

# DC11

OFF LINE DIAGNOSTIC TEST  
MD-11-DZDCA-B

EP DZDCA B DL A  
COPYRIGHT 1977  
FICHE 1 OF 1

MAR 1977  
**digital**  
MADE IN USA

This section contains a grid of 150 small diagnostic test data tables, arranged in 10 columns and 15 rows. Each table is a small grid of text, likely representing test results for various components or systems. The text is too small to read clearly but appears to be organized into columns and rows within each individual table.

A small diagnostic test data table located in the bottom right corner of the page. It consists of a few lines of text arranged in a grid format, similar to the other tables in the document.

B01

EOF1DZACBSEQ

00010000

770224

PDP10 411

HDR1DZDCABSEQ

00010000

770224

.REM%WSCODE

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDCA-B-D  
PRODUCT NAME: DC11 OFF LINE DIAGNOSTIC TEST  
DATE: JANUARY 1977  
MAINTAINER: DIAGNOSTIC GROUP

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.

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

COPYRIGHT 1970 ,1977 BY DIGITAL EQUIPMENT CORPORATION

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

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

1. ABSTRACT

TWO SEPARATE DIAGNOSTIC PROGRAMS ARE PROVIDED FOR THE DC-11 (ASYNCHRONOUS MODEM INTERFACE), MAINDEC-11-DZDCAA (DC-11 OFF LINE TESTS) AND MAINDEC-11-DZDCBA (DC-11 ON LINE TESTS). THE OFF LINE TESTS TEST ALL DC11 LOGIC AND MAY BE USED TO INDIVIDUALLY TEST UP TO 32 DC-11'S. THE OFF LINE TESTS DO NOT REQUIRE THE USE OF A MODEM, HOWEVER A SPECIAL JUMPER CONNECTOR IS REQUIRED. THE ON LINE TESTS ARE ESSENTIALLY DATA RELIABILITY TESTS REQUIRING THE USE OF MODEMS AND A SUITABLE TERMINAL DEVICE.

THIS DOCUMENT DESCRIBES THE OFF LINE TESTS.

THE AVAILABLE TESTS ARE:

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

2. REQUIREMENTS

2.1 EQUIPMENT

- A. PDP 11/20 SYSTEM
- B. DC11
- C. SPECIAL JUMPER CONNECTOR (SEE DC11 MAINTENANCE MANUAL FOR DETAILED DESCRIPTION)

2.2 STORAGE

THIS PROGRAM USES ALL OF CORE (4K) EXCEPT THAT AREA RESERVED FOR THE BOOTSTRAP AND ABSOLUTE LOADERS.

3. LOADING PROCEDURE

THE ABSOLUTE LOADER IS USED TO LOAD THE PROGRAM.

94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144

4. USE PROCEDURE  
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, I.E. SR 0-2=0-4, OTHERWISE PROGRAM OP-  
ERATION IS UNSPECIFIED. RELOAD PROGRAM AND START OVER.
- 4.1 PRGD INPUT/OUTPUT LOGIC TESTS
- A. LOAD ADDRESS = 000200 (RESTART LOAD ADDR. = 000200)  
LOAD SR 0-2 = 0, AND PRESS START SWITCH.  
THE DIAGNOSTIC WILL IDENTIFY THE PROGRAM YOU SELECTED.  
'PRGD-INPUT-OUTPUT LOGIC TESTS. DISCONNECT DC11 FROM MODEM  
AND CONNECT JUMPER TO CABLE'  
DISCONNECT THE DC11 FROM THE MODEM AND INSERT THE JUMPER CON-  
NECTOR IN THE MODEM END OF THE CABLE, AND PRESS CONTINUE.  
NOTE, IF THE CABLE IS LEFT CONNECTED TO THE MODEM THE FOL-  
LOWING TESTS WILL FAIL:  
AT5, AT34-AT42, AT44, AND AT144
- B. THE PROGRAM WILL NOW REQUEST THE LINE # (8) YOU WISH TO  
TEST. LOAD THE LINE # AS REQUESTED AND PRESS CONTINUE.  
LINE NUMBER REFERS TO THE ADDRESSES TO WHICH THE DC11 RESPONDS.
- |        |        |         |        |         |        |         |        |
|--------|--------|---------|--------|---------|--------|---------|--------|
| LINE 0 | 77400X | LINE 10 | 77410X | LINE 20 | 77420X | LINE 30 | 77430X |
| LINE 1 | 77401X | LINE 11 | 77411X | LINE 21 | 77421X | LINE 31 | 77431X |
| LINE 2 | 77402X | LINE 12 | 77412X | LINE 22 | 77422X | LINE 32 | 77432X |
| LINE 3 | 77403X | LINE 13 | 77413X | LINE 23 | 77423X | LINE 33 | 77433X |
| LINE 4 | 77404X | LINE 14 | 77414X | LINE 24 | 77424X | LINE 34 | 77434X |
| LINE 5 | 77405X | LINE 15 | 77415X | LINE 25 | 77425X | LINE 35 | 77435X |
| LINE 6 | 77406X | LINE 16 | 77416X | LINE 26 | 77426X | LINE 36 | 77436X |
| LINE 7 | 77407X | LINE 17 | 77417X | LINE 27 | 77427X | LINE 37 | 77437X |
- C. 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)  
SR8 DISABLE STALL MODE  
SR9 LOOP SELECTED ROUTINE  
SR10 HALT AT END OF PROGRAM  
SR11 INHIBIT ITERATION  
SR13 INHIBIT PRINTOUT  
SR14 SCOPE  
SR15 HALT ON ERROR.
- D. THE PROGRAM WILL NOW BEGIN TESTING THE DC11 YOU SELECTED.
- E. REFER TO SECTION 5.1.2 FOR ERROR DESCRIPTION
- F. AFTER ONE COMPLETE PASS PRGEND WILL BE TYPED OUT

145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183

## 4.2 PRG1 - TRANSMITTER SCOPE LOOP

- A. LOAD ADDRESS = 000200  
LOAD SR 0-2 = 1, AND PRESS START SWITCH.  
THE DIAGNOSTIC WILL IDENTIFY THE PROGRAM YOU SELECTED, AND  
REQUEST THE LINE # YOU WISH TO TEST. LOAD THE LINE # AS RE-  
QUESTED AND PRESS CONTINUE.
- B.  
THE PROGRAM WILL REQUEST THE DC11 LINE PARAMETERS. LOAD THE  
PARAMETERS AS REQUESTED AND PRESS CONTINUE.
- C. THE PROGRAM WILL REQUEST A CHARACTER CODE, AND A DELAY  
TIME. THE CHARACTER CODE IS THE DATA THE DC11 WILL TRANSMIT  
AND THE DELAY IS THE TIME ELAPSED BETWEEN SUCCESSIVE TRANS-  
MISSIONS OF ONE CHARACTER. PRESS CONTINUE WHEN THIS IS DONE.
- D. THE PROGRAM WILL RUN WITHOUT ERROR OR END TYPEOUTS.

## 4.3 PRG2 - RECEIVER SCOPE LOOP

- A. LOAD ADDRESS = 000200  
LOAD SR 0-2 = 2, AND PRESS START.  
THE DIAGNOSTIC WILL IDENTIFY THE PROGRAM YOU SELECTED, AND  
REQUEST THE LINE # YOU WISH TO TEST. LOAD THE LINE # AS REQ-  
UESTED AND PRESS CONTINUE.
- B. THE PROGRAM WILL REQUEST THE DC11 LINE PARAMETERS. LOAD THE  
PARAMETERS AS REQUESTED AND PRESS CONTINUE.
- C. THE PROGRAM WILL REQUEST A TEST CHARACTER CODE, AND A DELAY  
TIME. THE CHARACTER CODE IS THE DATA THAT THE DC11 WILL BE  
TRANSMITTING AND THE DELAY IS THE ELAPSED TIME BETWEEN SUCCES-  
SIVE CHARACTERS. PRESS CONTINUE WHEN THIS IS DONE.
- D. THE PROGRAM WILL NOW RUN WITHOUT ERROR OR END TYPEOUTS.

184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226

## 4.4 PRG3 - SINGLE CHARACTER MAINT MODE DATA TEST

- A. LOAD ADDRESS = 000200  
LOAD SR 0-2 = 3, AND PRESS START.  
THE DIAGNOSTIC WILL IDENTIFY THE PROGRAM YOU SELECTED, AND  
REQUEST THE LINE # YOU WISH TO TEST. LOAD THE LINE # AS REQ-  
UESTED AND PRESS CONTINUE.
- B. THE PROGRAM WILL REQUEST THE DC11 LINE PARAMETERS. LOAD THE  
PARAMETERS AS REQUESTED AND PRESS CONTINUE.
- C. THE PROGRAM WILL REQUEST A TEST CHARACTER. LOAD THE TEST CHAR-  
ACTER AND PRESS CONTINUE.
- D. THE PROGRAM WILL NOW RUN CONTINUOUSLY REPORTING ANY DATA FAIL-  
URES.

## 4.5 PRG4 - SPECIAL BINARY COUNT MAINT. MODE DATA TEST

- A. LOAD ADDRESS = 000200  
LOAD SR 0-2 = 4, AND PRESS START.  
THE DIAGNOSTIC WILL IDENTIFY THE PROGRAM YOU SELECTED, AND  
REQUEST THE LINE # YOU WISH TO TEST. LOAD THE LINE # AS REQ-  
UESTED AND PRESS CONTINUE.
- B. THE PROGRAM WILL NOW REQUEST THE DC11 LINE PARAMETERS. LOAD THE  
PARAMETERS AS REQUESTED AND PRESS CONTINUE.
- C. THE PROGRAM WILL BEGIN TESTING THE LINE YOU SELECTED.  
AND REPORT ANY DATA ERRORS.

## 5. PROGRAM DESCRIPTIONS

## 5.1 PRG0 - INPUT/OUTPUT LOGIC TESTS

THE INPUT/OUTPUT LOGIC TESTS CONSIST OF 103(8) ROUTINES WHICH  
MAY BE RUN IN SEQUENTIAL ORDER OR INDIVIDUALLY LOOPED (SEE  
SECT 4.1, C FOR SWITCH SETTINGS). THE JUMPER CONNECTOR MUST  
BE INSERTED BEFORE STARTING.

227  
 228  
 229  
 230  
 231  
 232  
 233  
 234  
 235  
 236  
 237  
 238  
 239  
 240  
 241  
 242  
 243  
 244  
 245  
 246  
 247  
 248  
 249  
 250  
 251  
 252  
 253  
 254  
 255  
 256  
 257  
 258  
 259  
 260

5.1.1 ROUTINE DESCRIPTIONS

ROUTINE	TESTS
AT0-AT3 AT4-AT42	ADDRESSABILITY OF CSRS & DBRS DIDDLES ALL BITS IN THE CSRS AND CHECKS THAT THEY CAN BE READ/WITTEN PROPERLY.
AT43-AT44 AT45-AT51 AT52-AT63	PROPER OPERATION OF RESET INSTRUCTION PROPER OPERATION OF READY BIT PROPER OPERATION OF CHAR LENGTH, SPEED CONTROL AND STOP CODE BITS.
AT64	PROPER OPERATION OF DATA OVERFLOW BIT
AT65-AT74	PROPER OPERATION OF INTERRUPTS
AT75	DATA OVERFLOW CLEARS DONE
AT76	ERROR CAUSES INTERRUPT
AT77	PROPER OPERATION OF PARITY BIT
AT100-AT137	DATA TESTS THESE TESTS TEST ALL POSSIBLE COMBINATIONS OF CHARACTER LENGTH SPEED AND STOP CODES USING MAINT. MODE.
AT140	DATA TEST HIGH SPEED (JUMPER)
AT141	PROPER OPERATION OF BREAK BIT

## 5.1.2 ERROR DESCRIPTION

IF A ROUTINE FAILS AND THE INHIBIT PRINTOUT SWITCH IS NOT ENABLED (SR13) A PRINTOUT RESULTS. THE PRINTOUT FORMAT IS:

P(PROG#) T(ROUTINE#) PC=(PC OF ERROR CALL) AND AN ADDITIONAL/MESSAGE (IF APPLICABLE)

P00 T005 PC=XXXX INDICATING THAT TXCSR BIT 1 WAS SET (SHOULD'VE BEEN CLEAR)

P00 T122 PC=XXXX DATA ERR S/B:---WAS---  
INDICATING A DATA ERROR  
WHEN DC11 PARAMETERS  
WERE SET AT CHAR LENGTH=6  
SPEED=00, AND STOP CODE=1

TO RESUME TESTING PRESS CONTINUE. IF ROUTINES 65 OR 71 FAIL DUE TO A BAD TRAP VECTOR, I.E. THE VECTOR PROVIDED BY THE INTERRUPTING DC11 IS INCORRECT THE PROGRAM WILL HALT AND DISPLAY THE VECTOR+2 THAT WAS PROVIDED BY THE INTERRUPTING DC11. TO RECOVER FROM THIS TYPE OF ERROR IT WILL BE NECESSARY TO PUT INTO THE INCORRECT VECTOR ADDRESS THE ADDRESS TO RUN THE ROUTINE. I.E. ADDRESS ATAA AND AXAA FOR ROUTINES 65 AND 71 RESPECTIVELY.

## 5.1.3 JUMPER CONNECTOR

THE JUMPER CONNECTOR TESTS THOSE F/F'S, GATES (RING INDICATOR, CARRIER TRANSITION, CLEAR TO SEND, AND SUPERVISORY RECEIVE DATA) WHICH CANNOT BE TESTED UNLESS A DATA SET IS ACTUALLY CONNECTED TO THE DC11. IN ADDITION TO TESTING DC11 LOGIC THE JUMPER ALSO TESTS CABLE WIRING TO/FROM THE DC11/DATA SET. THE FOLLOWING TESTS WILL FAIL IF THE CABLE IS NOT INSTALLED IN THE DC11:

AT5, AT34-AT42, AT44  
AT140 WILL LOOP CONTINUOUSLY

IF THE JUMPER IS REMOVED FROM THE END OF THE CABLE AND THE CABLE IS LEFT CONNECTED TO THE DC11 THE ABOVE TESTS WILL FAIL WITH THE PROBABLE EXCEPTIONS OF AT35 AND AT36.

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

306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350

## 5.2 PRG1-TRANSMITTER SCOPE LOOP

THE PURPOSE OF PRG1 IS TO ALLOW SCOPING OF TRANSMITTER FUNCTIONS IN A RUN CONDITION USING USER SPECIFIED DC11 PARAMETERS AND DATA. NO ERROR PRINTOUTS ARE PROVIDED.

## 5.3 PRG2-RECEIVER SCOPE LOOP

THE PURPOSE OF PRG2 IS TO ALLOW SCOPING OF RECEIVER FUNCTIONS IN A RUN CONDITION USING USER SPECIFIED DC11 PARAMETERS AND DATA. NO ERROR PRINTOUTS ARE PROVIDED.

## 5.4 PRG3-SINGLE CHARACTER MAINT MODE DATA TEST

PRG3 TRANSMITS, RECEIVES AND CHECKS RECEIVED DATA USING USER SPECIFIED DC11, DC11 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 77 AND 100 EXCEPT THAT THE USER SPECIFIES DC11 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.

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



L01

407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458

: COPYRIGHT 1971, DIGITAL EQUIPMENT CORP., MAYNARD, MASS.  
: 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- INHIBIT TRACE  
: SR11- INHIBIT ITERATION.  
: SR10- HALT AT END OF PROGRAM  
: SR9- SELECT ROUTINE.  
: SR8- DISABLE STALL MODE AND RUN FULL SPEED.  
: SR5 THROUGH SR0 - NUMBER OF ROUTINE TO BE SELECTED.  
: DATA TEST PARAMETERS

	CHAR LENGTH	SPEED	STOP CODE
: NOTE0	8	00	22222222
: NOTE1	7	00	
: NOTE2	6	00	
: NOTE3	5	00	
: NOTE4	8	01	
: NOTE5	7	01	
: NOTE6	6	01	
: NOTE7	5	01	
: NOTE10	8	10	
: NOTE11	7	10	
: NOTE12	6	10	
: NOTE13	5	10	
: NOTE14	8	11	
: NOTE15	7	11	
: NOTE16	6	11	
: NOTE17	5	11	
: NOTE20	8	03	11111111
: NOTE21	7	00	11111111
: NOTE22	6	00	11111111
: NOTE23	5	00	11111111
: NOTE24	8	01	11111111
: NOTE25	7	01	11111111
: NOTE26	6	01	11111111
: NOTE27	5	01	11111111
: NOTE30	8	10	11111111
: NOTE31	7	10	11111111
: NOTE32	6	10	11111111
: NOTE33	5	10	11111111
: NOTE34	8	11	11111111
: NOTE35	7	11	11111111
: NOTE36	6	11	11111111
: NOTE37	5	11	11111111

459		000000	. = 0	
460	000000	000002	. + 2	; UNASSIGNED TRAP
461	000002	000000	HALT	
462	000004	000006	. + 2	; SP OVERFLOW, BUS ERROR TRAP
463	000006	000000	HALT	
464	000010	000012	. + 2	; RESERVED INSTRUCTION TRAP
465	000012	000000	HALT	
466	000014	000016	. + 2	; TRACE TRAP
467	000016	000000	HALT	
468	000020	000022	. + 2	; TRAP TO CALL IOX
469	000022	000002	. 2	
470	000024	000026	. + 2	; POWER FAIL TRAP
471	000026	000000	HALT	
472	000030	002106	EMTINT	; EMT TRAP
473	000032	000340	PRTY7	
474	000034	000036	. + 2	
475	000036	000000	HALT	
476	000040	000042	. + 2	
477	000042	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
478	000044	000046	. + 2	
479	000046	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
480	000050	000052	. + 2	
481	000052	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
482	000054	000056	. + 2	
483	000056	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
484	000060	000062	. + 2	
485	000062	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
486	000064	000066	. + 2	
487	000066	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
488	000070	000072	. + 2	
489	000072	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
490	000074	000076	. + 2	
491	000076	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
492	000100	000102	. + 2	
493	000102	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
494	000104	000106	. + 2	
495	000106	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
496	000110	000112	. + 2	
497	000112	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
498	000114	000116	. + 2	
499	000116	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
500	000120	000122	. + 2	
501	000122	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
502	000124	000126	. + 2	
503	000126	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
504	000130	000132	. + 2	
505	000132	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
506	000134	000136	. + 2	
507	000136	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
508	000140	000142	. + 2	
509	000142	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
510	000144	000146	. + 2	
511	000146	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
512	000150	000152	. + 2	
513	000152	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
514	000154	000156	. + 2	

MACHER:

NO1

515	000156	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
516	000160	000162	.+2	
517	000162	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
518	000164	000166	.+2	
519	000166	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
520	000170	000172	.+2	
521	000172	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
522	000174	000176	.+2	
523	000176	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
524	000200	000202	.+2	
525	000202	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
526	000204	000206	.+2	
527	000206	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
528	000210	000212	.+2	
529	000212	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
530	000214	000216	.+2	
531	000216	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
532	000220	000222	.+2	
533	000222	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
534	000224	000226	.+2	
535	000226	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
536	000230	000232	.+2	
537	000232	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
538	000234	000236	.+2	
539	000236	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
540	000240	000242	.+2	
541	000242	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
542	000244	000246	.+2	
543	000246	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
544	000250	000252	.+2	
545	000252	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.

546				
547	000254	000256	.+2	
548	000256	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
549	000260	000262	.+2	
550	000262	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
551	000264	000266	.+2	
552	000266	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
553	000270	000272	.+2	
554	000272	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
555	000274	000276	.+2	
556	000276	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
557	000300	000302	.+2	
558	000302	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
559	000304	000306	.+2	
560	000306	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
561	000310	000312	.+2	
562	000312	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
563	000314	000316	.+2	
564	000316	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
565	000320	000322	.+2	
566	000322	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
567	000324	000326	.+2	
568	000326	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
569	000330	000332	.+2	
570	000332	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
571	000334	000336	.+2	
572	000336	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
573	000340	000342	.+2	
574	000342	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
575	000344	000346	.+2	
576	000346	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
577	000350	000352	.+2	
578	000352	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
579	000354	000356	.+2	
580	000356	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
581	000360	000362	.+2	
582	000362	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
583	000364	000366	.+2	
584	000366	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
585	000370	000372	.+2	
586	000372	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
587	000374	000376	.+2	
588	000376	000000	HALT	; TRAPPED TO PREVIOUS ADDRESS.
589				
590				
591		177570		
592		177776		
593		177776		
594		001076		
595		000240		
596		000000		
597		100000		
598		100000		
599		040000		
600		020000		
601		010000		

```

;EQUATE STATEMENTS
SR=177570
CC=177776
PSW=177776
SPBOT=1076
NOP=240
OPEN=0
MANUAL=BIT15
BIT15=100000
BIT14=40000
BIT13=20000
BIT12=10000
    
```

602	004000	BIT11=4000
603	002000	BIT10=2000
604	001000	BIT9=1000
605	000400	BIT8=400
606	000200	BIT7=200
607	000100	BIT6=100
608	000040	BIT5=40
609	000020	BIT4=20
610	000010	BIT3=10
611	000004	BIT2=4
612	000002	BIT1=2
613	000001	BIT0=1
614	005726	POPSP=5726
615	022626	POPSP2=022626
616	000340	PRTY7=340
617	000300	PRTY6=300
618	000240	PRTY5=240
619	000200	PRTY4=200
620	000140	PRTY3=140
621	000100	PRTY2=100
622	000040	PRTY1=40
623	000000	PRTY0=0
624	104000	TYPE=EMT+0
625	104001	TYPES=EMT+1
626	104002	STALL=EMT+2
627	104003	ERROR=EMT+3
628	104004	DATCHK=EMT+4
629	104005	CHALT=EMT+5
630	104006	STRXV=EMT+6
631	104007	STTXV=EMT+7
632	104010	EHALT=EMT+10
633	104011	SRESET=EMT+11
634	104012	SCOPE=EMT+12
635	104013	SAVREG=EMT+13
636	104014	RSTREG=EMT+14
637	104015	ERROR1=EMT+15
638	104016	DELAY=EMT+16
639	000007	BELL=007
640	177777	ATLAST=-1

```

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

```

641						
642		000200		.=200		
643	000200	000167	001356	JMP	START	;GO TO START OF PROGRAM.
644		001100		.=1100		
645	001100	174000		RXCSR:		;RECEIVER CSR
646	001102	174002		RXBUF:		;RECEIVER BUFFER
647	001104	174004		TXCSR:		;TRANSMITTER CSR
648	001106	174006		TXBUF:		;TRANSMITTER BUFFER
649	001110	000000		RXVTR:	OPEN	;RECEIVER VECTOR
650	001112	000240		RXLVL:	PRTY5	;RECEIVER PRIORITY LEVEL
651	001114	000304		TXVTR:	304	;TRANSMITTER VECTOR
652	001116	000240		TXLVL:	PRTY5	;TRANSMITTER PRIORITY LEVEL
653	001120	177560		TKS:	177560	;LSR CSR
654	001122	177562		TKB:	177562	;LSR BUFFER
655	001124	177564		TPS:	177564	;LSP CSR
656	001126	177566		TPB:	177566	;LSP BUFFER
657	001130	000060		TKVTR:	60	;LSR INTERRUPT VECTOR
658	001132	000200		TKLVL:	PRTY4	;LSR PRIORITY LEVEL
659	001134	000064		TPVTR:	64	;LSP INTERRUPT VECTOR
660	001136	000200		TPLVL:	PRTY4	;LSP PRIORITY LEVEL
661	001140	000000		PRGNUM:	OPEN	;CONTAINS CURRENT PROGRAM#
662	001142	000000		KSTART:	OPEN	;CURRENT PROGRAM START ADDRESS.
663	001144	000000		CURTST:	OPEN	;CONTAINS ADDR OF CURRENT TEST.
664	001146	000000		RTNNO:	OPEN	;CONTAINS CURRENT TEST #.
665	001150	000000		NXTST:	OPEN	;CONTAINS ADDR OF NEXT TEST.
666	001152	000000		ICTR:	OPEN	;CONTAINS CURRENT ITERATION COUNT
667	001154	000000		SCOPTR:	OPEN	;CONTAINS CURRENT SCOPE POINTER.
668	001156	003654		PRGTAB:	PRG0	;PRG0 START ADDRESS
669	001160	014344			PRG1	;PRG1 START ADDRESS
670	001162	014410			PRG2	;PRG2 START ADDRESS
671	001164	014504			PRG3	;PRG3 START ADDRESS
672	001166	014542			PRG4	;PRG4 START ADDRESS
673	001170	002410		EMTTAB:	TYP	;POINTER TO TYPEOUT ROUTINE
674	001172	002532			TYP5	;POINTER TO CHAINED MESSAGES ROUTINE
675	001174	002664			STAL	;POINTER TO RANDOM STALL ROUTINE
676	001176	001406			ERR	;POINTER TO ERROR ROUTINE
677	001200	001344			DTCHK	
678	001202	000000			OPEN	
679	001204	002240			STLSRV	
680	001206	002270			STLSPV	
681	001210	001332			EHLT	;POINTER TO ERROR HALT ROUTINE.
682	001212	002320			SRSETT	
683	001214	001752			CHAINN	
684	001216	002140			SAVRG	
685	001220	002200			RSTRG	
686	001222	001430			ERR1	
687	001224	002616			DLY	
688						
689						
690						

```

691
692 001226 000000 RCNT: OPEN
693 001230 000000 CRBUF: OPEN
694 001232 000000 CRBUFA: OPEN
695 001234 000000 CARMSK: OPEN
696 001236 000000 CHR1: OPEN
697 001240 000000 CHR2: OPEN
698 001242 000000 CHR3: OPEN
699 001244 000000 ERCTR: OPEN
700 001246 000000 CTRA: OPEN
701 001250 000000 CTRB: OPEN
702 001252 000000 CTCR: OPEN
703 001254 000000 CTRD: OPEN
704 001256 000000 TXCSRT: OPEN
705 001260 000000 RXCSRT: OPEN
706 001262 000000 RXBUFT: OPEN
707 001264 000000 TEMP: OPEN
708 001266 000000 SRT: OPEN
709 001270 177740 STLMSK: 177740
710 001272 104000 SETSR: TYPE ;TYPE SELECT OPTION MESSAGE.
711 001274 016355 ASETSR ;COMMON HALT.
712 001276 000000 HALT ;EXIT.
713 001300 000207 RTS %7 ;TYPE INCORRECT ROUTINE SELECTED.
714 001302 104000 INCRTN: TYPE
715 001304 016453 AINCRT
716 001306 000000 HALT ;COMMON HALT.
717 001310 000207 RTS %7 ;EXIT.
718 001312 104000 PRGEND: TYPE ;TYPE PROGRAM END.
719 001314 016510 APGEND
720 001316 032767 002000 176244 BIT #BIT10,SR ;TEST END OF PROGRAM HALT OPTION
721 001324 001401 BEQ .+4 ;BRANCH IF NOT SELECTED
722 001326 000000 HALT
723 001330 000207 RTS %7 ;EXIT.
724
725 ;CONDITIONAL ERROR HALT ROUTINE.
726 001332 005767 176232 EHLT: TST SR ;CHECK FOR HALT ON ERROR.
727 001336 100001 BPL EHLTA ;BRANCH IF NO HALT DESIRED.
728 001340 000000 HALT ;HALT.
729 001342 000002 EHLTA: RTI ;IN DATA LIGHTS.
730
731 ;DATA CHECK ROUTINE.
732 001344 026767 177660 177660 DTCHK: CMP CRBUF,CRBUFA ;COMPARE EXPECTED AND RECEIVED
733 001352 001414 BEQ DTCHKA ;CHARS. BRANCH IF SAME.
734 001354 004567 001652 JSR %5,OACNV ;GO TO OCTAL TO ASCII CONVERT.
735 001360 001230 CRBUF ;SOURCE ADDR.
736 001362 016346 AWAS ;DESTINATION ADDR.
737 001364 000003 3 ;#OF DIGITS TO CONVERT.
738 001366 004567 001640 JSR %5,OACNV ;GO TO OCTAL TO ASCII CONVERT.
739 001372 001232 CRBUFA ;SOURCE ADDR.
740 001374 016334 AASB ;DESTINATION ADDR.
741 001376 000003 3 ;#OF DIGITS TO CONVERT.
742 001400 104015 ERROR1
743 001402 016313 ERDAT
744 001404 000002 DTCHKA: RTI ;EXIT.
745
746

```

747	001406	012767	177777	000126	ERR:	MOV	#-1,ERRB	;SET UP ONE MESSAGE CALL.
748	001414	012767	000240	000122		MOV	#240,ERRB+2	
749	001422	005067	000132			CLR	ERRE	
750	001426	000413				BR	ERRA	
751	001430	011667	000106		ERR1:	MOV	@%6,ERRB	;DEVELOP ADDT'L MESSAGE ADDR.
752	001434	017767	000102	000100		MOV	@ERRB,ERRB	;STORE AT ERRB.
753	001442	012767	177777	000074		MOV	#-1,ERRB+2	
754	001450	012767	000002	000102		MOV	#2,ERRE	
755	001456	032767	020000	176104	ERRA:	BIT	#BIT13,SR	;INHIBIT ERROR PRINT?
756	001464	001030				BNE	ERRC	;BRANCH TO INHIBIT PRINT.
757	001466	011667	000064			MOV	@%6,ERRD	;DEVELOP CALLING ADDR.
758	001472	162767	000002	000056		SUB	#2,ERRD	
759	001500	004567	001526			JSR	%5,OACNV	;GO TO OCTAL TO ASCII CONVERT.
760	001504	001556				ERRD		;SOURCE ADDR.
761	001506	015222				APC		;DESTINATION ADDR.
762	001510	000006				6		;#OF DIGITS TO CONVERT.
763	001512	004567	001514			JSR	%5,OACNV	;GO TO OCTAL TO ASCII CONVERT.
764	001516	001140				PRGNUM		;SOURCE ADDR.
765	001520	015205				APNUMB		;DESTINATION ADDR.
766	001522	000002				2		;#OF DIGITS TO CONVERT.
767	001524	004567	001502			JSR	%5,OACNV	;GO TO OCTAL TO ASCII CONVERT.
768	001530	001146				RTNNO		;SOURCE ADDR.
769	001532	015213				ATNUMB		;DESTINATION ADDR.
770	001534	000003				3		;#OF DIGITS TO CONVERT.
771	001536	104001				TYPES		;TYPE:
772	001540	015202				EMO		;ERROR HEADER,
773	001542	000000			ERRB:	OPEN		;ADDT'L ERROR MESSAGE IF ANY.
774	001544	177777				-1		
775	001546	104010			ERRC:	EHALT		;GO ERR HALT IF DESIRED.
776	001550	066716	000004			ADD	ERRE,@%6	
777	001554	000002				RTI		;EXIT.
778	001556	000000			ERRD:	OPEN		
779	001560	000000			ERRE:	OPEN		

780											
781	001562	012706	001076		START:	MOV	#SPBOT,%6				;SET BOTTOM OF SP STACK.
782	001566	005067	176204			CLR	PSW				
783	001572	012767	000006	176204		MOV	#6,MACHER				
784	001600	005067	177342			CLR	RTNNO				
785	001604	016700	175760			MOV	SR,%0				;(SR) TO RD
786	001610	042700	177770			BIC	#177770,%0				;LIMIT (SR) TO BITS 3-0
787	001614	010067	177320			MOV	%0,PRGNUM				;SAVE PROGRAM #
788	001620	006300				ASL	%0				
789	001622	012767	003576	176174		MOV	#PFAIL,24				
790	001630	012767	000340	176170		MOV	#PTY7,26				
791	001636	000170	001156			JMP	@PRGTAB(0)				;GO TO SELECTED PROGRAM.
792	001642	016767	177274	177300	GETRDY:	MOV	KSTART,NXTST				;ADDR OF 1ST ROUTINE TO NXTST
793	001650	012767	000006	176126	GTRDYX:	MOV	#6,MACHER				;RESET MACHER TRAP.
794	001656	005067	176114			CLR	PSW				
795	001662	012706	001076			MOV	#SPBOT,%6				;SET BOTTOM OF STACK.
796	001666	104011				SRESET					;ISSUE RESET.
797	001670	004767	000142		GTRDYA:	JSR	%7,FORWD				;ROLL FORWARD TO "NEXT" ROUTINE.
798	001674	032767	001000	175666	GTRDYB:	BIT	#BIT9,SR				;CHECK SELECT ROUTINE SWITCH
799	001702	001003				BNE	GTRDYC				;BRANCH IF SELECT ROUTINE SWITCH IS SET.
800	001704	000177	177234			JMP	@CURTST				;GO RUN CURRENT ROUTINE.
801	001710	000437				BR	CHNB				;NO GO. MANUAL RTN BYPASSED.
802	001712	016700	175652		GTRDYC:	MOV	SR,%0				;(SR) TO RD
803	001716	042700	177600			BIC	#177600,%0				;MASK UNDESIRED BITS
804	001722	126700	177220			CMPB	RTNNO,%0				;COMPARE RTNNO TO (RD)
805	001726	001002				BNE	GTRDYD				;BRANCH IF ROUTINE NOT FOUND YET.
806	001730	000177	177210			JMP	@CURTST				;GO RUN ROUTINE.
807	001734	022767	177777	177206	GTRDYD:	CMP	#-1,NXTST				;NO. CHECK FOR LAST ROUTINE.
808	001742	001352				BNE	GTRDYA				;BRANCH IF NOT LAST ROUTINE.
809	001744	004767	177332			JSR	%7,INCRTN				;YES. INCORRECT ROUTINE SELECTED.
810	001750	000734				BR	GETRDY				;START OVER.
811	001752	032767	040000	175610	CHAINN:	BIT	#BIT14,SR				;CHECK FOR SCOPE OPTION.
812	001760	001403				BEQ	CHNA				;BRANCH IF SCOPE SW NOT SET.
813	001762	016716	177166		CHNAB:	MOV	SCOPTR,%6				;SET UP TO RETURN TO ROUTINE.
814	001766	000002				RTI					;RETURN TO ROUTINE.
815	001770	032767	004000	175572	CHNA:	BIT	#BIT11,SR				;TEST INHIBIT ITERATION SWITCH
816	001776	001003				BNE	CHNAA				;BRANCH IF INHIBIT ITERATION SW SET.
817	002000	005367	177146			DEC	ICTR				;DECREMENT ITERATION COUNT.
818	002004	001366				BNE	CHNAB				;BRANCH IF COUNT NOT 0.
819	002006	022626			CHNAA:	POPSP2					;POP STACK TWICE
820											
821	002010	032767	001000	175552	CHNB:	BIT	#BIT9,SR				;CHECK SELECT ROUTINE SWITCH
822	002016	001311				BNE	GETRDY				;BRANCH IF SELECT RTN SW SET
823	002020	022767	177777	177122		CMP	#-1,NXTST				;LAST TEST?
824	002026	001310				BNE	GTRDYX				;BRANCH IF NOT LAST TEST.
825	002030	004767	177256			JSR	%7,PRGEND				;PROGRAM END.
826	002034	000702				BR	GETRDY				
827	002036	016705	177106		FORWD:	MOV	NXTST,%5				;ADDR OF NEXT ROUTINE TO R5.
828	002042	012567	177100			MOV	(5)+,RTNNO				;GET NEXT ROUTINE NUMBER.
829	002046	012567	177076			MOV	(5)+,NXTST				;GET ADDR OF NEXT "NEXT" ROUTINE.
830	002052	012567	177074			MOV	(5)+,ICTR				;GET ITERATION COUNT.
831	002056	012567	177072			MOV	(5)+,SCOPTR				;GET SCOPE LOOP ENTRY POINTER.
832	002062	010567	177056		FORWDA:	MOV	%5,CURTST				;ADDR OF NOW CURRENT TEST TO CURTST.
833	002066	000207				RTS	%7				;EXIT FORWD SUBROUTINE.
834	002070	012767	177777	177056	FORWDB:	MOV	#-1,SCOPTR				;FORCE "NO SCOPE"
835	002076	012767	000001	177046		MOV	#1,ICTR				;FORCE I COUNT OF 1

836	002104	000766			BR	FORWDA		
837	002106	011646			EMTINT: MOV	Q%6, -(6)		;GET SAVED PC.
838	002110	162716	000002		SUB	#2, Q%6		;DECREMENT PC BY 2.
839	002114	017616	000000		MOV	Q(6), Q%6		
840	002120	006316			EMTA: ASL	Q%6		;EMT ARG X 2.
841	002122	042716	177001		BIC	#177001, Q%6		;REMOVE 7 MSB.
842	002126	062716	001170		ADD	#EMTTAB, Q%6		;FORM EMT RTN ADDR.
843	002132	017616	000000		MOV	Q(6), Q%6		
844	002136	000136			JMP	Q(6)+		;GO TO EMT ROUTINE.
845								
846								
847	002140	012667	000030		:SAVE REGS 0 TO 4 SUBROUTINE.			
848	002144	012667	000026		SAVRG: MOV	(6)+, SVRPC		;SAVE PC AND PSW.
849	002150	010446			MOV	(6)+, SVRPSW		
850	002152	010346			MOV	%4, -(6)		;SAVE REGS 0 - 4
851	002154	010246			MOV	%3, -(6)		;IN STACK.
852	002156	010146			MOV	%2, -(6)		
853	002160	010046			MOV	%1, -(6)		
854	002162	016746	000010		MOV	%0, -(6)		
855	002166	016746	000002		MOV	SVRPSW, -(6)		;RESTORE PC AND PSW.
856	002172	000002			MOV	SVRPC, -(6)		
857	002174	000000			RTI			;EXIT.
858	002176	000000			SVRPC: OPEN			
859					SVRPSW: OPEN			
860	002200	012667	000030		:RESTORE REGS 0 TO 4 SUBROUTINE.			
861	002204	012667	000026		RSTRG: MOV	(6)+, RSTPC		;SAVE PC AND PSW.
862	002210	012600			MOV	(6)+, RSTPSW		
863	002212	012601			MOV	(6)+, %0		;RESTORE REGS 0 - 4
864	002214	012602			MOV	(6)+, %1		;FROM STACK.
865	002216	012603			MOV	(6)+, %2		
866	002220	012604			MOV	(6)+, %3		
867					MOV	(6)+, %4		
868	002222	016746	000010		MOV	RSTPSW, -(6)		;RESTORE PC AND PSW.
869	002226	016746	000002		MOV	RSTPC, -(6)		
870	002232	000002			RTI			;EXIT
871	002234	000000			RSTPC: OPEN			
872	002236	000000			RSTPSW: OPEN			
873					:ROUTINE TO SET RECEIVER INTERRUPT VECTOR AND PRIORITY			
874	002240	017667	000000	000012	STLSRV: MOV	Q(6), STPRA+2		;MOVE VECTOR ADDR TO STPRA+2
875	002246	062716	000002		ADD	#2, Q%6		;SET UP EXIT
876	002252	016701	176632		MOV	RXVTR, %1		
877	002256	012721	000000		STPRA: MOV	#OPEN, (1)+		;SET VECTOR ADDRESS
878	002262	016721	176624		MOV	RXLVL, (1)+		;SET PRIORITY
879	002266	000002			RTI			;EXIT
880					:ROUTINE TO SET TRANSMITTER INTERRUPT VECTOR AND PRIORITY.			
881	002270	017667	000000	000012	STLSPV: MOV	Q(6), STPPA+2		;MOVE VECTOR ADDR TO STPPA+2
882	002276	062716	000002		ADD	#2, Q%6		;SET UP EXIT
883	002302	016701	176606		MOV	TXVTR, %1		
884	002306	012721	000000		STPPA: MOV	#OPEN, (1)+		;SET VECTOR ADDRESS.
885	002312	016721	176600		MOV	TXLVL, (1)+		;SET PRIORITY
886	002316	000002			RTI			;EXIT.
887					:ROUTINE TO ISSUE RESET.			
888	002320	012700	052525		SRSETT: MOV	#52525, %0		;DATA TO R0.
889	002324	005100			COM	%0		;COMPLEMENT (R0).
890	002326	010067	177770		MOV	%0, SRSETT+2		; (R0) TO SRSETT+2.
891	002332	000005			RESET			;ISSUE RESET. (R0) IS

```

892 002334 000002          RTI          ;DISPLAYED. EXIT.
893
894          ;RANDOM NUMBER GENERATOR. ROUTINE EXITS WITH NUMBER IN REGISTER 0.
895 002336 016700 000042  RNGEN:  MOV    RP1,%0
896 002342 006100          ROL    %0
897 002344 006100          ROL    %0
898 002346 066700 000034  ADD    RP2,%0
899 002352 010067 000026  MOV    %0,RP1
900 002356 006100          ROL    %0
901 002360 006100          ROL    %0
902 002362 066700 000020  ADD    RP2,%0
903 002366 006100          ROL    %0
904 002370 006100          ROL    %0
905 002372 010067 000010  MOV    %0,RP2
906 002376 016700 000002  MOV    RP1,%0
907 002402 000207          RTS    %7          ;EXIT. NUMBER IN R0
908 002404 001233  RP1:    1233
909 002406 007622  RP2:    7622
910          ;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER.
911 002410 011600          TYP:  MOV    @%6,%0          ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS.
912 002412 062716 000002  ADD    #2,@%6          ;SET UP EXIT.
913 002416 011000          MOV    @%0,%0          ;ADDRESS OF MESSAGE TO R0.
914 002420 112067 000104  TYPA:  MOVVB  (0)+,TYPDAT      ;GET CHARACTER
915 002424 122767 000100 000076  CMPB   #100,TYPDAT      ;CHECK FOR"@"CHARACTER
916 002432 001001          BNE    TYPC            ;BRANCH IF NOT"@".
917 002434 000002          RTI                    ;TERMINATOR CHAR. DONE. EXIT.
918 002436 122767 000045 000064  TYPC:  CMPB   #45,TYPDAT      ;CHECK FOR"%".
919 002444 001416          BEQ    TYPF            ;BRANCH IF"%".
920 002446 122767 000043 000054  CMPB   #43,TYPDAT      ;NOT"%".CHECK FOR"#".
921 002454 001417          BEQ    TYPG            ;BRANCH IF"#".
922 002456 004767 000002          JSR    %7,TYPD         ;TYPE CHAR IN TYPDAT
923 002462 000756          BR     TYPA
924 002464 116777 000040 176434  TYPD:  MOVVB  TYPDAT,@TPB    ;OUTPUT CHARACTER TO PRINTER
925 002472 105777 176426          TSTB   @TPS            ;WAIT FOR DONE FLAG.
926 002476 100375          BPL    -4
927 002500 000207          RTS    %7          ;EXIT
928 002502 112767 000015 000020  TYFF:  MOVVB  #15,TYPDAT      ;MOVE CARRIAGE RETURN CODE TO TYPDAT
929 002510 004767 177750          JSR    %7,TYPD         ;GO TYPE CHAR.
930 002514 112767 000012 000006  TYPG:  MOVVB  #12,TYPDAT      ;MOVE LF CODE TO TYPDAT.
931 002522 004767 177736          JSR    %7,TYPD         ;GO TYPE CHAR.
932 002526 000734          BR     TYPA
933 002530 000000  TYPDAT: OPEN
934          ;SUBROUTINE TO OUTPUT A SERIES OF ASCII MESSAGES ON TELETYPE PRINTER
935 002532 011600          TYPs:  MOV    @%6,%0          ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
936 002534 062716 000002  ADD    #2,@%6          ;UPDATE TO NEXT MESSAGE ADDRESS
937 002540 011067 000014          MOV    @%0,TYPsB      ;ADDRESS OF MESSAGE TO TYPsB
938 002544 022767 177777 000006  CMP    #-1,TYPsB      ;CHECK FOR TERMINATOR
939 002552 001001          BNE    TYPsA          ;BRANCH IF NOT TERMINATOR.
940 002554 000002          RTI                    ;TERMINATOR. EXIT
941 002556 104000  TYPsA:  TYPE                    ;CALL ON TYP SUB TO TYPE MESSAGE
942 002560 000000  TYPsB:  OPEN                    ;ADDRESS OF MESSAGE GOES HERE
943 002562 000763          BR     TYPs            ;GO PROCESS NEXT MESSAGE
944
945 002564 012701 000300  OVRLAY: MOV    #300,%1          ;GET DC11 VECTOR BASE ADDRESS
946 002570 012702 000302  MOV    #302,%2          ;GET NEXT ADDRESS
947 002574 010221  OVRLYA: MOV    %2,(1)+      ;LOAD VECTOR WITH ADDRESS OF NEXT ADDRESS
  
```

948	002576	005021		CLR	(1)+		;PUT A HALT IN THE NEXT ADDRESS
949	002600	020267	176174	CMP	%2,1000		;ALL VECTORS BEEN LOADED
950	002604	001403		BEQ	OVRLYB		
951	002606	062702	000004	ADD	#4,%2		;GET NEXT VECTOR ADDRESS
952	002612	000770		BR	OVRLYA		
953	002614	000207		OVRLYB: RTS	7		;EXIT
954							
955							;SUBROUTINE TO DELAY A SPECIFIED NUMBER OF MILLISECONDS
956	002616	011667	000040	DLY: MOV	%6,DLCNT		;GET DELAY COUNT ADDRESS.
957	002622	062716	000002	ADD	#2,%6		;SET UP EXIT ADDRESS
958	002626	017746	000030	MOV	%DLCNT,-(6)		;DELAY COUNT TO STACK
959	002632	001411		BEQ	DLYC		
960	002634	005067	175136	CLR	PSW		;SET PRIORITY 0
961	002640	012746	000554	DLYA: MOV	#554,-(6)		;1 MSEC COUNT TO STACK
962	002644	005316		DLYB: DEC	%6		;DECREMENT 1 MSEC COUNT
963	002646	001376		BNE	DLYB		;BRANCH IF NOT 0.
964	002650	005726		POPSP			;ZERO. UNCOVER MSECS. COUNT.
965	002652	005316		DEC	%6		;DECREMENT IT
966	002654	001371		BNE	DLYA		;BR IF NOT DONE DELAYING
967	002656	005726		DLYC: POPSP			;DONE
968	002660	000002		RTI			;EXIT.
969	002662	000000		DLCNT: OPEN			;CONTAINS MILLISECONDS COUNT ADDRESS.
970							;SUBROUTINE TO STALL A RANDOM NUMBER OF MILLISECONDS. MAXIMUM STALL
971							;DETERMINED BY CONTENTS OF LOC STLMSK.
972	002664	004767	177446	STAL: JSR	%7,RNGEN		;GO GET RANDOM NUMBER.
973	002670	046700	176374	BIC	STLMSK,%0		;# IN RD. APPLY STALL MASK.
974	002674	001404		BEQ	STALB		;BRANCH IF RESULT IS 0.
975	002676	010067	000002	MOV	%0,STALA		
976	002702	104016		DELAY			;DELAY
977	002704	000000		STALA: OPEN			;DELAY COUNT
978	002706	000002		STALB: RTI			;DONE. EXIT.
979							;SUBROUTINE TO GENERATE RANDOM CHARACTER COUNT
980	002710	004767	177422	GRCNT: JSR	%7,RNGEN		;GET RANDOM NUMBER
981	002714	046700	000010	BIC	RCMSK,%0		;APPLY MASK
982	002720	001773		BEQ	GRCNT		;TRY AGAIN IF RESULT 0
983	002722	010067	000004	MOV	%0,RNCNT		;COUNT TO RNCNT
984	002726	000207		RTS	%7		;EXIT.
985	002730	000000		RCMSK: OPEN			;RANDOM CHARACTER MASK.
986	002732	000000		RNCNT: OPEN			;RANDOM CHARACTER COUNT.
987							;SUBROUTINE TO SELECT LINE AND
988	002734	104000		LINSEL: TYPE			
989	002736	016732		LDLINE			
990	002740	000000		HALT			
991	002742	016701	174622	MOV	SR,%1		
992	002746	042701	177407	BIC	#177407,%1		
993	002752	010167	176306	MOV	%1,TEMP		
994	002756	012702	000770	MOV	#770,%2		
995	002762	012703	001100	MOV	#RXCSR,%3		
996	002766	012704	000004	MOV	#4,%4		
997	002772	040213		BIC	%2,(3)		
998	002774	050123		BIS	%1,(3)+		
999	002776	005304		DEC	%4		
1000	003000	001374		BNE	,-6		
1001	003002	006201		ASR	%1		;POSITION SELECTED LINE
1002	003004	006201		ASR	%1		
1003	003006	016101	015102	MOV	VECTAB(1),%1		;GET LINE VECTOR ADDRESS

1004	003012	010167	176072		MOV	%1,RXVTR	;LOAD INTO PROGRAM RXVTR
1005	003016	022121			CMP	(1)+,(1)+	;ADD +4
1006	003020	010167	176070		MOV	%1, TXVTR	;LOAD INTO PROGRAM TXVTR
1007	003024	006267	176234		ASR	TEMP	
1008	003030	006267	176230		ASR	TEMP	
1009	003034	006267	176224		ASR	TEMP	
1010	003040	004567	000166		JSR	5,OACNV	;TYPE LINE #
1011	003044	001264			TEMP		
1012	003046	017003			SELINE		
1013	003050	000002			2		
1014	003052	104000			TYPE		
1015	003054	016771			ALINE		
1016	003056	000205			RTS	5	
1017							
1018	003060	012767	177777	000014	:SUBROUTINE TO INITIALIZE BINARY COUNT PATTERNS		
1019	003066	004567	000222		INBIN: MOV	#-1,RIND	;SET ALL VARIABLES
1020	003072	003102			JSR	%5,BMOVE	;TO MINUS 1.
1021	003074	003103			RIND		
1022	003076	000013			RIND+1		
1023	003100	000207			11.		
1024	003102	000000			RTS	%7	;EXIT
1025	003104	000000			RIND:	OPEN	
1026	003106	000000			PTO:	OPEN	
1027	003110	000000			PT1:	OPEN	
1028	003112	000000			PIND:	OPEN	
1029	003114	000000			PTOP:	OPEN	
1030					PT1P:	OPEN	
1031	003116	016767	177762	177762	:SPECIAL BINARY COUNT PATTERN SUBROUTINE. EXITS WITH BIN CHAR IN RO		
1032	003124	005167	177756		GTBIN: MOV	PTO,PT1	;PREVIOUS BIN CHAR TO PT1
1033	003130	005167	177746		COM	PT1	
1034	003134	001002			COM	RIND	
1035	003136	005267	177744		BNE	.+6	
1036	003142	042767	177400	177736	INC	PT1	;MASK TO 8 BITS
1037	003150	016767	177732	177726	BIC	#177400,PT1	;SAVE BIN CHAR IN PTO
1038	003156	016700	177724		MOV	PT1,PTO	;BIN CHAR TO RO.
1039	003162	000207			MOV	PT1,%0	;EXIT.
1040	003164	016767	177722	177722	RTS	%7	;PREVIOUS BIN CHAR TO PT1P
1041	003172	005167	177716		GTBINP: MOV	PTOP,PT1P	
1042	003176	005167	177706		COM	PT1P	
1043	003202	001002			COM	PIND	
1044	003204	005267	177704		BNE	.+6	
1045	003210	042767	177400	177676	INC	PT1P	;MASK TO 8 BITS.
1046	003216	016767	177672	177666	BIC	#177400,PT1P	;SAVE BIN CHAR IN PTOP.
1047	003224	016701	177664		MOV	PT1P,PTOP	;BIN CHAR TO R1.
1048	003230	000207			MOV	PT1P,%1	;EXIT.
1049					RTS	%7	
1050	003232	013567	000054		:OCTAL TO ASCII CONVERT ROUTINE		
1051	003236	012501			OACNV: MOV	2(5)+,OACNVX	;GET OCTAL VALUE.
1052	003240	012502			MOV	(5)+,%1	;GET DESTINATION ADDR.
1053	003242	060201			MOV	(5)+,%2	;GET CONVERT COUNT.
1054	003244	016703	000042		ADD	%2,%1	;DEVELOP ADDR TO STORE 1ST CHAR.
1055	003250	042703	177770		OACNVA: MOV	OACNVX,%3	
1056	003254	062703	000060		BIC	#177770,%3	;ISOLATE LEAST SIGNIFICANT DIGIT.
1057	003260	110341			ADD	#60,%3	;CONVERT DIGIT TO ASCII.
1058	003262	042767	000007	000022	MOVB	%3,-(1)	;STORE ASCII CHARACTER.
1059	003270	006067	000016		BIC	#7,OACNVX	
					ROR	OACNVX	

1060	003274	006067	000012		ROR	OACNVX	
1061	003300	006067	000006		ROR	OACNVX	
1062	003304	005302			DEC	%2	;DONE ALL DIGITS?
1063	003306	001356			BNE	OACNVA	;BRANCH IF NOT DONE.
1064	003310	000205			RTS	%5	;DONE. EXIT.
1065	003312	000000			OACNVX: OPEN		
1066					;SUBROUTINE TO MOVE A VARIABLE NUMBER OF BYTES.		
1067	003314	104013			BMOVE: SAVREG		;SAVE REGS.
1068	003316	012501			MOV	(5)+,%1	;GET"FROM"ADDRESS
1069	003320	012502			MOV	(5)+,%2	;GET"TO"ADDRESS
1070	003322	012503			MOV	(5)+,%3	;GET COUNT
1071	003324	112122			BMOVA: MOVB	(1)+,(2)+	;MOVE BYTE
1072	003326	005303			DEC	%3	;DECREMENT COUNT
1073	003330	001375			BNE	BMOVA	;BRANCH IF NOT DONE.
1074	003332	104014			RSTREG		;RESTORE REGS.
1075	003334	000205			RTS	%5	;DONE EXIT
1076					;BINARY TO DECIMAL ASCII CONVERT		SUBROUTINE.
1077	003336	012700	003456		BDCNV: MOV	#DECVAL,%0	;SET UP ADDR TO STORE DECIMAL ASCII IN R0
1078	003342	013501			MOV	@(5)+,%1	;BINARY VALUE TO R1.
1079	003344	012702	003444		MOV	#ADTENP,%2	;ADDR OF TEN POWER STRING TO R2.
1080	003350	012767	000005	000060	MOV	#5,CNVCTR	;SET UP FOR 5 POWER CONVERSIONS.
1081	003356	012267	000060		BDCNVA: MOV	(2)+,TENPWR	;MOVE POWER OF TEN VALUE TO TENPWR.
1082	003362	004767	000010		JSR	%7,SUBTEN	;PERFORM CONVERSION
1083	003366	005367	000044		DEC	CNVCTR	;DONE 5 CONVERSIONS?
1084	003372	001371			BNE	BDCNVA	;BRANCH IF NOT YET 5.
1085	003374	000205			RTS	%5	;YES, EXIT.
1086	003376	005067	000036		SUBTEN: CLR	DIGIT	;CLEAR DIGIT
1087	003402	166701	000034		SUBTNA: SUB	TENPWR,%1	;SUBTRACT TEN POWER FROM BINARY VALUE.
1088	003406	103403			BCS	SUBTNB	;BRANCH IF UNSUCCESSFUL SUBTRACTION.
1089	003410	005267	000024		INC	DIGIT	
1090	003414	000772			BR	SUBTNA	
1091	003416	066701	000020		SUBTNB: ADD	TENPWR,%1	;RESTORE SUBTRACTED VALUE.
1092	003422	062767	000060	000010	ADD	#60,DIGIT	;CONVERT (DIGIT) TO ASCII
1093	003430	116720	000004		MOVB	DIGIT,(0)+	;MOVE ASCII CHAR TO DECVAL FIELD.
1094	003434	000207			RTS	%7	;EXIT.
1095	003436	000000			CNVCTR: OPEN		
1096	003440	000000			DIGIT: OPEN		
1097	003442	000000			TENPWR: OPEN		
1098	003444	023420			ADTENP: 10000.		
1099	003446	001750			1000.		
1100	003450	000144			100.		
1101	003452	000012			10.		
1102	003454	000001			1		
1103	003456	040	040	040	DECVAL: .BYTE	040,040,040,040,040,040	
1104	003461	040	040	040			
1105	003464	012567	175576		DATTST: MOV	(5)+,SRT	;GET PARAMETERS
1106	003470	004767	011220		JSR	7,STPARB	;LOAD PARAMETERS
1107	003474	042777	000001	175376	BIC	#BIT0,@RXCSR	;CLEAR DATA TERM. READY
1108	003502	052777	000004	175374	BIS	#BIT2,@TXCSR	;SET MAINTENANCE BIT
1109	003510	012767	000144	175530	MOV	#100,CTRA	;GET CHARACTER COUNT
1110	003516	105777	175362		DATAA: TSTB	@TXCSR	;WAIT FOR
1111	003522	100375			BPL	-4	;READY FLAG
1112	003524	004767	177434		JSR	7,GTBINP	;GET CHARACTER
1113	003530	110167	175476		MOVB	%1,CBUBA	;MOVE CHARACTER
1114	003534	046767	175474	175470	BIC	CARMSK,CBUBA	;MASK OFF NON TRANSMITTED BITS
1115	003542	110177	175340		MOVB	%1,@TXBUF	;TRANSMIT CHARACTER

1116	003546	105777	175326		TSTB	QXCSR		;WAIT FOR
1117	003552	100375			BPL	.-4		;DONE FLAG
1118	003554	117767	175322	175446	MOVB	QXBUF,CRBUF		;GET RECEIVED CHARACTER
1119	003562	104004			DATCHK			;CHK DATA
1120	003564	005367	175456		DEC	CTRA		;DECREMENT CHARACTER COUNT
1121	003570	001352			BNE	DAAA		
1122	003572	005726			TST	(6)+		;POP STACK
1123	003574	104012			SCOPE			
1124								
1125	003576	012767	003606	174220	PFAIL: MOV	#PWRUP,24		
1126	003604	000000			HALT			
1127								
1128	003606	000005			PWRUP: RESET			
1129	003610	012706	001076		MOV	#SPBOT,%6		
1130	003614	104003			ERROR			
1131	003616	016700	175316		RESTART: MOV	PRGNUM,%0		;GET PROGRAM NUMBER
1132	003622	006300			ASL	%0		
1133	003624	012767	003576	174172	MOV	#PFAIL,24		;RELOAD POWER FAIL VECTOR
1134	003632	004767	011056		JSR	7,STPARB		;RELOAD LINE PARAMETERS
1135	003636	000170	003642		JMP	QSTART(0)		;GO RESTART SELECTED PROGRAM
1136								
1137	003642	003700			RSTART: PRGOA			;PROGRAM 0 RESTART ADDRESS
1138	003644	014366			PRG1A			;PROGRAM 1 RESTART ADDRESS
1139	003646	014432			PRG2A			;PROGRAM 2 RESTART ADDRESS
1140	003650	014526			PRG3A			;PROGRAM 3 RESTART ADDRESS
1141	003652	014562			PRG4A			;PROGRAM 4 RESTART ADDRESS

```

1142
1143
1144 003654 012767 003704 175260 ;PRGO - INPUT-OUTPUT LOGIC TESTS
1145 003662 104000 PRGO:  MOV #ATO,KSTART
1146 003664 015233          TYPE                                ;TYPE TITLE AND INSTRUCTIONS
1147 003666 000000          POTIT
1148 003670 004567 177040          HALT
1149 003674 004767 175372          JSR 5,LINSEL          ;GO GET LINE # FROM USER
1150 003700 000167 175736          JSR 7,SETSR
1151 003700 177777          PRGOA: JMP GETRODY          ;GET STARTED.
1152                                     TX=-1
1153 003704 000000          ;*****
1154 003706 003736          ATO:  0                ;TEST NUMBER 0
1155 003710 001750          AT1                ;ADDRESS OF NEXT TEST
1156 003712 003714          1000.              ;TEST ITERATION COUNT
1157                                     AAA                ;SCOPE ENTRY POINT
1158                                     ;*****
1159 003714 012767 003730 174062 ;TEST ABILITY TO REFERENCE RECEIVER CSR WITHOUT TRAPPING
1160 003722 005777 175152 AAA:  MOV #AAE,MACHER ;SET UP MACHINE ERROR TRAP.
1161 003726 104012          TST @RXCSR          ;REFERENCE RXCSR
1162 003730 022626          AAB:  SCOPE          ;OK IF NO TRAP. SCOPE
1163 003732 104003          AAE:  POPSP2
1164 003734 000774          ERROR          ;TRAPPED WHEN REFERENCING RXCSR.
1165                                     BR AAB
1166 003736 000001          ;*****
1167 003740 003770          AT1:  1                ;TEST NUMBER 1
1168 003742 001750          AT2                ;ADDRESS OF NEXT TEST
1169 003744 003746          1000.              ;TEST ITERATION COUNT
1170                                     ABA                ;SCOPE ENTRY POINT
1171                                     ;*****
1172 003746 012767 003762 174030 ;TEST ABILITY TO REFERENCE RECEIVER BUFFER WITHOUT TRAPPING
1173 003754 005777 175122 ABA:  MOV #ABE,MACHER ;SET UP MACHINE ERROR TRAP.
1174 003760 104012          TST @RXBUF          ;REFERENCE RXBUF
1175 003762 022626          ABB:  SCOPE          ;OK IF NO TRAP SCOPE
1176 003764 104003          ABE:  POPSP2
1177 003766 000774          ERROR          ;TRAPPED WHEN REFERENCING RXBUF
1178                                     BR ABB
1179                                     ;*****
1180 003770 000002          AT2:  2                ;TEST NUMBER 2
1181 003772 004022          AT3                ;ADDRESS OF NEXT TEST
1182 003774 001750          1000.              ;TEST ITERATION COUNT
1183 003776 004000          ACA                ;SCOPE ENTRY POINT
1184                                     TX=TX+1
1185                                     ;*****
1186                                     ;TEST ABILITY TO REFERENCE TRANSMITTER CSR WITHOUT TRAPPING.
1187 004000 012767 004014 173776 ACA:  MOV #ACE,MACHER ;SET UP MACHINE ERROR TRAP.
1188 004006 005777 175072          TST @TXCSR          ;REFERENCE TXCSR
1189 004012 104012          ACB:  SCOPE          ;SCOPE
1190 004014 022626          ACE:  POPSP2
1191 004016 104003          ERROR          ;TRAPPED WHEN REFERENCING TXCSR
1192 004020 000774          BR ACB
1193

```

```

1194
1195
1196 004022 000003
1197 004024 004054
1198 004026 001750
1199 004030 004032
1200
1201
1202 004032 012767 004046 173744
1203 004040 005777 175042
1204 004044 104012
1205 004046 022626
1206 004050 104003
1207 004052 000774
1208
1209
1210 004054 000004
1211 004056 004154
1212 004060 000144
1213 004062 004064
1214
1215
1216 004064 032777 000001 175012
1217 004072 001402
1218 004074 104003
1219 004076 000421
1220 004100 052777 000001 174776
1221 004106 032777 000001 174770
1222 004114 001002
1223 004116 104003
1224 004120 000410
1225 004122 042777 000001 174754
1226 004130 032777 000001 174746
1227 004136 001401
1228 004140 104003
1229 004142 052777 000001 174734
1230 004150 104011
1231 004152 104012
1232
1233 004154 000005
1234 004156 004252
1235 004160 000144
1236 004162 004164
1237
1238
1239 004164 042777 000001 174706
1240 004172 032777 000002 174704
1241 004200 001402
1242 004202 104003
1243 004204 000421
1244 004206 052777 000001 174664
1245 004214 032777 000002 174662
1246 004222 001002
1247 004224 104003
1248 004226 000410
1249 004230 042777 000001 174642

```

```

;*****
AT3: 3 ;TEST NUMBER 3 *
      AT4 ;ADDRESS OF NEXT TEST *
      1000. ;TEST ITERATION COUNT *
      ADA ;SCOPE ENTRY POINT *
;*****
;TEST ABILITY TO REFERENCE TRANSMITTER BUFFER WITHOUT TRAPPING
ADA: MOV #ADE,MACHER ;SET UP MACHINE ERROR TRAP.
      TST @TXBUF ;REFERENCE TX BUF.
ADB: SCOPE ;SCOPE
ADE: POPSP2
      ERROR ;TRAPPED WHEN REFERENCING TXBUF
      BR ADB
;*****
AT4: 4 ;TEST NUMBER 4 *
      AT5 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AEA ;SCOPE ENTRY POINT *
;*****
;TEST THAT TXCSR BIT0 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
AEA: BIT #BIT0,@TXCSR ;SEE IF TXCSR BIT0 IS CLEAR.
      BEQ AEB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RESET DID NOT CLEAR TXCSR BIT0
      BR AED
AEB: BIS #BIT0,@TXCSR ;SET TXCSR BIT0.
      BIT #BIT0,@TXCSR ;SEE IF BIT IS SET.
      BNE AEC ;BRANCH IF BIT IS SET.
      ERROR ;TXCSR BIT0 FAILED TO SET.
      BR AED
AEC: BIC #BIT0,@TXCSR ;CLEAR TXCSR BIT0
      BIT #BIT0,@TXCSR ;SEE IF BIT IS CLEAR.
      BEQ AED
      ERROR ;TXCSR BIT0 FAILED TO CLEAR.
AED: BIS #BIT0,@TXCSR ;SET TXCSR BIT0.
      SRESET ;ISSUE RESET TO CLEAR BIT.
      SCOPE ;SCOPE
;*****
AT5: 5 ;TEST NUMBER 5 *
      AT6 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AFA ;SCOPE ENTRY POINT *
;*****
;TEST THAT TXCSR BIT1 (CLEAR TO SEND) CAN BE SET, AND CLEARED
AFA: BIC #BIT0,@RXCSR ;CLEAR DATA TERMINAL READY
      BIT #BIT1,@TXCSR ;SEE IF TXCSR BIT1 IS CLEAR.
      BEQ AFB ;BRANCH IF BIT IS CLEAR.
      ERROR ;TXCSR BIT1 IS NOT CLEAR.
      BR AFD ;EXIT TEST
AFB: BIS #BIT0,@RXCSR ;SET DATA TERM. RDY. (SETS CTS VIA JUMPER)
      BIT #BIT1,@TXCSR ;IS CLEAR TO SEND SET?
      BNE AFC ;BRANCH IF SET
      ERROR ;CTS NOT BEING SET VIA DTR
      BR AFD ;EXIT TEST
AFC: BIC #BIT0,@RXCSR ;CLEAR DATA TERM. RDY.

```

C03

.MAIN. MACY11 27(732) 29-SEP-76 14:25 PAGE 28  
DZDCAB.P11

1250	004236	032777	000002, 174640	BIT	#BIT1, @TXCSR	; IS CTS CLEAR?
1251	004244	001401		BEG	AFD	
1252	004246	104003		ERROR		; CTS FAILED TO CLEAR VIA DTR
1253	004250	104012		AFD:	SCOPE	; SCOPE

```

1254
1255
1256 004252 000006
1257 004254 004352
1258 004256 000144
1259 004260 004262
1260
1261
1262 004262 032777 000004 174614
1263 004270 001402
1264 004272 104003
1265 004274 000421
1266 004276 052777 000004 174600
1267 004304 032777 000004 174572
1268 004312 001002
1269 004314 104003
1270 004316 000410
1271 004320 042777 000004 174556
1272 004326 032777 000004 174550
1273 004334 001401
1274 004336 104003
1275 004340 052777 000004 174536
1276 004346 104011
1277 004350 104012
1278
1279 004352 000007
1280 004354 004452
1281 004356 000144
1282 004360 004362
1283
1284
1285 004362 032777 000010 174514
1286 004370 001402
1287 004372 104003
1288 004374 000421
1289 004376 052777 000010 174500
1290 004404 032777 000010 174472
1291 004412 001002
1292 004414 104003
1293 004416 000410
1294 004420 042777 000010 174456
1295 004426 032777 000010 174450
1296 004434 001401
1297 004436 104003
1298 004440 052777 000010 174436
1299 004446 104011
1300 004450 104012

;*****
AT6: 6 ;TEST NUMBER 6 *
      AT7 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AGA ;SCOPE ENTRY POINT *
;*****
;TEST THAT TXCSR BIT2 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
AGA: BIT #BIT2,@TXCSR ;SEE IF TXCSR BIT2 IS CLEAR.
      BEQ AGB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RESET DID NOT CLEAR TXCSR BIT2
      BR AGD
AGB: BIS #BIT2,@TXCSR ;SET TXCSR BIT2.
      BIT #BIT2,@TXCSR ;SEE IF BIT IS SET.
      BNE AGC ;BRANCH IF BIT IS SET.
      ERROR ;TXCSR BIT2 FAILED TO SET.
      BR AGD
AGC: BIC #BIT2,@TXCSR ;CLEAR TXCSR BIT2
      BIT #BIT2,@TXCSR ;SEE IF BIT IS CLEAR.
      BEQ AGD
      ERROR ;TXCSR BIT2 FAILED TO CLEAR.
AGD: BIS #BIT2,@TXCSR ;SET TXCSR BIT2.
      SRESET ;ISSUE RESET TO CLEAR BIT.
      SCOPE ;SCOPE
;*****
AT7: 7 ;TEST NUMBER 7 *
      AT10 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AHA ;SCOPE ENTRY POINT *
;*****
;TEST THAT TXCSR BIT3 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
AHA: BIT #BIT3,@TXCSR ;SEE IF TXCSR BIT3 IS CLEAR.
      BEQ AHB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RESET DID NOT CLEAR TXCSR BIT3
      BR AHD
AHB: BIS #BIT3,@TXCSR ;SET TXCSR BIT3.
      BIT #BIT3,@TXCSR ;SEE IF BIT IS SET.
      BNE AHC ;BRANCH IF BIT IS SET.
      ERROR ;TXCSR BIT3 FAILED TO SET.
      BR AHD
AHC: BIC #BIT3,@TXCSR ;CLEAR TXCSR BIT3
      BIT #BIT3,@TXCSR ;SEE IF BIT IS CLEAR.
      BEQ AHD
      ERROR ;TXCSR BIT3 FAILED TO CLEAR.
AHD: BIS #BIT3,@TXCSR ;SET TXCSR BIT3.
      SRESET ;ISSUE RESET TO CLEAR BIT.
      SCOPE ;SCOPE

```

```

1301
1302
1303 004452 000010
1304 004454 004552
1305 004456 000144
1306 004460 004462
1307
1308
1309 004462 032777 000020 174414
1310 004470 001402
1311 004472 104003
1312 004474 000421
1313 004476 052777 000020 174400
1314 004504 032777 000020 174372
1315 004512 001002
1316 004514 104003
1317 004516 000410
1318 004520 042777 000020 174356
1319 004526 032777 000020 174350
1320 004534 001401
1321 004536 104003
1322 004540 052777 000020 174336
1323 004546 104011
1324 004550 104012
1325
1326 004552 000011
1327 004554 004660
1328 004556 000144
1329 004560 004562
1330
1331
1332 004562 012767 000340 173206
1333 004570 032777 000100 174306
1334 004576 001402
1335 004600 104003
1336 004602 000421
1337 004604 052777 000100 174272
1338 004612 032777 000100 174264
1339 004620 001002
1340 004622 104003
1341 004624 000410
1342 004626 042777 000100 174250
1343 004634 032777 000100 174242
1344 004642 001401
1345 004644 104003
1346 004646 052777 000100 174230
1347 004654 104011
1348 004656 104012
1349

;*****
AT10: 10 ;TEST NUMBER 10 *
      AT11 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AIA ;SCOPE ENTRY POINT *
;*****
;TEST THAT TXCSR BIT4 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
AIA: BIT #BIT4,@TXCSR ;SEE IF TXCSR BIT4 IS CLEAR.
      BEQ AIB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RESET DID NOT CLEAR TXCSR BIT4
      BR AID
AIB: BIS #BIT4,@TXCSR ;SET TXCSR BIT4.
      BIT #BIT4,@TXCSR ;SEE IF BIT IS SET.
      BNE AIC ;BRANCH IF BIT IS SET.
      ERROR ;TXCSR BIT4 FAILED TO SET.
      BR AID
AIC: BIC #BIT4,@TXCSR ;CLEAR TXCSR BIT4
      BIT #BIT4,@TXCSR ;SEE IF BIT IS CLEAR.
      BEQ AID
      ERROR ;TXCSR BIT4 FAILED TO CLEAR.
AID: BIS #BIT4,@TXCSR ;SET TXCSR BIT4.
      SRESET ;ISSUE RESET TO CLEAR BIT.
      SCOPE ;SCOPE
;*****
AT11: 11 ;TEST NUMBER 11 *
      AT12 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AJA ;SCOPE ENTRY POINT *
;*****
;TEST THAT TXCSR BIT6 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
AJA: MOV #PTY7,PSW ;SET PRIORITY 7.
      BIT #BIT6,@TXCSR ;SEE IF TXCSR BIT6 IS CLEAR.
      BEQ AJB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RESET DID NOT CLEAR TXCSR BIT6
      BR AJD
AJB: BIS #BIT6,@TXCSR ;SET TXCSR BIT6.
      BIT #BIT6,@TXCSR ;SEE IF BIT IS SET.
      BNE AJC ;BRANCH IF BIT IS SET.
      ERROR ;TXCSR BIT6 FAILED TO SET.
      BR AJD
AJC: BIC #BIT6,@TXCSR ;CLEAR TXCSR BIT6
      BIT #BIT6,@TXCSR ;SEE IF BIT IS CLEAR.
      BEQ AJD
      ERROR ;TXCSR BIT6 FAILED TO CLEAR.
AJD: BIS #BIT6,@TXCSR ;SET TXCSR BIT6.
      SRESET ;ISSUE RESET TO CLEAR BIT.
      SCOPE ;SCOPE

```

```

1350
1351
1352 004660 000012
1353 004662 004702
1354 004664 000144
1355 004666 004670
1356
1357
1358
1359 004670 105777 174210
1360 004674 100401
1361 004676 104003
1362 004700 104012
1363
1364 004702 000013
1365 004704 005002
1366 004706 000144
1367 004710 004712
1368
1369
1370 004712 032777 000400 174164
1371 004720 001402
1372 004722 104003
1373 004724 000421
1374 004726 052777 000400 174150
1375 004734 032777 000400 174142
1376 004742 001002
1377 004744 104003
1378 004746 000410
1379 004750 042777 000400 174126
1380 004756 032777 000400 174120
1381 004764 001401
1382 004766 104003
1383 004770 052777 000400 174106
1384 004776 104011
1385 005000 104012
1386
1387
1388
1389 005002 000014
1390 005004 005026
1391 005006 000144
1392 005010 005012
1393
1394
1395 005012 032777 100000 174064
1396 005020 001401
1397 005022 104003
1398 005024 104012
1399

;*****
AT12: 12 ;TEST NUMBER 12 *
      AT13 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AKA ;SCOPE ENTRY POINT *
;*****
;TEST THAT TXCSR BIT 7 (READY BIT) IS SET UPON ENTERING ROUTINE AND
;THAT IT CAN BE READ RELIABLY.
AKA: TSTB @TXCSR ;SEE IF TXCSR BIT 7 IS SET.
      BMI AKB ;BRANCH IF SET.
      ERROR ;TXCSR BIT 7 NOT SET.
AKB: SCOPE ;SCOPE
;*****
AT13: 13 ;TEST NUMBER 13 *
      AT14 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      ALA ;SCOPE ENTRY POINT *
;*****
;TEST THAT TXCSR BIT8 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
ALA: BIT #BIT8,@TXCSR ;SEE IF TXCSR BIT8 IS CLEAR.
      BEQ ALB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RESET DID NOT CLEAR TXCSR BITS
      BR ALD
ALB: BIS #BIT8,@TXCSR ;SET TXCSR BIT8.
      BIT #BIT8,@TXCSR ;SEE IF BIT IS SET.
      BNE ALC ;BRANCH IF BIT IS SET.
      ERROR ;TXCSR BIT8 FAILED TO SET.
      BR ALD
ALC: BIC #BIT8,@TXCSR ;CLEAR TXCSR BIT8
      BIT #BIT8,@TXCSR ;SEE IF BIT IS CLEAR.
      BEQ ALD
      ERROR ;TXCSR BIT8 FAILED TO CLEAR.
ALD: BIS #BIT8,@TXCSR ;SET TXCSR BIT8.
      SRESET ;ISSUE RESET TO CLEAR BIT.
      SCOPE ;SCOPE
;*****
AT14: 14 ;TEST NUMBER 14 *
      AT15 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AMA ;SCOPE ENTRY POINT *
;*****
;TEST THAT TXCSR BIT15 IS CLEAR AND CAN BE READ RELIABLY.
AMA: BIT #BIT15,@TXCSR ;SEE IF TXCSR BIT15 IS CLEAR.
      BEQ AMB ;BRANCH IF BIT IS CLEAR.
      ERROR ;TXCSR BIT15 IS NOT CLEAR.
AMB: SCOPE ;SCOPE

```

```

1400 ;*****
1401 005026 000015 AT15: 15 ;TEST NUMBER 15 *
1402 005030 005120 AT16 ;ADDRESS OF NEXT TEST *
1403 005032 000144 100. ;TEST ITERATION COUNT *
1404 005034 005036 ANA ;SCOPE ENTRY POINT *
1405 ;*****
1406 ;TEST THAT RXCSR BIT 0 (DATA TERMINAL READY) CAN BE SET, NOT CLEARED BY RESET, AND CLEAR
1407 005036 052777 000001 174034 ANA: BIS #BIT0,@RXCSR ;SET RXCSR BIT 0.
1408 005044 032777 000001 174026 BIT #BIT0,@RXCSR ;SEE IF BIT IS SET.
1409 005052 001002 BNE ANB ;BRANCH IF BIT IS SET.
1410 005054 104003 ERROR
1411 005056 000417 BR AND
1412 005060 104011 ANB: SRESET ;ISSUE RESET.
1413 005062 032777 000001 174010 BIT #BIT0,@RXCSR ;SEE IF BIT IS STILL SET.
1414 005070 001002 BNE ANC ;BRANCH IF BIT SET.
1415 005072 104003 ERROR ;RESET CLEARED RXCSR BIT 0.
1416 005074 000410 BR AND
1417 005076 042777 000001 173774 ANC: BIC #BIT0,@RXCSR ;CLEAR RXCSR BIT 0.
1418 005104 032777 000001 173766 BIT #BIT0,@RXCSR ;SEE IF BIT IS CLEAR.
1419 005112 001401 BEQ AND ;BRANCH IF BIT IS CLEAR.
1420 005114 104003 ERROR ;RXCSR BIT 0 FAILED TO CLEAR.
1421 005116 104012 AND: SCOPE ;SCOPE
1422 ;*****
1423 005120 000016 AT15: 16 ;TEST NUMBER 16
1424 005122 005220 AT17 ;ADDRESS OF NEXT TEST
1425 005124 000144 100. ;TEST ITERATION COUNT
1426 005126 005130 ANW ;SCOPE ENTRY POINT
1427 ;*****
1428 ;TEST THAT RXCSR BIT 1 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT
1429
1430 005130 032777 000002 173742 ANW: BIT #BIT1,@RXCSR ;SEE IF BIT 1 IS CLEAR
1431 005136 001402 BEQ ANX ;BRANCH IF CLEAR
1432 005140 104003 ERROR ;RESET DID NOT CLEAR RXCSR BIT 1
1433 005142 000421 BR ANZ
1434 005144 052777 000002 173726 ANX: BIS #BIT1,@RXCSR ;SET RXCSR BIT1
1435 005152 032777 000002 173720 BIT #BIT1,@RXCSR ;SEE IF BIT IS SET
1436 005160 001002 BNE ANY ;BRANCH IF SET
1437 005162 104003 ERROR ;RXCSR BIT 1 FAILED TO SET
1438 005164 000410 BR ANZ
1439 005166 042777 000002 173704 ANY: BIC #BIT1,@RXCSR ;CLEAR RXCSR BIT 1
1440 005174 032777 000002 173676 BIT #BIT1,@RXCSR ;SEE IF BIT IS CLEAR
1441 005202 001401 BEQ ANZ
1442 005204 104003 ERROR ;RXCSR BIT 1 FAILED TO CLEAR
1443 005206 052777 000002 173664 ANZ: BIS #BIT1,@RXCSR ;SET RXCSR BIT 1
1444 005214 104011 SRESET ;ISSUE RESET TO CLEAR BIT
1445 005216 104012 AND: SCOPE ;SCOPE
1446

```

```

1447
1448
1449
1450 005220 000017
1451 005222 005244
1452 005224 000144
1453 005226 005230
1454
1455
1456 005230 032777 000004 173642
1457 005236 001401
1458 005240 104003
1459 005242 104012
1460
1461 005244 000020
1462 005246 005344
1463 005250 000144
1464 005252 005254
1465
1466
1467 005254 032777 000010 173616
1468 005262 001402
1469 005264 104003
1470 005266 000421
1471 005270 052777 000010 173602
1472 005276 032777 000010 173574
1473 005304 001002
1474 005306 104003
1475 005310 000410
1476 005312 042777 000010 173560
1477 005320 032777 000010 173552
1478 005326 001401
1479 005330 104003
1480 005332 052777 000010 173540
1481 005340 104011
1482 005342 104012
1483
1484 005344 000021
1485 005346 005444
1486 005350 000144
1487 005352 005354
1488
1489
1490 005354 032777 000020 173516
1491 005362 001402
1492 005364 104003
1493 005366 000421
1494 005370 052777 000020 173502
1495 005376 032777 000020 173474
1496 005404 001002
1497 005406 104003
1498 005410 000410
1499 005412 042777 000020 173460
1500 005420 032777 000020 173452
1501 005426 001401
1502 005430 104003

```

```

;*****
AT17: 17 ;TEST NUMBER 17 *
      AT20 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      APA ;SCOPE ENTRY POINT *
;*****
;TEST THAT RXCSR BIT2 IS CLEAR AND CAN BE READ RELIABLY.
APA: BIT #BIT2,ARXCSR ;SEE IF RXCSR BIT2 IS CLEAR.
      BEQ APB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RXCSR BIT2 IS NOT CLEAR.
APB: SCOPE ;SCOPE
;*****
AT20: 20 ;TEST NUMBER 20 *
      AT21 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AQA ;SCOPE ENTRY POINT *
;*****
;TEST THAT RXCSR BIT3 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
AQA: BIT #BIT3,ARXCSR ;SEE IF RXCSR BIT3 IS CLEAR.
      BEQ AQB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RESET DID NOT CLEAR RXCSR BIT3
      BR AQA
AQB: BIS #BIT3,ARXCSR ;SET RXCSR BIT3.
      BIT #BIT3,ARXCSR ;SEE IF BIT IS SET.
      BNE AQC ;BRANCH IF BIT IS SET.
      ERROR ;RXCSR BIT3 FAILED TO SET.
      BR AQA
AQC: BIC #BIT3,ARXCSR ;CLEAR RXCSR BIT3
      BIT #BIT3,ARXCSR ;SEE IF BIT IS CLEAR.
      BEQ AQA
      ERROR ;RXCSR BIT3 FAILED TO CLEAR.
AQA: BIS #BIT3,ARXCSR ;SET RXCSR BIT3.
      SRESET ;ISSUE RESET TO CLEAR BIT.
      SCOPE ;SCOPE
;*****
AT21: 21 ;TEST NUMBER 21 *
      AT22 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      ARA ;SCOPE ENTRY POINT *
;*****
;TEST THAT RXCSR BIT4 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
ARA: BIT #BIT4,ARXCSR ;SEE IF RXCSR BIT4 IS CLEAR.
      BEQ ARB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RESET DID NOT CLEAR RXCSR BIT4
      BR ARA
ARB: BIS #BIT4,ARXCSR ;SET RXCSR BIT4.
      BIT #BIT4,ARXCSR ;SEE IF BIT IS SET.
      BNE ARC ;BRANCH IF BIT IS SET.
      ERROR ;RXCSR BIT4 FAILED TO SET.
      BR ARA
ARC: BIC #BIT4,ARXCSR ;CLEAR RXCSR BIT4
      BIT #BIT4,ARXCSR ;SEE IF BIT IS CLEAR.
      BEQ ARD
      ERROR ;RXCSR BIT4 FAILED TO CLEAR.

```

```

1503 005432 052777 000020 173440 ARD:  BIS      #BIT4, @RXCSR  ;SET RXCSR BIT4.
1504 005440 104011          SRESET          ;ISSUE RESET TO CLEAR BIT.
1505 005442 104012          SCOPE             ;SCOPE
1506
1507
1508 005444 000022          AT22:  22             ;TEST NUMBER 22
1509 005446 005470          AT23          ;ADDRESS OF NEXT TEST
1510 005450 000144          100.          ;TEST ITERATION COUNT
1511 005452 005454          ARBA          ;SCOPE ENTRY POINT
1512
1513          ;*****
1514          ;TEST THAT PARITY INDICATOR (BITS RXCSR) IS CLEAR
1515          ;AND CAN BE READ RELIABLY.
1516 005454 032777 000040 173416 ARBA:  BIT      #BITS, @RXCSR  ;SEE IF PARITY INDICATOR IS CLEAR
1517 005462 001401          BEQ      ARBB          ;BRANCH IF CLEAR
1518 005464 104003          ERROR          ;IS NOT CLEAR
1519 005466 104012          ARBB:  SCOPE         ;SCOPE
1520
1521          ;*****
1522 005470 000023          AT23:  23             ;TEST NUMBER 23
1523 005472 005576          AT24          ;ADDRESS OF NEXT TEST
1524 005474 000144          100.          ;TEST ITERATION COUNT
1525 005476 005500          ASA          ;SCOPE ENTRY POINT
1526
1527          ;*****
1528          ;TEST THAT RXCSR BIT6 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
1528 005500 012767 000340 172270 ASA:  MOV      #PRTY7, PSW  ;SET PRIORITY 7.
1529 005506 032777 000100 173364          BIT      #BIT6, @RXCSR  ;SEE IF RXCSR BIT6 IS CLEAR.
1530 005514 001402          BEQ      ASB          ;BRANCH IF BIT IS CLEAR.
1531 005516 104003          ERROR          ;RESET DID NOT CLEAR RXCSR BIT6
1532 005520 000421          BR      ASD          ;
1533 005522 052777 000100 173350 ASB:  BIS      #BIT6, @RXCSR  ;SET RXCSR BIT6.
1534 005530 032777 000100 173342          BIT      #BIT6, @RXCSR  ;SEE IF BIT IS SET.
1535 005536 001002          BNE      ASC          ;BRANCH IF BIT IS SET.
1536 005540 104003          ERROR          ;RXCSR BIT6 FAILED TO SET.
1537 005542 000410          BR      ASD          ;
1538 005544 042777 000100 173326 ASC:  BIC      #BIT6, @RXCSR  ;CLEAR RXCSR BIT6
1539 005552 032777 000100 173320          BIT      #BIT6, @RXCSR  ;SEE IF BIT IS CLEAR.
1540 005560 001401          BEQ      ASD          ;
1541 005562 104003          ERROR          ;RXCSR BIT6 FAILED TO CLEAR.
1542 005564 052777 000100 173306 ASD:  BIS      #BIT6, @RXCSR  ;SET RXCSR BIT6.
1543 005572 104011          SRESET          ;ISSUE RESET TO CLEAR BIT.
1544 005574 104012          SCOPE             ;SCOPE
1545
1546 005576 000024          AT24:  24             ;TEST NUMBER IS 24
1547 005600 005622          AT25          ;ADDRESS OF NEXT TEST
1548 005602 000144          100.          ;TEST ITERATION COUNT
1549 005604 005606          ATA          ;SCOPE ENTRY POINT
1550
1551          ;*****
1552          ;TEST THAT RXCSR BIT7 IS CLEAR AND CAN BE READ RELIABLY.
1552 005606 032777 000200 173264 ATA:  BIT      #BIT7, @RXCSR  ;SEE IF RXCSR BIT7 IS CLEAR.
1553 005614 001401          BEQ      ATB          ;BRANCH IF BIT IS CLEAR.
1554 005616 104003          ERROR          ;RXCSR BIT7 IS NOT CLEAR.
1555 005620 104012          ATB:  SCOPE         ;SCOPE
1556
1557          ;*****
1558 005622 000025          AT25:  25             ;TEST NUMBER 25
  
```

```

1559 005624 005722 AT26 ;ADDRESS OF NEXT TEST *
1560 005626 000144 100. ;TEST ITERATION COUNT *
1561 005630 005632 AUA ;SCOPE ENTRY POINT *
1562 ;*****
1563 ;TEST THAT RXCSR BIT8 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
1564 005632 032777 000400 173240 AUA: BIT #BIT8, @RXCSR ;SEE IF RXCSR BIT8 IS CLEAR.
1565 005640 001402 BEQ AVB ;BRANCH IF BIT IS CLEAR.
1566 005642 104003 ERROR ;RESET DID NOT CLEAR RXCSR BIT8
1567 005644 000421 BR AVD
1568 005646 052777 000400 173224 AVB: BIS #BIT8, @RXCSR ;SET RXCSR BIT8.
1569 005654 032777 000400 173216 AVB: BIT #BIT8, @RXCSR ;SEE IF BIT IS SET.
1570 005662 001002 BNE AVC ;BRANCH IF BIT IS SET.
1571 005664 104003 ERROR ;RXCSR BIT8 FAILED TO SET.
1572 005666 000410 BR AVD
1573 005670 042777 000400 173202 AVC: BIC #BIT8, @RXCSR ;CLEAR RXCSR BIT8
1574 005676 032777 000400 173174 AVC: BIT #BIT8, @RXCSR ;SEE IF BIT IS CLEAR.
1575 005704 001401 BEQ AVD
1576 005706 104003 ERROR ;RXCSR BIT8 FAILED TO CLEAR.
1577 005710 052777 000400 173162 AVD: BIS #BIT8, @RXCSR ;SET RXCSR BIT8.
1578 005716 104011 SRESET ;ISSUE RESET TO CLEAR BIT.
1579 005720 104012 SCOPE ;SCOPE
1580 ;*****
1581 005722 000026 AT26: 26 ;TEST NUMBER 26 *
1582 005724 006022 AT27 ;ADDRESS OF NEXT TEST *
1583 005726 000144 100. ;TEST ITERATION COUNT *
1584 005730 005732 AVA ;SCOPE ENTRY POINT *
1585 ;*****
1586 ;TEST THAT RXCSR BIT9 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
1587 005732 032777 001000 173140 AVA: BIT #BIT9, @RXCSR ;SEE IF RXCSR BIT9 IS CLEAR.
1588 005740 001402 BEQ AVB ;BRANCH IF BIT IS CLEAR.
1589 005742 104003 ERROR ;RESET DID NOT CLEAR RXCSR BIT9
1590 005744 000421 BR AVD
1591 005746 052777 001000 173124 AVB: BIS #BIT9, @RXCSR ;SET RXCSR BIT9.
1592 005754 032777 001000 173116 AVB: BIT #BIT9, @RXCSR ;SEE IF BIT IS SET.
1593 005762 001002 BNE AVC ;BRANCH IF BIT IS SET.
1594 005764 104003 ERROR ;RXCSR BIT9 FAILED TO SET.
1595 005766 000410 BR AVD
1596 005770 042777 001000 173102 AVC: BIC #BIT9, @RXCSR ;CLEAR RXCSR BIT9
1597 005776 032777 001000 173074 AVC: BIT #BIT9, @RXCSR ;SEE IF BIT IS CLEAR.
1598 006004 001401 BEQ AVD
1599 006006 104003 ERROR ;RXCSR BIT9 FAILED TO CLEAR.
1600 006010 052777 001000 173062 AVD: BIS #BIT9, @RXCSR ;SET RXCSR BIT9.
1601 006016 104011 SRESET ;ISSUE RESET TO CLEAR BIT.
1602 006020 104012 SCOPE ;SCOPE
1603 ;*****
1604 006022 000027 AT27: 27 ;TEST NUMBER 27 *
1605 006024 006122 AT30 ;ADDRESS OF NEXT TEST *
1606 006026 000144 100. ;TEST ITERATION COUNT *
1607 006030 006032 AWA ;SCOPE ENTRY POINT *
1608 ;*****
1609 ;TEST THAT RXCSR BIT10 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
1610 006032 032777 002000 173040 AWA: BIT #BIT10, @RXCSR ;SEE IF RXCSR BIT10 IS CLEAR.
1611 006040 001402 BEQ AWB ;BRANCH IF BIT IS CLEAR.
1612 006042 104003 ERROR ;RESET DID NOT CLEAR RXCSR BIT10
1613 006044 000421 BR AWD
1614 006046 052777 002000 173024 AWB: BIS #BIT10, @RXCSR ;SET RXCSR BIT10.
  
```

```

1615 006054 032777 002000 173016 BIT #BIT10,@RXCSR ;SEE IF BIT IS SET.
1616 006062 001002 BNE AWC ;BRANCH IF BIT IS SET.
1617 006064 104003 ERROR ;RXCSR BIT10 FAILED TO SET.
1618 006066 000410 BR AWD
1619 006070 042777 002000 173002 AWC: BIC #BIT10,@RXCSR ;CLEAR RXCSR BIT10
1620 006076 032777 002000 172774 BIT #BIT10,@RXCSR ;SEE IF BIT IS CLEAR.
1621 006104 001401 BEQ AWD
1622 006106 104003 ERROR ;RXCSR BIT10 FAILED TO CLEAR.
1623 006110 052777 002000 172762 AWD: BIS #BIT10,@RXCSR ;SET RXCSR BIT10.
1624 006116 104011 SRESET ;ISSUE RESET TO CLEAR BIT.
1625 006120 104012 SCOPE ;SCOPE
1626 *****
1627 006122 000030 AT30: 30 ;TEST NUMBER 30 *
1628 006124 006146 AT31 ;ADDRESS OF NEXT TEST *
1629 006126 000144 100. ;TEST ITERATION COUNT *
1630 006130 006132 AXA ;SCOPE ENTRY POINT *
1631 *****
1632 ;TEST THAT RXCSR BIT12 IS CLEAR AND CAN BE READ RELIABLY.
1633 006132 032777 010000 172740 AXA: BIT #BIT12,@RXCSR ;SEE IF RXCSR BIT12 IS CLEAR.
1634 006140 001401 BEQ AXB ;BRANCH IF BIT IS CLEAR.
1635 006142 104003 ERROR ;RXCSR BIT12 IS NOT CLEAR.
1636 006144 104012 AXB: SCOPE ;SCOPE
1637 *****
1638 ;TEST THAT RXCSR BIT13 IS CLEAR AND CAN BE READ RELIABLY.
1639 006146 000031 AT31: 31 ;TEST NUMBER 31 *
1640 006150 006172 AT32 ;ADDRESS OF NEXT TEST *
1641 006152 000144 100. ;TEST ITERATION COUNT *
1642 006154 006156 AYA ;SCOPE ENTRY POINT *
1643 *****
1644 ;TEST THAT RXCSR BIT13 IS CLEAR AND CAN BE READ RELIABLY.
1645 006156 032777 020000 172714 AYA: BIT #BIT13,@RXCSR ;SEE IF RXCSR BIT13 IS CLEAR.
1646 006164 001401 BEQ AYB ;BRANCH IF BIT IS CLEAR.
1647 006166 104003 ERROR ;RXCSR BIT13 IS NOT CLEAR.
1648 006170 104012 AYB: SCOPE ;SCOPE
1649

```

```

1650
1651
1652 006172 000032
1653 006174 006216
1654 006176 000144
1655 006200 006202
1656
1657
1658 006202 032777 040000 172670
1659 006210 001401
1660 006212 104003
1661 006214 104012
1662
1663 006216 000033
1664 006220 006242
1665 006222 000144
1666 006224 006226
1667
1668
1669 006226 032777 100000 172644
1670 006234 001401
1671 006236 104003
1672 006240 104012

;*****
AT32: 32 ;TEST NUMBER 32 *
      AT33 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AZA ;SCOPE ENTRY POINT *
;*****
;TEST THAT RXCSR BIT14 IS CLEAR AND CAN BE READ RELIABLY.
AZA: BIT #BIT14, @RXCSR ;SEE IF RXCSR BIT14 IS CLEAR.
      BEQ AZB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RXCSR BIT14 IS NOT CLEAR.
AZB: SCOPE ;SCOPE
;*****
AT33: 33 ;TEST NUMBER 33 *
      AT34 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AAAA ;SCOPE ENTRY POINT *
;*****
;TEST THAT RXCSR BIT15 IS CLEAR AND CAN BE READ RELIABLY.
AAAA: BIT #BIT15, @RXCSR ;SEE IF RXCSR BIT15 IS CLEAR.
      BEQ AAAB ;BRANCH IF BIT IS CLEAR.
      ERROR ;RXCSR BIT15 IS NOT CLEAR.
AAAB: SCOPE ;SCOPE

```

1673  
 1674  
 1675  
 1676  
 1677  
 1678  
 1679  
 1680  
 1681  
 1682  
 1683  
 1684  
 1685  
 1686  
 1687  
 1688  
 1689  
 1690  
 1691  
 1692  
 1693  
 1694  
 1695  
 1696  
 1697  
 1698  
 1699  
 1700  
 1701  
 1702  
 1703  
 1704  
 1705  
 1706  
 1707  
 1708  
 1709  
 1710  
 1711  
 1712  
 1713  
 1714  
 1715  
 1716  
 1717  
 1718  
 1719  
 1720  
 1721  
 1722  
 1723  
 1724  
 1725  
 1726  
 1727  
 1728

006242 000034  
 006244 006320  
 006246 000144  
 006250 006252  
  
 006252 052777 000001 172620  
 006260 032777 000004 172612  
 006266 001002  
 006270 104003  
 006272 000410  
 006274 042777 000001 172576  
 006302 032777 000004 172570  
 006310 001401  
 006312 104003  
 006314 104011  
 006316 104012  
  
 006320 000035  
 006322 006460  
 006324 000144  
 006326 006330  
  
 006330 042777 000001 172542  
 006336 017767 172536 172714  
 006344 032777 040000 172526  
 006352 001402  
 006354 104003  
 006356 000436  
 006360 005277 172514  
 006364 000004  
  
 006366 017767 172506 172664  
 006374 032767 040000 172656  
 006402 001002  
 006404 104003  
 006406 000422  
 006410 032777 040000 172462  
 006416 001402  
 006420 104003  
 006422 000414

```

;ALL PREVIOUS TESTS MUST HAVE BEEN RUN SUCCESSFULLY PRIOR
;TO RUNNING THE FOLLOWING TESTS. ALSO, THE JUMPER CONNECTOR
;MUST BE INSERTED IN THE DC11 CABLE. TO THE MODEM. COMMENTS
;REFER TO OPERATION WITH JUMPER INSERTED.
;
;*****
AT34: 34 ;TEST NUMBER 34
      AT35 ;ADDRESS OF NEXT TEST
      100. ;TEST ITERATION COUNT
      AFBA ;SCOPE ENTRY POINT
;*****
;TEST THAT CARRIER DETECT SETS AND CLEARS WHEN DATA TERMINAL
;READY SETS AND CLEARS.
;
AFBA: BIS #BIT0, @RXCSR ;SET DATA TERMINAL READY
      BIT #BIT2, @RXCSR ;TEST CARRIER DETECT
      BNE AFBB ;SHOULD BE SET
      ERROR ;WASN'T
      BR AFBC
AFBB: BIC #BIT0, @RXCSR ;CLEAR DATA TERMINAL READY
      BIT #BIT2, @RXCSR ;TEST CARRIER DETECT
      BEQ AFBC
      ERROR ;WAS SET, ERROR
      SRESET
      SCOPE
;*****
AT35: 35 ;TEST NUMBER 35
      AT36 ;ADDRESS OF NEXT TEST
      100. ;TEST ITERATION COUNT
      AGBA ;SCOPE ENTRY POINT
;*****
;TEST THAT CARRIER TRANSITION (BIT 14) SETS WHEN CARRIER DETECT
;CHANGES STATE, AND IS CLEARED WHEN RXCSR IS READ.
;
AGBA: BIC #BIT0, @RXCSR ;CLEAR DATA TERMINAL READY
      MOV @RXCSR, RXCSRT ;READ RXCSR
      BIT #BIT14, @RXCSR ;TEST CARRIER TRANSITION
      BEQ AGBB ;WAS CLEAR GO TO AGBB
      ERROR ;WASN'T CLEAR
      BR AGBE ;GO TO SCOPE
AGBB: INC @RXCSR ;SETTING DATA TERMINAL READY
      IOT ;CAUSES CARRIER DETECT TO SET
      ;WHICH CAUSES CARRIER TRANSITION
      ;TO SET.
      MOV @RXCSR, RXCSRT ;MOVE RXCSR TO TEMPORARY LOCATION
      BIT #BIT14, RXCSRT ;TEST CARRIER TRANSITION
      BNE AGBC ;SHOULD BE SET GO TO AGBC
      ERROR ;WAS CLEAR
      BR AGBE ;GO TO SCOPE
AGBC: BIT #BIT14, @RXCSR ;CARRIER TRANSITION BIT SHOULD
      ;HAVE BEEN CLEARED
      BEQ AGBD ;IT WAS GO TO AGBD
      ERROR ;IT WASN'T
      BR AGBE ;GO TO SCOPE
  
```

```

1729 006424 042777 000001 172446 AGBD: BIC #BIT0,@RXCSR ;CLEARING DATA TERMINAL READY
1730 ;CAUSES CARRIER DETECT TO CLEAR
1731 ;BUT CARRIER TRANSITION
1732 ;WILL NOT SET
1733 006432 017767 172442 172620 MOV @RXCSR,RXCSRT ;MOV RXCSR TO TEMPORARY LOCATION
1734 006440 032767 040000 172612 BIT #BIT14,RXCSRT ;TEST CARRIER TRANSITION
1735 006446 001402 BEQ AGBE ;SHOULD BE CLEAR
1736 006450 104003 ERROR ;IT WASN'T
1737 006452 000400 BR AGBE
1738 006454 104011 AGBE: SRESET ;ISSUE RESET
1739 006456 104012 SCOPE ;SCOPE
1740 ;*****
1741
1742 006460 000036 AT36: 36 ;TEST NUMBER 36
1743 006462 006574 AT37 ;ADDRESS OF NEXT TEST
1744 006464 000144 100 ;TEST ITERATION COUNT
1745 006466 006470 AMBA ;SCOPE ENTRY POINT
1746 ;*****
1747 ;TEST THAT CARRIER TRANSITION SETTING CAUSES ERROR (BIT 15 RXCSR) TO
1748 ;SET AND THAT READING RXCSR CLEARS ERROR.
1749
1750 006470 042777 000001 172402 AMBA: BIC #BIT0,@RXCSR ;CLEAR DATA TERMINAL READY
1751 006476 005277 172376 INC @RXCSR ;SET DATA TERMINAL READY
1752 006502 000004 IOT
1753 006504 017767 172370 172546 MOV @RXCSR,+RXCSRT ;MOVE RXCSR TO TEMPORARY LOCATION
1754 006512 032767 100000 172540 BIT #BIT15,RXCSRT ;TEST ERROR BIT
1755 006520 001002 BNE AMBB ;ERROR BIT SHOULD BE SET
1756 006522 104003 ERROR
1757 006524 000421 BR AMBD
1758 006526 032777 100000 172344 AMBB: BIT #BIT15,@RXCSR ;TEST ERROR BIT
1759 006534 001402 BEQ AMBC ;SHOULD BE CLEAR
1760 006536 104003 ERROR
1761 006540 000413 BR AMBD
1762 006542 042777 000001 172330 AMBC: BIC #BIT0,@RXCSR ;CLEAR DATA TERMINAL READY
1763 006550 017767 172324 172502 MOV @RXCSR,RXCSRT ;MOV RXCSR TO TEMPORARY LOCATION
1764 006556 032767 100000 172474 BIT #BIT15,RXCSRT ;TEST ERROR BIT
1765 006564 001401 BEQ AMBD ;SHOULD BE CLEAR
1766 006566 104003 ERROR
1767 006570 104011 AMBD: SRESET ;ISSUE RESET
1768 006572 104012 SCOPE ;SCOPE
  
```

```

1769
1770 006574 000037
1771 006576 006670
1772 006600 000144
1773 006602 006604
1774
1775
1776
1777
1778 006604 042777 000001 172266 AJBA: BIC #BIT0, @RXCSR ;CLEAR DATA TERMINAL READY
1779 006612 032777 000004 172264 BIT #BIT2, @TXCSR ;TEST CLEAR TO SEND
1780 006620 001400 BEQ AJBB
1781
1782
1783 006622 052777 000001 172250 AJBB: BIS #BIT0, @RXCSR ;SET DATA TERMINAL READY
1784 006630 032777 000002 172246 BIT #BIT1, @TXCSR ;TEST CLEAR TO SEND
1785 006636 001002 BNE AJBC ;BRANCH IF SET
1786 006640 104003 ERROR ;CLEAR TO SEND SHOULD BE SET
1787 006642 000410 BR AJBD
1788 006644 042777 000001 172226 AJBC: BIC #BIT0, @RXCSR ;CLEAR DATA TERMINAL READY
1789 006652 032777 000002 172224 BIT #BIT1, @TXCSR ;TEST CLEAR TO SEND
1790 006660 001401 BEQ AJBD
1791 006662 104003 ERROR ;CLEAR TO SEND SHOULD BE CLEAR
1792 006664 104011 AJBD: SRESET ;ISSUE RESET
1793 006666 104012 SCOPE ;SCOPE
1794
1795 006670 000040
1796 006672 007006
1797 006674 000144
1798 006676 006700
1799
1800
1801
1802
1803
1804 006700 042777 000001 172176 AKBA: BIC #BIT0, @TXCSR ;CLEAR REQUEST TO SEND
1805 006706 052777 000001 172170 AKBB: BIS #BIT0, @TXCSR ;SET REQUEST TO SEND
1806 006714 042777 000001 172162 BIC #BIT0, @TXCSR
1807 006722 032777 020000 172150 BIT #BIT13, @RXCSR ;TEST RING
1808 006730 001002 BNE AKBC ;RING SHOULD BE SET
1809 006732 104003 ERROR
1810 006734 000422 BR AKBE
1811 006736 032777 020000 172134 AKBC: BIT #BIT13, @RXCSR ;TEST RING
1812 006744 001402 BEQ AKBD ;RING SHOULD BE CLEAR
1813 006746 104003 ERROR
1814 006750 000414 BR AKBE
1815 006752 052777 000001 172124 AKBD: BIS #BIT0, @TXCSR ;SET
1816 006760 042777 000001 172116 BIC #BIT0, @TXCSR ;RING
1817 006766 000005 RESET
1818 006770 032777 020000 172102 BIT #BIT13, @RXCSR ;TEST RING
1819 006776 001401 BEQ AKBE ;BRANCH IF CLEAR
1820 007000 104003 ERROR ;RING SHOULD BE CLEAR AFTER RESET
1821 ;BUT WAS SET
1822 007002 104011 AKBE: SRESET ;ISSUE RESET
1823 007004 104012 SCOPE ;SCOPE
1824

```

1825 007006 000041  
 1826 007010 007114  
 1827 007012 000144  
 1828 007014 007016  
 1829  
 1830  
 1831  
 1832 007016 042777  
 1833 007024 032777  
 1834 007032 001402  
 1835 007034 104003  
 1836 007036 000424  
 1837 007040 052777  
 1838 007046 042777  
 1839 007054 032777  
 1840 007062 001002  
 1841 007064 104003  
 1842 007066 000410  
 1843 007070 042777  
 1844 007076 032777  
 1845 007104 001401  
 1846 007106 104003  
 1847 007110 104011  
 1848 007112 104012

AT41: 41 ;TEST NUMBER 41  
 AT42 ;ADDRESS OF NEXT TEST  
 100. ;TEST ITERATION COUNT  
 A0BA ;SCOPE ENTRY POINT  
 ;\*\*\*\*\*  
 ;TEST THAT ERROR (BIT 15 RXCSR) SETS WHEN RING SETS.  
 A0BA: BIC #BIT0,@TXCSR ;SET REQUEST TO SEND  
 BIT #BIT15,@RXCSR ;TEST ERROR BIT  
 BEQ A0BB  
 ERROR  
 BR A0BD  
 A0BB: BIS #BIT0,@TXCSR ;SET REQUEST TO SEND  
 BIC #BIT0,@TXCSR ;CLEAR REQUEST TO SEND  
 BIT #BIT15,@RXCSR ;TEST ERROR BIT  
 BNE A0BC  
 ERROR  
 BR A0BD  
 A0BC: BIC #BIT0,@TXCSR ;CLEAR REQUEST TO SEND  
 BIT #BIT15,@RXCSR ;TEST ERROR BIT  
 BEQ A0BD  
 ERROR  
 A0BD: SRESET ;ISSUE RESET  
 SCOPE ;SCOPE

```

1849
1850
1851 007114 000042
1852 007116 007214
1853 007120 000144
1854 007122 007124
1855
1856
1857
1858
1859 007124 042777 000400 171746 ALBA: BIC #BIT8,@TXCSR ;CLEAR SUPERVISOR XMIT DATA
1860 007132 032777 100000 171744 BIT #BIT15,@TXCSR ;TEST SUPERVISORY RECEIVE DATA.
1861 007140 001402 BEQ ALBB
1862 007142 104003 ERROR ;SHOULD HAVE BEEN CLEAR
1863 007144 000421 BR ALBD
1864 007146 052777 000400 171724 ALBB: BIS #BIT8,@TXCSR ;SET SUPERVISORY XMIT DATA
1865 007154 032777 100000 171722 BIT #BIT15,@TXCSR ;TEST SUPERVISORY RECEIVE DATA
1866 007162 001002 BNE ALBC
1867 007164 104003 ERROR ;SHOULD HAVE BEEN SET
1868 007166 000410 BR ALBD
1869 007170 042777 000400 171702 ALBC: BIC #BIT8,@TXCSR ;CLEAR SUPERVISORY XMIT DATA
1870 007176 032777 100000 171700 BIT #BIT15,@TXCSR ;TEST SUPERVISORY RECEIVE DATA
1871 007204 001401 BEQ ALBD
1872 007206 104003 ERROR ;SHOULD HAVE BEEN CLEAR
1873 007210 104011 ALBD: SRESET ;ISSUE RESET
1874 007212 104012 SCOPE ;SCOPE
1875
1876 007214 000043
1877 007216 007320
1878 007220 000144
1879 007222 007224
1880
1881
1882 007224 012767 000340 170544 ABAA: MOV #PRTY7,PSW ;SET PRIORITY 7.
1883 007232 012777 177777 171644 MOV #-1,@TXCSR ;SET ALL POSSIBLE BITS IN TXCSR
1884 007240 104011 SRESET ;ISSUE RESET TO CLEAR BITS
1885 007242 022777 000200 171634 CMP #BIT7,@TXCSR ;SEE IF ONLY BIT 7 IS SET.
1886 007250 001422 BEQ ABAB ;BRANCH IF ONLY BIT 7 IS SET
1887 007252 017767 171626 171776 MOV @TXCSR,TXCRT ;SAVE CONTENTS OF TXCSR
1888 007260 012767 000200 171776 MOV #BIT7,TEMP ;MOVE EXPECTED TXCSR TO TEMP.
1889 007266 004567 173740 JSR %5,0ACNV ;GO TO OCTAL TO ASCII CONVERT.
1890 007272 001264 TEMP ;SOURCE ADDR.
1891 007274 015403 ATXSB ;DESTINATION ADDR.
1892 007276 000006 6 ;#OF DIGITS TO CONVERT.
1893 007300 004567 173726 JSR %5,0ACNV ;GO TO OCTAL TO ASCII CONVERT.
1894 007304 001256 TXCRT ;SOURCE ADDR.
1895 007306 015420 ATXWAS ;DESTINATION ADDR.
1896 007310 000006 6 ;#OF DIGITS TO CONVERT.
1897 007312 104015 ERROR1 ;RESET FAILED TO CLEAR ALL BITS EXCEPT
1898 007314 015370 ATXCSR ;BIT 7 - SEE PRINTOUT
1899 007316 104012 ABAB: SCOPE ;SCOPE

```

```

1900 ;*****
1901 007320 000044 AT44: 44 ;TEST NUMBER 44 *
1902 007322 007474 AT45 ;ADDRESS OF NEXT TEST *
1903 007324 000144 100. ;TEST ITERATION COUNT *
1904 007326 007330 ACAA ;SCOPE ENTRY POINT *
1905 ;*****
1906 ;TEST THAT RESET CLEARS ALL RXCSR BITS EXCEPT BIT 0 (DATA TERMINAL READY)
1907 ;RING, AND THE BREAK BIT.
1908 007330 012767 000340 170440 ACAA: MOV #PTY7,PSW ;SET PRIORITY 7
1909 007336 042777 000001 171534 BIC #BIT0,@RXCSR ;CLEAR DATA TERM.READY
1910 007344 012777 177775 171526 MOV #177775,@RXCSR ;SET ALL POSSIBLE BITS IN RXCSR
1911 007352 052777 000030 171524 BIS #30,@TXCSR ;SET MAINT BIT
1912 007360 005077 171522 CLR @TXBUF ;TRANSMIT A CHAR
1913 007364 105777 171514 TSTB @TXCSR ;WAIT FOR
1914 007370 100375 BPL .-4 ;TRANSMITTER TO FINISH
1915 007372 012777 000001 171506 MOV #1,@TXBUF ;TRANSMIT ANOTHER CHAR.
1916 007400 105777 171500 TSTB @TXCSR ;WAIT FOR
1917 007404 100375 BPL .-4 ;TRANSMITTER TO FINISH
1918 007406 104011 SRESET ;ISSUE RESET TO CLEAR BITS.
1919 007410 017767 171464 171642 MOV @RXCSR,RXCSRT ;MOVE RXCSR CONTENTS TO RXCSRT
1920 007416 022767 000005 171634 CMP #5,RXCSRT ;SEE IF ONLY BIT 0 IS SET
1921 007424 001417 BEQ ACAB ;BRANCH IF ONLY BIT 0 IS SET.
1922 007426 012767 000005 171630 MOV #5,TEMP
1923 007434 004567 173572 JSR %5,OACNV ;GO TO OCTAL TO ASCII CONVERT.
1924 007440 001264 TEMP ;SOURCE ADDR.
1925 007442 015442 ARXSB ;DESTINATION ADDR.
1926 007444 000006 6 ;#OF DIGITS TO CONVERT.
1927 007446 004567 173560 JSR %5,OACNV ;GO TO OCTAL TO ASCII CONVERT.
1928 007452 001260 RXCSRT ;SOURCE ADDR.
1929 007454 015457 ARXWAS ;DESTINATION ADDR.
1930 007456 000006 6 ;#OF DIGITS TO CONVERT.
1931 007460 104015 ERROR1 ;RESET FAILED TO CLEAR ALL BITS EXCEPT
1932 007462 015427 ARXCSR ;BIT 0. SEE ERROR PRINTOUT.
1933 007464 042777 000001 171406 ACAB: BIC #BIT0,@RXCSR ;CLEAR DATA TERM. READY
1934 007472 104012 SCOPE ;SCOPE
1935 ;*****
1936 007474 000045 AT45: 45 ;TEST NUMBER 45 *
1937 007476 007524 AT46 ;ADDRESS OF NEXT TEST *
1938 007500 000144 100. ;TEST ITERATION COUNT *
1939 007502 007504 ADAA ;SCOPE ENTRY POINT *
1940 ;*****
1941 ;TEST THAT LOADING TXBUF (TRANSMITTER BUFFER) CLEARS TXCSR BIT 7 (READY)
1942 007504 005077 171376 ADAA: CLR @TXBUF ;LOAD TXBUF
1943 007510 105777 171370 TSTB @TXCSR ;TEST TXCSR BIT 7 (READY BIT)
1944 007514 100001 BPL ADAB ;BRANCH IF BIT NOT SET.
1945 007516 104003 ERROR ;ERROR. LOADING TXBUF FAILED TO CLEAR READY.
1946 007520 104011 SRESET ;ISSUE RESET TO SET READY.
1947 007522 104012 SCOPE ;SCOPE.

```

```

1948
1949 007524 000046
1950 007526 007556
1951 007530 000012
1952 007532 007534
1953
1954
1955
1956 007534 005077 171346
1957 007540 104016
1958 007542 001750
1959 007544 105777 171334
1960 007550 100401
1961 007552 104003
1962
1963 007554 104012
1964
1965
1966 007556 000047
1967 007560 007616
1968 007562 000012
1969 007564 007566
1970
1971
1972
1973 007566 052777 000010 171310
1974 007574 005077 171306
1975 007600 104016
1976 007602 000764
1977 007604 105777 171274
1978 007610 100401
1979 007612 104003
1980
1981 007614 104012
1982
1983 007616 000050
1984 007620 007656
1985 007622 000012
1986 007624 007626
1987
1988
1989
1990 007626 052777 000020 171250
1991 007634 005077 171246
1992 007640 104016
1993 007642 000620
1994 007644 105777 171234
1995 007650 100401
1996 007652 104003
1997
1998 007654 104012

;*****
AT46: 46 ;TEST NUMBER 46 *
      AT47 ;ADDRESS OF NEXT TEST *
      10. ;TEST ITERATION COUNT *
      AEA4 ;SCOPE ENTRY POINT *
;*****
;TEST THAT READY BIT (TXCSR BIT 7) BECOMES SET NO LATER THAN 1000 MSECS AFTER
;LOADING TXBUF WITH TRANSMIT SPEED SET TO 00 (TXCSR BITS 3 AND 4)
AEA4: CLR @TXBUF ;LOAD TXBUF
      DELAY ;DELAY 1000 MSECS APPROX.
      1000.
      TSTB @TXCSR ;SEE IF READY BIT IS SET
      BMI AEA4 ;BRANCH IF READY IS SET
      ERROR ;READY NOT SET 200 MSECS AFTER BUFFER
      ;LOAD TX SPEED = 00.
AEA4: SCOPE ;SCOPE

;*****
AT47: 47 ;TEST NUMBER 47 *
      AT50 ;ADDRESS OF NEXT TEST *
      10. ;TEST ITERATION COUNT *
      AFA4 ;SCOPE ENTRY POINT *
;*****
;TEST THAT READY BIT (TXCSR BIT 7) BECOMES SET NO LATER THAN 500 MSECS AFTER
;LOADING TXBUF, WITH TRANSMIT SPEED SET TO 01 (TXCSR BITS 3 AND 4).
AFA4: BIS #10,@TXCSR ;SET TX SPEED TO 01.
      CLR @TXBUF ;LOAD TXBUF
      DELAY ;DELAY 500 MSECS
      500.
      TSTB @TXCSR ;SEE IF READY BIT IS SET
      BMI AFAB ;BRANCH IF READY IS SET
      ERROR ;READY NOT SET 200 MSECS AFTER BUFFER
      ;LOAD TX SPEED = 01.
AFAB: SCOPE ;SCOPE.

;*****
AT50: 50 ;TEST NUMBER 50 *
      AT51 ;ADDRESS OF NEXT TEST *
      10. ;TEST ITERATION COUNT *
      AGAA ;SCOPE ENTRY POINT *
;*****
;TEST THAT READY BIT (TXCSR BIT 7) BECOMES SET NO LATER THAN 400 MSECS AFTER
;LOADING TXBUF, WITH TRANSMIT SPEED SET TO 10 (TXCSR BITS 3 AND 4).
AGAA: BIS #20,@TXCSR ;SET TX SPEED TO 10.
      CLR @TXBUF ;LOAD TXBUF
      DELAY ;DELAY 400 MSECS
      400.
      TSTB @TXCSR ;SEE IF READY BIT IS SET
      BMI AGAB ;BRANCH IF READY BIT IS SET
      ERROR ;READY NOT SET 200 MSECS AFTER BUFFER
      ;LOAD TX SPEED = 10.
AGAB: SCOPE ;SCOPE
  
```

```

1999
2000 007656 000051
2001 007660 007716
2002 007662 000012
2003 007664 007666
2004
2005
2006
2007 007666 052777 000030 171210
2008 007674 005077 171206
2009 007700 104016
2010 007702 000372
2011 007704 105777 171174
2012 007710 100401
2013 007712 104003
2014
2015 007714 104012
2016
2017
2018 007716 000052
2019 007720 010134
2020 007722 000144
2021 007724 007726
2022
2023
2024
2025 007726 005067 171314
2026 007732 005067 171312
2027 007736 005067 171310
2028 007742 005067 171306
2029 007746 042777 000030 171130
2030 007754 004767 000110
2031 007760 066767 000146 171260
2032 007766 052777 000010 171110
2033 007774 004767 000070
2034 010000 066767 000126 171242
2035 010006 042777 000030 171070
2036 010014 052777 000020 171062
2037 010022 004767 000042
2038 010026 066767 000100 171216
2039 010034 052777 000030 171042
2040 010042 004767 000022
2041 010046 066767 000060 171200
2042 010054 004767 004176
2043 010060 000402
2044 010062 104015
2045 010064 015466
2046 010066 104012
2047 010070 005067 000036
2048 010074 105777 171004
2049 010100 100375
2050 010102 104016
2051 010104 000024
2052 010106 005077 170774
2053 010112 104016
2054 010114 000001

;*****
AT51: 51 ;TEST NUMBER 51 *
      AT52 ;ADDRESS OF NEXT TEST *
      10. ;TEST ITERATION COUNT *
      AHAA ;SCOPE ENTRY POINT *
;*****
;TEST THAT READY BIT (TXCSR BIT 7) BECOMES SET NO LATER THAN 250 MSECS AFTER
;LOADING TXBUF, WITH TRANSMIT SPEED SET TO 11 (TXCSR BITS 3 AND 4).
AHAA: BIS #30,@TXCSR ;SET TX SPEED TO 30.
      CLR @TXBUF ;LOAD TXBUF
      DELAY ;DELAY 250 MSECS.
      250.
      TSTB @TXCSR ;SEE IF READY BIT IS SET.
      BMI AHAB ;BRANCH IF READY BIT IS SET.
      ERROR ;READY NOT SET 200 MSECS AFTER
            ;BUFFER LOAD. TX SPEED = 11.
AHAB: SCOPE ;SCOPE
;*****
AT52: 52 ;TEST NUMBER 52 *
      AT53 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AIAA ;SCOPE ENTRY POINT *
;*****
;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: CLR CTRA ;CLEAR CTRA THROUGH CTRD
      CLR CTRB ;(USED TO COUNT ELAPSED TIME.)
      CLR CTCR
      CLR CTRD
      BIC #30,@TXCSR ;SELECT TX SPEED 0
      JSR %7,AIAS ;OUTPUT CHAR AND TIME.
      ADD AIAST,CTRA ;ADD ELAPSED TIME TO CTRA.
      BIS #10,@TXCSR ;SELECT TX SPEED 1
      JSR %7,AIAS ;OUTPUT CHAR AND TIME.
      ADD AIAST,CTRB ;ADD ELAPSED TIME TO CTRB.
      BIC #30,@TXCSR ;SELECT TX SPEED 2
      BIS #20,@TXCSR
      JSR %7,AIAS ;OUTPUT CHAR AND TIME.
      ADD AIAST,CTRC ;ADD ELAPSED TIME TO CTCR.
      BIS #30,@TXCSR ;SELECT TX SPEED 3
      JSR %7,AIAS ;OUTPUT CHAR AND TIME.
      ADD AIAST,CTRD ;ADD ELAPSED TIME TO CTRD.
      JSR %7,CMPT ;CHECK THAT CTRA THROUGH CTRD CONTAIN
            ;DESCENDING VALUES
            ;TRANSMIT SPEEDS NOT ARRANGED IN
            ;ASCENDING ORDER.
AIAF: SCOPE ;SCOPE
AIAA: CLR AIAST ;CLEAR ELAPSED TIME COUNTER.
      TSTB @TXCSR ;WAIT FOR TX READY.
      BPL .-4
      DELAY ;WAIT 20 MSECS.
      20.
AIAA: CLR @TXBUF ;LOAD TXBUF.
      DELAY ;DELAY 1 MSEC.
      1

```

```

2055 010116 005267 000010          INC      AIAST      ;INCREMENT ELAPSED TIME COUNTER.
2056 010122 105777 170756          TSTB     @TXCSR    ;READY SET?
2057 010126 100371                   BPL      AIASA    ;BRANCH IF READY NOT SET.
2058 010130 000207                   RTS      %7       ;EXIT.
2059 010132 000000                   AIAST:  OPEN
2060                                     ;*****
2061 010134 000053                   AT53:   53        ;TEST NUMBER 53 *
2062 010136 010320                   AT54    ;ADDRESS OF NEXT TEST *
2063 010140 000144                   100.    ;TEST ITERATION COUNT *
2064 010142 010144                   AJAA    ;SCOPE ENTRY POINT *
2065                                     ;*****
2066                                     ;TEST FOR CORRECT OPERATION OF STOP CODE BIT (TXCSR BIT 8) BY CHECKING THAT TIME.
2067                                     ;REQUIRED TO COMPLETE TRANSMISSION OF 2 CONSECUTIVE CHARACTERS WITH STOP BIT
2068                                     ;SET TO 0 IS LONGER THAN TIME REQUIRED WITH STOP CODE BIT SET TO A 1.
2069 010144 005067 171076          AJAA:   CLR      CTRA    ;CLEAR CTRA AND CTRB
2070 010150 005067 171074          CLR      CTRB    ;(ELAPSED TIME COUNTERS).
2071 010154 042777 000400 170722          BIC      #BIT8,@TXCSR ;SET STOP CODE TO 0 (2 STOP CODES)
2072 010162 004767 000044          JSR      %7,AJAS ;OUTPUT CHAR AND TIME
2073 010166 066767 000124 171052          ADD      AJAST,CTRA ;ADD ELAPSED TIME TO CTRA
2074 010174 052777 000400 170702          BIS      #BIT8,@TXCSR ;SET STOP CODE TO 1 (1 STOP CODE)
2075 010202 004767 000024          JSR      %7,AJAS ;OUTPUT CHARACTER AND TIME.
2076 010206 066767 000104 171034          ADD      AJAST,CTRB ;ADD ELAPSED TIME TO CTRB
2077 010214 026767 171026 171026          CMP      CTRA,CTRB ;SEE IF CTRA IS GREATER THAN CTRB
2078 010222 101002                   BHI      AJAB    ;BRANCH IF CTRA IS GREATER.
2079 010224 104015                   ERROR1  ;ERROR. ELAPSED TIME FOR 2 STOP CODE
2080 010226 015530                   ESTPCD  ;OPERATION NOT GREATER THAN FOR 1 STOP
2091                                     ;CODE.
2082 010230 104012                   AJAB:   SCOPE    ;SCOPE
2083 010232 005067 000060          AJAS:   CLR      AJAST ;CLEAR ELAPSED TIME COUNTER AJAST
2084 010236 105777 170642          TSTB     @TXCSR    ;WAIT FOR TX READY.
2085 010242 100375                   BPL      -.4
2086 010244 104016                   DELAY   ;WAIT 20 MSECS.
2087 010246 000024                   20.
2088 010250 005077 170632          AJASA: CLR      @TXBUF ;LOAD TXBUF
2089 010254 104016                   DELAY   ;DELAY 1 MSEC
2090 010256 000001                   1
2091 010260 005267 000032          INC      AJAST    ;INCREMENT ELAPSED TIME COUNTER
2092 010264 105777 170614          TSTB     @TXCSR    ;READY SET?
2093 010270 100371                   BPL      AIASA    ;BRANCH IF READY NOT SET.
2094 010272 005077 170610          CLR      @TXBUF   ;LOAD TXBUF.
2095 010276 104016                   AJASB: DELAY   ;DELAY 1 MSEC.
2096 010300 000001                   1
2097 010302 005267 000010          INC      AJAST    ;INCR ELAPSED TIME COUNTER.
2098 010306 105777 170572          TSTB     @TXCSR    ;READY SET?
2099 010312 100371                   BPL      AJASB    ;BRANCH IF READY NOT SET.
2100 010314 000207                   RTS      %7       ;EXIT
2101 010316 000000                   AJAST:  OPEN      ;ELAPSED TIME COUNTER.

```

2102									
2103									
2104	010320	000054							
2105	010322	010544							
2106	010324	000144							
2107	010326	010330							
2108									
2109									
2110									
2111									
2112	010330	005067	170712						
2113	010334	005067	170710						
2114	010340	005067	170706						
2115	010344	005067	170704						
2116	010350	042777	003000	170522					
2117	010356	004767	000116						
2118	010362	066767	000154	170656					
2119	010370	042777	003000	170502					
2120	010376	052777	001000	170474					
2121	010404	004767	000070						
2122	010410	066767	000126	170632					
2123	010416	042777	003000	170454					
2124	010424	052777	002000	170446					
2125	010432	004767	000042						
2126	010436	066767	000100	170606					
2127	010444	052777	003000	170426					
2128	010452	004767	000022						
2129	010456	066767	000060	170570					
2130	010464	004767	003566						
2131	010470	000402							
2132	010472	104015							
2133	010474	015601							
2134	010476	104012							
2135	010500	005067	000036						
2136	010504	105777	170374						
2137	010510	100375							
2138	010512	104016							
2139	010514	000024							
2140	010516	005077	170364						
2141	010522	104016							
2142	010524	000001							
2143	010526	005267	000010						
2144	010532	105777	170346						
2145	010536	100371							
2146	010540	000207							
2147	010542	000000							

```

*****
AT54: 54 ;TEST NUMBER 54 *
      AT55 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AKAA ;SCOPE ENTRY POINT *
*****
;TEST FOR CORRECT OPERATION OF CHARACTER LENGTH SELECTION (RXCSR BITS 9 AND 10)
;BY CHECKING THAT TIME REQUIRED FOR OUTPUTTING A CHARACTER IS LONGEST FOR
;8 BIT CODE THAN FOR 7 BIT CODE ETC.
AKAA: CLR CTRA ;CLEAR CTRA THROUGH CTRD.
      CLR CTRB ;(ELAPSED TIME COUNTERS).
      CLR CTCR
      CLR CTRD
      BIC #3000, @RXCSR ;SET CHAR LENGTH TO 00 (8 BIT CODE).
      JSR %7, AKAS ;OUTPUT CHARACTER AND TIME.
      ADD AKAST, CTRA ;ADD ELAPSED TIME TO CTRA
      BIC #3000, @RXCSR ;SET CHAR LENGTH TO 01 (7 BIT CODE).
      BIS #1000, @RXCSR
      JSR %7, AKAS ;OUTPUT CHARACTER AND TIME.
      ADD AKAST, CTRB ;ADD ELAPSED TIME TO CTRB.
      BIC #3000, @RXCSR ;SET CHAR LENGTH TO 10 (6 BIT CODE)
      BIS #2000, @RXCSR
      JSR %7, AKAS ;OUTPUT CHARACTER AND TIME.
      ADD AKAST, CTCR ;ADD ELAPSED TIME TO CTCR
      BIS #3000, @RXCSR ;SET CHAR LENGTH TO 11 (5 BIT CODE)
      JSR %7, AKAS ;OUTPUT CHARACTER AND TIME
      ADD AKAST, CTRD ;ADD ELAPSED TIME TO CTRD
      JSR %7, CMPT ;CHECK THAT CTRA THROUGH CTRD
      BR AKAB ;DESCENDING VALUES.
      ERROR1 ;TX CHARACTER LENGTH NOT ARRANGED
      ETCLGT ;IN DESCENDING ORDER.
AKAB: SCOPE
AKAS: CLR AKAST ;CLEAR ELAPSED TIME COUNTER AKAST
      TSTB @TXCSR ;WAIT FOR TX READY.
      BPL -.4
      DELAY 20. ;WAIT 20 MSECS.
AKASA: CLR @TXBUF ;LOAD TXBUF
        DELAY 1 MSEC ;DELAY 1 MSEC
        INC AKAST ;INCREMENT ELAPSED TIME COUNTER
        TSTB @TXCSR ;READY SET?
        BPL AKASA ;BRANCH IF READY NOT SET
        RTS %7 ;EXIT
AKAST: OPEN ;ELAPSED TIME COUNTER
  
```

```

2148
2149
2150 010544 000055
2151 010546 010620
2152 010550 000144
2153 010552 010554
2154
2155
2156
2157
2158 010554 052777 000004 170322
2159 010562 005077 170320
2160 010566 104016
2161 010570 000310
2162 010572 105777 170302
2163 010576 100402
2164 010600 104003
2165 010602 000405
2166 010604 104011
2167 010606 105777 170266
2168 010612 100001
2169 010614 104003
2170 010616 104012
2171
2172 010620 000056
2173 010622 010666
2174 010624 000144
2175 010626 010630
2176
2177
2178
2179 010630 052777 000004 170246
2180 010636 005077 170244
2181 010642 105777 170232
2182 010646 100375
2183 010650 005777 170226
2184 010654 105777 170220
2185 010660 100001
2186 010662 104003
2187 010664 104012

;*****
AT55: 55 ;TEST NUMBER 55 *
      AT56 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      ALAA ;SCOPE ENTRY POINT *
;*****
;TEST THAT OUTPUTTING A CHARACTER WITH THE MAINTENANCE BIT SET (TXCSR BIT 2)
;RESULTS IN DONE BIT SETTING (RXCSR BIT 7) NO LATER THAN 200 MSECS, AND
;THAT RESET INSTRUCTION CLEARS THE DONE BIT
ALAA: BIS #BIT2,@TXCSR ;SET MAINTENANCE (TXCSR BIT 2)
      CLR @TXBUF ;LOAD TXBUF
      DELAY ;WAIT 200 MSECS.
      200.
      TSTB @RXCSR ;SEE IF DONE BIT IS SET
      BMI ALAB ;BRANCH IF DONE BIT IS SET
      ERROR ;DONE BIT FAILED TO SET
ALAB: BR ALAC
      SRESET ;ISSUE RESET TO CLEAR DONE BIT
      TSTB @RXCSR ;SEE IF DONE BIT IS CLEARED
      BPL ALAC ;BRANCH IF DONE BIT IS CLEARED
      ERROR ;RESET FAILED TO CLEAR DONE BIT
ALAC: SCOPE ;SCOPE
;*****
AT56: 56 ;TEST NUMBER 56 *
      AT57 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AMAA ;SCOPE ENTRY POINT *
;*****
;TEST THAT DONE BIT (RXCSR BIT 7) IS CLEARED BY READING RXBUF.
;DONE SET BY OUTPUTTING CHARACTER WITH MAINTENANCE BIT SET (TXCSR BIT 2)
AMAA: BIS #BIT2,@TXCSR ;SET MAINTENANCE BIT (TXCSR BIT 2)
      CLR @TXBUF ;LOAD TXBUF
AMAB: TSTB @RXCSR ;WAIT FOR DONE BIT TO SET.
      BPL AMAB
      TST @RXBUF ;READ RXBUF TO CLEAR DONE BIT
      TSTB @RXCSR ;SEE IF DONE BIT IS CLEAR
      BPL AMAC ;BRANCH IF DONE BIT IS CLEAR
AMAC: ERROR ;READING RXBUF FAILED TO CLEAR DONE BIT
      SCOPE ;SCOPE

```

```

2188
2189
2190 010666 000057
2191 010670 010754
2192 010672 000012
2193 010674 010676
2194
2195
2196
2197
2198 010676 042777 000030 170200
2199 010704 052777 000014 170172
2200 010712 042777 000030 170160
2201 010720 052777 000010 170152
2202 010726 005077 170154
2203 010732 104016
2204 010734 000764
2205 010736 105777 170136
2206 010742 100401
2207 010744 104003
2208 010746 005777 170130
2209 010752 104012
2210
2211
2212 010754 000060
2213 010756 011042
2214 010760 000012
2215 010762 010764
2216
2217
2218
2219
2220 010764 042777 000030 170112
2221 010772 052777 000024 170104
2222 011000 042777 000030 170072
2223 011006 052777 000020 170064
2224 011014 005077 170066
2225 011020 104016
2226 011022 000620
2227 011024 105777 170050
2228 011030 100401
2229 011032 104003
2230 011034 005777 170042
2231 011040 104012
2232
2233 011042 000061
2234 011044 011114
2235 011046 000012
2236 011050 011052
2237
2238
2239
2240
2241 011052 052777 000034 170024
2242 011060 052777 000030 170012
2243 011066 005077 170014

```

```

;*****
AT57: 57 ;TEST NUMBER 57 *
      AT60 ;ADDRESS OF NEXT TEST *
      10. ;TEST ITERATION COUNT *
      ANAA ;SCOPE ENTRY POINT *
;*****
;TEST THAT DONE BIT (RXCSR BIT 7) SETS NO LATER THAN 500 MSECS AFTER OUTPUTTING
;CHARACTER WITH MAINTENANCE BIT SET AND RECEIVE SPEED SET TO 01 (TRANSMIT
;SPEED ALSO SET TO 01
ANAA: BIC #30,@TXCSR ;SET MAINTENANCE BIT AND SET
      BIS #14,@TXCSR ;TX SPEED=01
      BIC #30,@RXCSR ;SET RX SPEED =01
      BIS #10,@RXCSR
      CLR @TXBUF ;LOAD TXBUF
      DELAY 500. ;DELAY 500 MSECS.
      TSTB @RXCSR ;SEE IF DONE BIT IS SET.
      BMI ANAB ;BRANCH IF DONE IS SET.
ANAB: ERROR ;DONE FAILED TO SET WITH RX SPEED=01.
      TST @RXBUF ;CLEAR DONE BIT IF SET.
      SCOPE ;SCOPE
;*****
AT60: 60 ;TEST NUMBER 60 *
      AT61 ;ADDRESS OF NEXT TEST *
      10. ;TEST ITERATION COUNT *
      AOAA ;SCOPE ENTRY POINT *
;*****
;TEST THAT DONE BIT (RXCSR BIT 7) SETS NO LATER THAN 400 MSECS AFTER OUTPUTTING
;CHARACTER WITH MAINTENANCE BIT SET AND RECEIVE SPEED SET TO 10 (TRANSMIT
;SPEED ALSO SET TO 10).
AOAA: BIC #30,@TXCSR ;SET MAINTENANCE BIT AND SET
      BIS #24,@TXCSR ;TX SPEED=10.
      BIC #30,@RXCSR ;SET RX SPEED=10.
      BIS #20,@RXCSR
      CLR @TXBUF ;LOAD TXBUF
      DELAY 400. ;DELAY 400 MSECS
      TSTB @RXCSR ;SEE IF DONE BIT IS SET.
      BMI AOAB ;BRANCH IF DONE BIT IS SET.
AOAB: ERROR ;DONE FAILED TO SET WITH RX SPEED=10.
      TST @RXBUF ;CLEAR DONE BIT IF SET
      SCOPE ;SCOPE
;*****
AT61: 61 ;TEST NUMBER 61 *
      AT62 ;ADDRESS OF NEXT TEST *
      10. ;TEST ITERATION COUNT *
      APAA ;SCOPE ENTRY POINT *
;*****
;TEST THAT DONE BIT (RXCSR BIT 7) SETS NO LATER THAN 250 MSECS AFTER OUTPUTTING
;CHARACTER WITH MAINTENANCE BIT SET AND RECEIVE SPEED SET 11 (TRANSMIT SPEED
;ALSO SET TO 11).
APAA: BIS #34,@TXCSR ;SET MAINT BIT AND TX SPEED=11
      BIS #30,@RXCSR ;SET RX SPEED=11
      CLR @TXBUF ;LOAD TXBUF

```

L04

2244	011072	104016	
2245	011074	000372	
2246	011076	105777	167776
2247	011102	100401	
2249	011104	104003	
2249	011106	005777	167770
2250	011112	104012	

	DELAY	
	250	
	TSTB	DRXCSR
	BMI	APAB
	ERROR	
APAB:	TST	DRXBUF
	SCOPE	

;DELAY 250 MSECS.  
;SEE IF DONE BIT IS SET.  
;BRANCH IF DONE BIT IS SET.  
;DONE FAILED TO SET WITH RX SPEED=11  
;CLEAR DONE BIT IF SET.  
;SCOPE

```

2251
2252
2253 011114 000062
2254 011116 011416
2255 011120 000144
2256 011122 011124
2257
2258
2259
2260
2261 011124 005067 170116
2262 011130 005067 170114
2263 011134 005067 170112
2264 011140 005067 170110
2265 011144 042777 000030 167732
2266 011152 052777 000004 167724
2267 011160 042777 000030 167712
2268 011166 004767 000154
2269 011172 066767 000216 170046
2270 011200 042777 000030 167676
2271 011206 052777 000010 167670
2272 011214 042777 000030 167656
2273 011222 052777 000010 167650
2274 011230 004767 000112
2275 011234 066767 000154 170006
2276 011242 042777 000030 167634
2277 011250 052777 000020 167626
2278 011256 042777 000030 167614
2279 011264 052777 000020 167606
2280 011272 004767 000050
2281 011276 066767 000112 167746
2282 011304 052777 000030 167572
2283 011312 052777 000030 167560
2284 011320 004767 000022
2285 011324 066767 000064 167722
2286 011332 004767 002720
2287 011336 000402
2288 011340 104015
2289 011342 015652
2290 011344 104012
2291 011346 005067 000042
2292 011352 105777 167526
2293 011356 100375
2294 011360 104016
2295 011362 000024
2296 011364 005777 167512
2297 011370 005077 167512
2298 011374 104016
2299 011376 000001
2300 011400 005267 000010
2301 011404 105777 167470
2302 011410 100371
2303 011412 000207
2304 011414 000000

```

```

*****
AT62: 62 ;TEST NUMBER 62 *
      AT63 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AQAA ;SCOPE ENTRY POINT *
*****
;TEST THAT RECEIVE SPEEDS ARE ARRANGED IN ASCENDING ORDER BY CHECKING THAT TIME
;ELAPSED TO DONE BIT SETTING (RXCSR BIT 7) DECREASES AS A HIGHER SPEED
;IS SELECTED. THE TRANSMIT SPEED SELECTED WILL CORRESPOND TO THE SELECTED RECEIVE SPEED
AQAA: CLR CTRA ;CLEAR CTRA THROUGH CTRD
      CLR CTRB ;(ELAPSED TIME COUNTERS)
      CLR CTRC
      CLR CTRD
      BIC #30, @TXCSR ;SELECT TX SPEED 00
      BIS #BIT2, @TXCSR ;SET MAINTENANCE BIT
      BIC #30, @RXCSR ;SELECT RX SPEED 00
      JSR %7, AQAS ;OUTPUT CHARACTER AND TIME DONE BIT
      ADD AQAST, CTRA ;ADD ELAPSED TIME TO CTRA
      BIC #30, @TXCSR ;SELECT TX SPEED 01
      BIS #10, @TXCSR
      BIC #30, @RXCSR ;SELECT RX SPEED 01.
      BIS #10, @RXCSR
      JSR %7, AQAS ;OUTPUT CHARACTER AND TIME DONE BIT
      ADD AQAST, CTRB ;ADD ELAPSED TIME TO CTRB
      BIC #30, @TXCSR ;SELECT TX SPEED 10
      BIS #20, @TXCSR
      BIC #30, @RXCSR ;SELECT RX SPEED 10
      BIS #20, @RXCSR
      JSR %7, AQAS ;OUTPUT CHARACTER AND TIME DONE BIT.
      ADD AQAST, CTRC ;ADD ELAPSED TIME TO CTRC.
      BIS #30, @TXCSR ;SELECT TX SPEED 11
      BIS #30, @RXCSR ;SELECT RX SPEED 11
      JSR %7, AQAS ;OUTPUT CHARACTER AND TIME DONE BIT
      ADD AQAST, CTRD ;ADD ELAPSED TIME TO CTRD.
      JSR %7, CMPT ;CHECK THAT CTRA THROUGH CTRD CONTAIN
      BR AQAB ;DESCENDING VALUES.
      ERROR1 ;RECEIVE SPEEDS NOT ARRANGED IN
      ERXTIM ;ASCENDING ORDER.
AQAB: SCOPE ;SCOPE
AQAS: CLR AQAST ;CLEAR ELAPSED TIME COUNTER AQAST
      TSTB @TXCSR ;WAIT FOR TX READY.
      BPL .-4
      DELAY 20. ;WAIT 20 MSECS.
      TST @RXBUF ;CLEAR DONE BIT IF SET
      CLR @TXBUF ;LOAD TXBUF
AQASA: DELAY 1 ;DELAY 1 MSEC
      INC AQAST ;INCREMENT ELAPSED TIME COUNTER
      TSTB @RXCSR ;DONE SET?
      BPL AQASA ;BRANCH IF DONE NOT SET
      RTS ;EXIT
AQAST: OPEN ;ELAPSED TIME COUNTER

```

2305					
2306					
2307	011416	000063			
2308	011420	011654			
2309	011422	000144			
2310	011424	011426			
2311					
2312					
2313					
2314					
2315	011426	005067	167614		
2316	011432	005067	167612		
2317	011436	005067	167610		
2318	011442	005067	167606		
2319	011446	042777	003000	167424	
2320	011454	004767	000116		
2321	011460	066767	000166	167560	
2322	011466	042777	003000	167404	
2323	011474	052777	001000	167376	
2324	011502	004767	000070		
2325	011506	066767	000140	167534	
2326	011514	042777	003000	167356	
2327	011522	052777	002000	167350	
2328	011530	004767	000042		
2329	011534	066767	000112	167510	
2330	011542	052777	003000	167330	
2331	011550	004767	000022		
2332	011554	066767	000072	167472	
2333	011562	004767	002470		
2334	011566	000402			
2335	011570	104015			
2336	011572	015714			
2337	011574	104012			
2338	011576	005067	000050		
2339	011602	105777	167276		
2340	011606	100375			
2341	011610	104016			
2342	011612	000024			
2343	011614	005777	167262		
2344	011620	052777	000004	167256	
2345	011626	005077	167254		
2346					
2347	011632	104016			
2348	011634	000001			
2349	011636	005267	000010		
2350	011642	105777	167232		
2351	011646	100371			
2352	011650	000207			
2353	011652	000000			

```

*****
AT63: 63 ;TEST NUMBER 63 *
      AT64 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      ARAA ;SCOPE ENTRY POINT *
*****
;TEST FOR CORRECT OPERATION OF CHARACTER LENGTH SELECTION DURING RECEIVE
;(RXCSR BITS 9 AND 10) BY CHECKING THAT TIME REQUIRED TO RECEIVE A CHARACTER
;IS LONGEST FOR 8 BIT CODE THAN FOR 7 BIT CODE ETC.
ARAA: CLR CTRA ;CLEAR CTRA THROUGH CTRD
      CLR CTRB ;(ELAPSED TIME COUNTERS)
      CLR CTCR
      CLR CTRD
      BIC #3000, @RXCSR ;SET CHAR LENGTH TO 00 (8 BIT CODE)
      JSR %7, ARAS ;OUTPUT CHAR AND TIME DONE BIT.
      ADD ARAST, CTRA ;ADD ELAPSED TIME TO CTRA
      BIC #3000, @RXCSR ;SET CHAR LENGTH TO 01 (7 BIT CODE)
      BIS #1000, @RXCSR
      JSR %7, ARAS ;OUTPUT CHAR AND TIME DONE BIT
      ADD ARAST, CTRB ;ADD ELAPSED TIME TO CTRB
      BIC #3000, @RXCSR ;SET CHAR LENGTH TO 10 (6 BIT CODE)
      BIS #2000, @RXCSR
      JSR %7, ARAS ;OUTPUT CHAR AND TIME DONE BIT
      ADD ARAST, CTCR ;ADD ELAPSED TIME TO CTCR
      BIS #3000, @RXCSR ;SET CHAR LENGTH TO 11 (5 BIT CODE)
      JSR %7, ARAS ;OUTPUT CHAR AND TIME DONE BIT
      ADD ARAST, CTRD ;ADD ELAPSED TIME TO CTRD
      JSR %7, CMPT ;CHECK THAT CTRA THROUGH CTRD
      BR ARAB ;CONTAIN DESCENDING VALUES
      ERROR1 ;RECEIVE CHARACTER LENGTHS NOT ARRANGED
      ERCLGT ;IN DESCENDING ORDER
      SCOPE ;SCOPE
      ARAS: CLR ARAST ;CLEAR ELAPSED TIME COUNTER ARAST
      TSTB @TXCSR ;WAIT FOR TX READY.
      BPL .-4
      DELAY ;WAIT 20 MSECS.
      20
      TST @RXBUF ;CLEAR DONE BIT IF SET
      BIS #BIT2, @TXCSR ;SET MAINTENANCE BIT
      CLR @TXBUF ;LOAD TXBUF
      ARASA: DELAY ;DELAY 1 MSEC.
      1
      INC ARAST ;INCREMENT ELAPSED TIME COUNTER
      TSTB @RXCSR ;SEE IF DONE BIT IS SET.
      BPL ARASA ;BRANCH IF NOT SET
      RTS ;EXIT
      ARAST: OPEN ;ELAPSED TIME COUNTER
  
```

```

2354
2355
2356 011654 000064
2357 011656 011770
2358 011660 000144
2359 011662 011664
2360
2361
2362 011664 004767 000060
2363 011670 004767 000054
2364 011674 017767 167200 167356
2365 011702 032767 010000 167350
2366 011710 001002
2367 011712 104003
2368 011714 000412
2369 011716 005767 167336
2370 011722 100402
2371 011724 104003
2372
2373 011726 000405
2374 011730 032777 010000 167142
2375
2376 011736 001401
2377 011740 104003
2378
2379 011742 005777 167134
2380 011746 104012
2381 011750 052777 000004 167126
2382 011756 005077 167124
2383 011762 104016
2384 011764 000310
2385 011766 000207
2386
2387
2388 011770 000065
2389 011772 012042
2390 011774 000012
2391 011776 012010
2392
2393
2394
2395 012000 004767 170560
2396 012004 104007
2397 012006 012036
2398 012010 042777 000100 167066
2399 012016 005067 165754
2400 012022 052777 000104 167054
2401 012030 000240
2402 012032 104003
2403 012034 104012
2404 012036 022626
2405 012040 000775
2406

```

```

;*****
AT64: 64 ;TEST NUMBER 64 *
      AT65 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      ASAA ;SCOPE ENTRY POINT *
;*****
;TEST CORRECT OPERATION OF DATA OVERRUN BIT (RXCSR BIT 12)
ASAA: JSR %7,ASAS ;OUTPUT CHARACTER AND WAIT 200 MSECS
      JSR %7,ASAS ;OUTPUT CHARACTER AND WAIT 200 MSECS
      MOV @RXCSR,RXCSRT ;SAVE RXCSR CONTENTS
      BIT #BIT12,RXCSRT ;SEE IF DATA OVERRUN BIT WAS SET
      BNE ASAD ;BRANCH IF BIT WAS SET
      BR ASAD
ASAB: TST RXCSRT ;SEE IF ERROR BIT WAS SET (RXCSR BIT 15)
      BMI ASAC
      ERROR ;ERROR BIT FAILED TO SET
      ;WHEN DATA OVERRUN SET
ASAC: BR ASAD
      BIT #BIT12,@RXCSR ;SEE IF DATA OVERRUN WAS
      ;CLEARED WHEN RXCSR WAS READ
      BEQ ASAD ;BRANCH IF CLEAR
      ERROR ;READING RXCSR FAILED
      ;TO CLEAR DATA OVERRUN
ASAD: TST @RXBUF ;CLEAR DONE BIT (RXCSR BIT 7)
      SCOPE ;SCOPE
ASAS: BIS #BIT2,@TXCSR ;SET MAINTENANCE BIT
      CLR @TXBUF ;LOAD TXBUF
      DELAY 200. ;DELAY 200 MSECS
      RTS %7 ;EXIT
;*****
AT65: 65 ;TEST NUMBER 65 *
      AT66 ;ADDRESS OF NEXT TEST *
      10. ;TEST ITERATION COUNT *
      ATAA ;SCOPE ENTRY POINT *
;*****
;TEST THAT TRANSMITTER IS ABLE TO INTERRUPT. IF THE INTERRUPT IS SERVICED,
;IT WILL HAVE OCCURRED AT THE CORRECT VECTOR.
JSR 7,OVRLAY ;GO TO OVER LAY ROUTINE
STTXV ;SET TX INTERRUPT SERVICE
ATAC ;TO ATAC
ATAA: BIC #BIT6,@TXCSR ;DISABLE TX INTERRUPT
      CLR PSW ;SET PROCESSOR PRIORITY TO 0
      BIS #104,@TXCSR ;ENABLE TX INTERRUPT
      NOP
      ERROR ;READY DID NOT CAUSE AN INTERRUPT
ATAB: SCOPE ;SCOPE
ATAC: POPSP2 ;HERE IF INTERRUPT IS SERVICED. POP
      BR ATAB ;THE STACK TWICE

```

```

2407
2408
2409 012042 000066
2410 012044 012120
2411 012046 001750
2412 012050 012056
2413
2414
2415
2416 012052 104007
2417 012054 012110
2418 012056 016767 167034 165712 AUAA: MOV TXLVL,PSW ;SET PROCESSOR PRIORITY SAME AS TX PRIORITY
2419 012064 042777 000100 167012 BIC #BIT6,@TXCSR ;DISABLE TX INTERRUPTS
2420 012072 052777 000104 167004 BIS #104,@TXCSR ;ENABLE TX INTERRUPTS
2421 012100 000240 NOP
2422
2423
2424 012102 042777 000100 166774 AUAB: BIC #BIT6,@TXCSR ;OK IF NO INTERRUPT OCCURS. DISABLE INTERRUPTS
2425 012110 104012 AUAC: SCOPE ;SCOPE
2426 012112 022626 POPSP2 ;HERE IF INTERRUPT OCCURS. POP STOCK TWICE
2427 012114 104003 ERROR ;TX INTERRUPTED WITH PROCESSOR AT SAME
2428 012116 000774 BR AUAC ;PRIORITY AS THE TRANSMITTER
2429
2430
2431 012120 000067
2432 012122 012204
2433 012124 000012
2434 012126 012134
2435
2436
2437
2438 012130 104007
2439 012132 012172
2440 012134 042777 000100 166742 AVAA: BIC #BIT6,@TXCSR ;DISABLE TX INTERRUPTS
2441 012142 016767 166750 165626 MOV TXLVL,PSW ;SET PROCESSOR PRIORITY TO ONE LEVEL
2442 012150 162767 000040 165620 SUB #40,PSW ;LOWER THAN TX PRIORITY
2443 012156 052777 000104 166720 BIS #104,@TXCSR ;ENABLE TX INTERRUPTS
2444 012164 000240 NOP
2445 012166 104003 ERROR ;TX FAILED TO INTERRUPT
2446 012170 000401 BR AVAC
2447 012172 022626 AVAB: POPSP2 ;HERE IF INTERRUPT OCCURS. POP STOCK TWICE
2448 012174 042777 000100 166702 AVAC: BIC #BIT6,@TXCSR ;DISABLE TX INTERRUPTS
2449 012202 104012 SCOPE ;SCOPE
2450

```

```

2451
2452
2453 012204 000070
2454 012206 012302
2455 012210 000144
2456 012212 012214
2457
2458
2459
2460 012214 104007
2461 012216 012254
2462 012220 042777 000100 166656
2463 012226 005067 165544
2464 012232 052777 000104 166644
2465 012240 000240
2466 012242 104003
2467 012244 042777 000100 166632 AWAB:
2468 012252 104012
2469 012254 012777 012274 166632 AWAC:
2470 012262 012716 012270
2471 012266 000002
2472 012270 000240
2473 012272 000764
2474 012274 022626
2475 012276 104003
2476 012300 000761
2477
2478
2479 012302 000071
2480 012304 012360
2481 012306 000012
2482 012310 012326
2483
2484
2485
2486 012312 004767 170246
2487 012316 104006
2488 012320 012354
2489 012322 004767 001710
2490 012326 042777 000100 166544 AXAA:
2491 012334 005067 165436
2492 012340 052777 000100 166532
2493 012346 000240
2494 012350 104003
2495 012352 000401
2496 012354 022626
2497 012356 104012
2498

```

```

*****
AT70: 70 ;TEST NUMBER 70 *
      AT71 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AWAA ;SCOPE ENTRY POINT *
*****
;TEST THAT TRANSMITTER DOES NOT REINTERRUPT AFTER THE INITIAL INTERRUPT HAS
;OCCURRED AND HAS BEEN SERVICED.
AWAA: STTXV ;SET TX INTERRUPT SERVICE TO AWAC
      AWAC
      BIC #BIT6,@TXCSR ;DISABLE TX INTERRUPTS
      CLR PSW ;SET PROCESSOR PRIORITY TO 0
      BIS #104,@TXCSR ;ENABLE TX INTERRUPTS
      NOP
      ERROR ;TRANSMITTER FAILED TO INTERRUPT
      BIC #BIT6,@TXCSR ;DISABLE TX INTERRUPTS
      SCOPE ;SCOPE
      MOV #AWAE,@TXVTR ;HERE IF INTERRUPT OCCURS. CHANGE EXIT
      MOV #AWAD,@%6 ;POINTER TO AWAD AND EXIT INTERRUPT
      RTI
      AWAD: NOP ;OK IF NO INTERRUPT REOCCURS.
      BR AWAB
      AWAE: POPSP2 ;HERE IF INTERRUPT REOCCURS
      ERROR ;TX REINTERRUPTED AFTER RTI
      BR AWAB
*****
AT71: 71 ;TEST NUMBER 71 *
      AT72 ;ADDRESS OF NEXT TEST *
      10. ;TEST ITERATION COUNT *
      AXAA ;SCOPE ENTRY POINT *
*****
;TEST THAT RECEIVER DONE BIT IS ABLE TO INTERRUPT. IF THE INTERRUPT IS
;SERVICED IT WILL HAVE OCCURRED AT THE CORRECT VECTOR.
JSR 7,OVRLAY ;GO TO OVERLAY ROUTINE
STRXV ;SET RX INTERRUPT SERVICE TO AXAB
AXAB
JSR %7,STRXD ;SET RX DONE BIT
AXAA: BIC #BIT6,@RXCSR ;DISABLE RX INTERRUPTS
      CLR PSW ;SET PROCESSOR PRIORITY TO 0
      BIS #BIT6,@RXCSR ;ENABLE RX INTERRUPTS
      NOP
      ERROR ;RX FAILED TO INTERRUPT
      BR AXAC
      AXAB: POPSP2 ;HERE IF INTERRUPT OCCURS
      AXAC: SCOPE ;SCOPE

```

```

2499
2500
2501 012360 000072
2502 012362 012442
2503 012364 001750
2504 012366 012400
2505
2506
2507
2508 012370 104006
2509 012372 012434
2510 012374 004767 001636
2511 012400 042777 000100 166472 AYAA:
2512 012406 016767 166500 165362
2513 012414 052777 000100 166456
2514 012422 000240
2515 012424 042777 000100 166446 AYAB:
2516 012432 104012
2517 012434 022626 AYAC:
2518 012436 104003
2519 012440 000771
2520
2521
2522 012442 000073
2523 012444 012532
2524 012446 000012
2525 012450 012462
2526
2527
2528
2529 012452 104006
2530 012454 012520
2531 012456 004767 001554
2532 012462 042777 000100 166410 AZAA:
2533 012470 016767 166416 165300
2534 012476 162767 000040 165272
2535 012504 052777 000100 166366
2536 012512 000240
2537 012514 104003
2538 012516 000401
2539
2540 012520 022626 AZAB:
2541 012522 042777 000100 166350 AZAC:
2542 012530 104012

```

```

*****
AT72: 72 ;TEST NUMBER 72 *
      AT73 ;ADDRESS OF NEXT TEST *
      1000. ;TEST ITERATION COUNT *
      AYAA ;SCOPE ENTRY POINT *
*****
;TEST THAT RECEIVER DONE BIT DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR
;IS AT THE SAME PRIORITY LEVEL AS THE RECEIVER INTERRUPT REQUEST LEVEL
STRXV ;SET RX INTERRUPT SERVICE TO AYAC
AYAC
JSR %7,STRXD ;SET RX DONE BIT
AYAA: BIC #BIT6,ARXCSR ;DISABLE RX INTERRUPTS
MOV RXLVL,PSW ;SET PROCESSOR PRIORITY SAME AS RECEIVER'S
BIS #BIT6,ARXCSR ;ENABLE RX INTERRUPTS
NOP
AYAB: BIC #BIT6,ARXCSR ;OK IF NO INTERRUPT. DISABLE RX INTERRUPTS
SCOPE ;SCOPE
AYAC: POPSP2 ;HERE IF INTERRUPT OCCURS. POP STOCK TWICE
ERROR ;RX INTERRUPTED WITH PROCESOR AT SAME
BR AYAB ;PRIORITY AS THE RECEIVER
*****
AT73: 73 ;TEST NUMBER 73 *
      AT74 ;ADDRESS OF NEXT TEST *
      10. ;TEST ITERATION COUNT *
      AZAA ;SCOPE ENTRY POINT *
*****
;TEST THAT RECEIVER DONE BIT CAUSES INTERRUPT WHEN PROCESSOR IS AT PRIORITY
;ONE LEVEL LOWER THAN THE RECEIVER'S INTERRUPT REQUEST LEVEL
STRXV ;SET RX INTERRUPT TO AZAB
AZAB
JSR %7,STRXD ;SET RX DONE BIT
AZAA: BIC #BIT6,ARXCSR ;DISABLE RX INTERRUPTS
MOV RXLVL,PSW ;SET PROCESSOR PRIORITY ONE LEVEL
SUB #40,PSW ;LOWER THAN RECEIVER'S PRIORITY
BIS #BIT6,ARXCSR ;ENABLE RX INTERRUPTS
NOP
ERROR ;RX FAILED TO INTERRUPT WITH PROCESSOR AT
BR AZAC ;PRIORITY ONE LEVEL LOWER THAN RECEIVER'S
*****
AZAB: POPSP2 ;HERE IF INTERRUPT OCCURS
AZAC: BIC #BIT6,ARXCSR ;DISABLE RX INTERRUPTS
SCOPE ;SCOPE

```

```

2543
2544
2545 012532 000074
2546 012534 012630
2547 012536 000144
2548 012540 012546
2549
2550
2551
2552 012542 004767 001470
2553 012546 104006
2554 012550 012602
2555 012552 042777 000100 166320
2556 012560 052777 000100 166312
2557 012566 000240
2558 012570 104003
2559 012572 042777 000100 166300 AAB8:
2560 012600 104012
2561 012602 012777 012622 166300 AABC:
2562 012610 012716 012616
2563 012614 000002
2564 012616 000240 AABD:
2565 012620 000764
2566 012622 022626 AABE:
2567 012624 104003
2568 012626 000761
2569
2570
2571 012630 000075
2572 012632 012704
2573 012634 000144
2574 012636 012640
2575
2576
2577 012640 004767 001372
2578 012644 005077 166236
2579 012650 104016
2580 012652 000024
2581 012654 017767 166220 166376
2582 012662 105777 166212
2583 012666 100001
2584 012670 104003
2585 012672 104016 ABB8:
2586 012674 000310
2587 012676 005777 166200
2588 012702 104012
2589

;*****
AT74: 74 ;TEST NUMBER 74 *
      AT75 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      AABA ;SCOPE ENTRY POINT *
;*****
;TEST THAT RECEIVER DOES NOT INTERRUPT AFTER THE INITIAL INTERRUPT HAS
;OCCURED AND DONE BIT HAS NOT BEEN CLEARED
AABA: JSR %7,STRXD ;SET RX DONE BIT
      STRXV ;SET RX INTERRUPT SERVICE TO AABC
      AABC
      BIC #BIT6,@RXCSR ;DISABLE RX INTERRUPTS
      BIS #BIT6,@RXCSR ;ENABLE RX INTERRUPTS
      NOP
      ERROR ;RX FAILED TO INTERRUPT
AAB8: BIC #BIT6,@RXCSR ;DISABLE RX INTERRUPTS
      SCOPE ;SCOPE
AABC: MOV #AABE,@RXVTR ;HERE IF INTERRUPT OCCURS. CHANGE SERVICE TO
      MOV #AABD,@%6 ;AABE, SET EXIT POINTER TO AABD
      RTI ;EXIT INTERRUPT SERVICE
AABD: NOP ;OK IF NO INTERRUPT REOCCURS
AABE: BR AAB8 ;HERE IF INTERRUPT REOCCURS
      ERROR ;RX REINTERRUPTED AFTER RTI
      BR AAB8

;*****
AT75: 75 ;TEST NUMBER 75 *
      AT76 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      ABBA ;SCOPE ENTRY POINT *
;*****
;TEST THAT DATA OVERRUN (RXCSR BIT 12) CLEARS THE DONE BIT (RXCSR BIT 7)
ABBA: JSR %7,STRXD ;SET RX DONE BIT
      CLR @TXBUF ;LOAD TXBUF
      DELAY ;WAIT 20 MSECS.
      20.
      MOV @RXCSR,RXC8RT ;SAVE CONTENT OF RXCSR
      T8TB @RXCSR ;SEE IF DONE BIT IS CLEAR
      BPL ABB8 ;BRANCH IF DONE BIT IS CLEAR
      ERROR
ABBB: DELAY ;WAIT FOR RX DONE TO SET.
      200.
      T8T @RXBUF ;CLEAR DONE BIT IF SET
      SCOPE ;SCOPE
  
```

```

2590
2591
2592 012704 000076
2593 012706 012762
2594 012710 000144
2595 012712 012720
2596
2597
2598 012714 104006
2599 012716 012756
2600 012720 004767 001312
2601 012724 004767 001306
2602 012730 042777 000100 166142
2603 012736 005067 165034
2604 012742 052777 000100 166130
2605 012750 000240
2606 012752 104003
2607 012754 000401
2608 012756 022626
2609 012760 104012
2610
2611
2612
2613
2614 012762 000077
2615 012764 013144
2616 012766 000144
2617 012770 013004
2618
2619
2620 012772 004567 170062
2621 012776 052777 000004 166100
2622 013004 112767 000144 166234
2623 013012 112767 000010 166227
2624 013020 004567 170140
2625 013024 105777 166054
2626 013030 100375
2627 013032 010177 166050
2628 013036 105777 166036
2629 013042 100375
2630 013044 017767 166032 166160
2631 013052 005000
2632 013054 006067 166152
2633 013060 103001
2634 013062 005100
2635 013064 105367 166157
2636 013070 001371
2637 013072 032777 000040 166000
2638 013100 001403
2639 013102 005700
2640 013104 001403
2641 013106 000412
2642 013110 005700
2643 013112 001410
2644 013114 104003
2645 013116 004567 170110

;*****
AT76: 76 ;TEST NUMBER 76 *
      AT77 ;ADDRESS OF NEXT TEST *
      100. ;TEST ITERATION COUNT *
      ACBA ;SCOPE ENTRY POINT *
;*****
;TEST THAT ERROR BIT (RXCSR BIT 15) IS ABLE TO CAUSE AN INTERRUPT
      STRXV ;SET RX INTERRUPT SERVICE TO ACBB.
      ACBB
ACBA: JSR %7,STRXD ;SET RX DONE BIT
      JSR %7,STRXD ;SET RX DATA OFLOW
      BIC #BIT6,ARXCSR ;DISABLE RX INTERRUPTS
      CLR PSW ;SET PROCESSOR PRIORITY TO 0
      BIS #BIT6,ARXCSR ;ENABLE RX INTERRUPTS
      NOP
      ERROR ;RX ERROR BIT FAILED TO CAUSE INTERRUPT
      BR ACBC
ACBB: POPSP2 ;HERE IF INTERRUPT OCCURS. POP STOCK TWICE
ACBC: SCOPE ;SCOPE

;*****
AT77: 77 ;TEST NUMBER 77
      AT100 ;ADDRESS OF NEXT TEST
      100. ;TEST ITERATION COUNT
      ANBB ;SCOPE ENTRY POINT
;*****
;TEST THAT PARITY INDICATOR OPERATES CORRECT.
ANBA: JSR 5,INBIN ;INITIALIZE BINARY COUNT PATTERN
      BIS #BIT2,ARXCSR ;SET MAINTENANCE BIT
ANBB: MOV#B #100.,CTRA ;GET CHARACTER COUNT
ANBC: MOV#B #8.,CTRA+1 ;GET CHARACTER BIT COUNT
      JSR 5,GTBINP ;GET A CHARACTER (IN R1)
      TSTB ARXCSR ;WAIT FOR
      BPL -4 ;TRANSMITTER READY FLAG
      MOV %1,ATXBUF ;LOAD TRANSMITTER BUFFER
      TSTB ARXCSR ;WAIT FOR
      BPL -4 ;RECEIVER READY FLAG
      MOV ARXBUF,CRBUFA ;GET RECEIVED CHARACTER
      CLR %0 ;CLEAR WORKING REGISTER
ANBD: ROR CRBUFA ;LOOK AT CHARACTER BITS
      BCC .+4 ;AND COMPLEMENT R0 WHEN
      COM %0 ;A 1 IS RECEIVED
      DECB CTRA+1 ;IF R0=1'S, ODD#1'S RECEIVED
      BNE ANBD ;IF R0=0'S, EVEN #1'S RECEIVED
      BIT #BIT5,ARXCSR ;TEST PARITY INDICATOR
      BEQ ANBE ;BRANCH IF INDICATES EVEN
      TST %0 ;TEST RECEIVED PARITY (IN R0)
      BEQ ANBF ;ERROR BRANCH
      BR ANBG ;OK BRANCH
ANBE: TST %0 ;TEST RECEIVED PARITY (IN R0)
      BEQ ANBG ;OK BRANCH
ANBF: ERROR ;TYPE PC
      JSR 5,OACNV ;GO TO OCTAL
  
```

H05

```

2646 013122 001232          CRBUFA          ;TO ASCII
2647 013124 016346          AWAS          ;ROUTINE AND
2648 013126 000003          3           ;CONVERT DATA
2649 013130 104015          ERROR1       ;TYPE
2650 013132 016346          AWAS          ;DATA
2651 013134 105367 166106  ANBG:  DECB      CTRA      ;DECREMENT CHARACTER COUNT
2652 013140 001324          BNE        ANBC
2653 013142 104012          SCOPE
2654          000077          X=77
2655          000000          Y=0
2656          ;*****
2657 013144 000100  AT100: 100          ;ROUTINE #100 *
2658 013146 013162          AT101       ;ADDRESS OF NEXT TEST *
2659 013150 000003          3.         ;ITERATION COUNT *
2660 013152 013154          DAT0        ;SCOPE ENTRY POINT *
2661          000100          X=X+1
2662          ;*****
2663 013154 004567 170304  DAT0:  JSR      5,DATTST ;LOAD PARAMETERS & RUN TEST
2664 013160 000000          0           ;SEE NOTE 0 FOR DATA TEST PARAMETERS
2665          000001          Y=Y+1
2666          ;*****
2667 013162 000101  AT101: 101          ;ROUTINE #101 *
2668 013164 013200          AT102       ;ADDRESS OF NEXT TEST *
2669 013166 000003          3.         ;ITERATION COUNT *
2670 013170 013172          DAT1        ;SCOPE ENTRY POINT *
2671          000101          X=X+1
2672          ;*****
2673 013172 004567 170266  DAT1:  JSR      5,DATTST ;LOAD PARAMETERS & RUN TEST
2674 013176 000001          1           ;SEE NOTE 1 FOR DATA TEST PARAMETERS
2675          000002          Y=Y+1
2676          ;*****
2677 013200 000102  AT102: 102          ;ROUTINE #102 *
2678 013202 013216          AT103       ;ADDRESS OF NEXT TEST *
2679 013204 000003          3.         ;ITERATION COUNT *
2680 013206 013210          DAT2        ;SCOPE ENTRY POINT *
2681          000102          X=X+1
2682          ;*****
2683 013210 004567 170250  DAT2:  JSR      5,DATTST ;LOAD PARAMETERS & RUN TEST
2684 013214 000002          2           ;SEE NOTE 2 FOR DATA TEST PARAMETERS
2685          000003          Y=Y+1
2686          ;*****
2687 013216 000103  AT103: 103          ;ROUTINE #103 *
2688 013220 013234          AT104       ;ADDRESS OF NEXT TEST *
2689 013222 000003          3.         ;ITERATION COUNT *
2690 013224 013226          DAT3        ;SCOPE ENTRY POINT *
2691          000103          X=X+1
2692          ;*****
2693 013226 004567 170232  DAT3:  JSR      5,DATTST ;LOAD PARAMETERS & RUN TEST
2694 013232 000003          3           ;SEE NOTE 3 FOR DATA TEST PARAMETERS
2695          000004          Y=Y+1
2696          ;*****
2697 013234 000104  AT104: 104          ;ROUTINE #104 *
2698 013236 013252          AT105       ;ADDRESS OF NEXT TEST *
2699 013240 000003          3.         ;ITERATION COUNT *
2700 013242 013244          DAT4        ;SCOPE ENTRY POINT *
2701          000104          X=X+1

```

```

2702 ;*****
2703 013244 004567 170214 DAT4: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2704 013250 000004 ;SEE NOTE 4 FOR DATA TEST PARAMETERS
2705 000005 Y=Y+1
2706 ;*****
2707 013252 000105 AT105: 105 ;ROUTINE #105 *
2708 013254 013270 ;ADDRESS OF NEXT TEST *
2709 013256 000003 3. ;ITERATION COUNT *
2710 013260 013262 DAT5 ;SCOPE ENTRY POINT *
2711 000105 X=X+1
2712 ;*****
2713 013262 004567 170176 DAT5: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2714 013266 000005 ;SEE NOTE 5 FOR DATA TEST PARAMETERS
2715 000006 Y=Y+1
2716 ;*****
2717 013270 000106 AT106: 106 ;ROUTINE #106 *
2718 013272 013306 AT107 ;ADDRESS OF NEXT TEST *
2719 013274 000003 3. ;ITERATION COUNT *
2720 013276 013300 DAT6 ;SCOPE ENTRY POINT *
2721 000106 X=X+1
2722 ;*****
2723 013300 004567 170160 DAT6: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2724 013304 000006 ;SEE NOTE 6 FOR DATA TEST PARAMETERS
2725 000007 Y=Y+1
2726 ;*****
2727 013306 000107 AT107: 107 ;ROUTINE #107 *
2728 013310 013324 AT110 ;ADDRESS OF NEXT TEST *
2729 013312 000003 3. ;ITERATION COUNT *
2730 013314 013316 DAT7 ;SCOPE ENTRY POINT *
2731 000107 X=X+1
2732 ;*****
2733 013316 004567 170142 DAT7: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2734 013322 000007 ;SEE NOTE 7 FOR DATA TEST PARAMETERS
2735 000010 Y=Y+1
2736 ;*****
2737 013324 000110 AT110: 110 ;ROUTINE #110 *
2738 013326 013342 AT111 ;ADDRESS OF NEXT TEST *
2739 013330 000003 3. ;ITERATION COUNT *
2740 013332 013334 DAT10 ;SCOPE ENTRY POINT *
2741 000110 X=X+1
2742 ;*****
2743 013334 004567 170124 DAT10: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2744 013340 000010 ;SEE NOTE 10 FOR DATA TEST PARAMETERS
2745 000011 Y=Y+1
2746 ;*****
2747 013342 000111 AT111: 111 ;ROUTINE #111 *
2748 013344 013360 AT112 ;ADDRESS OF NEXT TEST *
2749 013346 000003 3. ;ITERATION COUNT *
2750 013350 013352 DAT11 ;SCOPE ENTRY POINT *
2751 000111 X=X+1
2752 ;*****
2753 013352 004567 170106 DAT11: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2754 013356 000011 ;SEE NOTE 11 FOR DATA TEST PARAMETERS
2755 000012 Y=Y+1
2756 ;*****
2757 013360 000112 AT112: 112 ;ROUTINE #112 *
  
```

```

2758 013362 013376          AT113          ;ADDRESS OF NEXT TEST          *
2759 013364 000003          3.          ;ITERATION COUNT              *
2760 013366 013370          DAT12          ;SCOPE ENTRY POINT            *
2761          000112          X=X+1
2762          ;*****
2763 013370 004567 170070  DAT12: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2764 013374 000012          12          ;SEE NOTE 12 FOR DATA TEST PARAMETERS
2765          000013          Y=Y+1
2766          ;*****
2767 013376 000113          AT113: 113          ;ROUTINE #113                  *
2768 013400 013414          AT114          ;ADDRESS OF NEXT TEST          *
2769 013402 000003          3.          ;ITERATION COUNT              *
2770 013404 013406          DAT13          ;SCOPE ENTRY POINT            *
2771          000113          X=X+1
2772          ;*****
2773 013406 004567 170052  DAT13: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2774 013412 000013          13          ;SEE NOTE 13 FOR DATA TEST PARAMETERS
2775          000014          Y=Y+1
2776          ;*****
2777 013414 000114          AT114: 114          ;ROUTINE #114                  *
2778 013416 013432          AT115          ;ADDRESS OF NEXT TEST          *
2779 013420 000003          3.          ;ITERATION COUNT              *
2780 013422 013424          DAT14          ;SCOPE ENTRY POINT            *
2781          000114          X=X+1
2782          ;*****
2783 013424 004567 170034  DAT14: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2784 013430 000014          14          ;SEE NOTE 14 FOR DATA TEST PARAMETERS
2785          000015          Y=Y+1
2786          ;*****
2787 013432 000115          AT115: 115          ;ROUTINE #115                  *
2788 013434 013450          AT116          ;ADDRESS OF NEXT TEST          *
2789 013436 000003          3.          ;ITERATION COUNT              *
2790 013440 013442          DAT15          ;SCOPE ENTRY POINT            *
2791          000115          X=X+1
2792          ;*****
2793 013442 004567 170016  DAT15: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2794 013446 000015          15          ;SEE NOTE 15 FOR DATA TEST PARAMETERS
2795          000016          Y=Y+1
2796          ;*****
2797 013450 000116          AT116: 116          ;ROUTINE #116                  *
2798 013452 013466          AT117          ;ADDRESS OF NEXT TEST          *
2799 013454 000003          3.          ;ITERATION COUNT              *
2800 013456 013460          DAT16          ;SCOPE ENTRY POINT            *
2801          000116          X=X+1
2802          ;*****
2803 013460 004567 170000  DAT16: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2804 013464 000016          16          ;SEE NOTE 16 FOR DATA TEST PARAMETERS
2805          000017          Y=Y+1
2806          ;*****
2807 013466 000117          AT117: 117          ;ROUTINE #117                  *
2808 013470 013504          AT120          ;ADDRESS OF NEXT TEST          *
2809 013472 000003          3.          ;ITERATION COUNT              *
2810 013474 013476          DAT17          ;SCOPE ENTRY POINT            *
2811          000117          X=X+1
2812          ;*****
2813 013476 004567 167762  DAT17: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
  
```

K05

```

2814 013502 000017          17          ;SEE NOTE 17 FOR DATA TEST PARAMETERS
2815          000020          Y=Y+1
2816          ;*****
2817 013504 000120  AT120: 120          ;ROUTINE #120          *
2818 013506 013522          AT121          ;ADDRESS OF NEXT TEST  *
2819 013510 000003          3.          ;ITERATION COUNT      *
2820 013512 013514          DAT20          ;SCOPE ENTRY POINT    *
2821          000120          X=X+1
2822          ;*****
2823 013514 004567 167744  DAT20: JSR      5,DATTST ;LOAD PARAMETERS & RUN TEST
2824 013520 000020          20          ;SEE NOTE 20 FOR DATA TEST PARAMETERS
2825          000021          Y=Y+1
2826          ;*****
2827 013522 000121  AT121: 121          ;ROUTINE #121          *
2828 013524 013540          AT122          ;ADDRESS OF NEXT TEST  *
2829 013526 000003          3.          ;ITERATION COUNT      *
2830 013530 013532          DAT21          ;SCOPE ENTRY POINT    *
2831          000121          X=X+1
2832          ;*****
2833 013532 004567 167726  DAT21: JSR      5,DATTST ;LOAD PARAMETERS & RUN TEST
2834 013536 000021          21          ;SEE NOTE 21 FOR DATA TEST PARAMETERS
2835          000022          Y=Y+1
2836          ;*****
2837 013540 000122  AT122: 122          ;ROUTINE #122          *
2838 013542 013556          AT123          ;ADDRESS OF NEXT TEST  *
2839 013544 000003          3.          ;ITERATION COUNT      *
2840 013546 013550          DAT22          ;SCOPE ENTRY POINT    *
2841          000122          X=X+1
2842          ;*****
2843 013550 004567 167710  DAT22: JSR      5,DATTST ;LOAD PARAMETERS & RUN TEST
2844 013554 000022          22          ;SEE NOTE 22 FOR DATA TEST PARAMETERS
2845          000023          Y=Y+1
2846          ;*****
2847 013556 000123  AT123: 123          ;ROUTINE #123          *
2848 013560 013574          AT124          ;ADDRESS OF NEXT TEST  *
2849 013562 000003          3.          ;ITERATION COUNT      *
2850 013564 013566          DAT23          ;SCOPE ENTRY POINT    *
2851          000123          X=X+1
2852          ;*****
2853 013566 004567 167672  DAT23: JSR      5,DATTST ;LOAD PARAMETERS & RUN TEST
2854 013572 000023          23          ;SEE NOTE 23 FOR DATA TEST PARAMETERS
2855          000024          Y=Y+1
2856          ;*****
2857 013574 000124  AT124: 124          ;ROUTINE #124          *
2858 013576 013612          AT125          ;ADDRESS OF NEXT TEST  *
2859 013600 000003          3.          ;ITERATION COUNT      *
2860 013602 013604          DAT24          ;SCOPE ENTRY POINT    *
2861          000124          X=X+1
2862          ;*****
2863 013604 004567 167654  DAT24: JSR      5,DATTST ;LOAD PARAMETERS & RUN TEST
2864 013610 000024          24          ;SEE NOTE 24 FOR DATA TEST PARAMETERS
2865          000025          Y=Y+1
2866          ;*****
2867 013612 000125  AT125: 125          ;ROUTINE #125          *
2868 013614 013630          AT126          ;ADDRESS OF NEXT TEST  *
2869 013616 000003          3.          ;ITERATION COUNT      *

```

```

2870 013620 013622          DAT25          ;SCOPE ENTRY POINT          *
2871          000125          X=X+1
2872          :*****
2873 013622 004567 167636 DAT25: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2874 013626 000025          25          ;SEE NOTE 25 FOR DATA TEST PARAMETERS
2875          000026          Y=Y+1
2876          :*****
2877 013630 000126          AT126: 126          ;ROUTINE #126          *
2878 013632 013646          AT127          ;ADDRESS OF NEXT TEST  *
2879 013634 000003          3.          ;ITERATION COUNT      *
2880 013636 013640          DAT26          ;SCOPE ENTRY POINT    *
2881          000126          X=X+1
2882          :*****
2883 013640 004567 167620 DAT26: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2884 013644 000026          26          ;SEE NOTE 26 FOR DATA TEST PARAMETERS
2885          000027          Y=Y+1
2886          :*****
2887 013646 000127          AT127: 127          ;ROUTINE #127          *
2888 013650 013664          AT130          ;ADDRESS OF NEXT TEST  *
2889 013652 000003          3.          ;ITERATION COUNT      *
2890 013654 013656          DAT27          ;SCOPE ENTRY POINT    *
2891          000127          X=X+1
2892          :*****
2893 013656 004567 167602 DAT27: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2894 013662 000027          27          ;SEE NOTE 27 FOR DATA TEST PARAMETERS
2895          000030          Y=Y+1
2896          :*****
2897 013664 000130          AT130: 130          ;ROUTINE #130          *
2898 013666 013702          AT131          ;ADDRESS OF NEXT TEST  *
2899 013670 000003          3.          ;ITERATION COUNT      *
2900 013672 013674          DAT30          ;SCOPE ENTRY POINT    *
2901          000130          X=X+1
2902          :*****
2903 013674 004567 167564 DAT30: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2904 013700 000030          30          ;SEE NOTE 30 FOR DATA TEST PARAMETERS
2905          000031          Y=Y+1
2906          :*****
2907 013702 000131          AT131: 131          ;ROUTINE #131          *
2908 013704 013720          AT132          ;ADDRESS OF NEXT TEST  *
2909 013706 000003          3.          ;ITERATION COUNT      *
2910 013710 013712          DAT31          ;SCOPE ENTRY POINT    *
2911          000131          X=X+1
2912          :*****
2913 013712 004567 167546 DAT31: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2914 013716 000031          31          ;SEE NOTE 31 FOR DATA TEST PARAMETERS
2915          000032          Y=Y+1
2916          :*****
2917 013720 000132          AT132: 132          ;ROUTINE #132          *
2918 013722 013736          AT133          ;ADDRESS OF NEXT TEST  *
2919 013724 000003          3.          ;ITERATION COUNT      *
2920 013726 013730          DAT32          ;SCOPE ENTRY POINT    *
2921          000132          X=X+1
2922          :*****
2923 013730 004567 167530 DAT32: JSR      5,DATTST      ;LOAD PARAMETERS & RUN TEST
2924 013734 000032          32          ;SEE NOTE 32 FOR DATA TEST PARAMETERS
2925          000033          Y=Y+1

```

```

2926 ;*****
2927 013736 000133 AT133: 133 ;ROUTINE #133 *
2928 013740 013754 AT134 ;ADDRESS OF NEXT TEST *
2929 013742 000003 3. ;ITERATION COUNT *
2930 013744 013746 DAT33 ;SCOPE ENTRY POINT *
2931 000133 X=X+1
2932 ;*****
2933 013746 004567 167512 DAT33: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2934 013752 000033 33 ;SEE NOTE 33 FOR DATA TEST PARAMETERS
2935 000034 Y=Y+1
2936 ;*****
2937 013754 000134 AT134: 134 ;ROUTINE #134 *
2938 013756 013772 AT135 ;ADDRESS OF NEXT TEST *
2939 013760 000003 3. ;ITERATION COUNT *
2940 013762 013764 DAT34 ;SCOPE ENTRY POINT *
2941 000134 X=X+1
2942 ;*****
2943 013764 004567 167474 DAT34: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2944 013770 000034 34 ;SEE NOTE 34 FOR DATA TEST PARAMETERS
2945 000035 Y=Y+1
2946 ;*****
2947 013772 000135 AT135: 135 ;ROUTINE #135 *
2948 013774 014010 AT136 ;ADDRESS OF NEXT TEST *
2949 013776 000003 3. ;ITERATION COUNT *
2950 014000 014002 DAT35 ;SCOPE ENTRY POINT *
2951 000135 X=X+1
2952 ;*****
2953 014002 004567 167456 DAT35: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2954 014006 000035 35 ;SEE NOTE 35 FOR DATA TEST PARAMETERS
2955 000036 Y=Y+1
2956 ;*****
2957 014010 000136 AT136: 136 ;ROUTINE #136 *
2958 014012 014026 AT137 ;ADDRESS OF NEXT TEST *
2959 014014 000003 3. ;ITERATION COUNT *
2960 014016 014020 DAT36 ;SCOPE ENTRY POINT *
2961 000136 X=X+1
2962 ;*****
2963 014020 004567 167440 DAT36: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2964 014024 000036 36 ;SEE NOTE 36 FOR DATA TEST PARAMETERS
2965 000037 Y=Y+1
2966 ;*****
2967 014026 000137 AT137: 137 ;ROUTINE #137 *
2968 014030 014044 AT140 ;ADDRESS OF NEXT TEST *
2969 014032 000003 3. ;ITERATION COUNT *
2970 014034 014036 DAT37 ;SCOPE ENTRY POINT *
2971 000137 X=X+1
2972 ;*****
2973 014036 004567 167422 DAT37: JSR 5,DATTST ;LOAD PARAMETERS & RUN TEST
2974 014042 000037 37 ;SEE NOTE 37 FOR DATA TEST PARAMETERS
2975 000040 Y=Y+1

```

```

2976
2977 014044 000140
2978 014046 014160
2979 014050 000012
2980 014052 014074
2981
2982
2983
2984
2985
2986 014054 012777 000430 165022
2987 014062 012777 003031 165010
2988 014070 004767 166764
2989 014074 012767 001750 165144
2990 014102 105777 164776
2991 014106 100375
2992 014110 004767 167050
2993 014114 110167 165112
2994 014120 042767 177740 165104
2995 014126 110177 164754
2996 014132 105777 164742
2997 014136 100375
2998 014140 117767 164736 165062
2999 014146 104004
3000 014150 005367 165072
3001 014154 001352
3002 014156 104012
3003
3004 014160 000141
3005 014162 177777
3006 014164 000144
3007 014166 014170
3008
3009
3010
3011
3012 014170 012777 000004 164706
3013 014176 012777 000002 164674
3014 014204 012777 000377 164674
3015 014212 105777 164662
3016 014216 100375
3017 014220 027727 164656 000000
3018 014226 001401
3019 014230 104003
3020 014232 104011
3021 014234 104012
3022
3023 014236 052777 000004 164640
3024 014244 005077 164636
3025 014250 104016
3026 014252 000310
3027 014254 000207
3028
3029 014256 026767 164764 164764
3030 014264 101424
3031 014266 026767 164754 164756

```

```

AT140: 140 ;TEST NUMBER 140
        AT141 ;ADDRESS OF NEXT TEST
        10. ;TEST ITERATION COUNT
        APBA ;SCOPE ENTRY POINT
;*****
;DATA TEST USING JUMPER CONNECTOR. TX SPEED = 11, RX SPEED = 11.
;CHAR LENGTH = 11, STOP CODE = 1. USES SPECIAL BINARY COUNT PATTERN
;FOR DATA. NO INTERRUPT.
        MOV #430, @TXCSR ;SET TX SPEED = 11, STOP CODE = 1
        MOV #3031, @RXCSR ;SET RX SPEED = 11, CHAR. LENGTH = 11
        JSR 7, INBIN ;INITIALIZE BINARY COUNT PATTERN
APBA: MOV #1000, CTRA ;SET CHARACTER COUNT TO 1000
APBB: TSTB @TXCSR ;WAIT FOR TX READY
        BPL -4
        JSR 7, GTBINP ;GET BINARY CHARACTER
        MOVB %1, CRBUFA ;SAVE CHAR IN CRBUFA AND
        BIC #177740, CRBUFA ;MASK OFF ALL BUT 5 LSB.
        MOVB %1, @TXBUF ;LOAD CHAR.
        TSTB @RXCSR ;WAIT FOR RECEIVER
        BPL -4 ;TO RECEIVE CHARACTER
        MOVB @RXBUF, CRBUF ;LOAD RECEIVED DATA INTO CRBUF
        DATCHK ;CHECK DATA
        DEC CTRA ;TESTED 1000 CHARACTERS
        BNE APBB ;BRANCH IF NOT
        SCOPE ;YES. SCOPE
;*****
AT141: 141 ;TEST NUMBER 141
        ATLAST ;ADDRESS OF NEXT TEST
        100. ;TEST ITERATION COUNT
        AQBA ;SCOPE ENTRY POINT
;*****
;TEST THAT WHEN RXCSR BIT 1 IS SET THAT THE OUTPUT DATA LINE
;IS PULLED TO A SPACE.
AQBA: MOV #BIT2, @TXCSR ;SET MAINTENANCE BIT IN TXCSR
        MOV #BIT1, @RXCSR
        MOV #377, @TXBUF ;LOAD BUFFER
        TSTB @RXCSR ;WAIT FOR RECEIVER
        BPL -4 ;TO RECEIVE CHARACTER
        CMP @RXBUF, #0 ;CHARACTER RECEIVED SHOULD BE 0
        BEQ .+4
        ERROR ;CHARACTER OTHER THAN 0
        SRESET ;ISSUE RESET
        SCOPE
;SUBROUTINE TO SET RXCSR DONE BIT.
STRXD: BIS #BIT2, @TXCSR ;SET MAINTENANCE BIT.
        CLR @TXBUF ;LOAD TXBUF.
        DELAY ;DELAY 200 MSECS.
        200.
        RTS %7 ;EXIT.
;SUBROUTINE TO CHECK THAT CTRA THROUGH CTRD CONTAIN DESCENDING VALUES.
CMPT: CMP CTRA, CTRB
        BLOS CMPTA
        CMP CTRA, CTRC

```

3032	014274	101420			BLOS	CMPTA
3033	014276	026767	164744	164750	CMP	CTRA,CTRD
3034	014304	101414			BLOS	CMPTA
3035	014306	026767	164736	164736	CMP	CTRB,CTRC
3036	014314	101410			BLOS	CMPTA
3037	014316	026767	164726	164730	CMP	CTRB,CTRD
3038	014324	101404			BLOS	CMPTA
3039	014326	026767	164720	164720	CMP	CTRC,CTRD
3040	014334	101002			BHI	CMPTB
3041	014336	062716	000002		ADD	#2,2%E
3042	014342	000207			RTS	%7

CMPTA: ADD #2,2%E  
CMPTB: RTS %7

```

3043
3044
3045 ;*****
3046 ;PRG1 - TRANSMITTER SCOPE LOOP
3047 ;*****
3047 014344 104000 PRG1: TYPE ;TYPE PROGRAM TITLE.
3048 014346 015765 PITIT
3049 014350 004567 166360 JSR 5,LINSEL ;GO GET LINE # FROM USER
3050 014354 004767 000302 JSR %7,SETPAR ;GO SET PARAMETERS.
3051 014360 104000 TYPE ;TYPE SELECT CHAR AND DELAY.
3052 014362 016217 SELCAD
3053 014364 000000 HALT ;WAIT FOR USER.
3054 014366 116767 163176 000010 PRG1A: MOVB SR,PRG1B ;DELAY COUNT TO PRG1B.
3055 014374 116777 163171 164504 MOVB SR+1,@TXBUF ;LOAD TXBUF.
3056 014402 104016 DELAY ;DELAY # OF MSECS. SET AT SR.
3057 014404 000000 PRG1B: OPEN
3058 014406 000767 BR PRG1A ;REPEAT.
3059
3060 ;*****
3061 ;PRG2 - RECEIVER SCOPE LOOP.
3062 ;*****
3062 014410 104000 PRG2: TYPE ;TYPE PROGRAM TITLE.
3063 014412 016025 P2TIT
3064 014414 004567 166314 JSR 5,LINSEL ;GO GET LINE # FROM USER
3065 014420 004767 000236 JSR %7,SETPAR ;GO SET PARAMETERS.
3066 014424 104000 TYPE ;TYPE SELECT CHAR AND DELAY.
3067 014426 016217 SELCAD
3068 014430 000000 HALT ;WAIT FOR USER.
3069 014432 004767 000256 PRG2A: JSR 7,STPARB ;RELOAD PARAMETERS
3070 014436 052777 000004 164440 BIS #BIT2,@TXCSR ;SET MAINTENANCE BIT.
3071 014444 116767 163120 000010 MOVB SR,PRG2B ;DELAY COUNT TO PRG2B.
3072 014452 116777 163113 164426 MOVB SR+1,@TXBUF ;LOAD TXBUF.
3073 014460 104016 DELAY ;DELAY # OF MSECS. SET IN SR.
3074 014462 000000 PRG2B: OPEN
3075 014464 017700 164412 MOV @RXBUF,%0 ;RXBUF CONTENTS TO RO.
3076 014470 000005 RESET ;DISPLAY CONTENTS OF RXBUF (IN RO),
3077 014472 000005 RESET ;BY ISSUING 5 RESET INSTRUCTIONS
3078 014474 000005 RESET
3079 014476 000005 RESET
3080 014500 000005 RESET
3081 014502 000753 BR PRG2A
3082
3083 ;*****
3084 ;PRG3 - SINGLE CHARACTER MAINTENANCE MODE DATA TEST.
3085 ;*****
3085 014504 104000 PRG3: TYPE ;TYPE PROGRAM TITLE.
3086 014506 016526 P3TIT
3087 014510 004567 166220 JSR 5,LINSEL ;GO GET LINE # FROM USER
3088 014514 004767 000142 JSR %7,SETPAR ;SET PARAMETERS.
3089 014520 104000 TYPE ;TYPE: SELECT CHARACTER.
3090 014522 016672 SELCAR
3091 014524 000000 HALT
3092 014526 116767 163036 164476 PRG3A: MOVB SR,CBUBA ;MOVE DATA CHAR TO CBUBA.
3093 014534 004767 000040 JSR %7,MOUTIN ;GO OUTPUT, RECEIVE, AND CHECK DATA.
3094 014540 000772 BR PRG3A
3095
3096 ;*****
3097 ;PRG4 - SPECIAL BINARY COUNT MAINTENANCE MODE DATA TEST.
3098 ;*****
3098 014542 104000 PRG4: TYPE ;TYPE PROGRAM TITLE.

```

3099	014544	016576				P4TIT		
3100	014546	004567	166162			JSR	5,LINSEL	;GO GET LINE # FROM USER
3101	014552	004767	000104			JSR	%7,SETPAR	;SET PARAMETERS.
3102	014556	004767	166276			JSR	%7,INBIN	;INITIALIZE BINARY COUNT.
3103	014562	004767	166376			JSR	%7,GTBINP	;GET BINARY CHARACTER.
3104	014566	110167	164440			MOVB	%1,CBUBFA	;SAVE AT CBUBFA.
3105	014572	004767	000002			JSR	%7,MOUTIN	;GO OUTPUT, RECEIVE, AND CHECK DATA.
3106	014576	000771				BR	PRG4A	;REPEAT.
3107								
3108	014600	032767	000400	162762		;SUBROUTINE TO OUTPUT, RECEIVE, AND CHECK DATA WITH MAINTENANCE BIT SET.		
3109	014606	001001				MOUTIN: BIT	#BIT8,SR	;SEE IF BIT 8 IS SET.
3110	014610	104002				BNE	+.4	;BRANCH IF SET.
3111	014612	105777	164266			STALL		;SET. DO A RANDOM STALL.
3112	014616	100375				TSTB	@TXCSR	;WAIT FOR TX READY.
3113	014620	052777	000004	164256		BPL	-.4	
3114	014626	016777	164400	164252		BIS	#BIT2,@TXCSR	;SET MAINTENANCE BIT.
3115	014634	046767	164374	164370		MOV	CBUBFA,@TXBUF	;LOAD TXBUF.
						BIC	CARMSK,CBUBFA	;MASK OFF NON-EXPECTED BITS.

E06

.MAIN. MACY11 27(732) 29-SEP-76 14:25 PAGE 69  
DZCCAB.P11

3116	014642	105777	164232	TSTB	@RXCSR	;WAIT FOR RECEIVER DONE BIT.
3117	014646	100375		BPL	.-4	
3118	014650	017767	164226 164352	MOV	@RXBUF,CRBUF	:MOVE CHAR IN RX BUFFER TO CRBUF
3119	014656	104004		DATCHK		:COMPARE EXPECTED AND RECEIVED DATA
3120	014660	000207		RTS	%7	:EXIT.

```

3121
3122
3123
3124 014662 104000
3125 014664 016062
3126 014666 000000
3127 014670 016767 162674 164370
3128 014676 004567 166330
3129 014702 001266
3130 014704 016667
3131 014706 000002
3132 014710 104000
3133 014712 016651
3134 014714 032767 000020 164344 STPARB: BIT #BIT4,SRT ;SEE IF SR BIT 4 IS SET.
3135 014722 001403 BEQ .+10 ;BRANCH IF NOT SET.
3136 014724 052777 000400 164152 BIS #BIT8,@TXCSR ;SET. SET STOP CODE TO A 1.
3137 014732 032767 000010 164326 BIT #BIT3,SRT ;SEE IF SR BIT 3 IS SET.
3138 014740 001406 BEQ .+16 ;BRANCH IF NOT SET.
3139 014742 052777 000020 164134 BIS #BIT4,@TXCSR ;SET BIT4 IN TXCSR AND RXCSR
3140 014750 052777 000020 164122 BIS #BIT4,@RXCSR ;(MSB OF SPEED SELECT BITS.)
3141 014756 032767 000004 164302 BIT #BIT2,SRT ;SEE IF SR BIT 2 IS SET.
3142 014764 001406 BEQ .+16 ;BRANCH IF NOT SET.
3143 014766 052777 000010 164110 BIS #BIT3,@TXCSR ;SET BIT3 IN TXCSR AND RXCSR
3144 014774 052777 000010 164076 BIS #BIT3,@RXCSR ;(LSB OF SPEED SELECT BITS.)
3145 015002 012767 177400 164224 MOV #177400,CARMSK ;SET CHARACTER MASK TO 8 BITS.
3146 015010 032767 000002 164250 BIT #BIT1,SRT ;SEE IF SR BIT 1 IS SET.
3147 015016 001421 BEQ STPARA ;BRANCH IF NOT SET.
3148 015020 012767 177700 164206 MOV #177700,CARMSK ;CHANGE CHAR MASK TO 6 BITS.
3149 015026 052777 002000 164044 BIS #BIT10,@RXCSR ;SET RXCSR BIT 10(MSB OF CHAR LENGTH BITS.)
3150 015034 032767 000001 164224 BIT #BIT0,SRT ;SEE IF SR BIT0 IS SET.
3151 015042 001406 BEQ .+16 ;BRANCH IF NOT SET.
3152 015044 012767 177700 164162 MOV #177700,CARMSK ;CHANGE CHAR MASK TO 5 BITS.
3153 015052 052777 000000 164020 BIS #BIT9,@RXCSR ;SET RXCSR BIT9 (LSB OF CHAR LENGTH BITS.)
3154 015060 000207 RTS %7 ;EXIT.
3155 015062 032767 000001 164176 STPARA: BIT #BIT0,SRT ;SEE IF SR BIT0 IS SET.
3156 015070 001773 BEQ STPARA-2 ;BRANCH IF NOT SET.
3157 015072 012767 177600 164134 MOV #177600,CARMSK ;CHANGE CHAR MASK TO 7 BITS.
3158 015100 000764 BR STPARA-10

```

:VECTOR ASSIGNMENT TABLE  
VECTAB:

LINE	VECTOR
0	VECTOR
1	VECTOR
2	"
3	"
4	"
5	"
6	"
7	"
8	"
9	"
10	"
11	"
12	"
13	"
14	"
15	"
16	"
17	"

3161	015102	000300
3162	015104	000310
3163	015106	000320
3164	015110	000330
3165	015112	000340
3166	015114	000350
3167	015116	000360
3168	015120	000370
3169	015122	000400
3170	015124	000410
3171	015126	000420
3172	015130	000430
3173	015132	000440
3174	015134	000450
3175	015136	000460
3176	015140	000470



3194						
3195						: ASCII MESSAGES
3196	015202	050045	040			EMO: .ASCII '%P'
3197	015205	040	020040	052040		APNUMB: .ASCII ' T '
3198	015212	040				
3199	015213	040	020040	050040		ATNUMB: .ASCII ' PC '
3200	015220	020103				
3201	015222	020040	020040	020040		APC: .ASCII ' @ '
3202	015230	020040	100			
3203	015233	045	050045	043522		POTIT: .ASCII '%XPRGO - INPUT-OUTPUT LOGIC TESTS. '
3204	015240	020060	020055	047111		
3205	015246	052520	026524	052517		
3206	015254	050124	052125	046040		
3207	015262	043517	041511	052040		
3208	015270	051505	051524	020056		
3209	015276	044504	041523	047117		.ASCII 'DISCONNECT DC11 FROM MODEM.%'
3210	015304	042516	052103	042040		
3211	015312	030503	020061	051106		
3212	015320	046517	046440	042117		
3213	015326	046505	022456			
3214	015332	047101	020104	047503		.ASCII 'AND CONNECT JUMPER TO CABLE.%@'
3215	015340	047116	041505	020124		
3216	015346	052512	050115	051105		
3217	015354	052040	020117	040503		
3218	015362	046102	027105	040045		
3219	015370	054124	051503	020122		ATXCSR: .ASCII 'TXCSR S/B: '
3220	015376	027523	035102	040		
3221	015403	040	020040	020040		ATXSB: .ASCII ' WAS: '
3222	015410	020040	053440	051501		
3223	015416	020072				
3224	015420	020040	020040	020040		ATXWAS: .ASCII ' @ '
3225	015426	100				
3226	015427	122	041530	051123		ARXCSR: .ASCII 'RXCSR S/B: '
3227	015434	051440	041057	020072		
3228	015442	020040	020040	020040		ARXSB: .ASCII ' WAS: '
3229	015450	020040	040527	035123		
3230	015456	040				
3231	015457	040	020040	020040		ARXWAS: .ASCII ' @ '
3232	015464	040040				
3233	015466	054124	051440	042520		ETXTIM: .ASCII 'TX SPEEDS NOT IN ASCENDING ORDER.@'
3234	015474	042105	020123	047516		
3235	015502	020124	047111	040440		
3236	015510	041523	047105	044504		
3237	015516	043516	047440	042122		
3238	015524	051105	040056			
3239	015530	044524	042515	043040		ESTPCD: .ASCII 'TIME FOR 2 STOP CODE OP LESS THAN FOR 1.@'
3240	015536	051117	031040	051440		
3241	015544	047524	020120	047503		
3242	015552	042504	047440	020120		
3243	015560	042514	051523	052040		
3244	015566	040510	020116	047506		
3245	015574	020122	027061	100		
3246	015601	124	020130	044103		ETCLGT: .ASCII 'TX CHAR LENGTHS NOT IN DESCENDING ORDER.@'
3247	015606	051101	046040	047105		
3248	015614	052107	051510	047040		
3249	015622	052117	044440	020116		

.MAIN. MACY11 27(732) 29-SEP-76 14:25 PAGE 73  
 0ZDCAB.P11

3250	015630	042504	041523	047105	
3251	015636	044504	043516	047440	
3252	015644	042122	051105	040056	
3253	015652	054122	051440	042520	ERXTIM: .ASCII 'RX SPEEDS NOT IN ASCENDING ORDER.3'
3254	015660	042105	020123	047516	
3255	015666	020124	047111	040440	
3256	015674	041523	047105	044504	
3257	015702	043516	047440	042122	
3258	015710	051105	040056		
3259	015714	054122	041440	040510	ERCLGT: .ASCII 'RX CHAR LENGTHS NOT IN DESCENDING ORDER.3'
3260	015722	020122	042514	043516	
3261	015730	044124	020123	047516	
3262	015736	020124	047111	042040	
3263	015744	051505	042503	042116	
3264	015752	047111	020107	051117	
3265	015760	042504	027122	100	
3266	015765	045	050045	043522	P1TIT: .ASCII '%PRG1 - TRANSMITTER SCOPE LOOP3'
3267	015772	020061	020055	051124	
3268	016000	047101	046523	052111	
3269	016006	042524	020122	041523	
3270	016014	050117	020105	047514	
3271	016022	050117	100		
3272	016025	045	050045	043522	P2TIT: .ASCII '%PRG2 - RECEIVER SCOPE LOOP3'
3273	016032	020062	020055	042522	
3274	016040	042503	053111	051105	
3275	016046	051440	047503	042520	
3276	016054	046040	047517	040120	
3277	016062	051445	052105	050040	SELPAR: .ASCII '%SET PARAMETERS IN SR AS FOLLOWS:'
3278	016070	051101	046501	052105	
3279	016076	051105	020123	047111	
3280	016104	051440	020122	051501	
3281	016112	043040	046117	047514	
3282	016120	051527	072		
3283	016123	045	051123	020064	.ASCII '%SR4 = STOP CODE%SR3 AND 2 = SPEED'
3284	016130	020075	052123	050117	
3285	016136	041440	042117	022505	
3286	016144	051123	020063	047101	
3287	016152	020104	020062	020075	
3288	016160	050123	042505	104	
3289	016165	045	051123	020061	.ASCII '%SR1 AND 0 = CHAR LENGTH%3'
3290	016172	047101	020104	020060	
3291	016200	020075	044103	051101	
3292	016206	046040	047105	052107	
3293	016214	022510	100		
3294	016217	045	042523	020124	SELCAD: .ASCII '%SET TEST CHAR CODE IN SR15-SR8, SET DELAY TIME IN SR7-SR0.3'
3295	016224	042524	052123	041440	
3296	016232	040510	020122	047503	
3297	016240	042504	044440	020116	
3298	016246	051123	032461	051455	
3299	016254	034122	020054	042523	
3300	016262	020124	042504	040514	
3301	016270	020131	044524	042515	
3302	016276	044440	020116	051123	
3303	016304	026467	051123	027060	
3304	016312	100			
3305	016313	040	042040	052101	ERDAT: .ASCII ' DATA ERR S/B: '

3306	016320	020101	051105	020122	
3307	016326	051440	041057	020072	
3308	016334	020040	020040	053440	AASB: .ASCII ' WAS: '
3309	016342	051501	020072		
3310	016346	020040	040040		AWAS: .ASCII ' @'
3311	016352	037445	100		AINPRG: .ASCII '%?@'
3312	016355	045	042523	020124	ASETSR: .ASCII '%SET DESIRED SR OPTIONS. NORMAL OPERATION '
3313	016362	042504	044523	042522	
3314	016370	020104	051123	047440	
3315	016376	052120	047511	051516	
3316	016404	020056	047516	046522	
3317	016412	046101	047440	042520	
3318	016420	040522	044524	047117	
3319	016426	040			
3320	016427	111	020123	044527	.ASCII 'IS WITH SR = 000000@'
3321	016434	044124	051440	020122	
3322	016442	020075	030060	030060	
3323	016450	030060	100		
3324	016453	045	047111	047503	AINCRT: .ASCII '%INCORRECT ROUTINE SELECTED.@'
3325	016460	051122	041505	020124	
3326	016466	047522	052125	047111	
3327	016474	020105	042523	042514	
3328	016502	052103	042105	040056	
3329	016510	050045	047522	051107	APGEN: .ASCII '%PROGRAM END.@'
3330	016516	046501	042440	042116	
3331	016524	040056			
3332	016526	022445	051120	031507	P3TIT: .ASCII '%%PRG3-SINGLE CHAR MAINT MODE DATA TEST@'
3333	016534	051455	047111	046107	
3334	016542	020105	044103	051101	
3335	016550	046440	044501	052116	
3336	016556	046440	042117	020105	
3337	016564	040504	040524	052040	
3338	016572	051505	040124		
3339	016576	022445	051120	032107	P4TIT: .ASCII '%%PRG4-SPEC BIN COUNT MAINT MODE DATA TEST@'
3340	016604	051455	042520	020103	
3341	016612	044502	020116	047503	
3342	016620	047125	020124	040515	
3343	016626	047111	020124	047515	
3344	016634	042504	042040	052101	
3345	016642	020101	042524	052123	
3346	016650	100			
3347	016651	045	040520	040522	PARMTS: .ASCII '%PARAMETERS = '
3348	016656	042515	042524	051522	
3349	016664	036440	040		
3350	016667	040	040040		APARM: .ASCII ' @'
3351	016672	051445	052105	052040	SELCAR: .ASCII '%SET TEST CHAR CODE IN SR7-SR0.@'
3352	016700	051505	020124	044103	
3353	016706	051101	041440	042117	
3354	016714	020105	047111	051440	
3355	016722	033522	051455	030122	
3356	016730	040056			
3357	016732	046045	040517	020104	LDLINE: .ASCII '%LOAD LINE NO. (8) INTO SR 3-7@'
3358	016740	044514	042516	047040	
3359	016746	027117	024040	024470	
3360	016754	044440	052116	020117	
3361	016762	051123	031440	033455	

K06

.MAIN. MACY11 27(732) 29-SEP-76 14:25 PAGE 75  
DZDCAB.P11

3362	016770	100					
3363	016771	040	046040	047111	ALINE: .ASCII	'	LINE NO.'
3364	016776	020105	047516	056			
3365	017003	040	020040	040527	SELINE: .ASCII	'	WAS SELECTED@'
3366	017010	020123	042523	042514			
3367	017016	052103	042105	100			
3368	017023	000001			DEND: .END		

AAA	003714	1156	1159#				
AAAA	006226	1666	1669#				
AAAB	006240	1670	1672#				
AAB	003726	1161#	1164				
AAZA	012546	2548	2553#				
AABB	012572	2559#	2565	2568			
AABC	012602	2554	2561#				
AABD	012616	2562	2564#				
AABE	012622	2561	2566#				
AAE	003730	1159	1162#				
AASB	016334	740	3308#				
ABA	003746	1169	1172#				
ABAA	007224	1879	1882#				
ABAB	007316	1886	1899#				
ABB	003760	1174#	1177				
ABBA	012640	2574	2577#				
ABBB	012672	2583	2585#				
ABE	003762	1172	1175#				
ACA	004000	1183	1187#				
ACAA	007330	1904	1908#				
ACAB	007464	1921	1933#				
ACB	004012	1189#	1192				
ACBA	012720	2595	2600#				
ACBB	012756	2599	2608#				
ACBC	012760	2607	2609#				
ACE	004014	1187	1190#				
ADA	004032	1199	1202#				
ADAA	007504	1939	1942#				
ADAB	007520	1944	1946#				
ADB	004044	1204#	1207				
ADE	004046	1202	1205#				
ADTENP	003444	1079	1098#				
AEA	004064	1213	1216#				
AEAA	007534	1952	1956#				
AEAB	007554	1960	1963#				
AEB	004100	1217	1220#				
AEC	004122	1222	1225#				
AED	004142	1219	1224	1227	1229#		
AFA	004164	1236	1239#				
AFAA	007566	1969	1973#				
AFAB	007614	1978	1981#				
AFB	004206	1241	1244#				
AFBA	006252	1683	1688#				
AFBB	006274	1690	1693#				
AFBC	006314	1692	1695	1697#			
AFC	004230	1246	1249#				
AFD	004250	1243	1248	1251	1253#		
AGA	004262	1259	1262#				
AGAA	007626	1986	1990#				
AGAB	007654	1995	1998#				
AGB	004276	1263	1266#				
AGBA	006330	1703	1708#				
AGBB	006360	1711	1714#				
AGBC	006410	1720	1723#				
AGBD	006424	1725	1729#				
AGBE	006454	1713	1722	1727	1735	1737	1738#

.MAIN. MACY11 27(732) 29-SEP-76 14:25 PAGE 78  
 DZDCAB.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

AGC	004320	1268	1271#					
AGD	004340	1265	1270	1273	1275#			
AHA	004362	1282	1285#					
AHAA	007666	2003	2007#					
AHAB	007714	2012	2015#					
AHB	004376	1286	1289#					
AHC	004420	1291	1294#					
AHD	004440	1288	1293	1296	1298#			
AIA	004462	1306	1309#					
AIAA	007726	2021	2025#					
AIAF	010066	2043	2046#					
AIAS	010070	2030	2033	2037	2040	2047#		
AIASA	010112	2053#	2057					
AIAST	010132	2031	2034	2038	2041	2047*	2055*	2059#
AIB	004476	1310	1313#					
AIC	004520	1315	1318#					
AID	004540	1312	1317	1320	1322#			
AINCRT	016453	715	3324#					
AINPRG	016352	3311#						
AJA	004562	1329	1332#					
AJAA	010144	2064	2069#					
AJAB	010230	2078	2082#					
AJAS	010232	2072	2075	2083#				
AJASA	010254	2089#	2093					
AJASB	010276	2095#	2099					
AJAST	010316	2073	2076	2083*	2091*	2097*	2101#	
AJB	004604	1334	1337#					
AJBA	006604	1773	1778#					
AJBB	006622	1780	1783#					
AJBC	006644	1785	1788#					
AJBD	006664	1787	1790	1792#				
AJC	004626	1339	1342#					
AJD	004646	1336	1341	1344	1346#			
AKA	004670	1355	1359#					
AKAA	010330	2107	2112#					
AKAB	010476	2131	2134#					
AKAS	010500	2117	2121	2125	2128	2135#		
AKASA	010522	2141#	2145					
AKAST	010542	2118	2122	2126	2129	2135*	2143*	2147#
AKB	004700	1360	1362#					
AKBA	006700	1798	1804#					
AKBB	006706	1805#						
AKBC	006736	1808	1811#					
AKBD	006752	1812	1815#					
AKBE	007002	1810	1814	1819	1822#			
ALA	004712	1367	1370#					
ALAA	010554	2153	2158#					
ALAB	010604	2163	2166#					
ALAC	010616	2165	2168	2170#				
ALB	004726	1371	1374#					
ALBA	007124	1854	1859#					
ALBB	007146	1861	1864#					
ALBC	007170	1866	1869#					
ALBD	007210	1863	1868	1871	1873#			
ALC	004750	1376	1379#					
ALD	004770	1373	1378	1381	1383#			

ALINE	016771	1015	3363#					
AMA	005012	1392	1395#					
AMAA	010630	2175	2179#					
AMAB	010642	2181#	2182					
AMAC	010664	2185	2187#					
AMB	005024	1396	1398#					
AMBA	006470	1745	1750#					
AMBB	006526	1755	1758#					
AMBC	006542	1759	1762#					
AMBD	006570	1757	1761	1765		1767#		
ANA	005036	1404	1407#					
ANAA	010676	2193	2198#					
ANAB	010746	2206	2208#					
ANB	005060	1409	1412#					
ANBA	012772	2620#						
ANBB	013004	2617	2622#					
ANBC	013012	2623#	2652					
ANBD	013054	2632#	2636					
ANBE	013110	2638	2642#					
ANBF	013114	2640	2644#					
ANBG	013134	2641	2643	2651#				
ANC	005076	1414	1417#					
AND	005116	1411	1416	1419		1421#		
ANW	005130	1426	1430#					
ANX	005144	1431	1434#					
ANY	005166	1436	1439#					
ANZ	005206	1433	1438	1441		1443#		
AOAA	010764	2215	2220#					
AOAB	011034	2228	2230#					
AOBA	007016	1828	1832#					
AOBB	007040	1834	1837#					
AOBC	007070	1840	1844#					
AOBD	007110	1836	1842	1845		1847#		
APA	005230	1453	1456#					
APAA	011052	2236	2241#					
APAB	011106	2247	2249#					
APARM	016667	3130	3350#					
APB	005242	1457	1459#					
APBA	014074	2980	2989#					
APBB	014102	2990#	3001					
APC	015222	761	3201#					
APGEND	016510	719	3329#					
APNUMB	015205	765	3197#					
AQA	005254	1464	1467#					
AQAA	011124	2256	2261#					
AQAB	011344	2287	2290#					
AQAS	011346	2268	2274	2280		2284	2291#	
AQASA	011374	2298#	2302					
AQAST	011414	2269	2275	2281		2285	2291*	2300* 2304#
AQB	005270	1468	1471#					
AQBA	014170	3007	3012#					
AQC	005312	1473	1476#					
AQD	005332	1477	1475	1478		1480#		
ARA	005354	1477	1490#					
ARAA	011426	2310	2315#					
ARAB	011574	2334	2337#					

ARAS	011576	2320	2324	2328	2331	2338#		
ARASA	011632	2347#	2351					
ARAST	011652	2321	2325	2329	2332	2338*	2349*	2353#
ARB	005370	1491	1494#					
ARBA	005454	1511	1516#					
ARBB	005466	1517	1519#					
ARC	005412	1496	1499#					
ARD	005432	1493	1498	1501	1503#			
ARXCSR	015427	1932	3226#					
ARXSB	015442	1925	3228#					
ARXWAS	015457	1929	3231#					
ASA	005500	1525	1528#					
ASAA	011664	2359	2362#					
ASAB	011716	2366	2369#					
ASAC	011730	2370	2374#					
ASAD	011742	2368	2373	2376	2379#			
ASAS	011750	2362	2363	2381#				
ASB	005522	1530	1533#					
ASC	005544	1535	1538#					
ASD	005564	1532	1537	1540	1542#			
ASETSR	016355	711	3312#					
ATA	005606	1549	1552#					
ATAA	012010	2391	2398#					
ATAB	012034	2403#	2405					
ATAC	012036	2397	2404#					
ATB	005620	1553	1555#					
ATLAST=	177777	640#	3005					
ATNUMB	015213	769	3199#					
ATXCSR	015370	1898	3219#					
ATXSB	015403	1891	3221#					
ATXWAS	015420	1895	3224#					
ATO	003704	1144	1153#					
AT1	003736	1154	1166#					
AT10	004452	1280	1303#					
AT100	013144	2615	2657#					
AT101	013162	2658	2667#					
AT102	013200	2668	2677#					
AT103	013216	2678	2687#					
AT104	013234	2688	2697#					
AT105	013252	2698	2707#					
AT106	013270	2708	2717#					
AT107	013306	2718	2727#					
AT11	004552	1304	1326#					
AT110	013324	2728	2737#					
AT111	013342	2738	2747#					
AT112	013360	2748	2757#					
AT113	013376	2758	2767#					
AT114	013414	2768	2777#					
AT115	013432	2778	2787#					
AT116	013450	2788	2797#					
AT117	013466	2798	2807#					
AT12	004660	1327	1352#					
AT120	013504	2808	2817#					
AT121	013522	2818	2827#					
AT122	013540	2828	2837#					
AT123	013556	2838	2847#					

.MAIN. MACY11 27(732) 29-SEP-76 14:25 PAGE 81  
 DZDCAB.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

AT124	013574	2848	2857#
AT125	013612	2858	2867#
AT126	013630	2868	2877#
AT127	013646	2878	2887#
AT13	004702	1353	1364#
AT130	013664	2888	2897#
AT131	013702	2898	2907#
AT132	013720	2908	2917#
AT133	013736	2918	2927#
AT134	013754	2928	2937#
AT135	013772	2938	2947#
AT136	014010	2948	2957#
AT137	014026	2958	2967#
AT14	005002	1365	1389#
AT140	014044	2968	2977#
AT141	014160	2978	3004#
AT15	005026	1390	1401#
AT16	005120	1402	1423#
AT17	005220	1424	1450#
AT2	003770	1167	1180#
AT20	005244	1451	1461#
AT21	005344	1462	1484#
AT22	005444	1485	1508#
AT23	005470	1509	1522#
AT24	005576	1523	1546#
AT25	005622	1547	1558#
AT26	005722	1559	1581#
AT27	006022	1582	1604#
AT3	004022	1181	1196#
AT30	006122	1605	1627#
AT31	006146	1628	1639#
AT32	006172	1640	1652#
AT33	006216	1653	1663#
AT34	006242	1664	1680#
AT35	006320	1681	1700#
AT36	006460	1701	1742#
AT37	006574	1743	1770#
AT4	004054	1197	1210#
AT40	006670	1771	1795#
AT41	007006	1796	1825#
AT42	007114	1826	1851#
AT43	007214	1852	1876#
AT44	007320	1877	1901#
AT45	007474	1902	1936#
AT46	007524	1937	1949#
AT47	007556	1950	1966#
AT5	004154	1211	1233#
AT50	007616	1967	1983#
AT51	007656	1984	2000#
AT52	007716	2001	2018#
AT53	010134	2019	2061#
AT54	010320	2062	2104#
AT55	010544	2105	2150#
AT56	010620	2151	2172#
AT57	010666	2173	2190#
AT6	004252	1234	1256#

.MAIN. MACY11 27(732) 29-SEP-76 14:25 PAGE 82  
 DZDCAB.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

AT60	010754	2191	2212#		
AT61	011042	2213	2233#		
AT62	011114	2234	2253#		
AT63	011416	2254	2307#		
AT64	011654	2308	2356#		
AT65	011770	2357	2388#		
AT66	012042	2389	2409#		
AT67	012120	2410	2431#		
AT7	004352	1257	1279#		
AT70	012204	2432	2453#		
AT71	012302	2454	2479#		
AT72	012360	2480	2501#		
AT73	012442	2502	2522#		
AT74	012532	2523	2545#		
AT75	012630	2546	2571#		
AT76	012704	2572	2592#		
AT77	012762	2593	2614#		
AUA	005632	1561	1564#		
AUAA	012056	2412	2418#		
AUAB	012102	2424#			
AUAC	012110	2417	2425#	2428	
AUB	005646	1565	1568#		
AUC	005670	1570	1573#		
AUD	005710	1567	1572	1575	1577#
AVA	005732	1584	1587#		
AVAA	012134	2434	2440#		
AVAB	012172	2439	2447#		
AVAC	012174	2446	2448#		
AVB	005746	1588	1591#		
AVC	005770	1593	1596#		
AVD	006010	1590	1595	1598	1600#
AWA	006032	1607	1610#		
AWAA	012214	2456	2460#		
AWAB	012244	2467#	2473	2476	
AWAC	012254	2461	2469#		
AWAD	012270	2470	2472#		
AWAE	012274	2469	2474#		
AWAS	016346	736	2647	2650	3310#
AWB	006046	1611	1614#		
AWC	006070	1616	1619#		
AWD	006110	1613	1618	1621	1623#
AXA	006132	1630	1633#		
AXAA	012326	2482	2490#		
AXAB	012354	2488	2496#		
AXAC	012356	2495	2497#		
AXB	006144	1634	1636#		
AYA	006156	1642	1645#		
AYAA	012400	2504	2511#		
AYAB	012424	2515#	2519		
AYAC	012434	2509	2517#		
AYB	006170	1646	1648#		
AZA	006202	1655	1658#		
AZAA	012462	2525	2532#		
AZAB	012520	2530	2540#		
AZAC	012522	2538	2541#		
AZB	006214	1659	1661#		









MAIN MACY11 27(732) 29-SEP-76 14:25 PAGE 87  
 DZDCAB.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

RTNNO	001146	664*	768	784*	804	828*								
RXBUF	001102	646*	1118	1173	2183	2208	2230	2249	2296	2343	2379	2587	2630	2998
RXBUFT	001262	3017	3075	3118										
RXCSR	001100	706*												
		645*	995	1107*	1116	1160	1239*	1244*	1249*	1407*	1408	1413	1417*	1418
		1430	1434*	1435	1439*	1440	1443*	1456	1467	1471*	1472	1476*	1477	1480*
		1490	1494*	1495	1499*	1500	1503*	1516	1529	1533*	1534	1538*	1539	1542*
		1552	1564	1568*	1569	1573*	1574	1577*	1587	1591*	1592	1596*	1597	1600*
		1610	1614*	1615	1619*	1620	1623*	1633	1645	1658	1669	1688*	1689	1693*
		1694	1708*	1709	1710	1714*	1718	1723	1729*	1733	1750*	1751*	1753	1758
		1762*	1763	1778*	1783*	1788*	1807	1811	1818	1833	1839	1844	1859*	1864*
		1869*	1909*	1910*	1919	1933*	2116*	2119*	2120*	2123*	2124*	2127*	2162	2167
		2181	2184	2200*	2201*	2205	2222*	2223*	2227	2242*	2246	2267*	2272*	2273*
		2278*	2279*	2283*	2301	2319*	2322*	2323*	2326*	2327*	2330*	2350	2364	2374
		2490*	2492*	2511*	2513*	2515*	2532*	2535*	2541*	2555*	2556*	2559*	2581	2582
		2602*	2604*	2628	2637	2987*	2996	3013*	3015	3116	3140*	3144*	3149*	3153*
RXCSRT	001260	705*	1709*	1718*	1719	1733*	1734	1753*	1754	1763*	1764	1919*	1920	1928
		2364*	2365	2369	2581*									
RXLVL	001112	650*	878	2512	2533									
RXVTR	001110	649*	876	1004*	2561*									
SAVREG=	104013	635*	1067											
SAVRG	002140	684	847*											
SCOPE =	104012	634*	1123	1161	1174	1189	1204	1231	1253	1277	1300	1324	1348	1362
		1385	1398	1421	1445	1459	1482	1505	1519	1544	1555	1579	1602	1625
		1636	1648	1661	1672	1698	1739	1769	1793	1823	1848	1874	1899	1934
		1947	1963	1981	1998	2015	2046	2082	2134	2170	2187	2209	2231	2250
		2290	2337	2380	2403	2425	2449	2468	2497	2516	2542	2560	2589	2609
		2653	3002	3021										
SCOPTR	001154	667*	813	831*	834*									
SELCAD	016217	3052	3067	3294*										
SELCAR	016672	3090	3351*											
SELINE	017003	1012	3365*											
SELPAR	016062	3125	3277*											
SETPAR	014662	3050	3065	3088	3101	3124*								
SETSR	001272	710*	1149											
SPBOT =	001076	594*	781	795	1129									
SR =	177570	591*	720	726	755	785	798	802	811	815	821	991	3054	3055
		3071	3072	3092	3108	3127								
SRESET=	104011	633*	796	1230	1276	1299	1323	1347	1384	1412	1444	1481	1504	1543
		1578	1601	1624	1697	1738	1767	1792	1822	1847	1873	1884	1918	1946
		2166	3020											
SRSETT	002320	682	888*	890*										
SRT	001266	708*	1105*	3127*	3129	3134	3137	3141	3146	3150	3155			
STAL	002664	675	972*											
STALA	002704	975*	977*											
STALB	002706	974	978*											
STALL =	104002	626*	3110											
START	001562	643	781*											
STLMSK	001270	709*	973											
STLSPV	002270	680	881*											
STLSRV	002240	679	874*											
STPARA	015062	3147	3155*	3156	3158									
STPARB	014714	1106	1134	3069	3134*									
STPPA	002306	881*	884*											
STPRA	002256	874*	877*											
STRXD	014236	2489	2510	2531	2552	2577	2600	2601	3023*					

STRXV = 104006	630#	2487	2508	2529	2553	2598								
STTXV = 104007	631#	2396	2416	2438	2460									
SUBTEN 003376	1082	1086#												
SUBTNA 003402	1087#	1090												
SUBTNB 003416	1088	1091#												
SVRPC 002174	847*	855	857#											
SVRPSW 002176	848*	854	858#											
TEMP 001264	707#	993*	1007*	1008*	1009*	1011	1888*	1890	1922*	1924				
TENPWR 003442	1081*	1087	1091	1097#										
TKB 001122	654#													
TKLVL 001132	658#													
TKS 001120	653#													
TKVTR 001130	657#													
TPB 001126	656#	924*												
TPLVL 001136	660#													
TPS 001124	655#	925												
TPVTR 001134	659#													
TX = 000000	1151#	1184#												
TXBUF 001106	648#	1115*	1203	1912*	1915*	1942*	1956*	1974*	1991*	2008*	2052*	2088*	2094*	
	2140*	2159*	2180*	2202*	2224*	2243*	2297*	2345*	2382*	2578*	2627*	2995*	3014*	
	3024*	3055*	3072*	3114*										
TXCSR 001104	647#	1108*	1110	1188	1216	1220*	1221	1225*	1226	1229*	1240	1245	1250	
	1262	1266*	1267	1271*	1272	1275*	1285	1289*	1290	1294*	1295	1298*	1309	
	1313*	1314	1318*	1319	1322*	1333	1337*	1338	1342*	1343	1346*	1359	1370	
	1374*	1375	1379*	1380	1383*	1395	1779	1784	1789	1804*	1805*	1806*	1815*	
	1816*	1832*	1837*	1838*	1843*	1860	1865	1870	1883*	1885	1887	1911*	1913	
	1916	1943	1959	1973*	1977	1990*	1994	2007*	2011	2029*	2032*	2035*	2036*	
	2039*	2048	2056	2071*	2074*	2084	2092	2098	2136	2144	2158*	2179*	2198*	
	2199*	2220*	2221*	2241*	2265*	2266*	2270*	2271*	2276*	2277*	2282*	2292	2339	
	2344*	2361*	2398*	2400*	2419*	2420*	2424*	2440*	2443*	2448*	2462*	2464*	2467*	
	2621*	2625	2986*	2990	3012*	3023*	3070*	3111	3113*	3136*	3139*	3143*		
	704#	1887*	1894											
TXCSRT 001256	652#	885	2418	2441										
TXLVL 001116	651#	883	1006*	2469*										
TXVTR 001114	673	911#												
TYP 002410	914#	923	932											
TYPA 002420	916	918#												
TYPC 002436	922	924#	929	931										
TYPD 002464	914*	915	918	920	924	928*	930*	933#						
TYPDAT 002530	624#	710	714	718	941	988	1014	1145	3047	3051	3062	3066	3085	
TYPE = 104000	3089	3098	3124	3132										
	625#	771												
TYPES = 104001	919	928#												
TYPF 002502	921	930#												
TYPG 002514	674	935#	943											
TYPS 002532	939	941#												
TYPSA 002556	937*	938	942#											
TYPSB 002560	1003	3161#												
VECTAB 015102	2654#	2656	2661#	2666	2671#	2676	2681#	2686	2691#	2696	2701#	2706	2711#	
X = 000137	2716	2721#	2726	2731#	2736	2741#	2746	2751#	2756	2761#	2766	2771#	2776	
	2781#	2786	2791#	2796	2801#	2806	2811#	2816	2821#	2826	2831#	2836	2841#	
	2846	2851#	2856	2861#	2866	2871#	2876	2881#	2886	2891#	2896	2901#	2906	
	2911#	2916	2921#	2926	2931#	2936	2941#	2946	2951#	2956	2961#	2966	2971#	
Y = 000040	2655#	2656	2665#	2666	2675#	2676	2685#	2686	2695#	2696	2705#	2706	2715#	
	2716	2725#	2726	2735#	2736	2745#	2746	2755#	2756	2765#	2766	2775#	2776	
	2785#	2786	2795#	2796	2805#	2806	2815#	2816	2825#	2826	2835#	2836	2845#	





M07

.\$POWE	1#
.\$RAND	1#
.\$RDDE	1#
.\$RDOC	1#
.\$READ	1#
.\$R2AZ	1#
.\$SAVE	1#
.\$SB2D	1#
.\$SB2O	1#
.\$SCOP	1#
.\$SIZE	1#
.\$SUPR	1#
.\$STRAP	1#
.\$STYPB	1#
.\$STYPD	1#
.\$STYPE	1#
.\$STYPO	1#
.\$4OCA	1#
.1170	1#

ADD	776 2034 2329	842 2038 2332	875 2041 3041	882 2073	898 2076	902 2118	912 2122	936 2126	951 2129	957 2269	1053 2275	1056 2281	1091 2285	1092 2321	2031 2325	
ASL	788	840	1132													
ASR	1001	1002	1007	1008	1009											
BCC	2633															
BCS	1088															
BEQ	721 1286 1491 1670 1886 2078 786	733 1296 1501 1695 1921 3040 803	812 1310 1517 1711 2376	919 1320 1530 1725 2638	921 1334 1540 1735 2640	950 1344 1553 1759 2643	959 1371 1565 1765 3018	974 1381 1575 1780 3135	982 1396 1588 1790 3138	1217 1419 1598 1812 3142	1227 1431 1611 1819 3147	1241 1441 1621 1834 3151	1251 1457 1634 1845 3156	1263 1468 1646 1861	1273 1478 1659 1871	
BHI	1249 1708 1933 2276 2532	1271 1729 2029 2278 2541	1294 1750 2035 2319 2555	973 1318 1762 2071 2322 2559	981 1342 1778 2116 2326 2602	992 1379 1788 2119 2398 2994	997 1417 1804 2123 2419 3115	1036 1439 1806 2198 2424	1045 1476 1816 2200 2440	1055 1499 1832 2220 2448	1058 1538 1838 2222 2462	1107 1573 1843 2265 2467	1114 1596 1859 2267 2490	1225 1619 1869 2270 2511	1239 1693 1909 2272 2515	
BIC	786	803	841	973	981	992	997	1036	1045	1055	1058	1107	1114	1225	1239	
BIS	1407 1688 2124 2282 2604	1434 1783 2127 2283 2621	1443 1805 2158 2323 3023	1229 1471 1815 2179 2327	1244 1480 1837 2199 2330	1266 1494 1864 2201 2344	1275 1503 1911 2221 2381	1289 1533 1973 2223 2400	1298 1542 1990 2241 2420	1313 1568 2007 2242 2443	1322 1577 2032 2266 2464	1337 1591 2036 2271 2492	1346 1600 2039 2273 2513	1374 1614 2074 2277 2535	1383 1623 2120 2279 2556	
BIT	720 1285 1418 1552 1694 1839	755 1290 1430 1564 1710 1844	798 1295 1435 1569 1719 1860	811 1309 1440 1574 1723 1865	815 1314 1456 1587 1734 1870	821 1319 1467 1592 1754 2365	1216 1333 1472 1597 1758 2374	1221 1338 1477 1610 1764 2637	1226 1343 1490 1615 1779 3108	1240 1370 1495 1620 1784 3134	1245 1375 1500 1633 1789 3137	1250 1380 1516 1645 1807 3141	1262 1395 1529 1658 1811 3146	1267 1408 1534 1669 1818 3150	1272 1413 1539 1689 1833 3155	
BLOS	3030	3032	3034	3036	3038											
BMI	1360	1960	1978	1995	2012	2163	2206	2228	2247	2370						
BNE	756 1063 1496 3001	799 1073 1535 3109	805 1084 1570	808 1121 1593	816 1222 1616	818 1246 1690	822 1268 1720	824 1291 1755	916 1315 1785	939 1339 1808	963 1376 1840	966 1409 1866	1000 1414 2366	1034 1436 2636	1043 1473 2652	
BPL	727 2182	926 2185	1111 2293	1117 2302	1914 2340	1917 2351	1944 2583	2049 2626	2057 2629	2085 2991	2093 2997	2099 3016	2137 3112	2145 3117	2168	
BR	750 1224 1433 1713 2165	801 1243 1438 1722 2287	810 1248 1470 1727 2334	826 1265 1475 1737 2368	836 1270 1493 1757 2373	923 1288 1498 1761 2405	932 1293 1532 1787 2428	943 1312 1537 1810 2446	952 1317 1567 1814 2473	1090 1336 1572 1836 2476	1164 1341 1590 1842 2495	1177 1373 1595 1863 2519	1192 1378 1613 1868 2538	1207 1411 1618 2043 2565	1219 1416 1692 2131 2568	
CLR	2607 749 2027 2159 2338	2641 782 2028 2180 2345	3058 784 2047 2202 2382	3081 794 2052 2224 2399	3094 948 2069 2243 2463	3106 960 2070 2261 2491	3158 1086 2083 2262 2578	1912 1942 2088 2263 2603	1942 2094 2264 2631 2077	1956 2112 2291 3024 3017	1974 2113 2297	1991 2114 2315	2008 2115 2316	2025 2135 2317	2026 2140 2318	
CMP	732 3039	807	823	938	949	1005	1885	1920	2077	3017	3029	3031	3033	3035	3037	
CMPB	804	915	918	920												
COM	889	1032	1033	1041	1042	2634										
DEC	817	962	965	999	1062	1072	1083	1120	3000							
DECB	2635	2651														
EMT	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	



C08

.MAIN. MACY11 27(732) 29-SEP-76 14:25 PAGE 96  
DZDCAB.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

ERRORS DETECTED: 0  
DEFAULT GLOBALS GENERATED: 0

\*DZDCAB DZDCAB.SEG/SOL/CRF/PAGNUM/NL:TOC=SYSMAC.SML(400,1066),DZDCAB(400,4571)  
RUN-TIME: 30 42 5 SECONDS  
RUN-TIME RATIO: 413/79=5.2  
CORE USED: 34K (67 PAGES)

EOF1DZDCABSEG                    00010000            770224                    PDP10 411