

DP11

DP11 SYNC INTFC
CZDPAC0

AH-8573C-MC
COPYRIGHT 73-79
FICHE 1 OF 1

SEP 1979
digital
MADE IN USA

This microfiche card contains a grid of frames, likely representing a sequence of technical diagrams or data. The frames are arranged in approximately 12 rows and 6 columns. Each frame contains a small-scale diagram or text, which is too small to read clearly. The diagrams appear to be related to the 'DP11 SYNC INTFC' and 'CZDPAC0' mentioned in the header. The right half of the card is mostly blank, with some faint, illegible markings.

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

.REM %

IDENTIFICATION

PRODUCT CODE: AC-8572C-MC
PRODUCT NAME: CZDPAC0 DP11 SYNC INTFC
DATE: JAN,1979
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 1973,1976,1979 BY DIGITAL EQUIPMENT CORPORATION

58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113

1. ABSTRACT

THIS MAINDEC CONSISTS OF TWO PROGRAMS. FIRSTLY, A DP11A EXERCISER WHICH IS RUN WITH A TEST CONNECTOR (DB25S) IN PLACE OF THE MODEM. THE FUNCTION OF THIS TEST IS TO CHECK OUT THE CABLE (BC01R-25) AND FUNCTIONAL INTERFACE WITH THE MODEM. THIS TEST IS RUN UNDER A SIMULATED (SOFTWARE) CLOCK THAT RUNS AT APPROXIMATELY 54KHZ.

SECONDLY, THERE IS THE PRINCIPLE DIAGNOSTIC FOR THE DP11. THIS PROGRAM RUNS IN THE MAINTENANCE MODE WITH THE BC01R-25 CABLE REMOVED FROM THE SYSTEM UNIT. THIS TEST PROVIDES COMPLETE DIAGNOSTICS FOR THE DP11-DA AND DP11-CA.

2. REQUIREMENTS

2.1 EQUIPMENT

ANY PDP11 FAMILY CPU
DB25S TEST CONNECTOR (IF CABLE TEST IS TO BE RUN)
DP11-DA
DP11-CA (OPTIONAL)

2.2 STORAGE

THIS PROGRAM USES MEMORY TO LOCATION 17500

3. LOADING PROCEDURE

THE PROGRAM MAY BE LOADED LIKE ANY OTHER PROGRAM SUCH AS: PAPER TAPE, DECTAPE, MAGTAPE, CASSETT, DISK, ETC. MOST COMMON WILL BE PAPER TAPE LOADING THROUGH THE USE OF ABSOLUTE LOADER.

3.1 DF11-L TESTING

THIS PROGRAM WILL EXERCISE THE DF11-L.
METHOD: CABLE MUST BE INSERTED INTO DP11, TEST CONNECTOR ON END OF CABLE, SA=210, SET SW00=1, HIT START AND CONT. AS PER 4.3.1.
NOTE: IN THIS TEST AN ERROR WILL CAUSE PRG TO START AT BEGGINING OF TEST

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

SWITCH 7 SET INDICATES SINGLE DP11 MODE
CLEAR INDICATES CYCLE MODE (MORE THAN ONE)

SWITCH 8 SET SELECTS THE DP11-CA OPTION FOR TEST

4.2 STARTING ADDRESSES

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
145
146
147
148
149
150
151
152

200 = START ADDRESS FOR MAINTENANCE MODE DIAGNOSTICS (CABLE
MUST BE PULLED FROM THE SYSTEM UNIT)

210 = START ADDRESS FOR BC01R-25 CABLE TEST (DB25S TEST CONNECTOR
MUST BE PLUGED IN)

4.3 OPERATOR ACTION

4.3.1 IF SWITCH 7 WAS LEFT UP THE PROGRAM WILL HALT WITH THE SWITCH REG IN
IN THE DATA LIGHTS. (EXCLUDING 11/05 AND 11/10 CPU)
AND REQUIRE THE FOLLOWING OPERATOR ACTION:

A) SW0-SW8 MUST BE SET TO THE VECTOR ADDRESS OF THE FIRST
DP11. NOTE FIRST DP11 VECTOR; NOT THE LINE SELECTED
UNLESS IT IS LINE 0

B) SW9-SW15 MUST BE SET TO THE OCTAL LINE NUMBER OF THE DP11
TO BE TESTED. E.G. THE FIRST DP11 IS LINE 0.

PRESS CONTINUE, SR IS IN DATA LIGHTS (EXCLUDING 11/05, 11/10) RESET SWITCH
AS PER 5.1.1 'SWITCH SETTINGS'. PRESS CONTINUE. THIS STARTS PROGRAM.
ALL DP11 ADDRESSES SHALL BE ASSIGNED FROM 774777 TO 774400 (CONTIGUOUSLY

I.E.	1ST DP11	774776	XMIT	BUFFER
		774774	XMIT	STATUS
		774773	SYNC	BUFFER
		774772	RCV	BUFFER
		774770	RCV	STATUS
	2ND	774766		
		774764		
		774762		
		774760		
	32ND	774406		
		774404		
		774402		
		774400		

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
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201

- 4.3.2 FOR CABLE TEST REMOVE BC01R-25 CABLE FROM MODEM AND PLUG THE DB25S TEST CONNECTOR SOCKET INTO THE CABLE.
- 4.3.3 FOR THE MAINTENANCE MODE TEST THE CABLE MUST BE REMOVED FROM THE DEVICE SYSTEM UNIT.
- 5. OPERATING PROCEDURE
 - 5.1.1 SWITCH SETTINGS (APPLICABLE TO BOTH TESTS)
 - SW15 = 1 OR UP ... HALT ON ERROR
 - SW14 = 1 OR UP ... SCOPE LOOP FOR WHOLE CURRENT TEST
 - SW13 = 1 OR UP ... INHIBIT ERROR PRINTOUT
 - SW12 = 1 OR UP ... INHIBIT ALL PRINTOUT, BELL ON ERROR.
 - SW11 = 1 OR UP ... INHIBIT ITERATION
 - SW10 = 1 OR UP ... ESCAPE TO NEXT TEST ON ERROR
 - SW08 = 1 OR UP ... GO TO TOP OF CURRENT TEST ON ERROR.
NOTE: THIS SWITCH IS VERY IMPORTANT FOR DATA ERRORS IN WHICH THE DP11 CLOCK IS RUNNING. THIS SWITCH MUST BE SET TO A 1 TO STOP AN AVALANCH OF ERRORS.....
- 6. ERRORS
 - 6.1 ERROR PRINTOUT
 - PRINTS ALL ERRORS UNLESS INHIBITED BY SWITCH 13 OR SWITCH 12.
 - ERROR PRINT OUT WILL LOOK LIKE:
TEST NO. XXX LINE NO. XX
PC: XXXXXX
 - DEPENDING ON THE ERROR AN ADDITIONAL MESSAGE MAY BE TYPED OUT.
 - 6.1.1 AS STATED ABOVE FOR ERRORS THAT ARE CAUSED BY A COMPARISON SUCH AS DATA COMPARISON, REGISTER COMPARISON, ETC. AND INTERRUPT ERRORS, THERE WILL BE ADDITIONAL INFORMATION IN THE ERROR REPORT.

202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237

- 6.2 ERROR RECOVERY
 - A. IF IN A SCOPE LOOP, SET SWITCH 14.
 - B. TO RECOVER FROM HALT ON ERROR, MAKE SURE SW14 IS SET THEN DEPRESS CONTINUE.
 - C. IT MAY BE DESIRABLE TO SET SW 13 OR SW 12 IN AN ERROR CONDITION.
- 7. RESTRICTIONS
 - 7.1 STARTING RESTRICTIONS
 - SEE 4.1 AND 4.3.1
 - 7.2 OPERATING RESTRICTIONS
 - 7.2.1 CABLE TEST (SA=210) MUST BE RUN WITH DB25S CONNECTOR IN PLACE OF A MODEM.
 - 7.2.2 MAINTENANCE MODE DIAGNOSTIC MUST BE RUN WITH CABLE REMOVED FROM SYSTEM UNIT.
 - 7.2.3 CYCLE MODE. IF SW 07=0 WHEN START IS PRESSED; PROGRAM WILL RUN SELECTED TEST (CABLE OR MAINTENANCE) ASSUMING THAT ALL DP11S IN THE SYSTEM ARE TO BE TESTED. PROGRAM STARTS WITH LINE 0 THEN SENSES FOR NEXT LINE; IF IT EXISTS IT WILL BE TESTED. THIS GOES ON UNTILL A NON-EXISTANT LINE IS FOUND THEN THE PROGRAM STARTS WITH LINE 0 AGAIN.
BASCSR AND BASVEC ARE DEFAULT VALUES. IF YOUR VALUES ARE DIFFERENT PATCH YOUR VALUES IN THERE.
- 8. MISCELLANEOUS

238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277

9. PROGRAM DESCRIPTION

9.1 INITIALIZATION ROUTINES

THE START CODE FOR BOTH THE MAINTENANCE MODE AND THE CABLE TEST INITIALIZES THE PROCESSOR PRIORITY TO LEVEL SEVEN AND SETS THE STACK POINTER TO ADDRESS 1100. CONSOL SWITCH SEVEN IS THEN EXAMINED TO DETERMINE IF THIS IS CYCLE MODE OR SINGLE LINE IS TEST. IF SWITCH SEVEN IS UP TWO SUBROUTINES (CLRVEC,LINE.N) ARE EXECUTED BEFORE THE TEST SECTION IS ENTERED.

9.1.1 CLRVEC, CLEAR-VECTOR-AREA

THE SUBROUTINE 'CLEAR-VECTOR-AREA' LOADS THE COMMUNICATION VECTOR AREA WITH .+2,HALT. THIS CAUSES ANY ILLEGAL INTERRUPTS TO TRAP TO THERE STATUS WORD.

9.1.2 LINE.N, LINE NUMBER

THE FUNCTION OF THIS SUBROUTINE IS TO SAVE SWITCH EIGHT OF THE CONSOL (SW8 SELECTS DP11-CA OPTION) AND WAIT FOR OPERATOR ACTION TO SPECIFY THE LINE NUMBER AND FIRST DP11 VECTOR ADDRESS. WHEN THE PROGRAM HALTS;
SWITCHES SW0 THRU SW8 MUST BE SET TO THE VECTOR ADDRESS OF THE FIRST DP11 AND SWITCHES SW9 THRU SW15 MUST BE SET TO THE OCTAL EQUIVILANT OF THE LINE NUMBER (E.G. THE FIRST LINE IS LINE 0(8) THE TENTH LINE IS LINE 11(8)). FOLLOWING THIS ACTION "CONTINUE" ENTERS THE PROGRAM INTO THE SELECTED TEST SECTION. IF SWITCH SEVEN IS NOT UP WHEN "START" IS DEPRESSED THE PROGRAM ASSUMES CYCLE MODE AND WILL START RUNNING WITH LINE 0 THRU ALL LINES. BASCSR AND BASVEC ARE USED AS DEFALT CONDITIONS.

9.2 MAINTENANCE MODE TESTS

IN AN EFFORT TO OPTIMIZE CORE UTILIZATION MANY OF THE DIAGONOSTIC TEST WERE WRITTEN IN SUBROUTINE FORMAT VERSUS MACROS.

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

9.2.1 BITST, BIT TEST

THIS SUBROUTINE IS ENTERED WITH A JSR R5, BITST. IMMEDIATELY FOLLOWING THIS INSTRUCTION IS THE BIT SELECTED FOR TEST. THE BIT NUMBER IS FETCHED BY THE SUBROUTINE AND STORED IN LOCATION 'BITS'. USING A SEQUENCE OF 'BIS' 'BIT' AND 'BIC' INSTRUCTIONS EACH READ/WRITE BIT OF THE TRANSMITTER AND RECEIVER CSR (TCSR,RCSR) IS TESTED TO VERIFY THAT AT LEAST THAT PARTICULAR BIT CAN BE REFERENCED AND IS IN FACT READ/WRITE. NO ATTEMPT IS MADE AT THIS POINT TO CHECK FOR ILLEGITIMATE INTERACTION.

9.2.2 RESET TEST

THIS IS A SIMPLE TEST THAT MERELY WRITES INTO ALL WRITEABLE BITS OF THE TCSR AND RCSR, CHECKS THAT THEY WERE SET, ISSUES 'RESET' AND CHECKS THAT ALL BITS THAT ARE SUPPOSED TO BE CLEARED BY RESET WERE.

9.2.3 VALID

THE FUNCTION OF THIS SUBROUTINE IS TO TEST FOR INTERACTION BETWEEN READ/WRITE BITS OF THE TCSR AND RCSR. THIS ROUTINE IS ENTERED WITH A JSR REGISTER FIVE, FOLLOWED BY THE BIT NUMBER. THE SELECTED BIT IS SET AND THEN THE ENTIRE CSR IS COMPARED WITH THE WORD (BITS) USED TO SET THE SELECTED BIT. IF ANY OTHER BIT IS SET AN ERROR IS REPORTED. LOCATION 'REG' CONTAINS THE ADDRESS OF THE CSR SELECTED FOR TEST. AN EXAMINATION OF THIS CSR SHOULD REVEAL A BIT SET OTHER THAN THE ONE IN LOCATION 'BITS'.

9.2.4 CLEAR

THIS SUBROUTINE IS ENTERED THE SAME WAY AS BITST, AND VALID ARE ENTERED. ITS FUNCTION IS TO TEST FOR INTERACTION BETWEEN ANY CSR BITS DURING A BIT CLEAR INSTRUCTION. THIS IS ACCOMPLISHED BY SETTING ALL READ/WRITE BITS OF THE SELECTED CSR AND MAKING A DUPLICATE BIT MAP IN TMPDAT. THEN 'BITS' IS USED TO CLEAR A SINGLE BIT IN THE CSR AND TMPDAT. FOLLOWING THIS THE CSR IS COMPARED WITH TMPDAT TO VERIFY THAT ONLY THAT BIT WAS CLEARED.

9.2.5 PRIORITY TESTS

WITH THE PROCESSOR PRIORITY AT LEVEL FIVE 'STATUS-INTERRUPT-ENABLE' (SIE) IS SET AND ALL THE BITS THAT SHOULD CAUSE A STATUS INTERRUPT ARE SET INDIVIDUALLY AND COLLECTIVELY SET. SECONDLY SIE IS REMOVED AND THE PROCESSOR PRIORITY IS LOWERED TO FOUR. AGAIN THE CSR BITS THAT CAUSE 'STATUS INTERRUPTS' ARE SET AND RESET. FINALLY THE SIE BIT IS SET WHILE THE PROCESSOR PRIORITY IS AT FOUR AND IT IS VERIFIED THAT EACH DISCRETE EVENT THAT SHOULD CAUSE A STATUS INTERRUPT DOES. THIS SEQUENCE TESTS THAT THE DP11 STATUS BITS INTERRUPT AT THE PROPER PROCESSOR PRIORITY. THE NEXT SEQUENCE OF PRIORITY TESTS VERIFY THE TRANSMITTER INTERRUPTS BY LOADING THE

334
335
336

TRANSMIT BUFFER, ENABLING THE MAINTENANCE MODE CLOCK AND
WAITING FOR AN INTERRUPT. IF NO INTERRUPT IS RECEIVED WITHIN
10 CHARACTER TIMES AN ERROR IS REPORTED.

337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377

9.2.6 SYNCHRONIZATION TESTS

THE FIRST SYNC TEST (TEST 24) VERIFIES THE READ/WRITE CAPABILITY OF THE SYNC REGISTER 174XX3 AND THE SYNC EXTENSION 174XX7 WHEN THE DP11-CA OPTION EXISTS. THIS IS ACCOMPLISHED BY WRITING AND READING ALL POSSIBLE SYNC CHARACTERS (0 THRU 377(8) FOR THE SYNC REG AND 0 THRU 17(8) FOR THE SYNC EXTENSION).

THE NEXT SYNC TEST ISSUES TWO OF EVERY POSSIBLE SYNC CHARACTER IN EACH OF THE AVAILABLE CHARACTER LENGTH AND CHECKS THAT TWO OF EACH SYNC RAISES 'RECEIVER ACTIVE'. THEN THE THIRD SYNC IS TRANSMITTED AS DATA. THIS CHECKS THE CAPABILITY OF THE RECEIVER TO INTERRUPT AND ALSO CHECKS THE RECEIVER BUFFER FOR DATA RECEPTION ACCURACY. THIS TEST IS FIRST RUN BY LOADING THE TRANSMIT BUFFER UNDER SOFTWARE CONTROL THEN IS REPEATED IN THE IDLE MODE. THIS CHECKS THAT EACH AND EVERY POSSIBLE SYNC CHARACTER CAN BE TRANSMITTED IN THE IDLE MODE IN THE EVENT THAT AN ERROR IS DETECTED IN THE LAST TWO SYNC TEST AND THE 'HALT-ON-ERROR' SWITCH IS UP A SCOPE LOOP MAY BE RUN. THIS IS ACCOMPLISHED BY REMOVING 'HALT-ON-ERROR' SW15, SETTING 'SCOPE', INHIBIT PRINT, SET SW09, AND PRESSING CONTINUE. THIS CAUSES INCREMENT INSTRUCTION TO BE SKIPPED AND THEREFORE LOOP ON THE SAME SYNC CHARACTER.

9.2.7 INTERRUPT DRIVEN SEQUENTIAL DATA TEST

SYNC IS ESTABLISHED THROUGH THE TRANSMISSION TWO SYNC CHARACTERS. ONCE SYNC IS ESTABLISHED A BINARY COUNT PATTERN IS TRANSMITTED THE SIZE OF WHICH IS DETERMINED BY THE MAXIMUM CHARACTER SELECTED FOR TEST (8 BITS/CHARACTER OR 12/8 BITS/CHARACTER IF THE CA OPTION EXISTS). AT THE COMPLETION OF THE BINARY COUNT PATTERN 'ACTIVE' IS DROPPED AND THE NEXT SHORTEST CHARACTER LENGTH IS SELECTED. THIS TEST IS REPEATED FOR THREE CHARACTER LENGTHS (12,11,10,OR 8,7,6).

FUNCTIONALLY THIS TEST VERIFIES THE CAPABILITY OF THE DP11 TO MAINTAIN SYNC OVER A LONG CHARACTER STRING.

IN THE EVENT THAT AN ERROR IS DETECTED AND 'HALT-ON-ERROR' IS UP REMOVE IT, SET 'INHIBIT PRINT' AND 'SCOPE' AND PRESS CONTINUE.

378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431

9.2.8 RANDOM DATA, RANDOM STALL

THIS INTERRUPT DRIVEN TEST TRANSMITS RANDOM DATA FOR A PERIOD OF TIME (0 TO 0.65 SECONDS, 0 TO 260 CHARACTERS) DETERMINED BY A RANDOM GENERATOR. AT THE EXPIRATION OF THE DATA TIME INTERVAL THE "IDLE" MODE IS ENTERED AND SYNC CHARACTERS ARE TRANSMITTED FOR A RANDOM PERIOD OF TIME. WHEN THE IDLE TIME TERMINATES THE DATA MODE IS RESTARTED AND NEW DATA IS TRANSMITTED FOR A NEW TIME INTERVAL. THIS VERIFIES THAT THE DP11 CAN SWITCH BETWEEN DATA MODE AND IDLE AT RANDOM.

9.2.9 PARITY TEST

THE PARITY TEST CONSISTS OF A TRANSMITTER INTERRUPT SERVICE ROUTINE THAT TRANSMITS A BINARY COUNT PATTERN AND A RECEIVER INTERRUPT SERVICE ROUTINE THAT CALCULATES THE PARITY ON THE EXPECTED DATA, COMPARES THE RECEIVED DATA WITH THE EXPECTED DATA, AND FINALLY TESTS THE PARITY BIT (BIT 12=0 FOR EVEN, 1 FOR ODD).

9.2.10 RECEIVER OVERRUN TEST

THIS TEST TRANSMITS TWO SYNC CHARACTERS TO RAISE "ACTIVE" FOLLOWED BY TWO DATA CHARACTERS. RECEIVER INTERRUPT ENABLE IS NOT SET THEREFORE "RECEIVER OVERRUN" SHOULD SET AND CAUSE A TRANSMITTER STATUS INTERRUPT. THIS SEQUENCE IS REPEATED FOR A FULL BINARY COUNT. (000-377)

9.2.11 HALF DUPLEX TEST

THE HALF DUPLEX BIT SHOULD PREVENT ANY DATA FROM ENTERING THE RECEIVER WHILE SEND-REQUEST IS UP. TO VERIFY THIS RECEIVER INTERRUPT ENABLE IS SET WHILE THE TRANSMITTER IDLES FOR APPROXIMATELY 30MS. FOR EACH POSSIBLE CHARACTER AVAILABLE IN THE 8 BIT/CHAR SET. ANY DATA ENTRY INTO THE RECEIVER WILL CAUSE A TRAP TO AN ERROR ROUTINE.

9.3 CABLE TEST

THE CABLE TEST REQUIRES THE LEAST AMOUNT OF EFFORT AND THEREFORE CAN BE RUN AS A QUICK CONFIDENCE CHECK. THE OPERATING PROCEDURE IS TO DISCONNECT THE BC01R-25 CABLE FROM THE MODEM AND PLUG IT INTO THE DB25S TEST CONNECTOR. FROM THIS POINT ON THE OPERATING PROCEDURE IS THE SAME AS THE MAINTENANCE MODE DIAGNOSTIC. THE PRINCIPLE DIFFERENCE BETWEEN THE CABLE TEST AND THE MAINTENANCE MODE TEST IS THE CLOCK. THE MAINTENANCE MODE TEST RUNS OFF OF A FREE RUNNING 3KHZ MULTI-VIBRATOR WHERE AS THE CABLE TEST OPERATES OFF A SOFTWARE CLOCK. SETTING BIT 3 OF THE TRANSMITTER STATUS RAISES THE CLOCK, CLEARING IT LOWERS THE CLOCK. THE SOFTWARE CLOCK THEREFORE HAS A FREQUENCY RANGE OF ZERO TO 56KHZ. THIS ENABLES THE PROGRAM TO STEP THROUGH THE TRANSMIT-RECEIVE SEQUENCE ONE BIT AT A TIME. IT ALSO VERIFIES THE 10KHZ CABLE SPEC AND 50 KHZ LOGIC SPEC.

432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483

9.3.2 CLOCK

CLOCK IS THE SUBROUTINE TO RUN THE SOFTWARE CLOCK. IT IS ENTERED BY A TRAP CALL, CLOCK FOLLOWED BY THE NUMBER OF CYCLES DESIRED. UPON ENTRY THE SUBROUTINE FETCHES THE CYCLE COUNT AND EXAMINE BIT8 OF SAVSR1 TO DETERMINE IF 8 BITS/CHAR OR 12 BITS/CHAR HAVE BEEN SELECTED FOR TEST. IF THE 12 BIT MODE HAS BEEN SELECTED 4 IS ADDED TO THE CLOCK COUNT AND BIT 10 OF THE RECEIVER STATUS IS SET BEFORE EXECUTING THE CLOCKING INSTRUCTIONS. BY CHANGING LOCATION 'FREQ' IT IS POSSIBLE TO SLOW DOWN THE CLOCK. WITH FREQ 1 THE SOFTWARE CLOCK RUNS AT APPROXIMATELY 25 KHZ. THIS ENABLES THE OPERATOR TO SLOW THE CLOCK DOWN TO ALMOST ZERO CPS. THIS CAN BE USEFULL IN DETERMING IF A BUG IF FREQUENCY DEPENDENT.

9.3.3 RXCLK

THIS TRAP CALL IS ANOTHER SOFTWARE CLOCK. IT WAS WRITTEN FOR CODE THAT IS INDEPENDENT OF 12/8 BITS PER CHARACTER OPTION. RXCLK N EXECUTES N SOFRWARE CYCLES. RXCLK ALSO HAS 'FREQ' EMBEDDED WITHIN ITS DEFINITION. SINCE IT IS A TRAP CALL THE DELAY INSTRUCTIONS CAN BE CHANGED BY CHANGING THE CONTENTS OF 'FREQ'.

9.3.4 REE

REE IS A UTILITY SUBROUTINE TO REINITIALIZE THE DP11 STATUS REGISTER, INTERRUPT VECTOR AND SELECT THE 12/8 BITS PER CHARACTER MODE. THE ENTRY REGISTER IS R5. THE ADDRESS TO WHICH THIS SUBROUTINE RETURNS IS A FUNCTION OF THE NUMBER OF BITS PER CHARACTER SELECTED FOR TEST. IF 8 BITS PER CHARACTER IS SELECTED THE SUBROUTINE RETURNS TO AN INSTRUCTION THAT SETS UP THE DATA LIMIT FOR THAT MODE. IF THE TWELVE BIT PER CHARACTER MODE IS SELECTED THE CONTENTS OF REGISTER 5 IS MODIFIED AND SUBROUTINE RETURNS TO THE INSTRUCTION THAT SETS UP THE TWELVE BIT LIMIT.

9.3.5 SYNCHRONIZATION CHARACTER TESTS

FOLLOWING STATUS REGISTER AND VECTOR INITIALIZATION THE CLOCK IS RUN FOR 30 CYCLES TO CLEAR OUT ANY PREVIOUS DATA THAT MAY BE RESIDING IN THE TRANSMIT OR RECEIVE BUFFERS. AT THIS POINT THE 'TRANSMITTER DONE' AND 'TRANSMITTER INTERRUPT ENABLE' ARE SET CAUSING AN INTERRUPT TO A SYNCHRONIZATION SUBROUTINE, TV18. TV18 LOADS THE TRANSMIT BUFFER WITH A SYNC CHARACTER. UPON RETURN FROM THE INTERRUPT SERVICE ROUTINE THE SOFTWARE CLOCK RUNS FOR 3 CYCLES. THIS SHOULD BR SUFFICIENT TO RAISE 'SEND REQUEST'; IF NOT AN ERROR IS REPORTED. THE SOFTWARE CLOCK THEN GENERATOR ENOUGH CYCLES TO TRANSMIT EXACTLY ONE CHARACTER AND EXAMINES 'RECEIVE ACTIVE'. IF 'RECEIVER ACTIVE' IS UP THE RECEIVER IS PREMATURELY ACTIVE AND AN ERROR IS REPORTED. THE NEXT SET OF CYCLES GENERATED IS ONE SHORT OF THE NUMBER REQUIRED TO TRANSMIT A FULL CHARACTER.

484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522

AGAIN 'RECEIVER ACTIVE' IS TESTED FOR A PREMATURE RESPONSE.

ONE MORE CYCLE IS THEN GENERATED AND 'ACTIVE' SHOULD BE UP. TO VERIFY THAT IT WAS THE TRANSMITTED DATA THAT RAISED 'ACTIVE' AND NOT NOISE A THIRD SYNC IS TRANSMITTED AND CHECKED AS DATA. THE SEQUENCE IS CONTINUED FOR COMPLETE SET OF POSSIBLE SYNC CHARACTERS (1 TO LIMIT), IN EACH OF THE THREE AVAILABLE CHARACTER LENGTHS. THIS SET OF TESTS ALSO HAS THE SCOPE LOOP FACILITY WHERE THE SAME SYNC CHARACTER IS NOT CHANGED IF THE SCOPE SWITCH SW14 IS UP.

9.3.6 SEQUENTIAL DATA

THIS TEST IS THE SAME AS THAT RUN IN THE MAINTENANCE MODE TEST. IT'S PRINCIPLE OBJECTIVE IS TO VERIFY THE 50KHZ SPEC OF THE DP11. THE SOFTWARE CLOCK THAT DRIVES THIS TEST OPERATES AT APPROXIMATELY 56KHZ.

9.3.7 RANDOM DATA, RANDOM IDLE

THIS TEST IS SIMILAR TO THE RANDOM DATA TEST RUN IN THE MAINTENANCE MODE. THE DIFFERENCE IS THAT IN GOING FROM 'IDLE' TO 'DATA', 'ACTIVE' IS DROPPED AND THE RECEIVER RESYNCD.

9.3.8 SEND-REQUEST TEST

IN THIS TEST 'SEND REQUEST' IS RAISED BY LOADING THE TRANSMIT BUFFER AND GENERATING 3 CYCLES. IF THE CABLE IS WIRED PROPERLY 'SEND REQUEST' SHOULD RAISE 'CLEAR-TO-SEND' AND 'MODEM READY' (DATA SET READY). THIS TEST IS REPEATED IN THE 8, 7, AND 6 BITS PER CHARACTER MODE.

9.3.9 TERMINAL READY

THE FUNCTION OF THIS TEST IS TO VERIFY THAT WITH THE DB25S TEST CONNECTOR IN PLACE, SETTING 'TERMINAL READY' RAISES 'CARRIER' AND 'RING FLAG'. THIS TEST ALSO VERIFIES 'RING FLAG' AND 'CARRIER DOWN FLAG' INTERRUPT.

.ENDR %

```
523
524      ;DP11A SYNCHRONOUS MODEM INTERFACE DIAGNOSTIC
525      ;*****MAINTENANCE MODE*****
526      ;COPYRIGHT, DIGITAL EQUIPMENT CORPORATION*****
527      ;MAYNARD, MASSACHUSETTS 01754
528      ;PROGRAMMER: JOHN FRIEDRICH
529      ;REVISED BY:
530      ;
531      ;           REVISION B CREATED BY JOHN EGOLF
532      ;           REVISION C CREATED BY ELLIOT GERBERG
533      ;           ( ALL CODE COMMENTED ;+ IS FROM REV. C )
534
535
536
537      ;ZZ-CZDPA-C
538
539
540      ;DP11
541      ;COPYRIGHT JUNE 1973,1979, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
542
543      ;STARTING PROCEDURE
544      ;LOAD PROGRAM
545      ;LOAD ADDRESS 000200
546      ;PRESS START
547      ;THIS IS FOR THE STATIC TEST
548      ;LOAD ADDRESS 000210
549      ;PRESS START
550      ;THIS IS FOR CABLE TEST
551      ;FOR MORE INFORMATION SEE THE
552      ;DOCUMENTATION IN FRONT OF THE LISTING
553
554
555
556      ;SWITCH REGISTER OPTIONS
557
558      100000      SW15=100000      :=1,HALT ON ERROR
559      040000      SW14=40000      :=1,LOOP ON CURRENT TEST
560      020000      SW13=20000      :=1,INHIBIT ERROR TYPEOUT
561      010000      SW12=10000      :=1,DELETE TYPEOUT/BELL ON ERROR.
562      004000      SW11=4000       :=1,INHIBIT ITERATIONS
563      002000      SW10=2000       :=1,ESCAPE TO NEXT TEST ON ERROR
564      001000      SW09=1000       :=1,LOOP WITH CURRENT DATA
565      000400      SW08=400        :=1,LOOP ON ERROR
566      000200      SW07=200        :=1,SINGLE SELECTED DP11. =0 CYCLE ALL DP11S
567      000100      SW06=100
568      000040      SW05=40
569      000020      SW04=20
570      000010      SW03=10
571      000004      SW02=4
572      000002      SW01=2
573      000001      SW00=1
```

```
574
575
576
577
578      000000      R0=%0      ;GENERAL REGISTER
579      000001      R1=%1      ;GENERAL REGISTER
580      000002      R2=%2      ;GENERAL REGISTER
581      000003      R3=%3      ;GENERAL REGISTER
582      000004      R4=%4      ;GENERAL REGISTER
583      000005      R5=%5      ;GENERAL REGISTER
584      000006      SP=%6      ;PROCESSOR STACK POINTER
585      000007      PC=%7      ;PROGRAM COUNTER
586
587
588
589      177570      SWR=177570 ;CONSOLE SWITCH REGISTER
590      177570      LIGHTS=177570 ;PDP-11/45 DISPLAY REGISTER
591      177776      PS=177776 ;PROCESSOR STATUS WORD
592      001050      STACK=1050 ;START OF PROCESSOR STACK
593
594
595
596      005746      PUSH1SP=5746 ;DECREMENT PROCESSOR STACK 1 WORD
597      005726      POP1SP=5726 ;INCREMENT PROCESSOR STACK 1 WORD
598      010046      PUSHRO=10046 ;SAVE R0 ON STACK
599      012600      POPRO=12600 ;RESTORE R0 FROM STACK
600      024646      PUSH.SP=24646 ;DECREMENT STACK TWICE
601      022626      POP.SP=22626 ;INCREMENT STACK TWICE
602
603
604
605      100000      BIT15=100000
606      040000      BIT14=40000
607      020000      BIT13=20000
608      010000      BIT12=10000
609      004000      BIT11=4000
610      002000      BIT10=2000
611      001000      BIT9=1000
612      000400      BIT8=400
613      000200      BIT7=200
614      000100      BIT6=100
615      000040      BIT5=40
616      000020      BIT4=20
617      000010      BIT3=10
618      000004      BIT2=4
619      000002      BIT1=2
620      000001      BIT0=1
621
622
```

```
623 ;TRAPCATCAER FOR ILLEGAL INTERRUPTS
624 000000 000000 . =0 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
625 000000 000002 HALT ;EXAMINE STACK TO FIND CAUSE
626 000002 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
627 000004 000006 HALT ;EXAMINE STACK TO FIND CAUSE
628 000006 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
629 000010 000012 HALT ;EXAMINE STACK TO FIND CAUSE
630 000012 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
631 000014 000016 HALT ;EXAMINE STACK TO FIND CAUSE
632 000016 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
633 000020 000022 HALT ;EXAMINE STACK TO FIND CAUSE
634 000022 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
635 000024 000026 HALT ;EXAMINE STACK TO FIND CAUSE
636 000026 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
637 000030 000032 HALT ;EXAMINE STACK TO FIND CAUSE
638 000032 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
639 000034 000036 HALT ;EXAMINE STACK TO FIND CAUSE
640 000036 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
641 000040 000042 HALT ;EXAMINE STACK TO FIND CAUSE
642 000042 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
643 000044 000046 HALT ;EXAMINE STACK TO FIND CAUSE
644 000046 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
645 000050 000052 HALT ;EXAMINE STACK TO FIND CAUSE
646 000052 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
647 000054 000056 HALT ;EXAMINE STACK TO FIND CAUSE
648 000056 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
649 000060 000062 HALT ;EXAMINE STACK TO FIND CAUSE
650 000062 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
651 000064 000066 HALT ;EXAMINE STACK TO FIND CAUSE
652 000066 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
653 000070 000072 HALT ;EXAMINE STACK TO FIND CAUSE
654 000072 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
655 000074 000076 HALT ;EXAMINE STACK TO FIND CAUSE
656 000076 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
657 000100 000102 HALT ;EXAMINE STACK TO FIND CAUSE
658 000102 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
659 000104 000106 HALT ;EXAMINE STACK TO FIND CAUSE
660 000106 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
661 000110 000112 HALT ;EXAMINE STACK TO FIND CAUSE
662 000112 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
663 000114 000116 HALT ;EXAMINE STACK TO FIND CAUSE
664 000116 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
665 000120 000122 HALT ;EXAMINE STACK TO FIND CAUSE
666 000122 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
667 000124 000126 HALT ;EXAMINE STACK TO FIND CAUSE
668 000126 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
669 000130 000132 HALT ;EXAMINE STACK TO FIND CAUSE
670 000132 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
671 000134 000136 HALT ;EXAMINE STACK TO FIND CAUSE
672 000136 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
673 000140 000142 HALT ;EXAMINE STACK TO FIND CAUSE
674 000142 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
675 000144 000146 HALT ;EXAMINE STACK TO FIND CAUSE
676 000146 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
677 000150 000152 HALT ;EXAMINE STACK TO FIND CAUSE
678 000152 000000 .+2 ;UNEXPECTED TRAP TO THIS LOCATION
```


679	000154	000156	.+2	:UNEXPECTED TRAP TO THIS LOCATION
680	000156	000000	HALT	:EXAMINE STACK TO FIND CAUSE
681	000160	000162	.+2	:UNEXPECTED TRAP TO THIS LOCATION
682	000162	000000	HALT	:EXAMINE STACK TO FIND CAUSE
683	000164	000166	.+2	:UNEXPECTED TRAP TO THIS LOCATION
684	000166	000000	HALT	:EXAMINE STACK TO FIND CAUSE
685	000170	000172	.+2	:UNEXPECTED TRAP TO THIS LOCATION
686	000172	000000	HALT	:EXAMINE STACK TO FIND CAUSE
687	000174	000176	.+2	:UNEXPECTED TRAP TO THIS LOCATION
688	000176	000000	HALT	:EXAMINE STACK TO FIND CAUSE
689	000200	000202	.+2	:UNEXPECTED TRAP TO THIS LOCATION
690	000202	000000	HALT	:EXAMINE STACK TO FIND CAUSE
691	000204	000206	.+2	:UNEXPECTED TRAP TO THIS LOCATION
692	000206	000000	HALT	:EXAMINE STACK TO FIND CAUSE
693	000210	000212	.+2	:UNEXPECTED TRAP TO THIS LOCATION
694	000212	000000	HALT	:EXAMINE STACK TO FIND CAUSE
695	000214	000216	.+2	:UNEXPECTED TRAP TO THIS LOCATION
696	000216	000000	HALT	:EXAMINE STACK TO FIND CAUSE
697	000220	000222	.+2	:UNEXPECTED TRAP TO THIS LOCATION
698	000222	000000	HALT	:EXAMINE STACK TO FIND CAUSE
699	000224	000226	.+2	:UNEXPECTED TRAP TO THIS LOCATION
700	000226	000000	HALT	:EXAMINE STACK TO FIND CAUSE
701	000230	000232	.+2	:UNEXPECTED TRAP TO THIS LOCATION
702	000232	000000	HALT	:EXAMINE STACK TO FIND CAUSE
703	000234	000236	.+2	:UNEXPECTED TRAP TO THIS LOCATION
704	000236	000000	HALT	:EXAMINE STACK TO FIND CAUSE
705	000240	000242	.+2	:UNEXPECTED TRAP TO THIS LOCATION
706	000242	000000	HALT	:EXAMINE STACK TO FIND CAUSE
707	000244	000246	.+2	:UNEXPECTED TRAP TO THIS LOCATION
708	000246	000000	HALT	:EXAMINE STACK TO FIND CAUSE
709	000250	000252	.+2	:UNEXPECTED TRAP TO THIS LOCATION
710	000252	000000	HALT	:EXAMINE STACK TO FIND CAUSE
711	000254	000256	.+2	:UNEXPECTED TRAP TO THIS LOCATION
712	000256	000000	HALT	:EXAMINE STACK TO FIND CAUSE
713	000260	000262	.+2	:UNEXPECTED TRAP TO THIS LOCATION
714	000262	000000	HALT	:EXAMINE STACK TO FIND CAUSE
715	000264	000266	.+2	:UNEXPECTED TRAP TO THIS LOCATION
716	000266	000000	HALT	:EXAMINE STACK TO FIND CAUSE
717	000270	000272	.+2	:UNEXPECTED TRAP TO THIS LOCATION
718	000272	000000	HALT	:EXAMINE STACK TO FIND CAUSE
719	000274	000276	.+2	:UNEXPECTED TRAP TO THIS LOCATION
720	000276	000000	HALT	:EXAMINE STACK TO FIND CAUSE
721	000300	000302	.+2	:UNEXPECTED TRAP TO THIS LOCATION
722	000302	000000	HALT	:EXAMINE STACK TO FIND CAUSE
723	000304	000306	.+2	:UNEXPECTED TRAP TO THIS LOCATION
724	000306	000000	HALT	:EXAMINE STACK TO FIND CAUSE
725	000310	000312	.+2	:UNEXPECTED TRAP TO THIS LOCATION
726	000312	000000	HALT	:EXAMINE STACK TO FIND CAUSE
727	000314	000316	.+2	:UNEXPECTED TRAP TO THIS LOCATION
728	000316	000000	HALT	:EXAMINE STACK TO FIND CAUSE
729	000320	000322	.+2	:UNEXPECTED TRAP TO THIS LOCATION
730	000322	000000	HALT	:EXAMINE STACK TO FIND CAUSE
731	000324	000326	.+2	:UNEXPECTED TRAP TO THIS LOCATION
732	000326	000000	HALT	:EXAMINE STACK TO FIND CAUSE
733	000330	000332	.+2	:UNEXPECTED TRAP TO THIS LOCATION
734	000332	000000	HALT	:EXAMINE STACK TO FIND CAUSE

735	000334	000336	.+2	:UNEXPECTED TRAP TO THIS LOCATION
736	000336	000000	HALT	:EXAMINE STACK TO FIND CAUSE
737	000340	000342	.+2	:UNEXPECTED TRAP TO THIS LOCATION
738	000342	000000	HALT	:EXAMINE STACK TO FIND CAUSE
739	000344	000346	.+2	:UNEXPECTED TRAP TO THIS LOCATION
740	000346	000000	HALT	:EXAMINE STACK TO FIND CAUSE
741	000350	000352	.+2	:UNEXPECTED TRAP TO THIS LOCATION
742	000352	000000	HALT	:EXAMINE STACK TO FIND CAUSE
743	000354	000356	.+2	:UNEXPECTED TRAP TO THIS LOCATION
744	000356	000000	HALT	:EXAMINE STACK TO FIND CAUSE
745	000360	000362	.+2	:UNEXPECTED TRAP TO THIS LOCATION
746	000362	000000	HALT	:EXAMINE STACK TO FIND CAUSE
747	000364	000366	.+2	:UNEXPECTED TRAP TO THIS LOCATION
748	000366	000000	HALT	:EXAMINE STACK TO FIND CAUSE
749	000370	000372	.+2	:UNEXPECTED TRAP TO THIS LOCATION
750	000372	000000	HALT	:EXAMINE STACK TO FIND CAUSE
751	000374	000376	.+2	:UNEXPECTED TRAP TO THIS LOCATION
752	000376	000000	HALT	:EXAMINE STACK TO FIND CAUSE
753	000400	000402	.+2	:UNEXPECTED TRAP TO THIS LOCATION
754	000402	000000	HALT	:EXAMINE STACK TO FIND CAUSE
755	000404	000406	.+2	:UNEXPECTED TRAP TO THIS LOCATION
756	000406	000000	HALT	:EXAMINE STACK TO FIND CAUSE
757	000410	000412	.+2	:UNEXPECTED TRAP TO THIS LOCATION
758	000412	000000	HALT	:EXAMINE STACK TO FIND CAUSE
759	000414	000416	.+2	:UNEXPECTED TRAP TO THIS LOCATION
760	000416	000000	HALT	:EXAMINE STACK TO FIND CAUSE
761	000420	000422	.+2	:UNEXPECTED TRAP TO THIS LOCATION
762	000422	000000	HALT	:EXAMINE STACK TO FIND CAUSE
763	000424	000426	.+2	:UNEXPECTED TRAP TO THIS LOCATION
764	000426	000000	HALT	:EXAMINE STACK TO FIND CAUSE
765	000430	000432	.+2	:UNEXPECTED TRAP TO THIS LOCATION
766	000432	000000	HALT	:EXAMINE STACK TO FIND CAUSE
767	000434	000436	.+2	:UNEXPECTED TRAP TO THIS LOCATION
768	000436	000000	HALT	:EXAMINE STACK TO FIND CAUSE
769	000440	000442	.+2	:UNEXPECTED TRAP TO THIS LOCATION
770	000442	000000	HALT	:EXAMINE STACK TO FIND CAUSE
771	000444	000446	.+2	:UNEXPECTED TRAP TO THIS LOCATION
772	000446	000000	HALT	:EXAMINE STACK TO FIND CAUSE
773	000450	000452	.+2	:UNEXPECTED TRAP TO THIS LOCATION
774	000452	000000	HALT	:EXAMINE STACK TO FIND CAUSE
775	000454	000456	.+2	:UNEXPECTED TRAP TO THIS LOCATION
776	000456	000000	HALT	:EXAMINE STACK TO FIND CAUSE
777	000460	000462	.+2	:UNEXPECTED TRAP TO THIS LOCATION
778	000462	000000	HALT	:EXAMINE STACK TO FIND CAUSE
779	000464	000466	.+2	:UNEXPECTED TRAP TO THIS LOCATION
780	000466	000000	HALT	:EXAMINE STACK TO FIND CAUSE
781	000470	000472	.+2	:UNEXPECTED TRAP TO THIS LOCATION
782	000472	000000	HALT	:EXAMINE STACK TO FIND CAUSE
783	000474	000476	.+2	:UNEXPECTED TRAP TO THIS LOCATION
784	000476	000000	HALT	:EXAMINE STACK TO FIND CAUSE
785	000500	000502	.+2	:UNEXPECTED TRAP TO THIS LOCATION
786	000502	000000	HALT	:EXAMINE STACK TO FIND CAUSE
787	000504	000506	.+2	:UNEXPECTED TRAP TO THIS LOCATION
788	000506	000000	HALT	:EXAMINE STACK TO FIND CAUSE
789	000510	000512	.+2	:UNEXPECTED TRAP TO THIS LOCATION
790	000512	000000	HALT	:EXAMINE STACK TO FIND CAUSE

791	000514	000516	.+2	:UNEXPECTED TRAP TO THIS LOCATION
792	000516	000000	HALT	:EXAMINE STACK TO FIND CAUSE
793	000520	000522	.+2	:UNEXPECTED TRAP TO THIS LOCATION
794	000522	000000	HALT	:EXAMINE STACK TO FIND CAUSE
795	000524	000526	.+2	:UNEXPECTED TRAP TO THIS LOCATION
796	000526	000000	HALT	:EXAMINE STACK TO FIND CAUSE
797	000530	000532	.+2	:UNEXPECTED TRAP TO THIS LOCATION
798	000532	000000	HALT	:EXAMINE STACK TO FIND CAUSE
799	000534	000536	.+2	:UNEXPECTED TRAP TO THIS LOCATION
800	000536	000000	HALT	:EXAMINE STACK TO FIND CAUSE
801	000540	000542	.+2	:UNEXPECTED TRAP TO THIS LOCATION
802	000542	000000	HALT	:EXAMINE STACK TO FIND CAUSE
803	000544	000546	.+2	:UNEXPECTED TRAP TO THIS LOCATION
804	000546	000000	HALT	:EXAMINE STACK TO FIND CAUSE
805	000550	000552	.+2	:UNEXPECTED TRAP TO THIS LOCATION
806	000552	000000	HALT	:EXAMINE STACK TO FIND CAUSE
807	000554	000556	.+2	:UNEXPECTED TRAP TO THIS LOCATION
808	000556	000000	HALT	:EXAMINE STACK TO FIND CAUSE
809	000560	000562	.+2	:UNEXPECTED TRAP TO THIS LOCATION
810	000562	000000	HALT	:EXAMINE STACK TO FIND CAUSE
811	000564	000566	.+2	:UNEXPECTED TRAP TO THIS LOCATION
812	000566	000000	HALT	:EXAMINE STACK TO FIND CAUSE
813	000570	000572	.+2	:UNEXPECTED TRAP TO THIS LOCATION
814	000572	000000	HALT	:EXAMINE STACK TO FIND CAUSE
815	000574	000576	.+2	:UNEXPECTED TRAP TO THIS LOCATION
816	000576	000000	HALT	:EXAMINE STACK TO FIND CAUSE
817	000600	000602	.+2	:UNEXPECTED TRAP TO THIS LOCATION
818	000602	000000	HALT	:EXAMINE STACK TO FIND CAUSE
819	000604	000606	.+2	:UNEXPECTED TRAP TO THIS LOCATION
820	000606	000000	HALT	:EXAMINE STACK TO FIND CAUSE
821	000610	000612	.+2	:UNEXPECTED TRAP TO THIS LOCATION
822	000612	000000	HALT	:EXAMINE STACK TO FIND CAUSE
823	000614	000616	.+2	:UNEXPECTED TRAP TO THIS LOCATION
824	000616	000000	HALT	:EXAMINE STACK TO FIND CAUSE
825	000620	000622	.+2	:UNEXPECTED TRAP TO THIS LOCATION
826	000622	000000	HALT	:EXAMINE STACK TO FIND CAUSE
827	000624	000626	.+2	:UNEXPECTED TRAP TO THIS LOCATION
828	000626	000000	HALT	:EXAMINE STACK TO FIND CAUSE
829	000630	000632	.+2	:UNEXPECTED TRAP TO THIS LOCATION
830	000632	000000	HALT	:EXAMINE STACK TO FIND CAUSE
831	000634	000636	.+2	:UNEXPECTED TRAP TO THIS LOCATION
832	000636	000000	HALT	:EXAMINE STACK TO FIND CAUSE
833	000640	000642	.+2	:UNEXPECTED TRAP TO THIS LOCATION
834	000642	000000	HALT	:EXAMINE STACK TO FIND CAUSE
835	000644	000646	.+2	:UNEXPECTED TRAP TO THIS LOCATION
836	000646	000000	HALT	:EXAMINE STACK TO FIND CAUSE
837	000650	000652	.+2	:UNEXPECTED TRAP TO THIS LOCATION
838	000652	000000	HALT	:EXAMINE STACK TO FIND CAUSE
839	000654	000656	.+2	:UNEXPECTED TRAP TO THIS LOCATION
840	000656	000000	HALT	:EXAMINE STACK TO FIND CAUSE
841	000660	000662	.+2	:UNEXPECTED TRAP TO THIS LOCATION
842	000662	000000	HALT	:EXAMINE STACK TO FIND CAUSE
843	000664	000666	.+2	:UNEXPECTED TRAP TO THIS LOCATION
844	000666	000000	HALT	:EXAMINE STACK TO FIND CAUSE
845	000670	000672	.+2	:UNEXPECTED TRAP TO THIS LOCATION
846	000672	000000	HALT	:EXAMINE STACK TO FIND CAUSE

```

847 000674 000676      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
848 000676 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
849 000700 000702      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
850 000702 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
851 000704 000706      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
852 000706 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
853 000710 000712      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
854 000712 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
855 000714 000716      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
856 000716 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
857 000720 000722      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
858 000722 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
859 000724 000726      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
860 000726 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
861 000730 000732      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
862 000732 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
863 000734 000736      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
864 000736 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
865 000740 000742      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
866 000742 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
867 000744 000746      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
868 000746 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
869 000750 000752      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
870 000752 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
871 000754 000756      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
872 000756 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
873 000760 000762      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
874 000762 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
875 000764 000766      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
876 000766 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
877 000770 000772      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
878 000772 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
879 000774 000776      .+2      ;UNEXPECTED TRAP TO THIS LOCATION
880 000776 000000      HALT      ;EXAMINE STACK TO FIND CAUSE
881                                     ;VECTOR INITIALIZATION
882                                     .=24
883 000024 013776      .PFAIL      ;POWER FAIL VECTOR
884 000026 000340      340        ;PRIORITY 7
885 000030 016164      .HLT
886 000032 000340      340
887 000034 000056      .TRPSRV
888 000036 000340      340
889
890                                     .=46
891 000046 012470      LOGICAL
892                                     .=52
893 000052 000000      0
894                                     .=56
895                                     ;TRAP DISPATCH SERVICE
896                                     ;ARGUMENT OF TRAP IS EXTRACTED
897                                     ;AND USED AS OFFSET TO OBTAIN POINTER
898                                     ;TO SELECTED SUBROUTINE
899
900 000056 011646      .TRPSR: MOV      (SP),-(SP)      ;GET PC OF RETURN
901 000060 162716 000002      SUB      #2,(SP)      ;=PC OF TRAP
902 000064 017616 000000      MOV      @ (SP),(SP)      ;GET TRP
  
```

```

903 000070 006316          TRPOK: ASL      (SP)          ;MULTIPLY TRAP ARG BY 2
904 000072 042716 177001    BIC      #177001,(SP)      ;CLEAR UNWANTED BITS
905 000076 062716 001202    ADD      #.TRPTAB,(SP)    ;POINTER TO SUBROUTINE ADDRESS
906 000102 017616 000000    MOV      @ (SP), (SP)     ;SUBROUTINE ADDRESS
907 000106 000136          JMP      @ (SP)+         ;GO TO SUBROUTINE
908
909                          . =200
910 000200 005037 001146    START1: CLR     XLINEX
911 000204 000137 001262    JMP      BEGIN1         ;SET UP CONSOL SWITCH REGISTER
912
913                          . =210
914 000210 005037 001146    CLR     XLINEX
915 000214 000137 007234    JMP      BEGIN2         ;DB25S CONNECTOR TEST
916
917                          . =1050
918
919
920                          ;INDIRECT POINTERS TO TELETYPE VECTORS AND REGISTERS
921
922 001050 177560          TKCSR: 177560          ;TELETYPE KEYBOARD CONTROL REGISTER
923 001052 177562          TKDBR: 177562          ;TELETYPE KEYBOARD DATA BUFFER
924 001054 177564          TPCSR: 177564          ;TELEPRINTER CONTROL REGISTER
925 001056 177566          TPDBR: 177566          ;TELEPRINTER DATA BUFFER
926
927                          ;PROGRAM CONTROL PARAMETERS
928
929 001060 000000          RETURN: 0              ;SCOPE ADDRESS FOR LOOP ON TEST
930 001062 000000          NEXT: 0                ;ADDRESS OF NEXT TEST TO BE EXECUTED
931 001064 000000          LOCK: 0                ;ADDRESS FOR LOCK ON CURRENT DATA
932 001066 000000          ICOUNT: 0             ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE
933 001070 000000          LPCNT: 0              ;NUMBER OF ITERATIONS COMPLETED
934 001072 000000          TSTNO: 0              ;NUMBER OF TEST IN PROGRESS
935 001074 000000          PASCNT: 0            ;NUMBER OF PASSES COMPLETED
936 001076 000000          ERRCNT: 0            ;TOTAL NUMBER OF ERRORS
937 001100 000000          LSTERR: 0            ;PC OF LAST ERROR CALL
938
939                          ;PROGRAM VARIABLES
940
941 001102 000000          TEMP1: 0              ;TEMPORARY STORAGE
942 001104 000000          TEMP2: 0              ;TEMPORARY STORAGE
943 001106 000000          TEMP3: 0              ;TEMPORARY STORAGE
944 001110 000000          TEMP4: 0              ;TEMPORARY STORAGE
945 001112 000000          TEMP5: 0              ;TEMPORARY STORAGE
946 001114 000000          SAVR0: 0              ;R0 STORAGE
947 001116 000000          SAVR1: 0              ;R1 STORAGE
948 001120 000000          SAVR2: 0              ;R2 STORAGE
949 001122 000000          SAVR3: 0              ;R3 STORAGE
950 001124 000000          SAVR4: 0              ;R4 STORAGE
951 001126 000000          SAVR5: 0              ;R5 STORAGE
952 001130 000000          SAVSP: 0             ;STACK POINTER STORAGE
953 001132 000000          SAVPC: 0             ;PROGRAM COUNTER STORAGE
954 001134 000000          SAVSR1: 0
955 001136 000000          TMPDAT: 0
956 001140 000000          SLIM: 0
957 001142 000000          BPC: 0
958 001144 000000          TSYNC: 0

```

959	001146	000000	XLINEX:	0
960	001150	000000	CABLE:	0
961	001152	000000	TDATA:	0
962	001154	000000	RDATA:	0
963	001156	000000	CHLEN:	0
964	001160	000000	LIMIT:	0
965	001162	000000	SCNT:	0
966	001164	000000	SAVSR2:	0
967	001166	000000	TIME:	0
968	001170	000000	TP:	0
969	001172	000000	RP:	0
970	001174	000000	BACK:	0
971				

972
973
974
975 001176 000
976 001177 000
977 001200 000
978 001201 000
979 000000
980
981
982
983
984
985
986
987 001202
988 001202 104400
989 001202 015672
990 001204 104401
991 001204 013374
992 001206 104402
993 001206 013434
994 001210 104403
995 001210 014332
996 001212 104404
997 001212 013004
998 001214 104405
999 001214 012722
1000 001216 104406
1001 001216 016104
1002 001220 104407
1003 001220 016122
1004 001222 104410
1005 001222 016730
1006 001224 104411
1007 001224 016770
1008 001226 104412
1009 001226 016530
1010 001230 104413
1011 001230 016534
1012
1013
1014
1015
1016
1017
1018
1019 001232 000001
1020 001234 000001
1021 001236 000001
1022 001240 000001
1023 001242 000001
1024 001244 000001
1025
1026
1027

;PROGRAM CONTROL FLAGS

INIFLG: .BYTE 0 ;PROGRAM INITIALIZATION FLAG
STFLG: .BYTE 0 ;TEST START FLAG
ERRFLG: .BYTE 0 ;ERROR OCCURED FLAG
LOKFLG: .BYTE 0 ;LOCK ON CURRENT TEST FLAG
\$Y=0

;DEFINITIONS FOR TRAP SUBROUTINE CALLS
;POINTERS TO SUBROUTINES CAN BE FOUND
;IN THE TABLE IMMEDIATLY FOLLOWING THE DEFINITIONS

.TRPTAB:
SCOPE=TRAP+0 ;CALL TO SCOPE LOOP AND ITERATION HANDLER
;SCOPE
CLOCK=TRAP+1 ;CALL TO CLOCK DEVICE
;CLOCK
RXCLK=TRAP+2 ;CALL TO CLOCK THE RX
;RXCLK
DELAY=TRAP+3 ;CALL TO DELAY FOR SPEC. TIME.
;DELAY
CLEAR=TRAP+4 ;CALL TO BIT CLEAR SPEC BIT
;CLEAR
VALID=TRAP+5 ;CALL TO MAKE SURE ONLY SPEC BIT CLR
;VALID
SCOPI=TRAP+6 ;CALL TO LOOP ON CURRENT DATA HANDLER
;SCOPI
TYPE=TRAP+7 ;CALL TO TELETYPE OUTPUT ROUTINE
;TYPE
SAV05=TRAP+10 ;CALL TO REGISTER SAVE ROUTINE
;SAV05
RES05=TRAP+11 ;CALL TO REGISTER RESTORE ROUTINE
;RES05
CONVRT=TRAP+12 ;CALL TO DATA OUTPUT ROUTINE
;CONVRT
CNVRT=TRAP+13 ;CALL TO DATA OUTPUT ROUTINE WITHOUT CR/LF.
;CNVRT

DPRS: .BLKW 1 ;DP11 RECEIVER STATUS
DPRB: .BLKW 1 ;DP11 RECEIVER BUFFER
SYNC: .BLKW 1 ;SYNC BUFFER
DPTS: .BLKW 1 ;DP11 TRANSMITTER STATUS
DPTB: .BLKW 1 ;DP11 TRANSMITTER BUFFER
SEXT: .BLKW 1 ;DP11 SYNC EXTENSION

1028 001246 000001
1029 001250 000001
1030 001252 000001
1031 001254 000001
1032 001256 000300
1033 001260 174770
1034
1035
1036
1037 001262 000005
1038 001264 005037 001150
1039 001270 012706 001050
1040 001274 012737 000340 177776
1041 001302 104407 017022
1042 001306 105737 177570
1043 001312 100005
1044 001314 004737 C12514
1045 001320 004737 012234
1046 001324 000404
1047 001326 004737 012514
1048 001332 004737 012300
1049 001336 005737 001150
1050 001342 001402
1051 001344 000137 007306
1052 001350 012737 001356 001060
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063 001356 012737 000001 001072
1064 001364 012737 001470 001062
1065 001372 012737 000340 177776
1066 001400 005077 177634
1067 001404 005077 177622
1068 001410 012777 014532 177634
1069 001416 012777 014536 177622
1070 001424 012777 000240 177616
1071 001432 012777 000240 177614
1072 001440 112777 000026 177570
1073 001446 052777 000004 177556
1074
1075 001454 032777 000004 177550
1076 001462 001001
1077 001464 104000
1078 001466 104400

DPRIV: .BLKW 1 ;DP11 RECEIVER INTERRUPT VECTOR
DPRP: .BLKW 1 ;DP11 RECEIVER PRIORITY
DPTIV: .BLKW 1 ;DP11 TRANSMITTER INTERRUPT VECTOR
DPTP: .BLKW 1 ;DP11 TRANSMITTER PRIORITY
BASVEC: 300 ;THIS IS THE FIRST VECTOR. PATCH FOR YOUR FIRST
BASCSR: 174770 ;FIRST CSR ADDRESS.MAKE IT YOURS.

BEGIN1: RESET ;CLEAR THE WORLD.
CLR CABLE ;SET FLAG FOR NO CABLE TEST.
MOV #STACK,SP ;SET UP STACK POINTER
MOV #340,PS ;SET PROCESSOR PRIORITY = 7
TYPE, MTITLE ;+TYPE TITLE OF PROGRAM
STAR: TSTB SWR ;IS SWITCH SEVEN SET??
BPL BGNO ;BR IF SW 07 NOT UP.
JSR PC,CLRVEC ;SET UP COMM VECTOR AREA.
JSR PC,LINE.N ;GO GET THE DESIRED LINE NO. AND VECTOR.
BR PART1 ;GO TO START THE TEST.
BGNO: JSR PC,CLRVEC ;SET UP COMM VECTORS
JSR PC,LINE.X ;GO AND AUTO CYCLE THROUGH DP11S
PART1: TST CABLE ;SHOULD I DO THE CABLE TEST OR MAINT. TEST??
BEQ .+6 ;BR IF MAINT. TEST
JMP PART2 ;GO DO THE CABLE TEST
MOV #TST1,RETURN ;SET RETURN ADDRESS

*****TEST 1: READ/WRITE ALL BITS OF STATUS*****

: TEST 1

TST1: MOV #1,TSTNO ;SET PROCESSOR STATUS TO 7
MOV #TST2,NEXT ;CLEAR TRANSMITTER STATUS
MOV #340,PS ;CLEAR RECEIVER STATUS
CLR @DPTS ;CLEAR RECEIVER STATUS
CLR @DPRS ;CLEAR RECEIVER STATUS
MOV #FTINT,@DPTIV ;SET UP TRANSMITTER TEST VECTOR 1
MOV #FRINT,@DPRIV ;SET UP RECEIVER TEST VECTOR 1
MOV #240,@DPRP ;SET UP RECEIVER PRIORITY=5
MOV #240,@DPTP ;SET UP TRANSMITTER PRIORITY=5
MOVB #26,@SYNC ;CLEAR NOISE FROM SYNC
BIS #BIT2,@DPRS ;SET MAINTENANCE MODE
;TO ENABLE INTERNAL CLOCK (3KHZ)
BIT #BIT2,@DPRS ;MAINT. SET
BNE .+4 ;YES
HLT ;REPORT ERROR
SCOPE

1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090 001470 012737 000002 001072
1091 001476 012737 001576 001062
1092 001504 005000
1093 001506 013737 001232 013110
1094 001514 004537 012542
1095 001520 000001
1096 001522 004537 012542
1097 001526 000002
1098 001530 004537 012542
1099 001534 000004
1100 001536 004537 012542
1101 001542 000100
1102 001544 004537 012542
1103 001550 000200
1104 001552 004537 012542
1105 001556 000400
1106 001560 004537 012542
1107 001564 001000
1108 001566 004537 012542
1109 001572 002000
1110 001574 104400

:AN ILLEGAL INTERRUPT WILL TRAP TO
:AN ERROR MESSAGE ROUTINE
:TEST ALL READ/WRITE BITS OF RECEIVER STATUS

:*****
: *
: TEST 2
: *
:*****
:*****

TST2: MOV #2,TSTNO
MOV #TST3,NEXT
CLR R0
MOV DPRS,REG ;TEST RECEIVER STATUS BITS
JSR R5,BITST ;SYNC STRIP
BIT0
JSR R5,BITST ;HALF DUPLEX
BIT1
JSR R5,BITST ;MAINTENANCE MODE
BIT2
JSR R5,BITST ;RECEIVER INTERRUPT ENABLE
BIT6
JSR R5,BITST ;DONE
BIT7
JSR R5,BITST ;BITS/CHAR
BIT8
JSR R5,BITST ;'' ''
BIT9
JSR R5,BITST ;'' ''
BIT10
SCOPE

:BIT3=MISCELLANEOUS RECEIVE=READ ONLY
:BIT11=RECEIVE ACTIVE=READ/WRITE ZERO
:BIT12=PARITY(VRC)=READ ONLY
:ALL OTHER BITS ARE NOT USED

:TEST ALL READ/WRITE BITS OF TRANSMITTER STATUS

:*****
: *
: TEST 3
: *
:*****
:*****

1127 001576 012737 000003 001072
1128 001604 012737 001726 001062
1129 001612 005000
1130 001614 013737 001240 013110
1131 001622 004537 012542
1132 001626 000001
1133 001630 042777 000004 177374
1134 001636 004537 012542

TST3: MOV #3,TSTNO
MOV #TST4,NEXT
CLR R0
MOV DPTS,REG ;TEST TRANS STATUS BITS
JSR R5,BITST ;TERMINAL READY R/W
BIT0
BIC #BIT2,@DPRS ;SHUT OFF CLOCK FOR IDLE SYNC
JSR R5,BITST ;IDLE SYNC R/W

1135	001642	000002			BIT1		
1136	001644	052777	000004	177360	BIS	#BIT2,@DPRS	;START CLOCK
1137	001652	004537	012542		JSR	R5,BITST	;SECONDARY TRANSMIT R/W
1138	001656	000010			BIT3		
1139	001660	004537	012542		JSR	R5,BITST	;STATUS INTERRUPT ENABLE R/W
1140	001664	000040			BIT5		
1141	001666	004537	012542		JSR	R5,BITST	;TRANSMITTER INTERRUPT R/W
1142	001672	000100			BIT6		
1143	001674	004537	012542		JSR	R5,BITST	;DONE
1144	001700	000200			BIT7		
1145	001702	004537	012542		JSR	R5,BITST	;RING FLAG R/W
1146	001706	020000			BIT13		
1147	001710	004537	012542		JSR	R5,BITST	;CARRIER DOWN
1148	001714	100000			BIT15		
1149	001716	004537	012542		JSR	R5,BITST	;RECEIVER OVERUN FLAG R/W
1150	001722	040000			BIT14		
1151	001724	104400			SCOPE		
1152							
1153							
1154							
1155							
1156							
1157							
1158							
1159							
1160							
1161							
1162							
1163							
1164							
1165							
1166							
1167							
1168							
1169							
1170							
1171							
1172							
1173							
1174	001726	012737	000004	001072			
1175	001734	012737	002006	001062			
1176	001742	005077	177272				
1177	001746	012777	000004	177256			
1178	001754	012777	143707	177250			
1179	001762	017701	177244				
1180	001766	042701	010000				
1181	001772	012700	003707				
1182	001776	020001					
1183	002000	001401					
1184	002002	104001					
1185	002004	104400					
1186							
1187							
1188							
1189							
1190							

:RESET TEST
:SET PROCESSOR PRIORITY TO 7
:SET ALL WRITE BITS IN T & R STATUS
:ISSUE RESET AND VERIFY ALL BITS THAT ARE
:TO BE CLEARED BY RESET--WERE

:NOTE: IF BITS/CHAR BITS ARE SET TO ALL 1'S RCV WILL
:NOT GO ACTIVE

:TEST READ/WRITE BITS OF RECEIVER STATUS
:SECTION 1

:*****
: *
: TEST 4
: *
:*****

:*****
TST4: MOV #4,TSTNO
MOV #TST5,NEXT
CLR @DPTS ;CLEAR TRANSMITTER STATUS
MOV #BIT2,@DPRS ;MAINTENANCE MODE
MOV #143707,@DPRS ;SET ALL RECEIVER STATUS BITS
MOV @DPRS,R1 ;SAVE THE RX STATUS
BIC #BIT12,R1 ;CLEAR THE PARITY BIT
MOV #3707,R0 ;SET R0 FOR ERROR MESSAGE
CMP R0,R1 ;IS THE STATUS WHAT I EXPECTED??
BEQ +4 ;BR IF STATUS IS OK.
HLT 1 ;ERROR RX STATUS NOT WHAT EXPECTED.
SCOPE ;SCOPE THIS TEST.

:TEST ALL READ/WRITE BITS OF THE TRANSMITTER STATUS
:SECTION 2

1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246

002006	012737	000005	001072
002014	012737	002066	001062
002022	012777	000004	177202
002030	012777	160353	177202
002036	005000		
002040	062700	000001	
002044	001375		
002046	017701	177166	
002052	012700	163353	
002056	020001		
002060	001401		
002062	104001		
002064	104400		

```
*****  
: TEST 5 *  
*****  
*****  
TST5: MOV #5,TSTNO  
      MOV #TST6,NEXT  
      MOV #BIT2,@DPRS ;SET MAINT MODE  
      MOV #160353,@DPTS ;SET R/W BITS ON TX  
      CLR R0 ;SET FOR A DELAY  
1$: ADD #1,R0 ;WAIT FOR CLEAR TO SEND AND REQUEST TO S  
   BNE 1$ ;TO COME UP.  
   MOV @DPTS,R1 ;SAVE THE TX STATUS.  
   MOV #163353,R0 ;SET R0 FOR ERROR MESSAGE.  
   CMP R0,R1 ;IS THE TX STATUS CORRECT??  
   BEQ .+4 ;BR IF GOOD  
   HLT 1 ;TX STATUS ERROR  
   SCOPE ;SCOPE THIS TEST
```

:ISSUE 'RESET' AND VERIFY ALL BITS ARE CLEARED
:SECTION 3

```
*****  
: TEST 6 *  
*****  
*****  
TST6: MOV #6,TSTNO  
      MOV #TST7,NEXT  
      MOV #BIT2,@DPRS ;SET MAINT MODE  
      BIS #143707,@DPRS ;WRITE THE STATUS REG  
      RESET ;ISSUE RESET INSTR.  
      COM R0 ;FLASH THE LIGHTS  
      MOV @DPRS,R1 ;SAVE THE STATUS  
      TST R1 ;IS IT ZERO??  
      BEQ .+6 ;BR IF GOOD  
      CLR R0 ;SET R0 FOR ERROR MESSAGE  
      HLT 1 ;RX CSR NOT CLEARED BY INIT.  
      SCOPE ;SCOPE THIS TEST
```

:VERIFY ALL READ/WRITE BITS OF TRANSMITTER STATUS ARE CLEAR
:SECTION 4

```
*****  
: TEST 7 *  
*****
```

1247
1248
1249 002140 012737 000007 001072
1250 002146 012737 002212 001062
1251 002154 012777 000004 177050
1252 002162 012777 160353 177050
1253 002170 000005
1254 002172 005100
1255 002174 017701 177040
1256 002200 005701
1257 002202 001402
1258 002204 005000
1259 002206 104001
1260 002210 104400
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272 002212 012737 000010 001072
1273 002220 012737 002272 001062
1274 002226 012777 160377 177004
1275 002234 000005
1276 002236 005100
1277 002240 017701 176774
1278 002244 105777 176770
1279 002250 100002
1280 002252 005000
1281 002254 104001
1282 002256 005777 176756
1283 002262 001402
1284 002264 005000
1285 002266 104001
1286 002270 104400
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
i301
1302 002272 012737 000011 001072

```
*****  
:*****  
TST7:  MOV #7,TSTNO  
      MOV #TST10,NEXT  
      MOV #BIT2,@DPRS ;SET MAINT MODE  
      MOV #160353,@DPTS ;WRITE TX STATUS  
      RESET ;ISSUE A RESET INSTRUCTION  
      COM R0 ;FLASH THE LIGHTS.  
      MOV @DPTS,R1 ;SAVE TX STATUS  
      TST R1 ;IS IT ZERO??  
      BEQ .+6 ;BR IF GOOD  
      CLR R0 ;SET R0 FOR ERROR MSG  
      HLT 1 ;TX STATUS REG NOT ZEROED BY INIT  
      SCOPE ;SCOPE THIS TEST
```

:TEST READY BIT CLEAR BEFORE READY CAN COME UP

```
*****  
: *  
: TEST 10  
: *  
:*****
```

```
*****  
TST10: MOV #10,TSTNO  
      MOV #TST11,NEXT  
      MOV #160377,@DPTS ;LOAD STATUS  
      RESET ;ISSUE RESET INSTR.  
      COM R0 ;FLASH THE LIGHTS  
      MOV @DPTS,R1 ;SAVE THE STATUS.  
      TSTB @DPTS ;READY CLEARED  
      BPL .+6 ;YES  
      CLR R0  
      HLT 1 ;REPORT ERROR  
      TST @DPTS ;STATUS CLEAR  
      BEQ .-6 ;YES  
      CLR R0  
      HLT 1 ;REPORT ERROR  
      SCOPE
```

:BIT INTERACTION TEST
:SET EACH BIT AND VERIFY THAT ONLY THAT BIT IS AFFECTED

:RECEIVER STATUS BIT VALIDITY TEST

```
*****  
: *  
: TEST 11  
: *  
:*****  
TST11: MOV #11,TSTNO
```

1303 002300 012737 002360 001062
1304 002306 005000
1305 002310 013737 001232 013110
1306 002316 104405
1307 002320 000001
1308 002322 104405
1309 002324 000002
1310 002326 104405
1311 002330 000004
1312 002332 104405
1313 002334 000100
1314 002336 104405
1315 002340 000200
1316 002342 104405
1317 002344 000400
1318 002346 104405
1319 002350 001000
1320 002352 104405
1321 002354 002000
1322 002356 104400
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332 002360 012737 000012 001072
1333 002366 012737 002456 001062
1334 002374 005000
1335 002376 013737 001240 013110
1336 002404 005077 176630
1337 002410 104405
1338 002412 000001
1339 002414 104405
1340 002416 000002
1341 002420 104405
1342 002422 000010
1343 002424 104405
1344 002426 000040
1345 002430 104405
1346 002432 000100
1347 002434 104405
1348 002436 000200
1349 002440 104405
1350 002442 020000
1351 002444 104405
1352 002446 040000
1353 002450 104405
1354 002452 100000
1355 002454 104400
1356
1357
1358

```
MOV #TST12,NEXT
CLR R0
MOV DPRS,REG
VALID ;TEST RCV
BIT0 ;STRIP SYNC
VALID ;HALF DUPLEX
BIT1 ;MAINTENANCE MODE
VALID ;INTERRUPT ENABLE
BIT2
BIT6 ;DONE
BIT7 ;BITS/CHAR
BIT8 ;" "
BIT9 ;" "
BIT10
SCOPE
```

;TRANSMITTER STATUS BIT VALIDITY TEST

```
*****
:
: TEST 12
:
: *****
: *****
```

```
TST12: MOV #12,TSTNO
MOV #TST13,NEXT
CLR R0
MOV DPTS,REG
CLR @DPTS
VALID ;TEST XMIT STATUS
BIT0 ;CLEAR TRANSMITTER STATUS
VALID ;TERMINAL READY
BIT1 ;IDLE SYNC
VALID ;MISC TRANSMIT
BIT3 ;STATUS ENABLE
BIT5 ;TRANSMITTER ENABLE
VALID ;TRANSMITTER DONE
BIT6 ;RING FLAG
BIT7 ;RECEIVER OVERRUN
VALID ;CARRIER DOWN
BIT13
BIT14
BIT15
SCOPE
```

1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370 002456 012737 000013 001072
1371 002464 012737 002564 001062
1372 002472 112777 000026 176536
1373 002500 013737 001232 013110
1374 002506 012737 003707 001136
1375 002514 012777 003707 176510
1376 002522 104404
1377 002524 000001
1378 002526 104404
1379 002530 000002
1380 002532 104404
1381 002534 000004
1382 002536 104404
1383 002540 000100
1384 002542 104404
1385 002544 000200
1386 002546 104404
1387 002550 000400
1388 002552 104404
1389 002554 001000
1390 002556 104404
1391 002560 002000
1392 002562 104400
1393
1394
1395
1396
1397

;VERIFY BIT CLEAR ONLY CLEARS SPECIFIED BIT

;RECEIVER TEST SECTION

```
*****  
: *  
: TEST 13  
: *  
:*****  
*****  
TST13: MOV #13,TSTNO  
MOV #TST14,NEXT  
MOVB #26,@SYNC ;LOAD SYNC WITH ANYTHING  
MOV DPRS,REG ;TEST RCV STATUS  
MOV #3707,TMPDAT ;STORE STATUS IMAGE  
MOV #3707,@DPRS ;SET UP STATUS  
CLEAR ;STRIP SYNC  
BIT0  
CLEAR ;HALF DUPLEX  
BIT1  
CLEAR ;MAINTENANCE MODE  
BIT2  
CLEAR ;RECEIVER INT ENB  
BIT6  
CLEAR ;RECEIVER DONE  
BIT7  
CLEAR ;BITS/CHAR  
BIT8  
CLEAR : " "  
BIT9  
CLEAR : " "  
BIT10  
SCOPE
```

;TRANSMITTER TEST SECTION

;NOTE: "SEND REQUEST" IS SET BY "IDLE SYNC"
; "CLEAR-TO-SEND" IS SET BY MAINTENANCE MODE

1398
 1399
 1400
 1401
 1402
 1403
 1404
 1405 002564 012737 000014 001072
 1406 002572 012737 003050 001062
 1407 002600 012737 000340 177776
 1408 002606 012777 000004 176416
 1409 002614 012777 160353 176416
 1410 002622 012737 163353 001136
 1411 002630 012700 005050
 1412 002634 005300
 1413 002636 001376
 1414 002640 032777 001000 176372
 1415 002646 001001
 1416 002650 104000
 1417 002652 013737 001240 013110
 1418 002660 104404
 1419 002662 000001
 1420
 1421
 1422 002664 042737 003002 001136
 1423 002672 042777 000002 176340
 1424 002700 012700 007020
 1425 002704 005300
 1426 002706 001376
 1427 002710 013700 001136
 1428 002714 017701 176320
 1429 002720 023777 001136 176312
 1430 002726 001401
 1431 002730 104001
 1432 002732 052737 003002 001136
 1433 002740 052777 000002 176272
 1434 002746 104404
 1435 002750 000010
 1436 002752 104404
 1437 002754 000040
 1438 002756 104404
 1439 002760 000100
 1440 002762 042777 000004 176242
 1441 002770 042737 002000 001136
 1442 002776 104404
 1443 003000 000200
 1444 003002 052777 000004 176222
 1445 003010 052737 002000 001136
 1446 003016 042777 004004 176206
 1447 003024 042737 002000 001136
 1448 003032 104404
 1449 003034 020000
 1450 003036 104404
 1451 003040 040000
 1452 003042 104404
 1453 003044 100000

```

*****
: TEST 14
*****
*****
TST14:  MOV #14,TSTNO
        MOV #TST15,NEXT
        MOV #340,PS           ;SET STATUS TO LEVEL SEVEN
        MOV #BIT2,@DPRS      ;SET MAINTENANCE MODE
        MOV #160353,@DPTS    ;SET UP STATUS
        MOV #163353,TMPDAT   ;STORE STATUS IMAGE
        MOV #2600.,R0        ;DELAY 6MS FOR SEND REQUEST
        DEC R0
        BNE .-2
        BIT #BIT9,@DPTS     ;SEND REQUEST UP
        BNE .+4             ;YES
        HLT                 ;REPORT ERROR
        MOV DPTS,REG        ;TEST TRANS
        CLEAR               ;TERMINAL READY
        BIT0
; IDLE SYNC
        BIC #3002,TMPDAT    ;CLEARING IDLE SYNC SHOULD CLEAR SEND REQUEST
        BIC #2,@DPTS        ;CLEAR IDLE SYNC
        MOV #3600.,R0      ;WAIT FOR 'CLEAR-TO-SEND' TO DROP
        DEC R0
        BNE .-2
        MOV TMPDAT,R0
        MOV @DPTS,R1
        CMP TMPDAT,@DPTS   ;IDLE SYNC AND SEND REQUEST CLEAR
        BEQ .+4            ;YES
        HLT 1               ;REPORT ERROR
        BIS #3002,TMPDAT   ;REINSTATE IMAGE
        BIS #2,@DPTS       ;REINSTATE STATUS
        CLEAR              ;MISC TRANSMIT
        BIT3
        CLEAR              ;STATUS INTERRUPT ENABLE
        BIT5
        CLEAR              ;TRANSMITTER INTERRUPT ENABLE
        BIT6
        BIC #BIT2,@DPRS    ;MAINT MODE OFF (STOP CLOCK)
        BIC #BIT10,TMPDAT  ;CLEAR 'CLEAR-TO-SEND'
        CLEAR              ;READY
        BIT7
        BIS #BIT2,@DPRS    ;MAINT MODE ON
        BIS #BIT10,TMPDAT  ;SET 'CLEAR-TO-SEND' TEST BIT
        BIC #4004,@DPRS    ;SHUT OFF CLOCK
        BIC #BIT10,TMPDAT  ;CLEAR -TO -SEND
        CLEAR              ;RING FLAG
        BIT13
        CLEAR              ;RECEIVER OVERRUN
        BIT14
        CLEAR              ;CARRIER DOWN
        BIT15
  
```

1454 003046 104400

SCOPE

1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468

:PRIORITY TESTS
:VERIFY THAT THERE ARE NO STATUS INTERRUPTS
:WHEN PS=5
:INTERRUPT WILL TRAP TO ERROR MESSAGE

:
: *
: TEST 15
: *
: *****

1469 003050 012737 000015 001072
1470 003056 012737 003264 001062
1471 003064 012737 000240 177776
1472 003072 012777 014532 176152
1473 003100 012777 014536 176140
1474 003106 052777 000004 176116
1475 003114 052777 000040 176116
1476 003122 052777 010000 176110
1477 003130 042777 020040 176102
1478 003136 052777 000040 176074
1479 003144 052777 040000 176066
1480 003152 042777 040040 176060
1481 003160 052777 000040 176052
1482 003166 052777 100000 176044
1483 003174 042777 100040 176036

TST15: MOV #15,TSTNO
MOV #TST16,NEXT
MOV #240,PS ;PRIORITY=5
MOV #FTINT,@DPTIV ;SET FOR UNEXPECTED INTERUPT.
MOV #FRINT,@DPRIV ;SET FOR UNEXPECTED INTERUPT.
BIS #BIT2,@DPRS ;SET MAINTENANCE MODE
BIS #BIT5,@DPTS ;STATUS INTERRUPT ENABLE (SIE)
BIS #BIT12,@DPTS ;RING FLAG
BIC #20040,@DPTS ;CLEAR CSR
BIS #BIT5,@DPTS ;INT ENB (STATUS)
BIS #BIT14,@DPTS ;RECEIVER OVERRUN FLAG
BIC #40040,@DPTS ;CLEAR CSR
BIS #BIT5,@DPTS ;INT ENB (STATUS)
BIC #100040,@DPTS ;CARRIER DOWN FLAG
;CLEAR CSR

1484
1485
1486
1487 003202 052777 000100 176030
1488 003210 052777 000001 176022
1489 003216 052777 000002 176014
1490 003224 052777 000004 176006
1491 003232 052777 000010 176000
1492 003240 052777 000020 175772
1493 003246 042777 000037 175764
1494 003254 042777 000100 175756
1495 003262 104400

:VERIFY NO TRANSMITTER READY INTERRUPTS

BIS #BIT6,@DPTS ;XMIT INT ENB
BIS #BIT0,@DPTS ;NOISE
BIS #BIT1,@DPTS ;MORE NOISE
BIS #BIT2,@DPTS ;MORE NOISE
BIS #BIT3,@DPTS ;MORE NOISE
BIS #BIT4,@DPTS ;MORE NOISE
BIC #37,@DPTS ;QUIET!
BIC #BIT6,@DPTS

1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507

:TEST FOR CONTROL OF STATUS INTERRUPT ENABLE BIT
:NO INTERRUPT SHOULD OCCUR IF INT ENB IS NOT SET
:INTERRUPT VECTOR POINTS TO ERROR MESSAGE ROUTINE

:
: *
: TEST 16
: *
: *****

1508 003264 012737 000016 001072
1509 003272 012737 003422 001062

TST16: MOV #16,TSTNO
MOV #TST17,NEXT


```

1510 003300 012737 000200 177776      MOV      #200,PS                ;PRIORITY=4
1511 003306 012777 014532 175736      MOV      #FTINT,@DPTIV         ;SET FOR UNEXPECTED TX INTER.
1512 003314 012777 014536 175724      MOV      #FRINT,@DPRIV         ;SET FOR UNEXPECTED RX INTER.
1513 003322 052777 020000 175710      1$:     BIS      #BIT13,@DPTS     ;RING FLAG
1514 003330 042777 020000 175702      BIC      #BIT13,@DPTS         ;CLEAR
1515 003336 052777 040000 175674      BIS      #BIT14,@DPTS         ;RECEIVER OVERRUN FLAG
1516 003344 042777 040000 175666      BIC      #BIT14,@DPTS         ;CLEAR
1517 003352 052777 100000 175660      BIS      #BIT15,@DPTS         ;CARRIER DOWN FLAG
1518 003360 042777 100000 175652      BIC      #BIT15,@DPTS
1519 003366 052777 160000 175644      BIS      #160000,@DPTS        ;SET ALL STATUS ERROR BITS
1520 003374 042777 160000 175636      BIC      #160000,@DPTS        ;CLEAR
1521 003402 162737 000040 177776      SUB      #40,PS                ;DECREASE PRIORITY LEVEL
1522 003410 032737 000340 177776      BIT      #340,PS
1523 003416 001341
1524 003420 104400      BNE      1$
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539 003422 012737 000017 001072      SCOPE
1540 003430 012737 003614 001062
1541 003436 012777 003474 175606
1542 003444 012737 000200 177776
1543 003452 052777 000040 175560
1544 003460 052777 020000 175552
1545 003466 000240
1546 003470 104003
1547 003472 024646
1548 003474 042777 020000 175536      1$:     BIC      #BIT13,@DPTS     ;CLEAR RING
1549 003502 022626
1550 003504 012777 003534 175540      MOV      #2$,@DPTIV           ;SET UP NEXT INTERRUPT VECTOR
1551 003512 042737 000040 177776      BIC      #BIT5,PS             ;SET PRIORITY TO 4
1552 003520 052777 040000 175512      BIS      #BIT14,@DPTS        ;RECEIVER OVERRUN FLAG
1553 003526 000240
1554 003530 104003
1555 003532 024646
1556 003534 042777 040000 175476      2$:     BIC      #BIT14,@DPTS     ;+ERROR INDICATOR/NOPOP.SP
1557 003542 022626
1558 003544 012777 003574 175500      POP.SP
1559 003552 042737 000040 177776      MOV      #3$,@DPTIV           ;SET UP NEXT INTERRUPT VECTOR
1560 003560 052777 100000 175452      BIC      #BIT5,PS             ;SET PRIORITY TO 4
1561 003566 000240
1562 003570 104003
1563 003572 024646
1564 003574 042777 100000 175436      3$:     BIS      #BIT15,@DPTS        ;CARRIER DOWN FLAG
1565 003602 022626      POP.SP

```

```

:VERIFY THAT ALL STATUS BITS INTERRUPT AT ALL LEVELS
:EQUAL TO OR LESS THAN 4
:IF THE DEVICE INTERRUPTS SUCCESSFULLY, THE
:INTERRUPT SERVICE ROUTINE WILL RETURN
:THE PROGRAM COUNTER TO THE INSTRUCTION AFTER
:THE HLT CALL

```

```

:*****
: *
: TEST 17
: *
:*****
:*****

```

```

TST17: MOV      #17,TSTNO
      MOV      #TST20,NEXT
      MOV      #1$,@DPTIV         ;SET UP INTERRUPT VECTOR=RTI
      MOV      #200,PS           ;PRIORITY=4
      BIS      #BIT5,@DPTS        ;INT ENB STATUS
      BIS      #BIT13,@DPTS       ;RING FLAG
      NOP
      HLT      3                 ;SHOULD INTERRUPT AFTER NOP
      PUSH.SP                    ;REPORT ERROR
      1$:     BIC      #BIT13,@DPTS ;+ERROR INDICATOR/NOPOP.SP
      POP.SP                      ;CLEAR RING
      MOV      #2$,@DPTIV         ;SET UP NEXT INTERRUPT VECTOR
      BIC      #BIT5,PS           ;SET PRIORITY TO 4
      BIS      #BIT14,@DPTS       ;RECEIVER OVERRUN FLAG
      NOP
      HLT      3                 ;REPORT ERROR
      PUSH.SP                    ;+ERROR INDICATOR/NOPOP.SP
      2$:     BIC      #BIT14,@DPTS ;CLEAR RCV 0 RUN
      POP.SP
      MOV      #3$,@DPTIV         ;SET UP NEXT INTERRUPT VECTOR
      BIC      #BIT5,PS           ;SET PRIORITY TO 4
      BIS      #BIT15,@DPTS       ;CARRIER DOWN FLAG
      NOP
      HLT      3                 ;REPORT ERROR
      PUSH.SP                    ;+ERROR INDICATOR/NOPOP.SP
      3$:     BIC      #BIT15,@DPTS ;CLEAR CARRIER DOWN FLAG
      POP.SP

```

1566 003604 012777 014532 175440
1567 003612 104400
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578 003614 012737 000020 001072
1579 003622 012737 003670 001062
1580 003630 005077 175376
1581 003634 012777 000200 175376
1582 003642 105777 175372
1583 003646 100401
1584 003650 104000
1585 003652 010077 175364
1586 003656 105777 175356
1587 003662 100001
1588 003664 104000
1589 003666 104400
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600 003670 012737 000021 001072
1601 003676 012737 003774 001062
1602 003704 012777 000004 175320
1603 003712 005077 175322
1604 003716 012777 003764 175326
1605 003724 012737 000200 177776
1606 003732 110077 175304
1607 003736 052777 000100 175274
1608 003744 105777 175270
1609 003750 100001
1610 003752 104000
1611 003754 104403 004704
1612 003760 104003
1613 003762 024646
1614 003764 005077 175250
1615 003770 022626
1616 003772 104400
1617
1618
1619
1620
1621

```
MOV #FTINT,@DPTIV ;FALSE INT TRAP
SCOPE ;YES

;DOES LOADING XMIT BUFFER CLEAR XMT DONE

:*****
: *
: TEST 20
: *
:*****
:*****
TST20: MOV #20,TSTNO
MOV #TST21,NEXT
CLR @DPRS ;CLEAR RCV STATUS
MOV #BIT7,@DPTS ;DONE
TSTB @DPTS ;DONE SET
BMI .+4 ;YES
HLT ;REPORT ERROR
MOV RO,@DPTB ;LOAD BUFFER
TSTB @DPTS ;DONE CLEARED
BPL .+4 ;YES
HLT ;REPORT ERROR
SCOPE

;VERIFY TRANSMITTER READY INTERRUPTS AT LEVEL 4
;AT 8 BITS PER CHARACTER

:*****
: *
: TEST 21
: *
:*****
:*****
TST21: MOV #21,TSTNO
MOV #TST22,NEXT
MOV #BIT2,@DPRS
CLR @DPTS ;CLR STATUS
MOV #1,@DPTIV ;TEST PASS VECTOR = RT1 TO IT3A
MOV #200,PS ;PRIORITY=4
MOVB RO,@DPTB ;LOAD XMIT BUFFER
BIS #BIT6,@DPTS ;XMIT INT ENB
TSTB @DPTS ;READY CLEARED BY BUFF LOAD
BPL .+4 ;YES
HLT ;REPORT ERROR
DELAY ,2500. ;25 MS
HLT ;TRANSMITTER FAILED TO INTERRUPT
PUSH.SP ;+ERROR INDICATOR/NOPOP.SP
1$: CLR @DPTS ;CLEAR INT ENB
POP.SP
SCOPE

;VERIFY READY INTERRUPTS AT LEVEL 4
;AT 7 BITS PER CHARACTER
```

1622
1623
1624
1625
1626
1627
1628
1629 003774 012737 000022 001072
1630 004002 012737 004112 001062
1631 004010 012777 000004 175214
1632 004016 005077 175216
1633 004022 012777 004076 175222
1634 004030 012737 000200 177776
1635 004036 052777 000400 175166
1636 004044 110077 175172
1637 004050 052777 000100 175162
1638 004056 105777 175156
1639 004062 100001
1640 004064 104000
1641 004066 104403 004704
1642 004072 104003
1643 004074 024646
1644 004076 005077 175130
1645 004102 005077 175132
1646 004106 022626
1647 004110 104400
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658 004112 012737 000023 001072
1659 004120 012737 004224 001062
1660 004126 012777 000004 175076
1661 004134 012737 000200 177776
1662 004142 005077 175072
1663 004146 012777 004214 175076
1664 004154 052777 001000 175050
1665 004162 110077 175054
1666 004166 052777 000100 175044
1667 004174 105777 175040
1668 004200 100001
1669 004202 104000
1670 004204 104403 004704
1671 004210 104003
1672 004212 024646
1673 004214 005077 175012
1674 004220 022626
1675 004222 104400
1676
1677

```
*****  
: TEST 22  
: *****  
: *****  
TST22: MOV #22,TSTNO  
MOV #TST23,NEXT  
MOV #BIT2,@DPRS  
CLR @DPTS ;CLR STATUS  
MOV #1$,@DPTIV ;TEST PASS VECTOR = IT4A  
MOV #200,PS ;PRIORITY=4  
BIS #BIT8,@DPRS ;7 BITS/CHARACTER  
MOVB RO,@DPTB ;LOAD XMIT BUFFER  
BIS #BIT6,@DPTS ;XMIT INT ENB  
TSTB @DPTS ;LOAD BUFFER CLEARED READY  
BPL .+4 ;YES  
HLT ;REPORT ERROR  
DELAY ,2500. ;25 MS  
HLT 3 ;TRANSMITTER FAILED TO INTERRUPT  
PUSH.SP ;+ERROR INDICATOR/NOPOP.SP  
1$: CLR @DPRS ;CLR 7 BITS/CHAR  
CLR @DPTS ;CLEAR INT ENB  
POP.SP  
SCOPE ;YES
```

:VERIFY READY INTERRUPTS AT LEVEL 4
:AT 6 BITS PER CHARACTER

```
*****  
: TEST 23  
: *****  
: *****  
TST23: MOV #23,TSTNO  
MOV #TST24,NEXT  
MOV #BIT2,@DPRS  
MOV #200,PS ;PRIORITY=4  
CLR @DPTS ;CLR STATUS  
MOV #1$,@DPTIV ;TEST PASS VECTOR = IT5A  
BIS #BIT9,@DPRS ;6 BITS/CHARACTER  
MOVB RO,@DPTB ;LOAD XMIT BUFFER  
BIS #BIT6,@DPTS ;XMIT INT ENB  
TSTB @DPTS ;LOAD BUFFER CLEARED READY  
BPL .+4 ;YES  
HLT ;REPORT ERROR  
DELAY ,2500. ;25 MS  
HLT 3 ;TRANSMITTER FAILED TO INTERRUPT  
PUSH.SP ;+ERROR INDICATOR/NOPOP.SP  
1$: CLR @DPRS ;CLR 6 BITS/CHAR  
POP.SP  
SCOPE
```

```

1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690 004224 012737 000024 001072
1691 004232 012737 004346 001062
1692 004240 012737 004276 001064
1693 004246 005037 001136
1694 004252 005077 174754
1695 004256 005077 174755
1696 004262 105077 174750
1697 004266 105777 174744
1698 004272 001401
1699 004274 104000
1700 004276 113777 001136 174732
1701 004304 113700 001136
1702 004310 117701 174722
1703 004314 123777 001136 174714
1704 004322 001401
1705 004324 104001
1706 004326 104406
1707 004330 105237 001136
1708 004334 001360
1709 004336 112777 000026 174672
1710 004344 104400
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721 004346 012737 000025 001072
1722 004354 012737 004502 001062
1723 004362 032737 000400 001134
1724 004370 001005
1725 004372 013737 001062 001060
1726 004400 000177 174454
1727 004404 005037 001136
1728 004410 112777 000017 174626
1729 004416 113777 001136 174620
1730 004424 117701 174614
1731 004430 113700 001136
1732 004434 127737 174604 001136
1733 004442 001401
  
```

```

:TEST SYNC BUFFER IS READ/WRITE
:NOTE: SW09=1 WILL FREEZE ON CUPRENT SYNC
  
```

```

:*****
: *
: TEST 24
: *
:*****
:*****
TST24: MOV #24,TSTNO
      MOV #TST25,NEXT
      MOV #1$,LOCK
      CLR TMPDAT ;CLEAR TEST DATA
      CLR @DPRS ;CLEAR RECEIVER STATUS
      CLR @DPTS
      CLRB @SYNC ;CLEAR SYNC
      TSTB @SYNC
      BEQ .+4 ;BRANCH IF SYNC CLEARED
      HLT ;REPORT ERROR
1$: MOVB TMPDAT,@SYNC ;LOAD SYNC
   MOVB TMPDAT,R0
   MOVB @SYNC,R1
   CMPB TMPDAT,@SYNC ;TEST IF LOAD OK
   BEQ .+4 ;BRANCH OK
   HLT 1 ;REPORT ERROR
   SCOP1
   INCB TMPDAT ;NEXT SYNC
   BNE 1$ ;NO,TEST NEXT SYNC
   MOVB #26,@SYNC ;ANY SYNC BUT ALL 1'S
   SCOPE
  
```

```

:READ/WRITE ALL CHARACTERS IN SYNC EXTENSION
:NOTE: SW09=1 WILL FREEZE ON CURRENT SYNC
  
```

```

:*****
: *
: TEST 25
: *
:*****
:*****
TST25: MOV #25,TSTNO
      MOV #TST26,NEXT
      BIT #BIT8,SAVSR1 ;12 BITS/CHAR
      BNE 1$ ;NO, BRANCH AROUND TEST
      MOV NEXT,RETURN
      JMP @RETURN
1$: CLR TMPDAT ;LOAD TMPDAT WITH ZEROS
   MOVB #17,@SEXT ;LOAD SYNC EXT WITH 1'S
2$: MOVB TMPDAT,@SEXT ;LOAD SYNC EXTENSION
   MOVB @SEXT,R1
   MOVB TMPDAT,R0
   CMPB @SEXT,TMPDAT ;DID SYNC LOAD CORRECTLY
   BEQ .+4 ;YES
  
```

1734	004444	104001			HLT	1	:REPORT ERROR
1735	004446	104406			SCOP1		
1736	004450	005237	001136		INC	TMPDAT	:NEXT SYNC
1737	004454	022737	000020	001136	CMP	#20,TMPDAT	:HAVE ALL SYNC'S BEEN TESTED
1738	004462	001355			BNE	2\$:NO, CONTINUE TEST
1739	004464	105077	174554		CLRB	@SEXT	:CLEAR SYNC EXT
1740	004470	105777	174550		TSTB	@SEXT	:TEST SYNC EXT
1741	004474	001401			BEQ	+.4	:BRANCH IF SYNC CLEARED
1742	004476	104000			HLT		:REPORT ERROR
1743	004500	104400			SCOPE		

:***** SYNC TESTS *****

:SYNCHRONIZATION CHARACTER TEST
 :ISSUE ALL SYNC CHARACTERS AND VERIFY THAT IT WAS THE
 :CORRECT SYNC

:*****
 : *
 : TEST 26
 : *
 :*****

1764	004502	012737	000026	001072	TST26:	MOV	#26,TSTNO	
1765	004510	012737	005120	001062		MOV	#TST27,NEXT	
1766	004516	105077	174522			CLRB	@SEXT	:CLEAR SYNC EXTENSION
1767	004522	005077	174512			CLR	@DPTS	:CLEAR TRANSMITTER STATUS
1768	004526	005077	174500			CLR	@DPRS	:CLEAR RECEIVER STATUS
1769	004532	012737	000200	177776		MOV	#200,PS	:PRIORITY=4
1770	004540	012737	000377	001140		MOV	#377,SLIM	:SYNC LIMIT FOR 8BITS/CHAR
1771	004546	012737	000400	001142		MOV	#400,BPC	:INDEX TO CHANGE BITS/CHAR
1772	004554	052777	000004	174450		BIS	#BIT2,@DPRS	:MAINT MODE
1773	004562	012737	000001	001144		MOV	#1,TSYNC	:FIRST SYNC = 1
1774	004570	012777	014536	174450	1\$:	MOV	#FRINT,@DPRIV	:SET UP RECEIVER INT VECTOR TO ERROR
1775	004576	104403	013560		2\$:	DELAY	.6000.	:10 CHAR TIMES FOR ALL 1'S IN BUFF
1776	004602	113777	001144	174426		MOVB	TSYNC,@SYNC	:LOAD SYNC BUFFER
1777	004610	117737	174422	001152		MOVB	@SYNC,TDATA	:STORE SYNC
1778	004616	113777	001144	174416		MOVB	TSYNC,@DPTB	:LOAD FIRST SYNC CHAR
1779	004624	105777	174410			TSTB	@DPTS	:READY FOR NEXT SYNC
1780	004630	100375				BPL	.-4	:NO TEST AGAIN
1781	004632	032777	004000	174372		BIT	#BIT11,@DPRS	:TEST FOR PREMATURE ACTIVE
1782	004640	001401				BEQ	+.4	:BRANCH IF NOT SET
1783	004642	104000				HLT		:PREMATURE ACTIVE
1784	004644	113777	001144	174370		MOVB	TSYNC,@DPTB	:LOAD SECOND SYNC BYTE
1785	004652	012777	004712	174366		MOV	#3\$,@DPRIV	:SET UP TEST VECTOR
1786	004660	105777	174354			TSTB	@DPTS	:TRANSMITTER READY
1787	004664	100375				BPL	.-4	:NO
1788	004666	013777	001144	174346		MOV	TSYNC,@DPTB	:XMIT 3ED SYNC AS DATA
1789	004674	052777	000100	174330		BIS	#BIT6,@DPRS	:RCV INT ENB

```
1790 004702 104403 005670          DELAY      ,3000.          ;STALL 10 CHARACTER TIMES
1791 004706 104004          HLT        4              ;REPORT ERROR
1792 004710 024646          PUSH.SP                    ;+ERROR INDICATOR/NOPOP.SP
1793 004712 017737 174316 001136    3$:  MOV      @DPRB,TMPDAT    ;SAVE DATA
1794 004720 013701 001136          MOV      TMPDAT,R1
1795 004724 013700 001144          MOV      TSYNC,R0
1796 004730 023737 001136 001144    CMP      TMPDAT,TSYNC      ;CORRECT SYNC CHARACTER
1797 004736 001404          BEQ      4$              ;YES
1798 004740 042777 000004 174264    BIC      #BIT2,@DPRS      ;NO,SHUT OFF CLOCK
1799 004746 104001          HLT        1              ;REPORT ERROR
1800 004750 105777 174256          4$:  TSTB    @DPRS          ;DONE CLEARED
1801 004754 100001          BPL      .+4             ;YES
1802 004756 104000          HLT        ;REPORT ERROR
1803 004760 042777 000100 174244    BIC      #BIT6,@DPRS      ;CLEAR REV INT ENB
1804 004766 032777 004000 174236    BIT      #BIT11,@DPRS     ;RECEIVER ACTIVE
1805 004774 001001          BNE      .+4             ;YES
1806 004776 104000          HLT        ;REPORT ERROR
1807 005000 112777 000026 174230    MOVB    #26,@SYNC        ;CHANGE SYNC
1808 005006 042777 004000 174216    BIC      #BIT11,@DPRS     ;CLEAR RECEIVER ACTIVE
1809 005014 032777 004000 174210    BIT      #BIT11,@DPRS     ;RCV ACTIVE CLEARED
1810 005022 001401          BEQ      .+4             ;YES
1811 005024 104000          HLT        ;REPORT ERROR
1812 005026 022626          POP.SP                    ;SET PRIORITY TO 4
1813 005030 042737 000040 177776    BIC      #BIT5,PS        ;TEST FOR SCOPE LOOP
1814 005036 032737 040000 177570    BIT      #BIT14,SWR      ;BRANCH AROUND INC IF SCOPE
1815 005044 001002          BNE      5$              ;NEXT SYNC
1816 005046 105237 001144          INCB    TSYNC            ;HAVE ALL SYNC'S BEEN TESTED
1817 005052 123737 001140 001144    5$:  CMPB    SLIM,TSYNC      ;NO
1818 005060 001246          BNE      2$              ;YES
1819 005062 005037 001144          CLR     TSYNC            ;DEC BITS/CHAR BY 1 BIT
1820 005066 053777 001142 174136    BIS     BPC,@DPRS        ;DECREASE #BITS/CHAR
1821 005074 006237 001140          ASR     SLIM             ;DEC BITS/CHAR BY 1 BIT
1822 005100 062737 000400 001142    ADD     #400,BPC         ;HAVE ALL CHAR SEIZES BEEN TESTED
1823 005106 022737 001400 001142    CMP     #1400,BPC        ;NO
1824 005114 001225          BNE     1$              ;NO
1825 005116 104400          SCOPE
1826
1827          ;*****SYNC EXTENSION TEST*****
1828
1829          ;*****
1830          :
1831          : TEST 27
1832          :
1833          :*****
1834          ;*****
1835 005120 012737 000027 001072    TST27: MOV     #27,TSTNO
1836 005126 012737 005534 001062    MOV     #TST30,NEXT
1837 005134 032737 000400 001134    BIT     #BIT8,SAVSR1     ;DOES TWELVE BIT OPTION EXIST
1838 005142 001005          BNE     1$              ;BRANCH IF NOT
1839 005144 013737 001062 001060    MOV     NEXT,RETURN
1840 005152 000177 173704          JMP     @NEXT
1841 005156 012777 002104 174046    1$:  MOV     #2104,@DPRS     ;SET STATUS TO 12 BITS/CHARACTER
1842 005164 012737 000400 001144    MOV     #400,TSYNC       ;FIRST SYNC CHARACTER
1843 005172 105077 174040          CLRB   @SYNC            ;LOAD SYNC BUFFERS WITH 400
1844 005176 112777 000001 174040    MOVB   #1,@SEXT         ;LOAD SYNC BUFFERS WITH 400
1845 005204 012737 007400 001140    MOV     #7400,SLIM       ;SET UP SYNC LIMIT
```

1846	005212	012737	002000	001142		MOV	#2000,BPC		:SET # BITS/CHAR TO 12
1847	005220	012777	014536	174020	2\$:	MOV	#FRINT,@DPRIV		:RCV INTERRUPT VECTOR = ERROR
1848	005226	104403	006000			DELAY	,6000		:WAIT FOR ALL 1'S
1849									:TO SHIFT INTO XMIT,RCV BUFS
1850	005232	105077	174000			CLRB	@SYNC		:CLEAR SYNC EXTENTION
1851	005236	113777	001145	174000		MOVB	TSYNC+1,@SEXT		:LOAD NEXT SYNC
1852	005244	013777	001144	173770		MOV	TSYNC,@DPTB		:TRANSMIT FIRST SYNC
1853	005252	105777	173762			TSTB	@DPTS		:WAIT FOR 'DONE'
1854	005256	100375				BPL	.-4		:
1855	005260	032777	004000	173752		BIT	#BIT11,@DPTS		:TEST FOR PREMATURE 'ACTIVE'
1856	005266	001401				BEQ	+.4		:NO
1857	005270	104000				HLT			:REPORT ERROR
1858	005272	013777	001144	173742		MOV	TSYNC,@DPTB		:TRANSMIT SECOND SYNC
1859	005300	105777	173734			TSTB	@DPTS		:WAIT FOR 'DONE'
1860	005304	100375				BPL	.-4		:
1861	005306	012777	005360	173732		MOV	#3\$,@DPRIV		:SET UP RECEIVER INT VECTOR
1862	005314	052777	000100	173710		BIS	#BIT6,@DPRS		:RCV INTERRUPT ENABLE
1863	005322	013777	001144	173712		MOV	TSYNC,@DPTB		:TRANSMIT 3ED SYNC AS DATA
1864	005330	105777	173704			TSTB	@DPTS		:WAIT FOR 'DONE'
1865	005334	100375				BPL	.-4		:
1866	005336	032777	004000	173666		BIT	#BIT11,@DPRS		:TEST FOR ACTIVE
1867	005344	001001				BNE	+.4		:OK
1868	005346	104000				HLT			:REPORT ERROR
1869	005350	104403	013560			DELAY	,6000.		:WAIT FOR INTERRUPT
1870	005354	104004				HLT	4		:DEVICE FAILED TO INTERRUPT
1871	005356	024646				PUSH.SP			:INDICATOR/NOPOP.SP
1872	005360	017737	173650	001136	3\$:	MOV	@DPRB,TMPDAT		:SAVE RECEIVED DATA
1873	005366	013700	001144			MOV	TSYNC,R0		
1874	005372	013701	001136			MOV	TMPDAT,R1		
1875	005376	023737	001144	001136		CMP	TSYNC,TMPDAT		:COMPARE SYNC
1876	005404	001401				BEQ	+.4		:BRANCH IF SYNC OK
1877	005406	104001				HLT	1		:REPORT ERROR
1878	005410	022626				POP.SP			
1879	005412	042737	000040	177776		BIC	#BIT5,PS		:LOWER PRIORITY
1880	005420	042777	004000	173604		BIC	#BIT11,@DPRS		:CLEAR ACTIVE
1881	005426	042777	000100	173576		BIC	#BIT6,@DPRS		:CLEAR INT ENB
1882	005434	032737	040000	177570		BIT	#BIT14,SWR		:TEST FOR SCOPE LOOP
1883	005442	001002				BNE	4\$:BRANCH AROUND INC IF SCOPE
1884	005444	105237	001145			INCB	TSYNC+1		:INC TO NEXT TEST SYNC
1885	005450	123737	001145	001141	4\$:	CMPB	TSYNC+1,SLIM+1		:HAVE ALL SYNC'S BEEN TESTED
1886	005456	001404				BEQ	5\$:YES
1887	005460	112777	000026	173550		MOVB	#26,@SYNC		
1888	005466	000654				BR	2\$		
1889	005470	005037	001144		5\$:	CLR	TSYNC		:CLEAR TSYNC
1890	005474	006237	001140			ASR	SLIM		:DECREASE SYNC LIMIT
1891	005500	043777	001142	173524		BIC	BPC,@DPRS		:CLEAR OLD CHAR SIZE
1892	005506	062737	000400	001142		ADD	#400,BPC		:INC BITS/CHAR TO NEXT SIZE
1893	005514	053777	001142	173510		BIS	BPC,@DPRS		:CHANGE BIT MODE
1894	005522	022737	003400	001142		CMP	#3400,BPC		:CHECK CHARACTER SIZE
1895	005530	001233				BNE	2\$:BRANCH UNTIL ALL SIZES HAVE BEEN TESTED
1896	005532	104400				SCOPE			

```

1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912 005534 012737 000030 001072
1913 005542 012737 006134 001062
1914 005550 012737 000026 001144
1915 005556 005077 173456
1916 005562 005077 173444
1917 005566 013703 001232
1918 005572 005203
1919 005574 052777 000004 173430
1920 005602 012702 006112
1921
1922 005606 104403 013560
1923 005612 012777 005666 173426
1924 005620 113777 001144 173410
1925 005626 113777 001145 173410
1926 005634 013777 001144 173400
1927 005642 052777 000002 173370
1928 005650 052777 000100 173354
1929
1930 005656 104403 027340
1931 005662 104004
1932 005664 024646
1933 005666 017737 173342 001136
1934 005674 022626
1935 005676 000412
1936 005700 013700 001144
1937 005704 013701 001136
1938 005710 023737 001136 001144
1939 005716 001401
1940 005720 104001
1941 005722 000415
1942 005724 105737 001137
1943 005730 001401
1944 005732 104000
1945 005734 013700 001144
1946 005740 013701 001136
1947 005744 123737 001136 001144
1948 005752 001401
1949 005754 104001
1950 005756 105777 173250
1951 005762 100001
1952 005764 104000
  
```

```

:*****
:IDLE SYNC TEST
:RAISE 'ACTIVE' BY IDLEING IN EACH AVAILABLE CHARACTER LENGTH
:*****
: TEST 30
:*****
:*****
TST30: MOV #30,TSTNO
      MOV #TST31,NEXT
      MOV #26,TSYNC ;LOAD TEST SYNC CHARACTER
      CLR @DPTS ;CLEAR STATUS REGISTERS
      CLR @DPRS
      MOV DPRS,R3 ;FETCH DEVICE ADRS
      INC R3 ;CHANGE ADRS TO HIGH BYTE OF STATUS
      BIS #BIT2,@DPRS ;START MAINTENANCE
      MOV #6$,R2 ;SET UP CHARACTER LENGTH SELECTOR

1$: DELAY ,6000. ;WAIT FOR ALL 1'S TO SHIFT IN
   MOV #2$,@DPRIV ;LOAD DP RCV INTERRUPT VECTOR
   MOVB TSYNC,@SYNC ;LOAD LOW BYTE OF SYNC
   MOVB TSYNC+1,@SEXT ;LOAD SYNC EXTENSION BITS
   MOV TSYNC,@DPTB ;LOAD XMIT BUFFER
   BIS #BIT1,@DPTS ;SET IDLE SYNC
   BIS #BIT6,@DPRS ;SET RCV INTERRUPT ENABLE

2$: MOV @DPRB,TMPDAT ;+ERROR INDICATOR/NOPOP.SP
   POP.SP ;SAVE RCV DATA

3$: BR 4$ ;CONTROL WORD 12 BITS=BR .+2
   MOV TSYNC,R0
   MOV TMPDAT,R1
   CMP TMPDAT,TSYNC ;DOES SYNC CHECK
   BEQ ,+4 ;YES
   HLT 1 ;REPORT ERROR

4$: TSTB TMPDAT+1 ;VERIFY ONLY 8 BITS WERE TRANSMITTED
   BEQ ,+4 ;BRANCH IF OK
   HLT ;REPORT ERROR

5$: MOV TSYNC,R0 ;CHECK SYNC IN LOW BYTE
   MOV TMPDAT,R1 ;BRANCH IF SYNC OK
   CMPB TMPDAT,TSYNC ;REPORT ERROR
   BEQ ,+4 ;DID READING RCV BUFF CLR DONE
   HLT 1 ;YES
   TSTB @DPRS ;REPORT ERROR
   BPL ,+4
   HLT
  
```


1953	005766	042777	004000	173236	BIC	#BIT11,@DPRS	:CLEAR ACTIVE
1954	005774	032777	004000	173230	BIT	#BIT11,@DPRS	:ACTIVE CLEARED?
1955	006002	001401			BEQ	.+4	:YES
1956	006004	104000			HLT		:REPORT ERROR
1957	006006	042777	000002	173224	BIC	#BIT1,@DPTS	:CLEAR IDLE
1958	006014	042737	000040	177776	BIC	#BIT5,PS	:LOWER PRIORITY TO 4
1959	006022	022626			POP.SP		:ADJUST STACK
1960	006024	142213			BICB	(R2)+,@R3	:CLEAR CHAR LENGTH
1961	006026	152213			BISB	(R2)+,@R3	:SELECT NEXT CHAR LENGTH
1962	006030	020227	006116		CMP	R2,#6\$+4	:END OF MODE?
1963	006034	001264			BNE	1\$:NO
1964	006036	032737	000400	001134	BIT	#BIT8,SAVSR1	:TEST 12 BITS/CHARACTER
1965	006044	001424			BEQ	7\$:NO
1966	006046	032777	002000	173156	BIT	#BIT10,@DPRS	:END OF 12 BIT TEST
1967	006054	001020			BNE	7\$:YES
1968	006056	052777	002000	173146	BIS	#BIT10,@DPRS	:NO
1969	006064	024242			CMP	-(R2),-(R2)	:ADJUST CHAR SELECTION
1970	006066	042777	001400	173136	BIC	#1400,@DPRS	:CLEAR CHAR LENGTH LSB'S
1971	006074	012737	000400	005676	MOV	#400,3\$:CHANCE CONTROL WORD
1972	006102	012737	001426	001144	MOV	#1426,TSYNC	:CHANGE SYNC
1973	006110	000636			BR	1\$	
1974	006112	000403			6\$:	403	:CHARACTER LENGTH SELECTION
1975	006114	001003				1003	:CHARACTER LENGTH SELECTION
1976	006116	012737	000026	001144	7\$:	MOV #26,TSYNC	:RESTORE TSYNC
1977	006124	012737	000412	005676		MOV #412,3\$:RESTORE CONTROL WORD
1978	006132	104400				SCOPE	
1979							
1980							
1981							
1982							
1983							
1984							
1985							
1986							
1987							
1988							
1989	006134	012737	000031	001072	TST31:	MOV #31,TSTNO	
1990	006142	012737	006346	001062		MOV #TST32,NEXT	
1991	006150	012737	006344	001174		MOV #3\$,BACK	
1992	006156	105077	173062			CLRB @SEXT	:CLEAR SYNC EXTENTION
1993	006162	005037	001154			CLR RDATA	:INIT RCV DATA
1994	006166	005037	001152			CLR TDATA	:INIT XMIT DATA
1995	006172	005077	173042			CLR @DPTS	:TRANSMITTER STATUS
1996	006176	012777	000005	173026		MOV #5,@DPRS	:CLOCK ON + STRIP SYNC
1997	006204	012737	000400	001156		MOV #400,CHLEN	:CHAR LENGTH INDEX
1998	006212	032737	000400	001134		BIT #BIT8,SAVSR1	:TEST 12 BIT CHAR MODE
1999	006220	001414				BEQ 1\$:NO
2000	006222	012737	010000	001160		MOV #10000,LIMIT	:SELECT END OF DATA
2001	006230	052777	002000	172774		BIS #BIT10,@DPRS	:SELECT 12 BITS/CHARACTER
2002	006236	012737	000426	001144		MOV #426,TSYNC	:SYNC FOR 12 BIT CHAR
2003	006244	105277	172774			INCB @SEXT	:PLACE MSB OF SYNC IN SYNC EXT
2004	006250	000406				BR 2\$	
2005	006252	012737	000400	001160	1\$:	MOV #400,LIMIT	:TEMPORARY CHARACTER LIMIT
2006	006260	012737	000026	001144		MOV #26,TSYNC	:INIT SYNC STORAGE
2007	006266	012777	014542	172756	2\$:	MOV #TV18,@DPTIV	:TRANSMITTER VECTOR
2008	006274	012777	014650	172744		MOV #RV18,@DPRIV	:RECEIVER VECTOR

:*****INTERRUPT DRIVEN SEQUENTIAL DATA TEST*****

```

:*****
:
: TEST 31
:
:*****
:*****

```

```

2009 006302 012737 000200 177776      MOV      #200,PS          ;PRIORITY=4
2010 006310 012737 000003 001162      MOV      #3,SCNT         ;SYNC COUNT=3
2011 006316 113777 001144 172712      MOV      TSYNC,@SYNC    ;LOAD SYNC
2012 006324 052777 000100 172700      BIS      #BIT6,@DPRS    ;RCV INT ENB
2013 006332 052777 000340 172700      BIS      #340,@DPTS     ;STATUS INT ENB
2014                                     ;TRANS INT ENB
2015                                     ;TRANS DONE
2016 006340 000001      WAIT                                     ;WAIT FOR INTERRUPTS
2017 006342 000776      BR      .-2
2018 006344 104400      3$:   SCOPE
2019
2020                                     ;*****RANDOM DATA, RANDOM STALL*****
2021
2022
2023                                     ;*****
2024                                     ; *
2025                                     ; TEST 32
2026                                     ; *
2027                                     ;*****
2028                                     ;*****
2029 006346 012737 000032 001072      TST32: MOV      #32,TSTNO
2030 006354 012737 006472 001062      MOV      #TST33,NEXT
2031 006362 005037 001156      CLR      CHLEN          ;SET CHAR LENGTH TO 8 BITS
2032 006366 012737 000026 001144      MOV      #26,TSYNC     ;SYNC = 26
2033 006374 113777 001144 172634      MOV      TSYNC,@SYNC   ;LOAD SYNC BUFFER
2034 006402 004737 013112      JSR      PC,AND        ;EXECUTE DATA + STALL MODES
2035
2036                                     ;REPEAT PREVIOUS TEST AT 7 BITS/CHAR
2037
2038 006406 012737 000200 001156      MOV      #200,CHLEN    ;SET CHAR LENGTH TO 7 BITS
2039 006414 052777 000400 172610      BIS      #BIT8,@DPRS   ;7 BITS/CHAR
2040 006422 042777 004000 172602      BIC      #BIT11,@DPRS  ;CLEAR ACTIVE
2041 006430 004737 013112      JSR      PC,AND        ;EXECUTE DATA + STALL MODES
2042
2043                                     ;REPEAT PREVIOUS TEST AT 6 BITS/CHAR
2044
2045 006434 012737 000300 001156      MOV      #300,CHLEN    ;SET CHAR LENGTH TO 6 BITS
2046 006442 042777 000400 172562      BIC      #BIT8,@DPRS   ;
2047 006450 052777 001000 172554      BIS      #BIT9,@DPRS   ;SET MODE TO 6 BITS/CHAR
2048 006456 042777 004000 172546      BIC      #BIT11,@DPRS  ;CLEAR ACTIVE
2049
2050 006464 004737 013112      JSR      PC,AND        ;EXECUTE DATA & STALL MODES
2051 006470 104400      SCOPE

```

2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062 006472 012737 000033 001072
2063 006500 012737 006676 001062
2064 006506 012737 006674 001174
2065 006514 005077 172512
2066 006520 005077 172514
2067 006524 012737 001401 015454
2068 006532 012737 000400 001160
2069 006540 005037 001154
2070 006544 005037 001152
2071 006550 105077 172470
2072 006554 012737 000026 001144
2073 006562 113777 001144 172446
2074 006570 012737 000003 001162
2075 006576 005077 172430
2076 006602 005077 172432
2077 006606 032737 000400 001134
2078 006614 001406
2079 006616 012737 010000 001160
2080 006624 052777 002000 172400
2081 006632 012737 000200 177776
2082 006640 012777 015320 172404
2083 006646 012777 015410 172372
2084 006654 052777 000105 172350
2085 006662 052777 000300 172350
2086 006670 000001
2087 006672 000776
2088 006674 104400
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098 006676 012737 000034 001072
2099 006704 012737 007136 001062
2100 006712 005037 001136
2101 006716 105077 172322
2102 006722 112777 000026 172306
2103 006730 032737 000400 001134
2104 006736 001403
2105 006740 012777 002000 172264
2106 006746 012777 007072 172276
2107 006754 052777 000004 172250

```
;*****PARITY TEST*****  
;VERIFY 'PARITY' BIT=1 FOR ODD PARITY AND=0 FOR EVEN  
  
:*****  
: *  
: TEST 33  
: *  
:*****  
:*****  
TST33: MOV #33,TSTNO  
MOV #TST34,NEXT  
MOV #2$,BACK  
CLR @DPRS  
CLR @DPTS  
MOV #1401,RPRT1 ;LOAD RPRT2WITH BEQ .+4  
MOV #400,LIMIT ;SET UP CHARACTER LIMIT  
CLR RDATA ;CLR RCV DATA  
CLR TDATA ;CLR XMIT DATA  
CLRB @SEXT ;CLEAR SYNC EXTENTION  
MOV #26,TSYNC ;SET UP SYNC  
MOVB TSYNC,@SYNC ;INIT SYNC  
MOV #3,SCNT ;3 SYNC'S  
CLR @DPRS ;CLR RECEIVER STATUS  
CLR @DPTS ;CLR TRANSMITTER STATUS  
BIT #BIT8,SAVSR1 ;8/12 BITS/CHAR  
BEQ 1$ ;BRANCH IF 8 BITS/CHAR  
MOV #10000,LIMIT ;SET LIMIT TO 12 BITS/CHAR  
BIS #BIT10,@DPRS ;SELECT 12 BIT MODE  
1$: MOV #200,PS ;PRIORITY = 4  
MOV #TPRTY,@DPTIV ;TRANSMITTER PARITY TEST VECTOR  
MOV #RPRTY,@DPRIV ;RECEIVER PARITY TEST VECTOR  
BIS #105,@DPRS ;RCV INT ENB, STRIP SYNC, CLOCK  
BIS #300,@DPTS ;XMIT INT ENB,DONE  
WAIT  
BR -2  
2$: SCOPE  
  
;RECEIVER OVERRUN TST  
  
:*****  
: *  
: TEST 34  
: *  
:*****  
:*****  
TST34: MOV #34,TSTNO  
MOV #TST35,NEXT  
CLR TMPDAT ;STOR TEST CHAR IN TMPDAT  
CLRB @SEXT ;CLEAR SYNC EXTENTION  
MOVB #26,@SYNC ;LOAD SYNC BUFFER  
BIT #BIT8,SAVSR1 ;8/12 BITS/CHAR  
BEQ 1$ ;BRANCH IF 8 BITS/CHAR  
MOV #BIT10,@DPRS ;SELECT 12 BITS/CHAR  
1$: MOV #3$,@DPTIV ;XMIT STATUS INT VECTOR=0'RUN  
BIS #BIT2,@DPRS ;TURN ON CLOCK
```

2108 006762 104403 013560
 2109 006766 052777 000200 172244
 2110 006774 105777 172240
 2111 007000 100375
 2112 007002 012777 000026 172232
 2113 007010 105777 172224
 2114 007014 100375
 2115 007016 012777 000026 172216
 2116 007024 105777 172210
 2117 007030 100375
 2118 007032 013777 001136 172202
 2119 007040 105777 172174
 2120 007044 100375
 2121 007046 013777 001136 172166
 2122 007054 052777 000040 172156
 2123 007062 104403 005670
 2124 007066 104000
 2125 007070 024646
 2126 007072 032777 040000 172140
 2127 007100 001001
 2128 007102 104000
 2129 007104 005077 172130
 2130 007110 042777 004000 172114
 2131 007116 022626
 2132 007120 042737 000040 177776
 2133 007126 105237 001136
 2134 007132 001313
 2135 007134 104400
 2136
 2137
 2138
 2139
 2140
 2141
 2142
 2143
 2144
 2145
 2146 007136 012737 000035 001072
 2147 007144 012737 007340 001062
 2148 007152 005077 172062
 2149 007156 012777 000103 172046
 2150 007164 012777 014536 172054
 2151 007172 005037 001144
 2152 007176 012737 000200 177776
 2153 007204 113777 001144 172024
 2154 007212 052777 000002 172020
 2155 007220 104403 005670
 2156 007224 105237 001144
 2157 007230 001365
 2158 007232 104400
 2159

```

2$:  DELAY      ,6000.           ;WAIT FOR BCV TO CLEAR
      BIS       #BIT7,@DPTS      ;DONE
      TSTB     @DPTS             ;TRANSMIT FIRST SYNC
      BPL      .-4
      MOV       #26,@DPTB        ;
      TSTB     @DPTS             ;
      BPL      .-4
      MOV       #26,@DPTB        ;TRANSMIT SECOND SYNC
      TSTB     @DPTS             ;
      BPL      .-4
      MOV       TMPDAT,@DPTB     ;TRANSMIT DATA CHAR #1
      TSTB     @DPTS             ;
      BPL      .-4
      MOV       TMPDAT,@DPTB     ;TRANSMIT DATA CHAR #2
      BIS       #BIT5,@DPTS      ;SET STATUS INT ENB
      DELAY     ,3000.           ;WAIT FOR O'RUN INTERRUPT
      HLT
      PUSH.SP
3$:  BIT        #BIT14,@DPTS     ;TEST FOR O'RUN
      BNE      .+4               ;BRANCH IF O'RUN CAUSED INT
      HLT
      CLR       @DPTS            ;CLEAR XMIT STATUS
      BIC      #BIT11,@DPRS      ;CLEAR ACTIVE
      POP.SP
      BIC      #BIT5,PS          ;LOWER PRIORITY TO 4
      INCB     TMPDAT           ;INC TO NEXT DATA
      BNE     2$                ;BRANCH IF NOT END
      SCOPE
  
```

;HALF DUPLEX TEST

```

:*****
:      *
:  TEST 35
:      *
:*****
:*****
  
```

```

TST35: MOV      #35,TSTNO
        MOV      #TST36,NEXT
        CLR      @DPTS           ;CLEAR TRANSMITTER STATUS
        MOV      #103,@DPRS      ;HALF DUPLEX,INT EN,TURN CLK ON
        MOV      #FRINT,@DPRIV   ;SETUP TEST VECTOR
        CLR      TSYNC          ;CLR TEST SYNC
        MOV      #200,PS         ;PRIORITY=4
1$:  MOVB     TSYNC,@SYNC        ;LOAD SYNC BUFFER
      BIS     #BIT1,@DPTS       ;IDLE SYNC
      DELAY   ,3000.           ;DELAY 20.1 MS
      INCB    TSYNC            ;HAVE ALL SYNC BEEN TESTED
      BNE    1$                ;NO
      SCOPE
  
```

;NOTE END OF THIS TEST.

2160
 2161
 2162 007234 000005
 2163 007236 012737 177777 001150
 2164 007244 012706 001050
 2165 007250 012737 000340 177776
 2166 007256 105737 177570
 2167 007262 100005
 2168 007264 004737 012514
 2169
 2170
 2171 007270 004737 012234
 2172 007274 000404
 2173 007276 004737 012514
 2174 007302 004737 012300
 2175 007306 032737 000001 001134
 2176 007314 001405
 2177 007316 012737 012022 001060
 2178 007324 000177 171530
 2179 007330 012737 007306 001060
 2180 007336 000240
 2181
 2182
 2183
 2184
 2185
 2186
 2187
 2188
 2189
 2190
 2191
 2192 007340 012737 000036 001072
 2193 007346 012737 007470 001062
 2194 007354 005737 001150
 2195 007360 001013
 2196 007362 032737 000001 001134
 2197 007370 001405
 2198 007372 012737 012022 001060
 2199 007400 000177 171454
 2200 007404 000137 016002
 2201 007410 112777 000026 171620
 2202 007416 005077 171616
 2203 007422 105077 171616
 2204 007426 005077 171600
 2205 007432 012777 014532 171612
 2206 007440 012777 014536 171600
 2207 007446 012777 000240 171574
 2208 007454 012777 000240 171572
 2209 007462 104401
 2210 007464 000030
 2211 007466 104400
 2212
 2213
 2214
 2215

```

:*****PART2 DB25S CONNECTOR TEST SECTION*****
BEGIN2: RESET
        MOV    #-1,CABLE
        MOV    #STACK,SP      ;SET UP STACK POINTER
        MOV    #340,PS        ;SET PROCESSOR PRIORITY = 7
        TSTB   SWR            ;TEST FOR CHANGE IN DP ADRS
        BPL    BGN0A         ;BRANCH IF NO CHANGE
        JSR    PC,CLRVEC     ;LOAD ENTIRE VECTOR AREA WITH
        ;      .+2
        ;      HALT
        JSR    PC,LINE.N     ;FETCH LINE NUMBER FROM SWR
BR      PART2
BGN0A:  JSR    PC,CLRVEC
PART2:  JSR    PC,LINE.X
        BIT    #BIT0,SAVSR1
        BEQ    1$
        MOV    #PART3,RETURN
        JMP    @RETURN
1$:     MOV    #PART2,RETURN
        NOP

:*****TEST 1: CABLE TESTS
:*****
:      *
:      TEST 36
:      *
:*****
:*****
TST36:  MOV    #36,TSTNO
        MCV    #TST37,NEXT
        TST    CABLE
        BNE    1$
        BIT    #BIT0,SAVSR1
        BEQ    .+14
        MCV    #PART3,RETURN
        JMP    @RETURN
        JMP    .EOP
1$:     MOVB   #26,@SYNC      ;CLEAR NOISE FROM SYNC
        CLR    @DPTS        ;CLEAR TRANSMITTER STATUS
        CLRB  @SEXT        ;CLEAR SYNC EXT
        CLR    @DPRS        ;CLEAR RECEIVER STATUS
        MOV    #FTINT,@DPTIV ;SET UP TRANSMITTER TEST VECTOR 1
        MOV    #FRINT,@DPRIV ;SET UP RECEIVER TEST VECTOR 1
        MOV    #240,@DPRP   ;SET UP RECEIVER PRIORITY=6
        MOV    #240,@DPTP   ;SET UP TRANSMITTER PRIORITY=6
        CLOCK
        30
        SCOPE
:SYNCHRONIZATION CHARACTER TEST 8/12 BITS/CHARACTER
:INTERRUPT ENABLE, COMPARE SYNC, TEST PARITY
  
```

2216
 2217
 2218
 2219
 2220
 2221
 2222
 2223 007470 012737 000037 001072
 2224 007476 012737 007760 001062
 2225 007504 004537 013524
 2226 007510 012737 000376 001160
 2227 007516 000403
 2228
 2229 007520 012737 007776 001160
 2230 007526 012737 007710 001174
 2231 007534 012777 014542 171510
 2232 007542 012737 000003 001162
 2233
 2234 007550 104401
 2235 007552 000030
 2236 007554 113777 001144 171454
 2237 007562 113777 001145 171454
 2238 007570 052777 000100 171434
 2239 007576 052777 000300 171434
 2240 007604 104402
 2241 007606 000003
 2242 007610 032777 001000 171422
 2243 007616 001001
 2244 007620 104000
 2245 007622 104401
 2246 007624 000010
 2247 007626 032777 004000 171376
 2248 007634 001401
 2249 007636 104000
 2250 007640 104401
 2251 007642 000007
 2252 007644 032777 004000 171366
 2253 007652 001401
 2254 007654 104000
 2255 007656 104402
 2256 007660 000001
 2257 007662 032777 004000 171342
 2258 007670 001001
 2259 007672 104000
 2260 007674 012777 015622 171344
 2261 007702 104401
 2262 007704 000010
 2263 007706 104000
 2264 007710 042777 004000 171314
 2265 007716 042777 000100 171314
 2266 007724 032737 040000 177570
 2267 007732 001300
 2268 007734 112777 000026 171274
 2269 007742 005237 001144
 2270 007746 023737 001160 001144
 2271 007754 001267

```

:*****
:      *
: TEST 37
:      *
:*****

```

```

*****
TST37:  MOV    #37,TSTNO
        MOV    #TST40,NEXT
        JSR    R5,REE
        MOV    #376,LIMIT
        BR     .+10
;REE WILL ENER HERE IF 12 BITS/CHAR
        MOV    #7776,LIMIT
        MOV    #2$,BACK
1$:     MOV    #TV18,@DPTIV
        MOV    #3,SCNT
;*****
        CLOCK
        30
        MOVB   TSYNC,@SYNC
        MOVB   TSYNC+1,@SEXT
        BIS    #BIT6,@DPRS
        BIS    #300,@DPTS
        RXCLK
        3
        BIT    #BIT9,@DPTS
        BNE    .+4
        HLT
        CLOCK
        10
        BIT    #BIT11,@DPRS
        BEQ    .+4
        HLT
        CLOCK
        ?
        BIT    #BIT11,@DPTS
        BEQ    .+4
        HLT
        RXCLK
        1
        BIT    #BIT11,@DPRS
        BNE    .+4
        HLT
        MOV    #SRV5,@DPRIV
        CLOCK
        10
        HLT
        BIC    #BIT11,@DPRS
        BIC    #BIT6,@DPTS
        BIT    #BIT14,SWR
        BNE    1$
        MOVB   #26,@SYNC
        INC    TSYNC
        CMP    LIMIT,TSYNC
        BNE    1$
;REINIT FOR TEST
;8 BIT SYNC LIMIT
;BRANCH AROUND 12 BIT LIMIT
;SET UP 12 BITS/CHAR LIMIT
;SET UP RCV SERVICE RETURN
;SET XMIT INT VECTOR TO SYNC
;SYNC COUNT = 3
;RUN CLOCK
;LOAD SYNC BUFFER
;LOAD SYNC EXT
;RCV INT ENB
;XMIT INT ENB
;SEND REQUEST UP?
;YES
;REPORT ERROR
;RUN CLOCK
;RECEIVER ACTIVE
;NO
;REPORT ERROR
;RUN CLOCK
;RCV ACTIVE?
;NO
;REPORT ERROR
;RCV ACTIVE?
;YES
;REPORT ERROR
;TEST PASS VECTOR
;RUN CLOCK
;REPORT ERROR
;CLEAR ACTIVE
;CLEAR INT ENB
;TEST FOR SCOPE LOOP
;BRANCH IF SCOPE LOOP
;CHANGE SYNC
;NEXT SYNC, ENTRY FROM INTERRUPT
;HAVE ALL SYNC'S BEEN TESTED

```

2272 007756 104400

SCOPE

:SYNCHRONIZATION CHARACTER TEST 7/11 BITS/CHARACTER
 :INTERRUPT ENABLE AND SYNC CHARACTER CHECK

2273
 2274
 2275
 2276
 2277
 2278
 2279
 2280
 2281
 2282
 2283
 2284 007760 012737 000040 001072
 2285 007766 012737 010256 001062
 2286 007774 004537 013524
 2287
 2288 010000 012737 000176 001160
 2289 010006 000403
 2290
 2291 010010 012737 003776 001160
 2292 010016 052777 000400 171206
 2293 010024 012737 010206 001174
 2294 010032 012777 014536 171206
 2295 010040 012777 014542 171204
 2296 010046 012737 000003 001162
 2297 010054 104401
 2298 010056 000024
 2299 010060 113777 001144 171150
 2300 010066 052777 000300 171144
 2301 010074 052777 000100 171130
 2302 010102 104402
 2303 010104 000003
 2304 010106 032777 001000 171124
 2305 010114 001001
 2306 010116 104000
 2307 010120 104401
 2308 010122 000007
 2309 010124 032777 004000 171100
 2310 010132 001401
 2311 010134 104000
 2312 010136 104401
 2313 010140 000006
 2314 010142 032777 004000 171070
 2315 010150 001401
 2316 010152 104000
 2317 010154 104402
 2318 010156 000001
 2319 010160 032777 004000 171044
 2320 010166 001001
 2321 010170 104000
 2322 010172 012777 015622 171046
 2323 010200 104401
 2324 010202 000007
 2325 010204 104000
 2326 010206 042777 004000 171016
 2327 010214 042777 000100 171016

```

:*****
: *
: TEST 40
: *
:*****
:*****
TST40:  MOV    #40,TSTNO
        MOV    #TST41,NEXT
        JSR    R5,REE                ;REINIT DP1i
:RETURN HERE IF 8 BITS/CHAR
        MOV    #176,LIMIT            ;7 BIT LIMIT
        BR     .+10                  ;BRANCH AROUND 12 BITS/CHAR LIMIT
:RTURN HERE IF 12 BITS/CHAR
        MOV    #3776,LIMIT           ;11 BITS/CHAR LIMIT
        BIS    #BIT8,@DPRS           ;7/11 BITS PER CHAR
        MOV    #2$,BACK              ;SET UP RCV SERVICE RETURN
        MOV    #FRINT,@DPRIV         ;FALSE INT TEST VECTOR
1$:     MOV    #TV18,@DPTIV          ;SET XMIT INT VECTOR TO SYNC
        MOV    #3,SCNT               ;XMIT 3 SYNCs
        CLOCK
        24                            ;RUN CLOCK
        MOVB   TSYNC,@SYNC           ;LOAD SYNC BUFFER
        BIS    #300,@DPTS            ;XMIT INT ENB
        BIS    #BIT6,@DPRS           ;RCINT ENB
        RXCLK
        3
        BIT    #BIT9,@DPTS           ;SEND REQUEST UP?
        BNE    .+4
        HLT
        CLOCK                          ;REPORT ERROR
        7                            ;RUN CLOCK
        BIT    #BIT11,@DPRS          ;RECEIVER ACTIVE
        BEQ    .+4                    ;NO
        HLT                            ;REPORT ERROR
        CLOCK                          ;RUN CLOCK
        6
        BIT    #BIT11,@DPTS          ;RCV ACTIVE?
        BEQ    .+4                    ;NO
        HLT                            ;REPORT ERROR
        RXCLK
        1
        BIT    #BIT11,@DPRS          ;RCV ACTIVE?
        BNE    .+4                    ;YES
        HLT                            ;REPORT ERROR
        MOV    #SRV5,@DPRIV          ;TEST PASS VECTOR
        CLOCK                          ;RUN CLOCK
        7
        HLT                            ;REPORT ERROR
2$:     BIC    #BIT11,@DPRS           ;CLEAR RCV ACTIVE
        BIC    #BIT6,@DPTS           ;CLEAR XMIT INT ENB
  
```

2328	010222	112777	000026	171006	MOVB	#26,@SYNC	:DUMMY SYNC
2329	010230	032737	040000	177570	BIT	#BIT14,SWR	:TEST FOR SCOPE LOOP
2330	010236	001300			BNE	1\$:BRANCH IF SCOPE
2331	010240	005237	001144		INC	TSYNC	:NEXT SYNC
2332	010244	023737	001160	001144	CMP	LIMIT,TSYNC	:HAVE ALL SYNC'S BEEN TESTED
2333	010252	001272			BNE	1\$	
2334	010254	104400			SCOPE		
2335							
2336							
2337							
2338							
2339							
2340							
2341							
2342							
2343							
2344							
2345							
2346	010256	012737	000041	001072	TST41:	MOV #41,TSTNO	
2347	010264	012737	010562	001062		MOV #TST42,NEXT	
2348	010272	004537	013524			JSR R5,REE	:REINIT DP11
2349	010276	012737	000075	001160		MOV #75,LIMIT	:6 BIT LIMIT
2350	010304	000403				BR +10	:BRANCH AROUND 10 BIT LIMIT
2351							
2352	010306	012737	001775	001160	:RETURN	HERE IF 10 BITS/CHAR	
2353	010314	052777	001000	170710		MOV #1775,LIMIT	:SET UP 10 BITS/CHAR
2354	010322	012737	010520	001174		BIS #BIT9,@DPRS	:6/10 BITS/CHAR
2355	010330	012777	014542	170714		MOV #2\$,BACK	:SET UP RCV SERVICE RETURN
2356	010336	012737	000003	001162	1\$:	MOV #TV18,@DPTIV	:SET XMIT INT VECTOR TO SYNC
2357	010344	042777	000100	170666		MOV #3,SCNT	:XMIT 3 SYNC'S
2358	010352	042777	004000	170652		BIC #BIT6,@DPTS	:CLEAR XMIT INT ENB
2359	010360	112777	000026	170650		BIC #BIT11,@DPRS	:CLEAR ACTIVE
2360	010366	104401				MOVB #26,@SYNC	:CHALGE SYNC
2361	010370	000021				CLOCK 21	:RUN CLOCK
2362	010372	113777	001144	170636		MOVB TSYNC,@SYNC	:LOAD SYNC BUFFER
2363	010400	052777	000100	170624		BIS #BIT6,@DPRS	:RCVINT ENB
2364	010406	052777	000300	170624		BIS #300,@DPTS	:XMIT INT ENB
2365	010414	104402				RXCLK 3	
2366	010416	000003				3	
2367	010420	032777	001000	170612		BIT #BIT9,@DPTS	:SEND REQUEST UP?
2368	010426	001001				BNE +4	
2369	010430	104000				HLT	:REPORT ERROR
2370	010432	104401				CLOCK 6	:RUN CLOCK
2371	010434	000006				6	
2372	010436	032777	004000	170566		BIT #BIT11,@DPRS	:RECEIVER ACTIVE
2373	010444	001401				BEQ +4	:NO
2374	010446	104000				HLT	:REPORT ERROR
2375	010450	104401				CLOCK 5	:RUN CLOCK
2376	010452	000005				5	
2377	010454	032777	004000	170556		BIT #BIT11,@DPTS	:RCV ACTIVE?
2378	010462	001401				BEQ +4	:NO
2379	010464	104000				HLT	:REPORT ERROR
2380	010466	104402				RXCLK 1	
2381	010470	000001				1	
2382	010472	032777	004000	170532		BIT #BIT11,@DPRS	:RCV ACTIVE?
2383	010500	001001				BNE +4	:YES

:SYNCHRONIZATION CHARACTER TEST 6/10 BITS/CHARACTER
 :INTERRUPT ENABLE AND SYNC CHARACTER CHECK

:*****
 : *
 : TEST 41
 : *
 :*****

:*****

2384 010502 104000
2385 010504 012777 015622 170534
2386 010512 104401
2387 010514 000006
2388 010516 104000
2389
2390 010520 042777 004000 170504
2391 010526 112777 000026 170502
2392 010534 032737 040000 177570
2393 010542 001272
2394 010544 005237 001144
2395 010550 023737 001160 001144
2396 010556 001264
2397 010560 104400
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407 010562 012737 000042 001072
2408 010570 012737 010604 001062
2409 010576 004737 014032
2410 010602 104400
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423 010604 012737 000043 001072
2424 010612 012737 010762 001062
2425 010620 005077 170414
2426 010624 005077 170402
2427 010630 105077 170410
2428 010634 012737 177400 001156
2429 010642 012737 000026 001144
2430 010650 113777 001144 170360
2431 010656 012701 015371
2432 010662 012702 072414
2433 010666 012703 004036
2434 010672 004737 013614
2435
2436
2437
2438
2439 010676 012737 177600 001156

```
HLT ;REPORT ERROR
MOV #SRV5,@DPRIV ;TEST VECTOR
CLOCK ;RUN CLOCK
6
HLT ;REPORT ERROR

2$: BIC #BIT11,@DPRS
MOVB #26,@SYNC
BIT #BIT14,SWR ;TEST FOR SCOPE LOOP
BNE 1$ ;BRANCH IF SCOPE
INC TSYNC ;NEXT SYNC
CMP LIMIT,TSYNC ;HAVE ALL SYNC'S BEEN TESTED
BNE 1$
SCOPE

;INTERRUPT DRIVEN SEQUENTIAL DATA TEST

:*****
: *
: TEST 42
: *
:*****
:*****
TST42: MOV #42,TSTNO
MOV #TST43,NEXT
JSR PC,SEQ.DATA ;DC THE TEST.
SCOPE

:*****

;RANDOM DATA RANDOM IDLE

:*****
: *
: TEST 43
: *
:*****
:*****
TST43: MOV #43,TSTNO
MOV #TST44,NEXT
CLR @DPTS
CLR @DPRS
CLRB @SEXT ;CLEAR SYNC EXTENTION
MOV #177400,CHLEN ;SET CHAR LENGTH TO 8 BITS
MOV #26,TSYNC ;SYNC = 26
MOVB TSYNC,@SYNC ;LOAD SYNC BUFFER
MOV #15371,R1 ;PRIME RANDOM # GEN
MOV #72414,R2
MOV #4036,R3
JSR PC,A2ND

;REPEAT PREVIOUS TEST AT 7 BITS/CHAR

MOV #177600,CHLEN ;SET CHAR LENGTH TO 7 BITS
```

2440 010704 052777 000400 170320
 2441 010712 042777 004000 170312
 2442 010720 004737 013614
 2443
 2444
 2445
 2446 010724 012737 177700 001156
 2447 010732 042777 000400 170272
 2448 010740 052777 001000 170264
 2449 010746 042777 004000 170256
 2450 010754 004737 013614
 2451 010760 104400

BIS #BIT8,@DPRS
 BIC #BIT11,@DPRS
 JSR PC,A2ND

:7 BITS/CHAR
 :CLEAR ACTIVE

:REPEAT PREVIOUS TEST AT 6 BITS/CHAR

MOV #177700,CHLEN
 BIC #BIT8,@DPRS
 BIS #BIT9,@DPRS
 BIC #BIT11,@DPRS
 JSR PC,A2ND
 SCOPE

:SET CHAR LENGTH TO 6 BITS
 :SET MODE TO 6 BITS/CHAR
 :CLEAR ACTIVE

2452
 2453
 2454
 2455
 2456
 2457
 2458
 2459
 2460
 2461
 2462
 2463
 2464
 2465
 2466
 2467 010762 012737 000044 C01072
 2468 010770 012737 011156 001062
 2469 010776 012737 000240 177775
 2470 011004 005077 170230
 2471 011010 005077 170216
 2472 011014 104401
 2473 011016 000010
 2474 011020 012777 014532 170224
 2475 011026 012777 014536 170212
 2476 011034 052777 000040 170176
 2477 011042 013777 001152 170172
 2478 011050 104401
 2479 011052 000003
 2480 011054 032777 001000 170156
 2481 011062 001001
 2482 011064 104000
 2483 011066 032777 002000 170144
 2484 011074 001001
 2485 011076 104000
 2486 011100 032777 010000 170132
 2487 011106 001001
 2488 011110 104000
 2489 011112 104401
 2490 011114 000012
 2491 011116 032777 001000 170114
 2492 011124 001401
 2493 011126 104000
 2494 011130 032777 002000 170102
 2495 011136 001401
 2496 011140 104000
 2497 011142 032777 010000 170070
 2498 011150 001401
 2499 011152 104000
 2500 011154 104400

:DB255 TEST CONNECTOR DISCRETE EVENTS TEST

:VERIFY 'SEND REQUEST' RAISES 'CLEAR-TO-SEND' AND 'MODEM READY'
 :WHEN TRANSMITTER BUFFER IS LOADED

```

:*****
:      *
:  TEST 44
:      *
:*****
  
```

```

*****
TST44:  MOV    #44,TSTNO
        MOV    #TST45,NEXT
        MOV    #240,PS          :PRIORITY=5
        CLR    @DPTS            :CLR XMIT STATUS
        CLR    @DPRS            :CLR RCV STATUS
        CLOCK          :RUN CLOCK
        10
        MOV    #FTINT,@DPTIV    :XMIT ERROR TRAP VECTOR
        MOV    #FRINT,@DPRIV    :RCV ERROR TRAP VECTOR
        BIS    #BITS,@DPTS      :STATUS+RDY INT ENB
        MOV    TDATA,@DPTB      :LOAD BUFFER
        CLOCK          :RUN CLOCK
        3
        BIT    #BIT9,@DPTS      :'SEND REQUEST' ON
        BNE    .+4              :YES
        HLT                    :REPORT ERROR
        BIT    #BIT10,@DPTS     :'CLEAR-TO-SEND' UP?
        BNE    .+4              :YES
        HLT                    :REPORT ERROR
        BIT    #BIT12,@DPTS     :'MODEM READY' UP
        BNE    .+4              :YES
        HLT                    :REPORT ERROR
        CLOCK          :RUN CLOCK
        12
        BIT    #BIT9,@DPTS      :'SEND REQUEST' DOWN
        BEQ    .+4              :YES
        HLT                    :REPORT ERROR
        BIT    #BIT10,@DPTS     :'CLEAR-TO-SEND' DOWN
        BEQ    .+4              :YES
        HLT                    :REPORT ERROR
        BIT    #BIT12,@DPTS     :'MODEM READY' DOWN
        BEQ    .+4              :YES
        HLT                    :REPORT ERROR
        SCOPE
  
```

2501
2502
2503
2504
2505
2506
2507
2508
2509 011156 012737 000045 001072
2510 011164 012737 011524 001062
2511 011172 012737 000240 177776
2512 011200 005077 170034
2513 011204 005077 170022
2514 011210 012777 014532 170034
2515 011216 012777 014536 170022
2516 011224 052777 000400 170000
2517 011232 052777 000040 170000
2518 011240 013777 001152 167774
2519 011246 104401
2520 011250 000003
2521 011252 032777 001000 167760
2522 011260 001001
2523 011262 104000
2524 011264 032777 002000 167746
2525 011272 001001
2526 011274 104000
2527 011276 032777 010000 167734
2528 011304 001001
2529 011306 104000
2530
2531 011310 104401
2532 011312 000011
2533 011314 032777 001000 167716
2534 011322 001401
2535 011324 104000
2536 011326 032777 002000 167704
2537 011334 001401
2538 011336 104000
2539 011340 032777 010000 167672
2540 011346 001401
2541 011350 104000
2542 011352 104400
2543
2544
2545
2546 011354 012737 000240 177776
2547 011362 005077 167652
2548 011366 005077 167640
2549 011372 012777 014532 167652
2550 011400 012777 014536 167640
2551 011406 052777 001000 167616
2552 011414 052777 000040 167616
2553 011422 013777 001152 167612
2554 011430 104401
2555 011432 000003
2556 011434 032777 001000 167576

;PERFORM PREVIOUS TEST AT 7/11 BITS/CHARACTER

: TEST 45

TST45: MOV #45,TSTNO
MOV #TST46,NEXT
MOV #240,PS ;PRIORITY = 5
CLR @DPTS ;CLR XMIT STATUS
CLR @DPRS ;CLR RCV STATUS
MOV #FTINT,@DPTIV ;XMIT ERROR TRAP VECTOR
MOV #FRINT,@DPRIV ;RCV ERROR TRAP VECTOR
BIS #BIT8,@DPRS ;7/11 BITS/CHARACTER
BIS #BIT5,@DPTS ;STATUS+RDY INT ENB
MOV TDATA,@DPTB ;LOAD BUFFER
CLOCK ;RUN CLOCK
3
BIT #BIT9,@DPTS ;'SEND REQUEST' ON
BNE .+4 ;YES
HLT ;REPORT ERROR
BIT #BIT10,@DPTS ;'CLEAR-TO-SEND' UP?
BNE .+4 ;YES
HLT ;REPORT ERROR
BIT #BIT12,@DPTS ;'MODEM READY' UP
BNE .+4 ;YES
HLT ;REPORT ERROR
CLOCK ;RUN CLOCK
11
BIT #BIT9,@DPTS ;'SEND REQUEST' DOWN
BEQ .+4 ;YES
HLT ;REPORT ERROR
BIT #BIT10,@DPTS ;'CLEAR-TO-SEND' DOWN
BEQ .+4 ;YES
HLT ;REPORT ERROR
BIT #BIT12,@DPTS ;'MODEM READY' DOWN
BEQ .+4 ;YES
HLT ;REPORT ERROR
SCOPE

;PERFORM PREVIOUS TEST AT 6/10 BITS/CHARACTER

MOV #240,PS ;PRIORITY = 5
CLR @DPTS ;CLR XMIT STATUS
CLR @DPRS ;CLR RCV STATUS
MOV #FTINT,@DPTIV ;XMIT ERROR TRAP VECTOR
MOV #FRINT,@DPRIV ;RCV ERROR TRAP VECTOR
BIS #BIT9,@DPRS ;6 BITS/CHARACTER
BIS #BIT5,@DPTS ;STATUS+RDY INT ENB
MOV TDATA,@DPTB ;LOAD BUFFER
CLOCK ;RUN CLOCK
3
BIT #BIT9,@DPTS ;'SEND REQUEST' ON

2557 011442 001001
 2558 011444 104000
 2559 011446 032777 010000 167564
 2560 011454 001001
 2561 011456 104000
 2562 011460 104401
 2563 011462 000010
 2564 011464 032777 001000 167546
 2565 011472 001401
 2566 011474 104000
 2567 011476 032777 002000 167534
 2568 011504 001401
 2569 011506 104000
 2570 011510 032777 010000 167522
 2571 011516 001401
 2572 011520 104000
 2573 011522 104400

BNE .+4 :YES
 HLT :REPORT ERROR
 BIT #BIT12,@DPTS :'MODEM READY' UP
 BNE .+4 :YES
 HLT :REPORT ERROR
 CLOCK :RUN CLOCK
 10
 BIT #BIT9,@DPTS :'SEND REQUEST' DOWN
 BEQ .+4 :YES
 HLT :REPORT ERROR
 BIT #BIT10,@DPTS :'CLEAR-TO-SEND' DOWN
 BEQ .+4 :YES
 HLT :REPORT ERROR
 BIT #BIT12,@DPTS :'MODEM READY' DOWN
 BEQ .+4 :YES
 HLT :REPORT ERROR
 SCOPE

:DB25S TEST CONNECT CONNECTOR TEST

:TERMINAL READY

:TERM RDY-RAISE 'CARRIER' AND 'RING FLAG', NO INT ENB

:*****

:*

: TEST 46

:*

:*****

:*****

2588 011524 012737 000046 001072
 2589 011532 012737 012022 001062
 2590 011540 012737 000240 177776
 2591 011546 005077 167466
 2592 011552 012777 014532 167460
 2593 011560 012777 000001 167452
 2594 011566 104403 001500
 2595 011572 032777 000001 167440
 2596 011600 001001
 2597 011602 104000
 2598 011604 032777 004000 167426
 2599 011612 001001
 2600 011614 104000
 2601 011616 032777 020000 167414
 2602 011624 001001
 2603 011626 104000

TST46: MOV #46,TSTNO
 MOV #TST47,NEXT
 MOV #240,PS :PRIORITY = 5
 CLR @DPTS :CLR XMIT STATUS
 MOV #FTINT,@DPTS :ERROR TRAP VECTOR
 MOV #BIT0,@DPTS :TERMINAL READY
 DELAY ,1500 :15MS PROPOGATION DELAY
 BIT #BIT0,@DPTS :TERMINAL READY
 BNE .+4 :YES
 HLT :REPORT ERROR
 BIT #BIT11,@DPTS :CARRIER
 BNE .+4 :YES
 HLT :REPORT ERROR
 BIT #BIT13,@DPTS :RING FLAG UP
 BNE .+4 :YES
 HLT :REPORT ERROR

:CLEAR TERMINAL READY

2604
 2605
 2606
 2607 011630 042777 000001 167402
 2608 011636 104403 001500
 2609 011642 032777 000001 167370
 2610 011650 001401
 2611 011652 104000
 2612 011654 032777 004000 167356

BIC #BIT0,@DPTS :CLEAR TERMINAL READY
 DELAY ,1500
 BIT #BIT0,@DPTS :TERM RDY DOWN
 BEQ .+4 :YES
 HLT :REPORT ERROR
 BIT #BIT11,@DPTS :CARRIER DOWN

2613	011662	001401			BEQ	.+4	:YES
2614	011664	104000			HLT		:REPORT ERROR
2615	011666	032777	100000	167344	BIT	#BIT15,@DPTS	: 'CARRIER DOWN' FLAG UP
2616	011674	001001			BNE	.+4	:YES
2617	011676	104000			HLT		:REPORT ERROR
2618	011700	042777	100000	167332	BIC	#BIT15,@DPTS	:CLEAR DOWN FLAG
2619							
2620							
2621	011706	012777	000240	167340			:VERIFY THAT 'RING' AND 'CARRIER DOWN' INTERRUPT
2622	011714	005077	167320		MOV	#240,@DPTP	: INTERRUPT PRIORITY 5.
2623	011720	005077	167306		RCD: CLR	@DPTS	:CLEAR XIMT STATUS
2624	011724	012777	014406	167320	CLR	@DPRS	:CLEAR RCV STATUS
2625	011732	012737	000200	177776	MOV	#TV24,@DPTIV	:TEST PASS VECTOR
2626	011740	012777	000041	167272	MOV	#200,PS	:PRIORITY = 4
2627	011746	104403	001500		MOV	#BIT5+BIT0,@DPTS	:STATUS INTERRUPT ENABLE
2628	011752	104000			DELAY	,1500	:15.75 MS DELAY
2629					HLT		:REPORT ERROR
2630							
2631	011754	012777	014460	167270			:VERIFY 'CARRIER DOWN' RAISES INTERRUPT
2632	011762	012737	012020	001174	RCD1: MOV	#TV25,@DPTIV	:NEXT TEST VECTOR
2633	011770	012737	000200	177776	MOV	#1\$,BACK	
2634	011776	042777	000001	167234	MOV	#200,PS	
2635	012004	104403	001500		BIC	#BIT0,@DPTS	:CLEAR TERM. RDY. SHOULD SET 'CARRIER DOWN'.
2636	012010	104000			DELAY	,1500	:15.75MS DELAY
2637	012012	042777	100000	167220	HLT		:REPORT ERROR
2638	012020	104400			BIC	#BIT15,@DPTS	:CLEAR 'CARRIER DOWN'
2639					1\$: SCOPE		
2640							

2641
 2642
 2643
 2644
 2645
 2646 012022
 2647
 2648
 2649
 2650
 2651
 2652
 2653
 2654
 2655
 2656
 2657
 2658
 2659 012022 012737 000047 001072
 2660 012030 012737 016002 001062
 2661 012036 032737 000001 001134
 2662 012044 001002
 2663 012046 000137 016002
 2664 012052 005077 167162
 2665 012056 005077 167150
 2666 012062 005002
 2667 012064 012700 001300
 2668 012070 010277 167146
 2669 012074 017701 167140
 2670 012100 042701 164777
 2671 012104 022701 013000
 2672 012110 001403
 2673 012112 005202
 2674 012114 001367
 2675 012116 104001
 2676
 2677 012120 005002
 2678 012122 005202
 2679 012124 001376
 2680 012126 032777 013000 167104
 2681 012134 001401
 2682 012136 104000
 2683
 2684
 2685
 2686
 2687
 2688
 2689 012140 012777 000001 167072
 2690 012146 104403 001500
 2691 012152 017701 167062
 2692 012156 042701 153776
 2693 012162 022701 024001
 2694 012166 001401
 2695 012170 104001
 2696 012172 042777 000001 167040

:CABLE TEST WITH OUT EXERCISING THE SOFTWARE CLOCK.
 :IN THIS TEST FUNCTIONS OF THE CABLE WILL BE
 :TESTED WITHOUT THE SOFTWARE CLOCK. THE CLOCK MUST BE
 :SUPPLIED BY THE DP11.

PART3:
 : DF11-L TEST!!!
 :VERIFY THAT LOADING THE TRANSMITTER BUFFER
 :BRINGS UP 'REQUEST TO SEND' WHICH IN TURN WILL
 :BRING UP 'CLEAR TO SEND' AND 'MODEM READY'.
 :VERIFY ALSO THAT THEY GO AWAY.

:*****
 : *
 : TEST 47
 : *
 :*****

```

*****
TST47:  MOV    #47,TSTNO
        MOV    #.EOP,NEXT
        BIT    #BIT0,SAVSR1
        BNE   .+6
        JMP    .EOP
        CLR   @DPTS           ;CLEAR THE TX STATUS
        CLR   @DPRS           ;CLEAR THE RX STATUS.
        CLR   R2              ;SET TIME OUT
        MOV   #1300,R0
        MOV   R2,@DPTB        ;LOAD THE TX BUFFER
1$:     MOV   @DPTS,R1
        BIC   #164777,R1
        CMP   #13000,R1
        BEQ   2$
        INC   R2              ;UPDATE DELAY
        BNE   1$              ;IS IT DONE?
        HLT   1               ;ERROR REQUEST TO SEND,CLEAR TO
                                ;SEND AND MODEM READY NOT UP.
2$:     CLR   R2              ;SET FOR TIME OUT
        INC   R2              ;DELAY
        BNE   .-2
        BIT   #13000,@DPTS    ;ARE THEY GONE?
        BEQ   .+4
        HLT
  
```

:VERIFY THAT THE SETTING OF 'TERMINAL READY' BRINGS
 :UP 'RING' AND 'CARRIER DOWN' ALSO VERIFY THAT
 :CLEARING 'TERMINAL READY' BRINGS DOWN 'RING'
 :AND 'CARRIER DOWN'.

```

MOV    #BIT0,@DPTS
DELAY  ,1500
MOV    @DPTS,R1
BIC    #153776,R1
CMP    #24001,R1
BEQ    .+4
HLT    1
BIC    #BIT0,@DPTS
  
```

2697 012200 104403 001500
2698 012204 022777 120000 167026
2699 012212 001401
2700 012214 104000
2701
2702
2703
2704
2705
2706 012216 005037 001102
2707 012222 005037 001104
2708 012226 004737 014032
2709 012232 104400

DELAY ,1500
CMP #120000,@DPTS
BEQ .+4
HLT

;NOW TEST THAT DATA CAN BE TRANSFERED.
;A BINARY COUNT PATTERN WILL BE TRANSMITTED AND RECEIVED
;WITH OUT THE USE OF THE SOFTWARE CLOCK.

CLR TEMP1
CLR TEMP2
JSR PC,SEQ.DATA
SCOPE


```

2710
2711
2712
2713
2714
2715
2716
2717 012234 005037 001076
2718 012240 013737 177570 001134
2719 012246 013700 177570
2720 012252 000000
2721
2722
2723 012254 013737 177570 001164
2724 012262 013700 177570
2725 012266 000000
2726 012270 005001
2727 012272 113701 001165
2728 012276 000405
2729 012300 013701 001146
2730 012304 005037 001134
2731 012310 006301
2732 012312 006201
2733 012314 010137 001146
2734 012320 006301
2735 012322 006301
2736 012324 006301
2737 012326 005777 166726
2738 012332 000240
2739 012334 105737 001134
2740 012340 100403
2741 012342 012737 012454 000004
2742 012350 013702 001260
2743 012354 160102
2744 012356 012703 001232
2745 012362 010223
2746 012364 005722
2747 012366 010223
2748 012370 005202
2749 012372 010223
2750 012374 005202
2751 012376 010223
2752 012400 005722
2753 012402 010223
2754 012404 005202
2755 012406 010223
2756 012410 013702 001164
2757 012414 042702 177000
2758 012420 105737 001134
2759 012424 100402
2760 012426 013702 001256
2761 012432 060102
2762 012434 010223
2763 012436 005722
2764 012440 010223
2765 012442 005722

;LINE.N SUBROUTINE TO FETCH THE LINE
;NUMBER AND FIRST DP11 VECTOR ADDRESS FROM
;THE CONSOL SWITCHES
;SW0-SW8=VECTOR ADDRESS OF FIRST DP11
;SW9-SW15=LINE NUMBER OF DP11 SELECTED FOR TEST

LINE.N: CLR      ERRCNT      ;CLEAR ERROR COUNT
        MOV      SWR,SAVSR1  ;SAVE CONSOL SWITCH SETTINGS
        MOV      SWR,R0
        HALT
        ;SET SWR TO LINE NUMBER
        ;LOW BYTE = FIRST DP VECTOR
        ;HIGH BYTE = LINE NUMBER(8)
        ;SAVE CONSOL SWITCHES

        MOV      SWR,SAVSR2
        MOV      SWR,R0
        HALT
        CLR      R1
        MOV      SAVSR2+1,R1 ;SAVE LINE NUMBER
        BR      XTAG
LINE.X: MOV      XLINEX,R1
        CLR      SAVSR1
        ASL      R1
XTAG:   ASR      R1          ;CLEAR LSB
        MOV      R1,XLINEX
        ASL      R1
        ASL      R1
        ASL      R1
        TST      @BASCSR    ;DOES LINE 0 REALLY EXIST??
        NOP
        TSTB    SAVSR1
        BMI     .+10
        MOV     #3$,@#4
        MOV     BASCSR,R2
        SUB     R1,R2
        MOV     #DPRS,R3
        MOV     R2,(R3)+
        TST    (R2)+
        MOV     R2,(R3)+
        INC    R2
        MOV     R2,(R3)+
        INC    R2
        MOV     R2,(R3)+
        INC    R2
        MOV     R2,(R3)+
        TST    (R2)+
        MOV     R2,(R3)+
        INC    R2
        MOV     R2,(R3)+
        MOV     SAVSR2,R2
        BIC    #177000,R2
        TSTB  SAVSR1
        BMI     2$
        MOV     BASVEC,R2
2$:    ADD     R1,R2
        MOV     R2,(R3)+
        TST    (R2)+
        MOV     R2,(R3)+
        TST    (R2)+
  
```

```

2766 012444 010223          MOV      R2,(R3)+          ;LOAD XMIT VECTOR ADRS
2767 012446 005722          TST      (R2)+            ;INC TO NEXT VECTOR
2768 012450 010213          MOV      R2,(R3)          ;LOAD XMIT PRIORITY ADRS
2769 012452 000414          BR       5$
2770 012454 005037 001146    3$:     CLR      XLINEX
2771 012460 013701 000042    MOV      @#42,R1
2772 012464 001405          BEQ     4$
2773 012466 000005          RESET
2774          012470          LOGICAL=
2775 012470 004711          JSR     PC,(R1)
2776 012472 000240          NOP
2777 012474 000240          NOP
2778 012476 000240          NOP
2779 012500 022626          4$:     POP     SP
2780 012502 000676          BR      LINE.X
2781 012504 012737 000006 000004    5$:     MOV      #6,@#4
2782 012512 000207          RTS     PC
2783          ;*****CLRVEC*****
2784
2785          ;CLRVEC,ROUTINE TO FILL COMMUNICATION VECTOR AREA WITH .+2,HALT
2786
2787 012514 012702 000300    CLRVEC: MOV      #300,R2          ;R2 COMM VECTOR AREA ADRS
2788 012520 012701 000302    MOV      #302,R1          ;INIT R1 WITH ADRS OF HALT
2789 012524 010122          1$:     MOV      R1,(R2)+      ;MOV .+2 TO PC
2790 012526 005022          CLR     (R2)+            ;MOV HALT TO PC
2791 012530 022121          CMP     (R1)+,(R1)+      ;INC TO NEXT VECTOR AREA
2792 012532 022701 000776    CMP     #776,R1          ;END OF VECTOR AREA
2793 012536 001372          BNE    1$                ;NO
2794 012540 000207          RTS     PC                ;RETURN
2795
2796
2797
2798          ;BITSR,ROUTINE TO TEST READ WRITE BITS OF STATUS
2799          ;THIS ROUTINE VERIFIES THAT EACH READ/WRITE BIT
2800          ;CAN BE SET AND CLEARED
2801          ;EXAMINE LOCATIONS
2802          ;BITS: FOR BIT UNDER TEST
2803          ;REG: FOR REGISTER UNDER TEST
2804
2805 012542 010537 001110    BITST:  MOV      R5,TEMP4
2806 012546 012537 013106    MOV      (R5)+,BITS      ;SAVE BIT NUMBER
2807 012552 053777 013106 000330    BIS     BITS,@REG        ;SET BIT
2808 012560 053700 013106    BIS     BITS,R0
2809 012564 017701 000320    MOV     @REG,R1
2810 012570 033777 013106 000312    BIT     BITS,@REG        ;IS BIT SET?
2811 012576 001001          BNE    +4                ;YES
2812 012600 104002          HLT    2                  ;REPORT ERROR
2813 012602 043777 013106 000300    BIC     BITS,@REG        ;CLEAR BIT
2814 012610 043700 013106    BIC     BITS,R0
2815 012614 017701 000270    MOV     @REG,R1
2816 012620 033777 013106 000262    BIT     BITS,@REG        ;IS BIT CLEARED
2817 012626 001401          BEQ    +4                ;YES
2818 012630 104002          HLT    2                  ;REPORT ERROR
2819 012632 053777 013106 000250    BIS     BITS,@REG        ;SET BIT
2820 012640 053700 013106    BIS     BITS,R0
2821 012644 017701 000240    MOV     @REG,R1

```

```

2822 012650 033777 013106 000232
2823 012656 001001
2824 012660 104002
2825 012662 005077 000222
2826 012666 00500C
2827 012670 017701 000214
2828 012674 033777 013106 000206
2829 012702 001401
2830 012704 104002
2831 012706 052777 000004 166316
2832 012714 052700 000004
2833 012720 000205
2834
2835
2836
2837
2838
2839
2840
2841 012722 011637 001110
2842 012726 017637 000000 013106
2843 012734 062716 000002
2844 012740 053777 013106 000142
2845 012746 053700 013106
2846 012752 017701 000132
2847 012756 023777 013106 000124
2848 012764 001401
2849 012766 104002
2850 012770 043777 013106 000112
2851 012776 053700 013106
2852 013002 000002
2853
2854
2855
2856 013004 011637 001110
2857 013010 017637 000000 013106
2858 013016 062716 000002
2859 013022 043777 013106 000060
2860 013030 043737 013106 001136
2861 013036 012700 013000
2862 013042 005300
2863 013044 001376
2864 013046 013700 001136
2865 013052 017701 000032
2866 013056 023777 001136 000024
2867 013064 001401
2868 013066 104002
2869 013070 053777 013106 000012
2870 013076 053737 013106 001136
2871 013104 000002
2872
2873 013106 000000
2874 013110 000000
2875
2876
2877
  
```

```

BIT BITS,@REG ;IS BIT SET
BNE .+4 ;YES
HLT 2 ;REPORT ERROR
CLR @REG ;CLEAR REG
CLR R0
MOV @REG,R1
BIT BITS,@REG ;IS BIT CLEARED
BEQ .+4 ;YES
HLT 2 ;REPORT ERROR
BIS #BIT2,@DPRS ;KEEP CLOCK HUMMING
BIS #BIT2,R0
RTS R5
  
```

```

;VALID,ROUTINE TO TEST FOR ANY INTERACTION BETWEEN BITS
;THIS ROUTINE CHECKS THAT WHEN EXECUTING A BIT SET INSTRUCTION
;ONLY THE SPECIFIED BIT IS SET
  
```

```

.VALID: MOV (SP),TEMP4
MOV @ (SP),BITS ;FETCH BIT NUMBER
ADD #2,(SP)
BIS BITS,@REG ;SET BIT
BIS BITS,R0
MOV @REG,R1
CMP BITS,@REG ;WAS ONLY THAT BIT SET?
BEQ .+4 ;YES
HLT 2 ;REPORT ERROR
BIC BITS,@REG ;RESTORE REG
BIS BITS,R0
RTI ;RETURN
  
```

```

;CLEAR,ROUTINE TO TEST THAT BIC ONLY CLEARS SPECIFIED BIT
  
```

```

.CLEAR: MOV (SP),TEMP4
MOV @ (SP),BITS ;FETCH BIT NUMBER
ADD #2,(SP)
BIC BITS,@REG ;CLEAR BIT
BIC BITS,TMPDAT ;CLEAR MASK
MOV #13000,R0
DEC R0
BNE .-2
MOV TMPDAT,R0
MOV @REG,R1
CMP TMPDAT,@REG ;WERE ANY OTHER BITS CLEARED
BEQ .+4 ;NO
HLT 2 ;REPORT ERROR
BIS BITS,@REG ;RESTORE REG
BIS BITS,TMPDAT ;RESTORE MASK
RTI ;RETURN
  
```

```

BITS: 0
REG: 0
  
```

```

;COMMON DATA AND IDLE SUBROUTINE
  
```

```

2878 013112 012701 015371      AND:  MOV    #15371,R1      ;PRIME RANDOM # GEN
2879 013116 012702 072414      MOV    #72414,R2      ;.. ..
2880 013122 012703 004036      MOV    #4036,R3       ;.. ..
2881 013126 042777 004300 166076  BIC    #4300,@DPRS    ;RCV INT ENB, RCV ACTIVE
2882 013134 042777 160342 166076  BIC    #160342,@DPTS  ;INT ENBS IDLE SYNC, ERRORS
2883 013142 012737 015610 001172  MOV    #BOTTOM,RP    ;SET UP RCV POINTER
2884 013150 013737 001172 001170  MOV    RP,TP         ;SET UP XMIT POINTER
2885 013156 012777 015224 166062  MOV    #RRRR,@DPRIV  ;RCV INT VECTOR
2886 013164 012777 015076 166060  MOV    #RRRT,@DPTIV  ;XMIT INT VECTOR
2887 013172 012737 000002 001162  MOV    #2,SCNT       ;SYNC COUNT = 2
2888 013200 110237 001152      MOVB   R2,TDATA      ;RANDOM DATA
2889 013204 052777 000100 166020  BIS    #100,@DPRS    ;RCV INT ENB
2890 013212 013777 001144 166022  1$:  MOV    TSYNC,@DPTB  ;LOAD BUFFER
2891 013220 052777 000340 166012  BIS    #340,@DPTS   ;XMIT DONE, INT ENB, STATUS ENB
2892 013226 010137 001166      MOV    R1,TIME      ;'ON' STALL
2893 013232 005337 001166      2$:  DEC    TIME        ;0.6 SEC MAX
2894 013236 001375      BNE    2$
2895 013240 042777 000140 165772  BIC    #140,@DPTS    ;TURN OFF INT ENB
2896 013246 052777 000002 165764  BIS    #BIT1,@DPTS   ;IDLE SYNC
2897 013254 004537 013342      JSR    R5,RNUM       ;GENERATE 'STALL' TIME
2898 013260 010137 001166      MOV    R1,TIME      ;FETCH RANDON STALL TIME
2899 013264 005337 001166      3$:  DEC    TIME        ;COUNT IDLE TIME
2900 013270 001375      BNE    3$           ;TIME OUT?
2901 013272 004537 013342      JSR    R5,RNUM       ;GENERATE 'ON' TIME + SYNC
2902 013276 042777 000002 165734  BIC    #BIT1,@DPTS   ;CLEAR IDLE
2903 013304 000240      NOP
2904 013306 042777 000100 165716  BIC    #100,@DPRS    ;CLEAR RCV INT ENB
2905 013314 000207      RTS    PC
2906
2907
2908
2909      ;SGEN,ROUTINE TO GENERATE A UNIQUE SYNC CHARACTER
2910 013316 112777 000026 165712  SGEN:  MOVB   #26,@SYNC  ;SET UP FILLER SYNC
2911 013324 104403 005670      DELAY  #3000         ;DELAY 10 CHAR
2912 013330 004537 013342      JSR    R5,RNUM       ;RANDOM #
2913 013334 110137 001144      MOVB   R1,TSYNC     ;LOAD SYNC
2914 013340 000207      RTS    PC
2915
2916
2917
2918      ;RNUM, PSEUDO RANDOM NUMBER GENERATOR
2919
2920 013342 032737 040000 177570  RNUM:  BIT    #BIT14,SWR ;TEST FOR SCOPE LOOP
2921 013350 001010      BNE    1$           ;EXIT IF SCOPE
2922 013352 060201      ADD    R2,R1
2923 013354 005501      ADC    R1
2924 013356 060102      ADD    R1,R2
2925 013360 005502      ADC    R2
2926 013362 060302      ADD    R3,R2
2927 013364 005502      ADC    R2
2928 013366 060203      ADD    R2,R3
2929 013370 005503      ADC    R3
2930 013372 000205      1$:  RTS    R5
2931
2932
2933

```

```

2934                                     ;CLOCK,SUBROUTINE TO RUN SOFTWARE CLOCK
2935                                     ;NUMBER OF CYCLES IS FETCHED CALL
2936
2937 013374 005037 001102                .CLOCK: CLR      TEMP1
2938 013400 032737 000400 001134        BIT      #BIT8,SAVSR1                ;12/8 BITS/CHAR
2939 013406 001412                       BEQ      .RXCLK                      ;BRANCH IF 8 BITS/CHAR
2940 013410 052777 002000 165614        BIS      #BIT10,@DPRS                ;SELECT 12 BIT MODE
2941 013416 052737 000400 001160        BIS      #BIT8,LIMIT                 ;9 BIT SYNC
2942 013424 062737 000004 001102        ADD      #4,TEMP1                    ;INCREASE CLOCK COUNT
2943 013432 000402                       BR      .+6
2944 013434 005037 001102                .RXCLK: CLR      TEMP1
2945 013440 005037 177776                CLR      PS
2946 013444 067637 000000 001102        ADD      @(SP),TEMP1
2947 013452 062716 000002                ADD      #2,(SP)
2948 013456 052777 000010 165554        1$:    BIS      #BIT3,@DPTS           ;SET CLOCK HIGH
2949 013464 013705 013522                MOV      FREQ,R5                     ;SET UP DELAY COUNT
2950
2951 013470 005305                       DEL=.
2952 013472 001376                       DEC      R5                           ;DECREMENT COUNT
2953 013474 042777 000010 165536        BNE     DEL                          ;BRANCH IF NO TIMEOUT
2954 013502 013705 013522                BIC      #BIT3,@DPTS                 ;SET CLOCK LOW
2955
2956 013506 005305                       DEL=.
2957 013510 001376                       DEC      R5                           ;DEC COUNT
2958 013512 005337 001102                BNE     DEL                          ;BRANCH IF NO TIMEOUT
2959 013516 001357                       DEC      TEMP1
2960 013520 000002                       BNE     1$
2961 013522 000030                       RTI
2962
2963                                     FREQ: 30                             ;NORMAL 12.8 US DELAY
2964                                                                         ;PATCH FOR 50 FT CABEL
2965
2966                                     ;REE, SUBROUTINE TO REINITIALIZE DP11 FOR NEXT TEST
2967 013524 012737 000200 177776        REE:  MOV      #200,PS                 ;SET PRIORITY TO 4
2968 013532 005077 165502                CLR      @DPTS                       ;CLEAR XMIT STATUS
2969 013536 005077 165470                CLR      @DPRS                       ;CLEAR RCV STATUS
2970 013542 105077 165476                CLR     @SEXT                         ;CLEAR SYNC EXTENTION
2971 013546 012737 000001 001144        MOV      #1,TSYNC                    ;INIT TEST SYNC
2972 013554 012777 014536 165464        MOV      #FRINT,@DPRIV               ;SET UP RVI INT VECTOR
2973 013562 012777 014542 165462        MOV      #TV18,@DPTIV               ;SET XMIT INT VECTOR TO SYNC
2974 013570 032737 000400 001134        BIT      #BIT8,SAVSR1                ;TEST FOR 8/12 BITS/CHAR
2975 013576 001405                       BEQ      1$                           ;EXIT IF 8 BITS
2976 013600 062705 000010                ADD      #10,R5                      ;SET RETURN ADRS FOR 12 BIT LIMIT
2977 013604 052777 002000 165420        BIS      #BIT10,@DPRS                ;SET 12 BIT/CHAR MODE
2978 013612 000205                       RTS      R5                           ;RETURN
2979
2980                                     ;COMMON DATA AND IDLE SUBROUTINE
2981
2982 013614 042777 004300 165410        A2ND: BIC      #4300,@DPRS              ;RCV INT ENB, RCV ACTIVE
2983 013622 042777 160342 165410        BIC      #160342,@DPTS               ;INT ENBS IDLE SYNC, ERRORS
2984 013630 012737 015610 001172        MOV      #BOTTOM,RP                 ;RP = BOTTOM OF TUMBLE TABLE
2985 013636 013737 001172 001170        MOV      RP,TP                       ;SET UP TRANSMIT POINTER
2986 013644 012777 015224 165374        MOV      #RRRR,@DPRIV                ;RCV INT VECTOR
2987 013652 012777 015076 165372        MOV      #RRRT,@DPTIV               ;XMIT INT VECTOR
2988 013660 012737 000002 001162        MCV      #2,SCNT                     ;SYNC COUNT = 2
2989 013666 110237 001152                MOV     R2,TDATA                     ;RANDOM DATA
  
```

```

2990 013672 013777 001144 165342      MOV    TSYNC,@DPTB      ;LOAD BUFFER
2991 013700 052777 000100 165324      BIS    #100,@DPRS      ;RCV INT ENB
2992 013706 052777 000140 165324      BIS    #140,@DPTS      ;XMIT INT ENB
2993 013714 010137 001166              MOV    R1,TIME         ;'ON' STALL
2994 013720 104402              1$:  RXCLK
2995 013722 000001              1
2996 013724 005337 001166      DEC    TIME            ;0.6 SEC AVERAGE
2997 013730 001373              BNE    1$
2998 013732 042777 000140 165300      BIC    #140,@DPTS      ;TURN OFF INT ENB
2999 013740 052777 000002 165272      BIS    #BIT1,@DPTS     ;IDLE SYNC
3000 013746 004537 013342      JSR    R5,RNUM         ;GENERATE 'STALL' TIME
3001 013752 010137 001166      MOV    R1,TIME
3002 013756 104402              2$:  RXCLK
3003 013760 000001              1
3004 013762 005337 001166      DEC    TIME            ;COUNT IDLE TIME
3005 013766 001373              BNE    2$              ;TIME OUT?
3006 013770 004537 013342      JSR    R5,RNUM         ;GENERATE 'ON' TIME + SYNC
3007 013774 000207      RTS    PC
3008
3009
3010
3011      ;POWER FAIL ROUTINE
3012 013776 012737 014006 000024      .PFAIL: MOV    #PWRUP,24      ;LOAD PFAIL VECTOR FOR POWER UP
3013 014004 000000      HALT
3014 014006 000005      PWRUP: RESET
3015 014010 012706 001050      MOV    #STACK,SP      ;WAIT TTY TO COME UP
3016 014014 012737 013776 000024      MOV    #.PFAIL,24     ;REINIT STACK POINTER
3017 014022 104407      TYPE
3018 014024 017074      MPOWER
3019 014026 000177 165026      JMP    @RETURN
3020
3021
3022      SEQ.DATA:
3023 014032 011637 001174      MOV    (SP),BACK
3024 014036 105077 165202      CLR    @SEXT
3025 014042 005037 001154      CLR    RDATA          ;CLEAR SYNC EXTENTION
3026 014046 005037 001152      CLR    TDATA          ;RECEIVER DATA
3027 014052 005077 165154      CLR    @DPRS          ;TRANSMITTER DATA
3028 014056 005077 165156      CLR    @DPTS          ;RECEIVER STATUS
3029 014062 052777 000001 165142      BIS    #BIT0,@DPRS    ;TRANSMITTER STATUS
3030 014070 012737 000400 001156      MOV    #400,CHLEN     ;STRIP SYNC
3031 014076 032737 000400 001134      BIT    #BIT8,SAVSR1   ;CHAR LENGTH INDEX
3032 014104 001414      BEQ    1$              ;TEST 12 BIT CHAR MODE
3033 014106 012737 010000 001160      MOV    #10000,LIMIT   ;NO
3034 014114 052777 002000 165110      BIS    #BIT10,@DPRS   ;SELECT END OF DATA
3035 014122 012737 000426 001144      MOV    #426,TSYNC     ;SELECT 12 BITS/CHARACTER
3036 014130 105277 165110      INCB   @SEXT          ;SYNC FOR 12 BIT CHAR
3037 014134 000406      BR     2$              ;PLACE MSB OF SYNC IN SYNC EXT
3038 014136 012737 000400 001160      1$:  MOV    #400,LIMIT   ;TEMPORARY CHARACTER LIMIT
3039 014144 012737 000026 001144      MOV    #26,TSYNC      ;INIT SYNC STORAGE
3040 014152 012777 014542 165072      2$:  MOV    #TV18,@DPTIV ;TRANSMITTER VECTOR
3041 014160 012777 014650 165060      MOV    #RV18,@DPRIV  ;RECEIVER VECTOR
3042 014166 012737 000200 177776      MOV    #200,PS        ;PRIORITY=4
3043 014174 012737 000004 001162      MOV    #4,SCNT        ;SYNC COUNT=4
3044 014202 113777 001144 165026      MOV    TSYNC,@SYNC   ;LOAD SYNC
3045 014210 052777 000100 165014      BIS    #BIT6,@DPRS    ;RCV INT ENB
  
```

3046	014216	032737	000001	001134		BIT	#BIT0,SAVSR1	
3047	014224	001004				BNE	.+12	
3048	014226	052777	000340	165004		BIS	#340,@DPTS	:STATUS INT ENB
3049	014234	000403				BR	.+10	
3050	014236	052777	000301	164774		BIS	#301,@DPTS	
3051								:TRANS INT ENB
3052								:TRANS DONE
3053	014244	032737	000001	001134		BIT	#BIT0,SAVSR1	
3054	014252	001017				BNE	5\$	
3055	014254	022737	000001	013522		CMP	#1,FREQ	:TEST FOR HIGH SPEED
3056	014262	001010				BNE	4\$:BRANCH IF NOT HIGH SPEED
3057	014264	052777	000010	164746	3\$:	BIS	#BIT3,@DPTS	:SET CLOCK HIGH
3058	014272	042777	000010	164740		BIC	#BIT3,@DPTS	:SET CLOCK LOW
3059	014300	000771				BR	3\$	
3060	014302	000000				HALT		
3061								
3062	014304	104402			4\$:	RXCLK		:RUN SLOW CLOCK
3063	014306	000001				1		
3064	014310	000775				BR	4\$	
3065	014312	005237	001102		5\$:	INC	TEMP1	
3066	014316	001375				BNE	5\$	
3067	014320	005337	001104			DEC	TEMP2	
3068	014324	001372				BNE	5\$	
3069	014326	104000				HLT		
3070	014330	000207			6\$:	RTS	PC	
3071								
3072	014332	017637	000000	001106	.DELAY:	MOV	@(SP),TEMP3	:SET UP COUNT
3073	014340	062716	000002			ADD	#2,(SP)	
3074	014344	011637	001110			MOV	(SP),TEMP4	
3075	014350	022626				POP	.SP	
3076	014352	005037	177776			CLR	PS	
3077	014356	162737	001106			SUB	(PC)+,TEMP3	:FOR A SMALLER DELAY TIME
3078	014362				LESS.TIME:			:PUT A NUMBER IN HERE TO BE
3079	014362	000000				000		:SUBTRACTED FROM TOTAL.
3080	014364	062737	001106			ADD	(PC)+,TEMP3	:FOR A LONGER DELAY TIME
3081	014370				MORE.TIME:			:PUT A NUMBER IN HERE TO BE ADDED
3082	014370	000000				000		:TO THE TOTAL TIME.
3083		014372			DEL=.			
3084	014372	162737	000001	001106		SUB	#1,TEMP3	:TIME OUT
3085	014400	001374				BNE	DEL	:NO
3086	014402	000177	164502			JMP	@TEMP4	

```

3087
3088
3089
3090      ; INTERRUPT SERVICE ROUTINES
3091      ;
3092      ; THESE ROUTINES MAY FUNCTION AS:
3093      ; 1. ERROR TRAPS FOR FALSE INTERRUPTS
3094      ; 2. POINTERS BACK TO THE MAIN LINE FOR VALID INTERRUPTS ;
3095      ; 3. FUNCTIONAL TEST WHICH ARE INTERRUPT DRIVEN
3096
3097      ; VERIFY THAT INTERRUPT RESULTED FROM RING.
3098 014406 032777 020000 164624 TV24: BIT #BIT13,@DPTS ;TEST FOR RING
3099 014414 001001          BNE 1$ ;BRANCH IF SET.
3100 014416 104000          HLT ;REPORT ERROR
3101
3102 014420 042777 020000 164612 1$: BIC #BIT13,@DPTS ;CLEAR RING FLAG.
3103 014426 032777 020000 164604 BIT #BIT13,@DPTS ;TEST IT
3104 014434 001401          BEQ 2$ ;BRANCH IF CLEAR.
3105 014436 104000          HLT ;REPORT ERROR
3106
3107 014440 032777 140200 164572 2$: BIT #140200,@DPTS ;NO OTHER STATUS FLAG ON?
3108 014446 001401          BEQ 3$
3109 014450 104000          HLT ;REPORT ERROR
3110 014452 022626          3$: POP.SP
3111 014454 000137 011754 JMP RCD1
3112
3113      ; VERIFY THAT INTERRUPT RESULTED FROM 'CARRIER DOWN' FLAG
3114 014460 032777 100000 164552 TV25: BIT #BIT15,@DPTS ;TEST FOR 'CARRIER DOWN' FLAG
3115 014466 001001          BNE 1$ ;BRANCH IF SET
3116 014470 104000          HLT ;REPORT ERROR
3117
3118 014472 042777 100000 164540 1$: BIC #BIT15,@DPTS ;CLEAR 'CARRIER DOWN' FLAG.
3119 014500 032777 100000 164532 BIT #BIT15,@DPTS ;TEST IT
3120 014506 001401          BEQ 2$ ;BRANCH IF CLEAR.
3121 014510 104000          HLT ;REPORT ERROR
3122
3123 014512 032777 060200 164520 2$: BIT #060200,@DPTS ;NO OTHER FLAGS ON?
3124 014520 001401          BEQ 3$
3125 014522 104000          HLT ;REPORT ERROR
3126
3127 014524 013716 001174 3$: MOV BACK,(SP)
3128 014530 000002 RTI
3129
3130
3131
3132
3133 014532 104005 FTINT: HLT 5 ;ERROR ROUTINE SHOULD RETURN TO MAINLINE
3134 014534 000002 RTI
3135
3136 014536 104006 FRINT: HLT 6 ;ERROR ROUTINE SHOULD RETURN TO MAINLINE
3137 014540 000002 RTI
3138
3139
3140
3141 014542 013777 001144 164472 TV18: MOV TSYNC,@DPTB ;XMIT SYNC
3142 014550 113777 001145 164466 MOVSB TSYNC+1,@SEXT ;LOAD SYNC EXT
  
```



```

3143 014556 005337 001162          DEC          SCNT          ;HAVE 2 SYNC'S BEEN XMITED
3144 014562 001003                    BNE          1$          ;NO
3145 014564 012777 014574 164460    MOV          #TV19,@DPTIV ;YES CHANGE VECTOR
3146 014572 000002                    RTI
3147
3148                                ;SEQUENTIAL DATA TRANSMISSION ROUTINE
3149
3150
3151 014574 032777 140000 164436    TV19:  BIT          #140000,@DPTS ;ANY STATUS ERRORS
3152 014602 001401                    BEQ          .+4         ;NO
3153 014604 104000                    HLT
3154 014606 105777 164426            TSTB        @DPTS        ;REPORT ERROR
3155 014612 100401                    BMI          .+4         ;TRANSMITTER READY
3156 014614 104000                    HLT          ;YES
3157 014616 013777 001152 164416    MOV          TDATA,@DPTB ;REPORT ERROR
3158 014624 005237 001152            INC          TDATA        ;LOAD BUFFER
3159 014630 023737 001160 001152    CMP          LIMIT,TDATA ;NEXT CHARACTER
3160 014636 001003                    BNE          1$          ;HAVE ALL CHARACTERS
3161 014640 042777 000140 164372    BIC          #140,@DPTS  ;NO
3162 014646 000002                    RTI          ;YES, CLEAR INTERRUPTS
3163
3164                                ;RECEIVE SEQUENTIAL DATA
3165
3166 014650 105777 164356            RV18:  TSTB        @DPRS        ;RECEVIER DONE??
3167 014654 100401                    BMI          .+4         ;YES
3168 014656 104000                    HLT          ;REPORT ERROR
3169 014660 013700 001154            MOV          RDATA,R0
3170 014664 017701 164344            MOV          @DPRB,R1
3171 014670 023777 001154 164336    CMP          RDATA,@DPRB ;CORRECT DATA
3172 014676 001404                    BEQ          1$
3173 014700 017737 164330 001136    MOV          @DPRB,TMPDAT ;STORE DATA
3174 014706 104001                    HLT          ;REPORT ERROR
3175 014710 042777 000001 164314    1$:  BIC          #BIT0,@DPRS ;CLEAR STRIP SYNC
3176 014716 005237 001154            INC          RDATA        ;NEXT CHARACTER
3177 014722 023737 001160 001154    CMP          LIMIT,RDATA
3178 014730 001047                    BNE          3$
3179 014732 005037 001154            CLR          RDATA
3180 014736 005037 001152            CLR          TDATA
3181 014742 006237 001160            ASR          LIMIT
3182 014746 012777 014542 164276    MOV          #TV18,@DPTIV ;DECREASE LIMIT TO 7 BITS
3183 014754 012737 000004 001162    MOV          #4,SCNT      ;SET UP SYNC TRANSMISSION
3184 014762 052777 000001 164242    BIS          #BIT0,@DPRS  ;SYNC COUNT =4
3185 014770 042777 004000 164234    BIC          #BIT11,@DPRS ;STRIP SYNC
3186 014776 032737 000001 001134    BIT          #BIT0,SAVSR1 ;CLEAR RCV ACTIVE
3187 015004 001004                    BNE          .+12
3188 015006 052777 000340 164224    BIS          #340,@DPTS  ;INT ENB + DONE
3189 015014 000403                    BR          .+10
3190 015016 052777 000301 164214    BIS          #301,@DPTS
3191 015024 053777 001156 164200    BIS          CHLEN,@DPRS ;CHANGE CHAR LENGTH
3192 015032 062737 000400 001156    ADD          #400,CHLEN  ;DECREASE CHAR LENGTH
3193 015040 022737 001400 001156    2$:  CMP          #1400,CHLEN ;HAVE ALL LENGTHS BEEN TESTED
3194 015046 001401                    BEQ          4$          ;YES
3195 015050 000002                    RTI          ;NO
3196 015052 005077 164162            3$:  CLR          @DPTS      ;CLR TRANSMITTER STATUS
3197 015056 005077 164150            4$:  CLR          @DPRS      ;CLR TECEIVER STATUS
3198 015062 005037 177776            CLR          PS
  
```

```

3199 015066 012706 001050      MOV      #STACK,SP
3200 015072 000177 164076      JMP      @BACK
3201                               ;SYNC ROUTINE FOR RANDOM DATA TEST
3202
3203
3204                               ;SYNC ROUTINE FOR RANDOM DATA TEST
3205
3206 015076      @RRRT:
3207 015076 105777 164136      TSTB    @DPTS      ;READY
3208 015102 100401      BMI     .+4        ;YES
3209 015104 104000      HLT     ;REPORT ERROR
3210 015106 013777 001144 164126  MOV     TSYNC,@DPTB ;TRANSMIT SYNC
3211 015114 005337 001162      DEC     SCNT       ;2 SYNC'S
3212 015120 001006      BNE    1$         ;NO
3213 015122 012777 015140 164122  MOV     #RRRT1,@DPTIV ;YES,SET UP DATA TRANSMIT VECTOR
3214 015130 042777 160000 164102  BIC     #160000,@DPTS ;CLEAR ERROR BITS
3215 015136      1$:
3216 015136 000002      RTI
3217
3218                               ;RRRT, RANDOM DATA, SYNC, RANDOM STALL
3219                               ;TRANSMITTER SERVICE ROUTINE
3220
3221 015140      RRT1:
3222 015140 105777 164074      TSTB    @DPTS      ;TRANSMITTER READY
3223 015144 100401      BMI     .+4        ;YES
3224 015146 104000      HLT     ;REPORT ERROR
3225 015150 004537 013342      JSR     R5,RNUM    ;GENERATE NEXT CHARACTER
3226 015154 010137 001152      MOV     R1,TDATA
3227 015160 043737 001156 001152  BIC     CHLEN,TDATA ;REDUCE DATA TO # BITS/CHAR
3228 015166 013777 001152 164046  MOV     TDATA,@DPTB ;TRANSMIT CHARACTER
3229 015174 013704 001170      MOV     TP,R4      ;SET UP TRANSMITTER POINTER
3230 015200 013724 001152      MOV     TDATA,(R4)+ ;MOV CHARACTER TO TUMBLE TABLE
3231 015204 020427 015620      CMP     R4,#TOP    ;END OF TUMBLE TABLE
3232 015210 001002      BNE    1$
3233 015212 012704 015610      MOV     #BOTTOM,R4
3234 015216 010437 001170      1$:  MOV     R4,TP      ;SAVE TRANSMITTER POINTER
3235 015222 000002      RTI
3236
3237                               ;RRRR, RANDOM DATA, RANDOM SYNC, RANDOM STALL, RECEIVER SERVICE
3238 015224 011637 001110      RRRR:  MOV     (SP),TEMP4
3239 015230 017737 164000 001154  MOV     @DPRB,RDATA ;SAVE RECEIVED DATA
3240 015236 013702 001172      MOV     RP,R2      ;SET UP RECEIVER POINTER
3241 015242 013701 001154      MOV     RDATA,R1
3242 015246 011200      MOV     (R2),R0
3243 015250 023722 001154      CMP     RDATA,(R2)+ ;IS DATA CORRECT
3244 015254 001406      BEQ    1$         ;YES
3245 015256 023737 001154 001144  CMP     RDATA,TSYNC ;IF NOT DATA IS IT SYNC
3246 015264 001401      BEQ    .+4        ;YES
3247 015266 104002      HLT     2         ;REPORT ERROR
3248 015270 005742      TST    -(R2)     ;ADJUST TUMBLE TABLE
3249 015272 022702 015620      1$:  CMP     #TOP,R2   ;TOP OF TUMBLE TABLE
3250 015276 001002      BNE    2$         ;NO
3251 015300 012702 015610      MOV     #BOTTOM,R2 ;YES,RAP AROUND
3252 015304 010237 001172      2$:  MOV     R2,RP     ;SAVE RECEIVER POINTER
3253 015310 042777 000001 163714  BIC     #BIT0,@DPRS ;CLEAR STRIP SYNC
3254 015316 000002      RTI

```

```

3255
3256
3257
3258
3259
3260
3261 015320
3262 015320 013777 001144 163714
3263 015326 005337 001162
3264 015332 001003
3265 015334 012777 015344 163710
3266 015342 000002
3267
3268 015344 032777 160000 163666
3269 015352 001401
3270 015354 104000
3271 015356 013777 001152 163656
3272 015364 005237 001152
3273 015370 023737 001152 001160
3274 015376 001401
3275 015400 000002
3276 015402 005077 163632
3277 015406 000002
3278
3279
3280
3281 000000
3282
3283 015410 017727 163616 000000
3284 015416 017737 163612 001136
3285 015424 013700 001154
3286 015430 013701 001136
3287 015434 023737 001136 001154
3288 015442 001401
3289 015444 104001
3290 015446 032737 010000 015414
3291 015454 001401
3292
3293 015456 104000
3294
3295
3296
3297 015460 005237 001154
3298 015464 005037 001104
3299 015470 012737 000020 001106
3300 015476 000241
3301 015500 006137 001154
3302 015504 103002
3303 015506 005137 001104
3304 015512 005337 001106
3305 015516 001370
3306
3307
3308
3309
3310 015520 006137 001154
  
```

;TRANSMITTER SERVICE ROUTINES FOR PARITY TEST

```

TPRTY:
      MOV    TSYNC,@DPTB    ;XMIT SYNC CHARACTER
      DEC    SCNT           ;DEC SYNC COUNT
      BNE    1$            ;BRANCH IF LESS THAN 2 SYNCs
      MOV    #2$,@DPTIV    ;SET VECTOR TO TRANSMIT DATA
1$:   RTI                    ;RETURN TO MAINLINE

2$:   BIT    #160000,@DPTS ;ANY ERRORS
      BEQ    .+4           ;NO
      HLT                    ;REPORT ERROR
      MOV    TDATA,@DPTB  ;TRANSMIT DATA
      INC    TDATA        ;INC TRANSMIT DATA
      CMP    TDATA,LIMIT  ;IS UPPER LIMIT REACHED
      BEQ    3$           ;YES, EXIT
      RTI                    ;NO, RETURN TO MAINLINE
3$:   CLR    @DPTS        ;CLEAR STATUS REGISTER
      RTI
  
```

;RECEIVER SERVICE ROUTINE FOR PARITY TEST

```

HERE=0
RPRTY: MOV    @DPRS,#HERE  ;SAVE RCV STATUS HERE
      MOV    @DPRB,TMPDAT ;SAVE RCV DATA
      MOV    RDATA,R0
      MOV    TMPDAT,R1
      CMP    TMPDAT,RDATA ;CHECK FOR CORRECT DATA
      BEQ    .+4           ;BRANCH IF DATA OK
      HLT                    ;REPORT ERROR
      BIT    #BIT12,RPRTY+4 ;TEST PARITY
RPRT1: BEQ    .+4           ;(RPRT1)=BEQ .+4 FOR EVEN PARITY
      ;(RPRT1)=BNE .+4 FOR ODD PARITY
      HLT                    ;REPORT ERROR
      ;EXAMIN #HERE FOR STATUS
      ;      TMPDAT FOR DATA
      ;      RPRT1 FOR ODD/EVEN
      INC    RDATA        ;INC TO NEXT EXPECTED DATA
      CLR    TEMP2        ;PARITY FLAG
      MOV    #16.,TEMP3   ;SET ROTATE COUNT TO 16.
      CLC                    ;CLEAR CARRY
RPRT2: ROL    RDATA        ;ROTATE DATA
      BCC    RPRT3        ;BRANCH IF BIT IS A '0'
      COM    TEMP2        ;SET FLAG TO A '1' FOR ODD PARITY
RPRT3: DEC    TEMP3        ;DEC ROTATE COUNT
      BNE    RPRT2        ;BRANCH IF 16 BIT WORD NOT CHECKED
  
```

;IF FLAG=1 EXPECTED DATA SHOULD CAUSE ODD PARITY
 ;BIT 12='1'

```

      ROL    RDATA        ;RESTORE EXPECTED DATA
  
```



```

3354
3355
3356
3357 015672 032737 040000 177570
3358 015700 001407
3359 015702 000432
3360 015704 105777 163140
3361 015710 100027
3362 015712 017700 163134
3363 015716 000412
3364 015720 032737 004000 177570
3365 015726 001006
3366 015730 005237 001070
3367 015734 023737 001070 001066
3368 015742 001012
3369 015744 105037 001200
3370 015750 005037 001070
3371 015754 012737 000012 001066
3372 015762 013737 001062 001060
3373 015770 013716 001060
3374 015774 000002
3375 015776 001407
3376 016000 000432
3377
3378
3379
3380
3381
3382
3383
3384
3385 016002 005037 001100
3386 016006 005037 001200
3387 016012 005237 001074
3388 016016 104407
3389 016020 017111
3390 016022 104413
3391 016024 016076
3392 016026 105737 001134
3393 016032 100402
3394 016034 005237 001146
3395 016040 013737 001074 177570
3396 016046 012737 001336 001060
3397 016054 012706 001050
3398 016060 105737 001134
3399 016064 100002
3400 016066 000137 001336
3401 016072 000137 001326
3402 016076 000001
3403 016100 006 002
3404 016102 001232
3405
3406
3407
3408 016104 032737 001000 177570
3409 016112 001402

```

```

;SCOPE LOOP AND ITERATION HANDLER
.SCOPE: BIT #BIT14,SWR
TST: BEQ 1$
BR 3$
TSTB @TKCSR
BPL 3$
MOV @TKDBR,R0
BR 2$
1$: BIT #SW11,SWR
BNE 2$
INC LPCNT
CMP LPCNT,ICOUNT
BNE 3$
2$: CLRB ERRFLG
CLR LPCNT
MOV #10,ICOUNT
MOV NEXT,RETURN
3$: MOV RETURN,(SP)
RTI
BRW: 1407
BRX: 432

;END OF PASS
;TYPE 'END OF PASS CSR: XXXXXX'
;UPDATE PASS COUNT
;UPDATE LINE NUMBER
;IF IN CYCLE MODE
;RESTART TEST

.EOP: CLR LSTERR ;CLEAR LAST ERROR PC
CLR ERRFLG ;CLEAR ERROR FLAG
INC PASCNT ;UPDATE PASS COUNT
TYPE
MEPASS
CNVRT
XCSR
TSTB SAVSR1
BMI .+6
INC XLINEX
MOV PASCNT,I.LIGHTS ;DISPLAY PASS COUNT
RESTR: MOV #PART1,RETURN
MOV #STACK,SP
TSTB SAVSR1
BPL .+6
JMP PART1
JMP BGNO
XCSR: 1
.BYTE 6,2
DPRS

;CHECK FOR FREEZE ON CURRENT DATA
.SCOPE1: BIT #SW09,SWR
BEQ 1$

```

3410	016114	013716	001064						
3411	016120	000002			1\$:	MOV	LOCK,(SP)		
3412						RTI			
3413									
3414							;TELETYPE OUTPUT ROUTINE		
3415	016122	017605	000000		.TYPE:	MOV	@(SP),R5		
3416	016126	062716	000002			ADD	#2,(SP)		
3417	016132	032737	010000	177570	1\$:	BIT	#SW12,SWR		
3418	016140	001010				BNE	3\$		
3419	016142	105715				TSTB	(R5)		
3420	016144	001406				BEQ	3\$		
3421	016146	105777	162702		2\$:	TSTB	@TPCSR		
3422	016152	100375				BPL	2\$		
3423	016154	112577	162676			MOVB	(R5)+,@TPDBR		
3424	016160	000764				BR	1\$		
3425	016162	000002			3\$:	RTI			
3426									
3427							;ERROR HANDLER		
3428									
3429	016164	032737	010000	177570	.HLT:	BIT	#SW12,SWR		
3430	016172	001406				BEQ	XBX		
3431	016174	105777	162654			TSTB	@TPCSR		
3432	016200	100003				BPL	XBX		
3433	016202	112777	000207	162646		MOVB	#207,@TPDBR		
3434	016210	032737	020000	177570	XBX:	BIT	#SW13,SWR		
3435	016216	001075				BNE	HALTS		
3436	016220	021637	001100			CMP	(SP),LSTERR		
3437	016224	001404				BEQ	1\$		
3438	016226	011637	001100			MOV	(SP),LSTERR		
3439	016232	105037	001200			CLRB	ERRFLG		
3440	016236	104410			1\$:	SAV05			
3441	016240	011605				MOV	(SP),R5		
3442	016242	162705	000002			SUB	#2,R5		
3443	016246	011504				MOV	(R5),R4		
3444	016250	006304				ASL	R4		
3445	016252	061504				ADD	(R5),R4		
3446	016254	006304				ASL	R4		
3447	016256	042704	177001			BIC	#177001,R4		
3448	016262	062704	017352			ADD	#.ERRTAB,R4		
3449	016266	012437	016362			MOV	(R4)+,ERRMSG		
3450	016272	012437	016374			MOV	(R4)+,DATAHD		
3451	016276	011437	016406			MOV	(R4),DATABP		
3452	016302	105737	001200			TSTB	ERRFLG		
3453	016306	001403				BEQ	TYPMSG		
3454	016310	005737	016406			TST	DATABP		
3455	016314	001030				BNE	TYPDAT		
3456	016316	104407			TYPMSG:	TYPE			
3457	016320	017134				MTSTN			
3458	016322	104413				CNVRT			
3459	016324	016514				XTSTN			
3460	016326	104407				TYPE			
3461	016330	017150				MLINE			
3462	016332	104413				CNVRT			
3463	016334	016522				ZLINE			
3464	016336	104407				TYPE			
3465	016340	017157				MERRPC			

3466 016342 104413
3467 016344 016506
3468 016346 104407
3469 016350 017166
3470 016352 112737 177777 001200
3471 016360 104407
3472 016362 000000
3473 016364 005737 016374
3474 016370 001402
3475 016372 104407
3476 016374 000000
3477 016376 005737 016406
3478 016402 001402
3479 016404 104412
3480 016406 000000
3481 016410 104411
3482 016412 005737 177570
3483 016416 100005
3484 016420 010046
3485 016422 016600 000002
3486 016426 000000
3487 016430 012600
3488 016432 005237 001076
3489 016436 032737 000001 001134
3490 016444 001013
3491 016446 032737 000400 177570
3492 016454 001007
3493 016456 032737 002000 177570
3494 016464 001407
3495 016466 013737 001062 001060
3496 016474 012706 011050
3497 016500 000177 162354
3498 016504 000002
3499 016506 000001
3500 016510 006 002
3501 016512 001132
3502 016514 000001
3503 016516 003 002
3504 016520 001072
3505 016522 000001
3506 016524 002 002
3507 016526 001146
3508
3509
3510
3511 016530 104407
3512 016532 017166
3513 016534 017601 000000
3514 016540 013737 017454 001106
3515 016546 062716 000002
3516 016552 012137 016722
3517 016556 112137 016724
3518 016562 112137 016725
3519 016566 013137 016726
3520 016572 013704 016726
3521 016576 113705 016724

CNVRT
ERTABO
TYPE
MCRLF
MOVB #-1,ERRFLG
TYPE
ERRMSG: 0
TST DATAHD
BEQ TYPDAT
TYPE
DATAHD: 0
TYPDAT: TST DATABP
BEQ RESREG
CONVRT
DATABP: 0
RESREG: RES05
HALTS: TST SWR
BPL EXITER
PUSHRO
MOV 2(SP),R0
HALT
POPRO
EXITER: INC ERRCNT
BIT #BIT0,SAVSR1
BNE 1\$
BIT #SW08,SWR
BNE 1\$
BIT #SW10,SWR
BEQ 2\$
MOV NEXT,RETURN
1\$: MOV #STACK,SP
JMP @RETURN
2\$: RTI
ERTABO: 1
.BYTE 6,2
SAVPC
XTSTN: 1
.BYTE 3,2
TSTNO
ZLINE: 1
.BYTE 2,2
XLINEX
;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
.CONVR: TYPE
MCRLF
.CNVRT: MOV @(SP),R1
MOV TEMP,TEMP3
ADD #2,(SP)
MOV (R1)+,WRDCNT
1\$: MOVB (R1)+,CHRCNT
MOVB (R1)+,SPACNT
MOV @(R1)+,BINWRD
2\$: MOV BINWRD,R4
MOVB CHRCNT,R5

3522	016602	012700	017454		
3523	016606	010403		3\$:	MOV #TEMP,R0
3524	016610	042703	177770		MOV R4,R3
3525	016614	062703	000260		BIC #177770,R3
3526	016620	110320			ADD #260,R3
3527	016622	000241			MOVB R3,(R0)+
3528	016624	006004			CLC
3529	016626	000241			ROR R4
3530	016630	006004			CLC
3531	016632	000241			ROR R4
3532	016634	006004			CLC
3533	016636	005305			ROR R4
3534	016640	001362			DEC R5
3535	016642	012703	017516		BNE 3\$
3536	016646	114023		4\$:	MOV #MDATA,R3
3537	016650	105337	016724		MOVB -(R0),(R3)+
3538	016654	001374			DECB CHRCNT
3539	016656	105737	016725		BNE 4\$
3540	016662	001405			TSTB SPACNT
3541	016664	112723	000240		BEQ 6\$
3542	016670	105337	016725	5\$:	MOVB #240,(R3)+
3543	016674	001373			DECB SPACNT
3544	016676	105013		6\$:	BNE 5\$
3545	016700	104407			CLRB (R3)
3546	016702	017516			TYPE
3547	016704	005337	016722		MDATA
3548	016710	001322			DEC WRDCNT
3549	016712	013737	001106 017454		BNE 1\$
3550	016720	000002			MOV TEMP3,TEMP
3551	016722	000000			RTI
3552	016724	000000			WRDCNT: 0
3553		016725			CHRCNT: 0
3554	016726	000000			SPACNT=CHRCNT+1
3555					BINWRD: 0
3556					:SAVE PC OF TEST THAT FAILED AND R0-R5
3557					
3558	016730	016637	000004 001132	.SAV05:	MOV 4(SP),SAVPC
3559					
3560					:SAVE R0-R5
3561					
3562	016736	010537	001126	SV05:	MOV R5,SAVR5
3563	016742	010437	001124		MOV R4,SAVR4
3564	016746	010337	001122		MOV R3,SAVR3
3565	016752	010237	001120		MOV R2,SAVR2
3566	016756	010137	001116		MOV R1,SAVR1
3567	016762	010037	001114		MOV R0,SAVR0
3568	016766	000002			RTI
3569					
3570					:RESTORE R0-R5
3571					
3572	016770	013700	001114	.RES05:	MOV SAVR0,R0
3573	016774	013701	001116		MOV SAVR1,R1
3574	017000	013702	001120		MOV SAVR2,R2
3575	017004	013703	001122		MOV SAVR3,R3
3576	017010	013704	001124		MOV SAVR4,R4
3577	017014	013705	001126		MOV SAVR5,R5

3578	017020	000002		
3579	017022	005015	055103	050104
3580	017030	041501	020040	050104
3581	017036	030461	020101	054523
3582	017044	041516	046440	042117
3583	017052	046505	044440	052116
3584	017060	051105	040506	042503
3585	017066	042040	040511	000107
3586	017074	005015	053520	020122
3587	017102	040506	046111	042105
3588	017110	000		
3589	017111	007	006407	042412
3590	017116	042116	050040	051501
3591	017124	020123	051503	035122
3592	017132	000040		
3593	017134	005015	042524	052123
3594	017142	047040	027117	000040
3595	017150	044514	042516	020072
3596	017156	000		
3597	017157	015	050012	035103
3598	017164	000040		
3599	017166	005015	000	
3600	017171	015	042412	052116
3601	017176	051105	042105	043040
3602	017204	047522	020115	000040
3603	017212	005015	051124	047101
3604	017220	046523	052111	042524
3605	017226	020122	000040	
3606	017232	005015	042522	042503
3607	017240	053111	051105	020040
3608	017246	000		
3609	017247	105	050130	041505
3610	017254	042524	020104	051040
3611	017262	041505	044505	042526
3612	017270	020104	000	
3613	017275	106	044501	042514
3614	017300	020104	047524	044440
3615	017306	052116	051105	050125
3616	017314	027124	000040	
3617	017320	047111	042524	052522
3618	017326	052120	042105	052440
3619	017334	042516	050130	041505
3620	017342	042524	046104	027131
3621	017350	000		
3622				
3623		017352		
3624	017352			
3625	017352	000000		
3626	017354	000000		
3627	017356	000000		
3628	017360	017166		
3629	017362	017247		
3630	017364	017424		
3631				
3632	017366	017171		
3633	017370	017247		

RTI
 MTITLE: .ASCIZ <15><12>/CZDPAC DP11A SYNC MODEM INTERFACE DIAG/

MPOWER: .ASCIZ <15><12>/PWR FAILED/

MEPASS: .ASCIZ <7><7><15><12>/END PASS CSR: /

MTSTN: .ASCIZ <15><12>/TEST NO. /

MLINE: .ASCIZ /LINE: /

MERRPC: .ASCIZ <15><12>/PC: /

MCRLF: .ASCIZ <15><12>
 EM1: .ASCIZ <15><12>/ENTERED FROM /

EM2: .ASCIZ <15><12>/TRANSMITTER /

EM3: .ASCIZ <15><12>/RECEIVER /

DH0: .ASCIZ /EXPECTED RECEIVED /

DH1: .ASCIZ /FAILED TO INTERRUPT. /

DH2: .ASCIZ /INTERRUPTED UNEXPECTEDLY./

.EVEN
 .ERRTAB:
 0
 0
 0
 MCRLF
 DH0 ;HALT 1
 DT0
 EM1
 DH0 ;HALT 2

3634	017372	017436			DT1	
3635						
3636	017374	017212			EM2	
3637	017376	017273			DH1	:HALT 3
3638	017400	000000			0	
3639						
3640	017402	017232			EM3	
3641	017404	017273			DH1	:HALT 4
3642	017406	000000			0	
3643						
3644	017410	017212			EM2	
3645	017412	017320			DH2	:HALT 5
3646	017414	000000			0	
3647						
3648	017416	017232			EM3	
3649	017420	017320			DH2	:HALT 6
3550	017422	000000			0	
3651						
3652	017424	000002			DT0:	2
3653	017426	006	004		.BYTE	6.4
3654	017430	001114			SAVR0	
3655	017432	006	002		.BYTE	6.2
3656	017434	001116			SAVR1	
3657						
3658	017436	000003			DT1:	3
3659	017440	006	010		.BYTE	6.8.
3660	017442	001110			TEMP4	
3661	017444	006	004		.BYTE	6.4
3662	017446	001114			SAVR0	
3663	017450	006	002		.BYTE	6.2
3664	017452	001116			SAVR1	
3665						
3666	017454	000000			TEMP:	0
3667		017516			.=.+40	
3668	017516	000000			MDATA:	0
3669		017560			.=.+40	
3670		000001			.END	

BITSR	534#														
HLT	602#	1077	1184	1209	1233	1259	1281	1285	1416	1431	1546	1554	1562	1584	1588
	1610	1612	1640	1642	1669	1671	1699	1705	1734	1742	1783	1791	1799	1802	1806
	1811	1857	1868	1870	1877	1931	1940	1944	1949	1952	1956	2124	2128	2244	2249
	2254	2259	2263	2306	2311	2316	2321	2325	2369	2374	2379	2384	2388	2482	2485
	2488	2493	2496	2499	2523	2526	2529	2535	2538	2541	2558	2561	2566	2569	2572
	2597	2600	2603	2611	2614	2617	2628	2636	2675	2682	2695	2700	2812	2818	2824
	2830	2849	2868	3069	3100	3105	3109	3116	3121	3125	3133	3136	3153	3156	3168
	3174	3209	3224	3247	3270	3289	3293	3343	3348						
\$SCATCH	1#	623													
\$CONVR	1#	3508													
\$DELAY	534#	3072													
\$EOP	1#	3377													
\$HEADE	1#	539													
\$HLT	1#	3426													
\$MSG	1#														
\$REG	2#	3555													
\$SCOPE	1#	3354													
\$SCOPI	1#	3405													
\$SYMBO	1#	555													
\$STRAPS	1#	979													
\$TRPDE	1#	988	990	992	994	996	998	1000	1002	1004	1006	1008	1010		
\$TRPSR	2#	895													
\$TSTN	1#	1057	1084	1121	1168	1192	1217	1243	1266	1296	1326	1364	1399	1463	1502
	1533	1572	1594	1623	1652	1684	1715	1758	1829	1906	1983	2023	2056	2092	2140
	2186	2217	2278	2340	2401	2417	2461	2503	2582	2653					
\$TYPE	1#	3412													
\$VARIA	1#	918													

. ABS. 017560 000

ERRORS DETECTED: 0

CZDPAC.BIN,CZDPAC.SEQ/CRF/SOL/NL:TOC=CZDPAC.MCL,CZDPAC.P11
 RUN-TIME: 6 10 1 SECONDS
 RUN-TIME RATIO: 60/18=3.2
 CORE USED: 13K (25 PAGES)