051040	041130	043125		.ASCII	' RXBUF: '
3032	016114	020072					
3033	016116	020040	020040	020040	ARXBUF:	.ASCII	' 3'

3034	016124	040040				
3035	016126	051445	052105	042040	ASETSR: .ASCII	'%SET DESIRED SR OPTIONS. NORMAL OPERATION '
3036	016134	051505	051111	042105		
3037	016142	051440	020122	050117		
3038	016150	044524	047117	027123		
3039	016156	047040	051117	040515		
3040	016164	020114	050117	051105		
3041	016172	052101	047511	020116		
3042	016200	051511	053440	052111	.ASCII	'IS WITH SR = 000000%'
3043	016206	020110	051123	036440		
3044	016214	030040	030060	030060		
3045	016222	022460	100			
3046	016225	045	047111	047503	AINCRT: .ASCII	'%INCORRECT ROUTINE SELECTED, PLACE CORRECT PROGRAM'
3047	016232	051122	041505	020124		
3048	016240	047522	052125	047111		
3049	016246	020105	042523	042514		
3050	016254	052103	042105	020054		
3051	016262	046120	041501	020105		
3052	016270	047503	051122	041505		
3053	016276	020124	051120	043517		
3054	016304	040522	115			
3055	016307	045	047111	051440	.ASCII	'%IN SR 0-2 AND PRESS CONTINUE.'
3056	016314	020122	026460	020062		
3057	016322	047101	020104	051120		
3058	016330	051505	020123	047503		
3059	016336	052116	047111	042525		
3060	016344	040056				
3061	016346	044445	053116	046101	AINCPG: .ASCII	'%INVALID PROGRAM SELECTED.'
3062	016354	042111	050040	047522		
3063	016362	051107	046501	051440		
3064	016370	046105	041505	042524		
3065	016376	027104	100			
3066	016401	207			APGEND: .BYTE	207
3067	016402	042445	042116	050040	.ASCII	'%END PASS = '
3068	016410	051501	020123	020075		
3069	016416	020040	020040	020040	APCNT: .ASCII	' LINE = '
3070	016424	020040	044514	042516		
3071	016432	036440	020040			
3072	016436	020040	020040	054122	ACLIN: .ASCII	' RXCSR = '
3073	016444	051503	020122	020075		
3074	016452	020040	020040	020040	APRXC: .ASCII	' VECTOR = '
3075	016460	020040	042526	052103		
3076	016466	051117	036440	040		
3077	016473	040	020040	020040	APVEC: .ASCII	' '
3078	016500	020040	040040			
3079	016504	022445	051120	031507	P3TIT: .ASCII	'%PRG3-SINGLE CHAR MAINT MODE DATA TESTS'
3080	016512	051455	047111	046107		
3081	016520	020105	044103	051101		
3082	016526	046440	044501	052116		
3083	016534	046440	042117	020105		
3084	016542	040504	040524	052040		
3085	016550	051505	040124			
3086	016554	022445	051120	032107	P4TIT: .ASCII	'%PRG4-SPEC BIN COUNT MAINT MODE DATA TESTS'
3087	016562	051455	042520	020103		
3088	016570	044502	020116	047503		
3089	016576	047125	020124	040515		

.MAIN. MACY11 27(732) 25-FEB-76 10:49 PAGE 60
 DZDLAF.P11

3090	016604	047111	020124	047515	
3091	016612	042504	042040	052101	
3092	016620	020101	042524	052123	
3093	016626	100			
3094	016627	045	054524	042520	SELCAR: .ASCII '%TYPE IN TEST CHAR. CODE. %a'
3095	016634	044440	020116	042524	
3096	016642	052123	041440	040510	
3097	016650	027122	041440	042117	
3098	016656	027105	020040	020040	
3099	016664	040045			
3100	016666	052045	050131	020105	LDLINE: .ASCII '%TYPE IN LINE NO. a'
3101	016674	047111	046040	047111	
3102	016702	020105	047516	020056	
3103	016710	020040	040040		
3104	016714	046045	047111	020105	ALINE: .ASCII '%LINE NO.'
3105	016722	047516	056		
3106	016725	040	020040	040527	SELINE: .ASCII ' WAS SELECTEDa'
3107	016732	020123	042523	042514	
3108	016740	052103	042105	100	
3109	016745	045	042522	042503	MSETRX: .ASCII '%RECEIVER SPEED CHECKa'
3110	016752	053111	051105	051440	
3111	016760	042520	042105	041440	
3112	016766	042510	045503	100	
3113	016773	045	051124	047101	MSETTX: .ASCII '%TRANSMIT SPEED CHECKa'
3114	017000	046523	052111	051440	
3115	017006	042520	042105	041440	
3116	017014	042510	045503	100	
3117	017021	045	042523	020124	MSETC: .ASCII '%SET CLOCK SWITCHES TO POSITION, THEN PRESS CONTINUE.a'
3118	017026	046103	041517	020113	
3119	017034	053523	052111	044103	
3120	017042	051505	052040	020117	
3121	017050	047520	044523	044524	
3122	017056	047117	020054	044124	
3123	017064	047105	050040	042522	
3124	017072	051523	041440	047117	
3125	017100	044524	052516	027105	
3126	017106	100			
3127	017107	045	051105	047522	MTERR: .ASCII '%ERROR - UNEXPECTED TRAP'
3128	017114	020122	020055	047125	
3129	017122	054105	042520	052103	
3130	017130	042105	052040	040522	
3131	017136	120			
3132	017137	045	051124	050101	.ASCII '%TRAPPED TO '
3133	017144	042520	020104	047524	
3134	017152	020040			
3135	017154	020040	020040	020040	MTO: .ASCII ' , '
3136	017162	020040			
3137	017164	052045	040522	050120	.ASCII '%TRAPPED FROM PC '
3138	017172	042105	043040	047522	
3139	017200	020115	041520	020040	
3140	017206	020040	020040	020040	MFROM: .ASCII ' a'
3141	017214	020040	100		
3142	017217	045	047516	042040	MNOLIN: .ASCII '%NO DEVICE PRESENT - THIS LINE NO.a'
3143	017224	053105	041511	020105	
3144	017232	051120	051505	047105	
3145	017240	020124	020055	044124	

3146	017246	051511	046040	047111			
3147	017254	020105	047516	040056			
3148	017262	022477	100		DTERR:	.ASCII	'?%@'
3149	017265	045	047516	044440	INTER:	.ASCII	'%NO INTERRUPT@'
3150	017272	052116	051105	052522			
3151	017300	052120	100				
3152	017303	045	051503	036440	MS0:	.ASCII	'%CS = 0@'
3153	017310	030040	100				
3154	017313	045	051503	036440	MS1:	.ASCII	'%CS = 1@'
3155	017320	030440	100				
3156	017323	045	051503	036440	MS2:	.ASCII	'%CS = 2@'
3157	017330	031040	100				
3158	017333	045	051503	036440	MS3:	.ASCII	'%CS = 3@'
3159	017340	031440	100				
3160	017343	045	051503	036440	MS4:	.ASCII	'%CS = 4@'
3161	017350	032040	100				
3162	017353	045	051503	036440	MS5:	.ASCII	'%CS = 5@'
3163	017360	032440	100				
3164	017363	045	051503	036440	MS6:	.ASCII	'%CS = 6@'
3165	017370	033040	100				
3166	017373	045	051503	036440	MS7:	.ASCII	'%CS = 7@'
3167	017400	033440	100				
3168	017403	045	042522	047503	MPWRF:	.ASCII	'%RECOVERED FROM POWER FAILURE@'
3169	017410	042526	042522	020104			
3170	017416	051106	046517	050040			
3171	017424	053517	051105	043040			
3172	017432	044501	052514	042522			
3173	017440	100					
3174		017442				.EVEN	
3175		000001				.END	

AINCPG	016346	1234	3061#								
AINCRT	016225	1230	3046#								
AJA	007072	1609	1613#								
AJB	007114	1615	1619#								
AJBA	010336	1925	1930#								
AJBB	010364	1933	1936#								
AJBC	010412	1939	1942#								
AJBD	010436	1935	1941	1945		1947#					
AJC	007136	1620	1623#								
AJD	007156	1617	1622	1625		1627#					
AKA	007200	1636	1641#								
AKB	007212	1642	1645#								
AKBA	010450	1954	1959#								
AKBC	010506	1964	1966#								
ALA	007224	1652	1656#								
ALAA	012006	2240	2247#								
ALAB	012036	2252	2255#								
ALAC	012050	2254	2257	2259#							
ALBA	010656	2006	2011#								
ALBB	010704	2014	2017#								
ALBC	010732	2020	2023#								
ALBD	010756	2016	2022	2026		2028#					
ALINE	016714	1100	3104#								
ALY	007254	1659	1662#								
ALZ	007274	1661	1664	1666#							
AMAA	012062	2266	2271#								
AMAC	012114	2277	2279#								
AMBA	010770	2035	2039#								
AMBB	011030	2044	2047#								
AMBC	011044	2048	2051#								
AMBD	011072	2054	2057#								
AMBE	011116	2046	2050	2056		2060		2062#			
AOAA	012132	2286	2292#								
AOAB	012150	2295#	2299								
AOAB1	012164	2296	2298#								
AOAC	012206	2301	2304#								
AOAD	012220	2303	2305	2307#							
AOBA	010544	1978	1982#								
AOBB	010572	1985	1988#								
AOBC	010620	1991	1994#								
AOBD	010644	1987	1993	1997		1999#					
APA	007306	1673	1677#								
APB	007322	1678	1681#								
APBA	014022	2703	2709#								
APBB	014030	2710#	2721								
APC	015247	1341	2952#								
APCNT	016416	1242	3069#								
APCX	007344	1683	1686#								
APD	007364	1680	1685	1688		1690#					
APGEND	016401	1257	3066#								
APRXC	016452	1250	3074#								
APVEC	016473	1254	3077#								
AQA	007406	1699	1703#								
AQAA	012236	2315	2323#								
AQAB	012462	2368	2371#								
AQAS	012464	2330	2335	2340		2345	2350	2355	2360	2365	2373#

AT35	011776	2159	2237#
AT36	012052	2238	2263#
AT37	012116	2264	2283#
AT4	006662	1538	1553#
AT40	012226	2284	2312#
AT41	012530	2313	2387#
AT42	012674	2388	2427#
AT43	012762	2428	2451#
AT44	013040	2452	2473#
AT45	013124	2474	2496#
AT46	013222	2497	2524#
AT47	013306	2525	2548#
AT5	006762	1554	1580#
AT50	013370	2549	2572#
AT51	013452	2573	2595#
AT52	013542	2596	2619#
AT53	013640	2620	2647#
AT54	013700	2648	2664#
AT55	013764	2665	2687#
AT56	014006	2688	2700#
AT57	014100	2701	2726#
AT6	007062	1581	1606#
AT60	014164	2727	2751#
AT61	014234	2752	2772#
AT62	014270	2773	2787#
AT63	014324	2788	2801#
AT7	007170	1607	1633#
AUAA	012776	2454	2461#
AUAB	013022	2465#	2469
AUAC	013032	2460	2467#
AVAA	013054	2476	2483#
AVAB	013112	2482	2490#
AVAC	013114	2489	2491#
AWAA	013134	2499	2504#
AWAB	013164	2511#	2517
AWAC	013174	2505	2513#
AWAD	013210	2514	2516#
AWAE	013214	2513	2518#
AWAS	016103	1310	3030#
AXA	007750	1794	1798#
AXAA	013246	2527	2536#
AXAB	013274	2534	2542#
AXAC	013276	2541	2543#
AXB	007764	1799	1802#
AX1A	013326	2551	2559#
AX1B	013356	2558	2566#
AX1C	013360	2565	2567#
AYA	007776	1809	1813#
AYAA	013410	2575	2583#
AYAB	013434	2587#	2591
AYAC	013444	2581	2589#
AYB	010012	1814	1817#
AZA	010024	1824	1828#
AZAA	013472	2598	2606#
AZAB	013530	2604	2613#
AZAC	013532	2612	2614#

2520

K06

.MAIN. MACY11 27(732) 25-FEB-76 10:49 PAGE 77
DZDLAF.P11 CROSS REFERENCE TABLE -- MACRO NAMES

TSTA	686#	1494	1500	1518	1534	1550	1577	1603	1630	1646	1667	1693	1719	1746	1773
	1788	1803	1818	1833	1854	1877	1919	1948	1972	2000	2029	2063	2091	2129	2155
	2234	2260	2280	2309	2394	2424	2448	2470	2493	2521	2545	2569	2592	2616	2644
	2661	2694	2697	2723	2748	2769	2784								
TSTAA	677#	1485	1501	1519	1535	1551	1578	1604	1631	1647	1668	1694	1720	1747	1774
	1789	1804	1819	1834	1855	1878	1920	1949	1973	2001	2030	2054	2092	2130	2156
	2235	2261	2281	2310	2385	2425	2449	2471	2494	2522	2546	2570	2593	2617	2645
	2662	2685	2698	2724	2749	2770	2785	2799							

ADD	839	876	885	902	906	928	953	968	1002	1016	1058	1142	1145	1193	1194
ASL	1264	1356	1398	1400	1402	1406	2942								
ASR	755	837	998	999	1000	1076	1263	1405	1410	1459					
BCS	731	1284	1364	1365	1366										
BEQ	1190														
	705	725	741	785	800	935	937	970	988	990	1018	1034	1043	1055	1080
	1092	1259	1273	1280	1307	1389	1562	1572	1588	1598	1615	1625	1664	1678	1688
	1704	1714	1731	1741	1758	1768	1784	1799	1814	1829	1844	1874	1892	1906	1933
	1945	1969	1985	1997	2014	2026	2048	2077	2115	2296	2305	2412	2416	2816	
BGT	992														
BHI	2841														
BIC	706	753	775	802	838	983	1033	1042	1074	1087	1093	1123	1132	1144	1147
	1207	1297	1299	1338	1367	1411	1440	1452	1570	1596	1623	1656	1662	1686	1712
	1739	1766	1871	1888	1909	1930	1942	1959	1966	1982	1994	2011	2023	2039	2051
	2103	2127	2438	2443	2445	2462	2465	2483	2491	2506	2511	2536	2543	2559	2567
	2583	2587	2606	2614	2630	2634	2675	2682							
BIS	711	819	1086	1088	1208	1439	1441	1565	1574	1591	1600	1618	1627	1657	1681
	1690	1707	1716	1734	1743	1761	1770	1865	1895	1936	1961	1988	2017	2041	2057
	2105	2247	2271	2292	2419	2440	2463	2486	2508	2538	2561	2585	2609	2631	2677
	2810	2811	2822	2870	2913										
BIT	727	766	784	794	799	804	1064	1079	1238	1258	1279	1282	1304	1333	1418
	1446	1561	1566	1571	1587	1592	1597	1614	1619	1624	1658	1663	1677	1682	1687
	1703	1708	1713	1730	1735	1740	1757	1762	1767	1783	1798	1813	1828	1843	1867
	1873	1891	1900	1904	1914	1932	1938	1944	1963	1968	1984	1990	1996	2013	2019
	2025	2043	2047	2053	2059	2150	2295	2304	2397	2405	2411	2415	2908		
BLE	997														
BLOS	2829	2831	2833	2835	2837	2839									
BLT	994														
BMI	1512	1642	2252	2402	2657	2743									
BNE	697	728	737	767	777	780	795	797	805	807	932	956	1022	1025	1065
	1121	1130	1152	1164	1179	1221	1239	1268	1283	1305	1334	1419	1435	1445	1447
	1476	1567	1593	1620	1659	1683	1709	1736	1763	1868	1901	1915	1939	1964	1991
	2020	2044	2054	2060	2151	2231	2301	2398	2407	2721	2739	2745	2767	2909	
BPL	769	771	789	818	942	981	985	1057	1211	1217	1291	2145	2218	2221	2226
	2257	2277	2299	2375	2381	2763	2779	2794							
BR	646	710	713	722	734	773	782	809	813	939	948	960	971	1003	1007
	1051	1083	1192	1261	1328	1431	1499	1517	1533	1549	1564	1569	1590	1595	1617
	1622	1661	1680	1685	1706	1711	1733	1738	1760	1765	1870	1894	1903	1908	1917
	1935	1941	1987	1993	2016	2022	2046	2050	2056	2147	2211	2254	2303	2368	2447
	2469	2489	2517	2520	2541	2565	2591	2612	2640	2643	2680	2859	2881	2895	2906
CLR	652	673	692	693	701	707	743	747	748	761	764	916	917	918	919
	920	921	922	923	978	979	1019	1053	1066	1085	1188	1237	1270	1271	1327
	1393	1437	1438	1495	2106	2140	2143	2216	2219	2222	2248	2272	2293	2294	2373
	2377	2420	2439	2507	2537	2560	2676	2740	2823						
CMP	649	670	704	736	779	806	811	955	969	996	1267	1306	1434	2076	2114
	2300	2828	2830	2832	2834	2836	2838	2840							
CMPB	776	931	934	936	987	989	991	993	2815						
COM	893	1119	1120	1128	1129	1298									
DEC	796	1021	1024	1151	1163	1178	1220	2230	2720						
EMT	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462
	463	464													
HALT	407	744	803	1227	1231	1235	1292	1380	1424	1479	2172	2177	2182	2187	2192
	2197	2202	2207	2329	2334	2339	2344	2349	2354	2359	2364	2297	2379	2736	2738
INC	700	721	995	1054	1122	1131	1191	1260	1269	1454	2224				
	2744	2761	2766												
JMP	473	475	661	674	742	745	756	772	778	841	1236	1381	1420	1451	1460

N06.

.MAIN. MACY11 27(732) 25-FEB-76 10:49 PAGE 81
DZDLAF.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

.WORD 668

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

*DZDLAF,DZDLAF/SOL/CRF=DZDLAF
RUN-TIME: 10 22 5 SECONDS
RUN-TIME RATIO: 74/37=1.9
CORE USED: 12K (23 PAGES)

