

DP11

SYNCHRONOUS INTERFACE
MD-11-DZDPA-B

EP-DZDPA-B-DL-A
COPYRIGHT © 1976
FICHE 1 OF 1

NOV 1976
digital
MADE IN U.S.A.

The microfiche card displays a grid of 100 frames of data, arranged in 10 rows and 10 columns. Each frame contains a small, dense grid of characters, likely representing a data table or a series of small diagrams. The text is very small and difficult to read, but the overall structure is a regular grid of information.

E01

PAGE 04

152
151
150
149
148
147
146
145
144
143
142
141
140
139
138
137
136
135
134
133
132
131
130
129
128
127
126
125
124
123
122
121
120
119
118
117
116
115
114
113
112
111
110
109
108
107
106
105
104
103
102
101
100
99
98
97
96
95
94
93
92
91
90
89
88
87
86
85
84
83
82
81
80
79
78
77
76
75
74
73
72
71
70
69
68
67
66
65
64
63
62
61
60
59
58
57
56
55
54
53
52
51
50
49
48
47
46
45
44
43
42
41
40
39
38
37
36
35
34
33
32
31
30
29
28
27
26
25
24
23
22
21
20
19
18
17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1

- 4.3.2 FOR CABLE TEST REMOVE BCO1R-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.

037
038
039
040
041
042
043
044
045
046
047
048
049
050
051
052
053
054
055
056
057
058
059
060
061
062
063
064
065
066
067
068
069
070
071
072
073
074
075
076

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(9)). 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.

0277
 0278
 0279
 0280
 0281
 0282
 0283
 0284
 0285
 0286
 0287
 0288
 0289
 0290
 0291
 0292
 0293
 0294
 0295
 0296
 0297
 0298
 0299
 0300
 0301
 0302
 0303
 0304
 0305
 0306
 0307
 0308
 0309
 0310
 0311
 0312
 0313
 0314
 0315
 0316
 0317
 0318
 0319
 0320
 0321
 0322
 0323
 0324
 0325
 0326
 0327
 0328
 0329
 0330
 0331
 0332

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

I01

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 8
DZDPAB.DOC

333
334
335

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.

336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376

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.

377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430

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.

431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481

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 BITS OF SAVSRI 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 SOFTWARE 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.

NO1.

DZDP MACY11 27(732) 20-APR-75 14:14 PAGE 14
DZDPAB.MCL

522

676	000154	000156	.+2	; UNEXPECTED TRAP TO THIS LOCATION
677	000156	000000	HALT	; EXAMINE STACK TO FIND CAUSE
678	000160	000162	.+2	; UNEXPECTED TRAP TO THIS LOCATION
679	000162	000000	HALT	; EXAMINE STACK TO FIND CAUSE
680	000164	000166	.+2	; UNEXPECTED TRAP TO THIS LOCATION
681	000166	000000	HALT	; EXAMINE STACK TO FIND CAUSE
682	000170	000172	.+2	; UNEXPECTED TRAP TO THIS LOCATION
683	000172	000000	HALT	; EXAMINE STACK TO FIND CAUSE
684	000174	000176	.+2	; UNEXPECTED TRAP TO THIS LOCATION
685	000176	000000	HALT	; EXAMINE STACK TO FIND CAUSE
686	000200	000202	.+2	; UNEXPECTED TRAP TO THIS LOCATION
687	000202	000000	HALT	; EXAMINE STACK TO FIND CAUSE
688	000204	000206	.+2	; UNEXPECTED TRAP TO THIS LOCATION
689	000206	000000	HALT	; EXAMINE STACK TO FIND CAUSE
690	000210	000212	.+2	; UNEXPECTED TRAP TO THIS LOCATION
691	000212	000000	HALT	; EXAMINE STACK TO FIND CAUSE
692	000214	000216	.+2	; UNEXPECTED TRAP TO THIS LOCATION
693	000216	000000	HALT	; EXAMINE STACK TO FIND CAUSE
694	000220	000222	.+2	; UNEXPECTED TRAP TO THIS LOCATION
695	000222	000000	HALT	; EXAMINE STACK TO FIND CAUSE
696	000224	000226	.+2	; UNEXPECTED TRAP TO THIS LOCATION
697	000226	000000	HALT	; EXAMINE STACK TO FIND CAUSE
698	000230	000232	.+2	; UNEXPECTED TRAP TO THIS LOCATION
699	000232	000000	HALT	; EXAMINE STACK TO FIND CAUSE
700	000234	000236	.+2	; UNEXPECTED TRAP TO THIS LOCATION
701	000236	000000	HALT	; EXAMINE STACK TO FIND CAUSE
702	000240	000242	.+2	; UNEXPECTED TRAP TO THIS LOCATION
703	000242	000000	HALT	; EXAMINE STACK TO FIND CAUSE
704	000244	000246	.+2	; UNEXPECTED TRAP TO THIS LOCATION
705	000246	000000	HALT	; EXAMINE STACK TO FIND CAUSE
706	000250	000252	.+2	; UNEXPECTED TRAP TO THIS LOCATION
707	000252	000000	HALT	; EXAMINE STACK TO FIND CAUSE
708	000254	000256	.+2	; UNEXPECTED TRAP TO THIS LOCATION
709	000256	000000	HALT	; EXAMINE STACK TO FIND CAUSE
710	000260	000262	.+2	; UNEXPECTED TRAP TO THIS LOCATION
711	000262	000000	HALT	; EXAMINE STACK TO FIND CAUSE
712	000264	000266	.+2	; UNEXPECTED TRAP TO THIS LOCATION
713	000266	000000	HALT	; EXAMINE STACK TO FIND CAUSE
714	000270	000272	.+2	; UNEXPECTED TRAP TO THIS LOCATION
715	000272	000000	HALT	; EXAMINE STACK TO FIND CAUSE
716	000274	000276	.+2	; UNEXPECTED TRAP TO THIS LOCATION
717	000276	000000	HALT	; EXAMINE STACK TO FIND CAUSE
718	000300	000302	.+2	; UNEXPECTED TRAP TO THIS LOCATION
719	000302	000000	HALT	; EXAMINE STACK TO FIND CAUSE
720	000304	000306	.+2	; UNEXPECTED TRAP TO THIS LOCATION
721	000306	000000	HALT	; EXAMINE STACK TO FIND CAUSE
722	000310	000312	.+2	; UNEXPECTED TRAP TO THIS LOCATION
723	000312	000000	HALT	; EXAMINE STACK TO FIND CAUSE
724	000314	000316	.+2	; UNEXPECTED TRAP TO THIS LOCATION
725	000316	000000	HALT	; EXAMINE STACK TO FIND CAUSE
726	000320	000322	.+2	; UNEXPECTED TRAP TO THIS LOCATION
727	000322	000000	HALT	; EXAMINE STACK TO FIND CAUSE
728	000324	000326	.+2	; UNEXPECTED TRAP TO THIS LOCATION
729	000326	000000	HALT	; EXAMINE STACK TO FIND CAUSE
730	000330	000332	.+2	; UNEXPECTED TRAP TO THIS LOCATION
731	000332	000000	HALT	; EXAMINE STACK TO FIND CAUSE

732	000334	000336	.+2	:UNEXPECTED TRAP TO THIS LOCATION
733	000336	000000	HALT	:EXAMINE STACK TO FIND CAUSE
734	000340	000342	.+2	:UNEXPECTED TRAP TO THIS LOCATION
735	000342	000000	HALT	:EXAMINE STACK TO FIND CAUSE
736	000344	000346	.+2	:UNEXPECTED TRAP TO THIS LOCATION
737	000346	000000	HALT	:EXAMINE STACK TO FIND CAUSE
738	000350	000352	.+2	:UNEXPECTED TRAP TO THIS LOCATION
739	000352	000000	HALT	:EXAMINE STACK TO FIND CAUSE
740	000354	000356	.+2	:UNEXPECTED TRAP TO THIS LOCATION
741	000356	000000	HALT	:EXAMINE STACK TO FIND CAUSE
742	000350	000352	.+2	:UNEXPECTED TRAP TO THIS LOCATION
743	000362	000000	HALT	:EXAMINE STACK TO FIND CAUSE
744	000364	000366	.+2	:UNEXPECTED TRAP TO THIS LOCATION
745	000366	000000	HALT	:EXAMINE STACK TO FIND CAUSE
746	000370	000372	.+2	:UNEXPECTED TRAP TO THIS LOCATION
747	000372	000000	HALT	:EXAMINE STACK TO FIND CAUSE
748	000374	000376	.+2	:UNEXPECTED TRAP TO THIS LOCATION
749	000376	000000	HALT	:EXAMINE STACK TO FIND CAUSE
750	000400	000402	.+2	:UNEXPECTED TRAP TO THIS LOCATION
751	000402	000000	HALT	:EXAMINE STACK TO FIND CAUSE
752	000404	000406	.+2	:UNEXPECTED TRAP TO THIS LOCATION
753	000406	000000	HALT	:EXAMINE STACK TO FIND CAUSE
754	000410	000412	.+2	:UNEXPECTED TRAP TO THIS LOCATION
755	000412	000000	HALT	:EXAMINE STACK TO FIND CAUSE
756	000414	000416	.+2	:UNEXPECTED TRAP TO THIS LOCATION
757	000416	000000	HALT	:EXAMINE STACK TO FIND CAUSE
758	000420	000422	.+2	:UNEXPECTED TRAP TO THIS LOCATION
759	000422	000000	HALT	:EXAMINE STACK TO FIND CAUSE
760	000424	000426	.+2	:UNEXPECTED TRAP TO THIS LOCATION
761	000426	000000	HALT	:EXAMINE STACK TO FIND CAUSE
762	000430	000432	.+2	:UNEXPECTED TRAP TO THIS LOCATION
763	000432	000000	HALT	:EXAMINE STACK TO FIND CAUSE
764	000434	000436	.+2	:UNEXPECTED TRAP TO THIS LOCATION
765	000436	000000	HALT	:EXAMINE STACK TO FIND CAUSE
766	000440	000442	.+2	:UNEXPECTED TRAP TO THIS LOCATION
767	000442	000000	HALT	:EXAMINE STACK TO FIND CAUSE
768	000444	000446	.+2	:UNEXPECTED TRAP TO THIS LOCATION
769	000446	000000	HALT	:EXAMINE STACK TO FIND CAUSE
770	000450	000452	.+2	:UNEXPECTED TRAP TO THIS LOCATION
771	000452	000000	HALT	:EXAMINE STACK TO FIND CAUSE
772	000454	000456	.+2	:UNEXPECTED TRAP TO THIS LOCATION
773	000456	000000	HALT	:EXAMINE STACK TO FIND CAUSE
774	000460	000462	.+2	:UNEXPECTED TRAP TO THIS LOCATION
775	000462	000000	HALT	:EXAMINE STACK TO FIND CAUSE
776	000464	000466	.+2	:UNEXPECTED TRAP TO THIS LOCATION
777	000466	000000	HALT	:EXAMINE STACK TO FIND CAUSE
778	000470	000472	.+2	:UNEXPECTED TRAP TO THIS LOCATION
779	000472	000000	HALT	:EXAMINE STACK TO FIND CAUSE
780	000474	000476	.+2	:UNEXPECTED TRAP TO THIS LOCATION
781	000476	000000	HALT	:EXAMINE STACK TO FIND CAUSE
782	000500	000502	.+2	:UNEXPECTED TRAP TO THIS LOCATION
783	000502	000000	HALT	:EXAMINE STACK TO FIND CAUSE
784	000504	000506	.+2	:UNEXPECTED TRAP TO THIS LOCATION
785	000506	000000	HALT	:EXAMINE STACK TO FIND CAUSE
786	000510	000512	.+2	:UNEXPECTED TRAP TO THIS LOCATION
787	000512	000000	HALT	:EXAMINE STACK TO FIND CAUSE

788	000514	000516	.+2	; UNEXPECTED TRAP TO THIS LOCATION
789	000516	000000	HALT	; EXAMINE STACK TO FIND CAUSE
790	000520	000522	.+2	; UNEXPECTED TRAP TO THIS LOCATION
791	000522	000000	HALT	; EXAMINE STACK TO FIND CAUSE
792	000524	000526	.+2	; UNEXPECTED TRAP TO THIS LOCATION
793	000526	000000	HALT	; EXAMINE STACK TO FIND CAUSE
794	000530	000532	.+2	; UNEXPECTED TRAP TO THIS LOCATION
795	000532	000000	HALT	; EXAMINE STACK TO FIND CAUSE
796	000534	000536	.+2	; UNEXPECTED TRAP TO THIS LOCATION
797	000536	000000	HALT	; EXAMINE STACK TO FIND CAUSE
798	000540	000542	.+2	; UNEXPECTED TRAP TO THIS LOCATION
799	000542	000000	HALT	; EXAMINE STACK TO FIND CAUSE
800	000544	000546	.+2	; UNEXPECTED TRAP TO THIS LOCATION
801	000546	000000	HALT	; EXAMINE STACK TO FIND CAUSE
802	000550	000552	.+2	; UNEXPECTED TRAP TO THIS LOCATION
803	000552	000000	HALT	; EXAMINE STACK TO FIND CAUSE
804	000554	000556	.+2	; UNEXPECTED TRAP TO THIS LOCATION
805	000556	000000	HALT	; EXAMINE STACK TO FIND CAUSE
806	000560	000562	.+2	; UNEXPECTED TRAP TO THIS LOCATION
807	000562	000000	HALT	; EXAMINE STACK TO FIND CAUSE
808	000564	000566	.+2	; UNEXPECTED TRAP TO THIS LOCATION
809	000566	000000	HALT	; EXAMINE STACK TO FIND CAUSE
810	000570	000572	.+2	; UNEXPECTED TRAP TO THIS LOCATION
811	000572	000000	HALT	; EXAMINE STACK TO FIND CAUSE
812	000574	000576	.+2	; UNEXPECTED TRAP TO THIS LOCATION
813	000576	000000	HALT	; EXAMINE STACK TO FIND CAUSE
814	000600	000602	.+2	; UNEXPECTED TRAP TO THIS LOCATION
815	000602	000000	HALT	; EXAMINE STACK TO FIND CAUSE
816	000604	000606	.+2	; UNEXPECTED TRAP TO THIS LOCATION
817	000606	000000	HALT	; EXAMINE STACK TO FIND CAUSE
818	000610	000612	.+2	; UNEXPECTED TRAP TO THIS LOCATION
819	000612	000000	HALT	; EXAMINE STACK TO FIND CAUSE
820	000614	000616	.+2	; UNEXPECTED TRAP TO THIS LOCATION
821	000616	000000	HALT	; EXAMINE STACK TO FIND CAUSE
822	000620	000622	.+2	; UNEXPECTED TRAP TO THIS LOCATION
823	000622	000000	HALT	; EXAMINE STACK TO FIND CAUSE
824	000624	000626	.+2	; UNEXPECTED TRAP TO THIS LOCATION
825	000626	000000	HALT	; EXAMINE STACK TO FIND CAUSE
826	000630	000632	.+2	; UNEXPECTED TRAP TO THIS LOCATION
827	000632	000000	HALT	; EXAMINE STACK TO FIND CAUSE
828	000634	000636	.+2	; UNEXPECTED TRAP TO THIS LOCATION
829	000636	000000	HALT	; EXAMINE STACK TO FIND CAUSE
830	000640	000642	.+2	; UNEXPECTED TRAP TO THIS LOCATION
831	000642	000000	HALT	; EXAMINE STACK TO FIND CAUSE
832	000644	000646	.+2	; UNEXPECTED TRAP TO THIS LOCATION
833	000646	000000	HALT	; EXAMINE STACK TO FIND CAUSE
834	000650	000652	.+2	; UNEXPECTED TRAP TO THIS LOCATION
835	000652	000000	HALT	; EXAMINE STACK TO FIND CAUSE
836	000654	000656	.+2	; UNEXPECTED TRAP TO THIS LOCATION
837	000656	000000	HALT	; EXAMINE STACK TO FIND CAUSE
838	000660	000662	.+2	; UNEXPECTED TRAP TO THIS LOCATION
839	000662	000000	HALT	; EXAMINE STACK TO FIND CAUSE
840	000664	000666	.+2	; UNEXPECTED TRAP TO THIS LOCATION
841	000666	000000	HALT	; EXAMINE STACK TO FIND CAUSE
842	000670	000672	.+2	; UNEXPECTED TRAP TO THIS LOCATION
843	000672	000000	HALT	; EXAMINE STACK TO FIND CAUSE

844	000674	000676		.+2	;UNEXPECTED TRAP TO THIS LOCATION
845	000676	000000		HALT	;EXAMINE STACK TO FIND CAUSE
846	000700	000702		.+2	;UNEXPECTED TRAP TO THIS LOCATION
847	000702	000000		HALT	;EXAMINE STACK TO FIND CAUSE
848	000704	000706		.+2	;UNEXPECTED TRAP TO THIS LOCATION
849	000706	000000		HALT	;EXAMINE STACK TO FIND CAUSE
850	000710	000712		.+2	;UNEXPECTED TRAP TO THIS LOCATION
851	000712	000000		HALT	;EXAMINE STACK TO FIND CAUSE
852	000714	000716		.+2	;UNEXPECTED TRAP TO THIS LOCATION
853	000716	000000		HALT	;EXAMINE STACK TO FIND CAUSE
854	000720	000722		.+2	;UNEXPECTED TRAP TO THIS LOCATION
855	000722	000000		HALT	;EXAMINE STACK TO FIND CAUSE
856	000724	000726		.+2	;UNEXPECTED TRAP TO THIS LOCATION
857	000726	000000		HALT	;EXAMINE STACK TO FIND CAUSE
858	000730	000732		.+2	;UNEXPECTED TRAP TO THIS LOCATION
859	000732	000000		HALT	;EXAMINE STACK TO FIND CAUSE
860	000734	000736		.+2	;UNEXPECTED TRAP TO THIS LOCATION
861	000736	000000		HALT	;EXAMINE STACK TO FIND CAUSE
862	000740	000742		.+2	;UNEXPECTED TRAP TO THIS LOCATION
863	000742	000000		HALT	;EXAMINE STACK TO FIND CAUSE
864	000744	000746		.+2	;UNEXPECTED TRAP TO THIS LOCATION
865	000746	000000		HALT	;EXAMINE STACK TO FIND CAUSE
866	000750	000752		.+2	;UNEXPECTED TRAP TO THIS LOCATION
867	000752	000000		HALT	;EXAMINE STACK TO FIND CAUSE
868	000754	000756		.+2	;UNEXPECTED TRAP TO THIS LOCATION
869	000756	000000		HALT	;EXAMINE STACK TO FIND CAUSE
870	000760	000762		.+2	;UNEXPECTED TRAP TO THIS LOCATION
871	000762	000000		HALT	;EXAMINE STACK TO FIND CAUSE
872	000764	000766		.+2	;UNEXPECTED TRAP TO THIS LOCATION
873	000766	000000		HALT	;EXAMINE STACK TO FIND CAUSE
874	000770	000772		.+2	;UNEXPECTED TRAP TO THIS LOCATION
875	000772	000000		HALT	;EXAMINE STACK TO FIND CAUSE
876	000774	000776		.+2	;UNEXPECTED TRAP TO THIS LOCATION
877	000776	000000		HALT	;EXAMINE STACK TO FIND CAUSE
878			06400	;VECTOR	INITIALIZATION
879		000024	06900	.=24	
880	000024	013744	07000	.PFAIL	;POWER FAIL VECTOR
881	000026	000340	07100	340	;PRIORITY 7
882	000030	016132	07200	.HLT	
883	000032	000340	07300	340	
884	000034	000056	07400	.TRPSRV	
885	000036	000340	07500	340	
886			07600		
887		000046	07650	.=46	
888	000046	012436	07655	LOGICAL	
889		000052	07660	.=52	
890	000052	000000	07665	0	
891		000056	07700	.=56	
892					;TRAP DISPATCH SERVICE
893					;ARGUMENT OF TRAP IS EXTRACTED
894					;AND USED AS OFFSET TO OBTAIN POINTER
895					;TO SELECTED SUBROUTINE
896					
897	000056	011646		.TRPSR: MOV	(SP), -(SP) ;GET PC OF RETURN
898	000060	162716	000002	SUB	#2, (SP) ;=PC OF TRAP
899	000064	017616	000000	MOV	2(SP), (SP) ;GET TRP

900	000070	006316		TRPOK:	ASL	(SP)	;MULTIPLY TRAP ARG BY 2
901	000072	042716	177001		BIC	#177001,(SP)	;CLEAR UNWANTED BITS
902	000076	062716	001202		ADD	#.TRPTAB,(SP)	;POINTER TO SUBROUTINE ADDRESS
903	000102	017616	000000		MOV	Q(SP),(SP)	;SUBROUTINE ADDRESS
904	000106	000136			JMP	Q(SP)+	;GO TO SUBROUTINE
905			07900				
906		000200	08000				
907	000200	005037	001146	.=200	START1:	CLR	XLINEX
908	000204	000137	001262			JMP	BEGIN1
909			08100				;SET UP CONSOL SWITCH REGISTER
910		000210	08200				
911	000210	005037	001146	.=210		CLR	XLINEX
912	000214	000137	007202			JMP	BEGIN2
913			08300				;DB25S CONNECTOR TEST
914		001050	08400				
915			08500				
916			08600				
917			08700				
918			08800	.=1050			
919	001050	177560					;INDIRECT POINTERS TO TELETYPE VECTORS AND REGISTERS
920	001052	177562		TKCSR:	177560		;TELETYPE KEYBOARD CONTROL REGISTER
921	001054	177564		TKDBR:	177562		;TELETYPE KEYBOARD DATA BUFFER
922	001056	177566		TPCSR:	177564		;TELEPRINTER CONTROL REGISTER
923				TPDBR:	177566		;TELEPRINTER DATA BUFFER
924							;PROGRAM CONTROL PARAMETERS
925				RETURN:	0		;SCOPE ADDRESS FOR LOOP ON TEST
926	001060	000000		NEXT:	0		;ADDRESS OF NEXT TEST TO BE EXECUTED
927	001062	000000		LOCK:	0		;ADDRESS FOR LOCK ON CURRENT DATA
928	001064	000000		ICOUNT:	0		;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE
929	001066	000000		LPCNT:	0		;NUMBER OF ITERATIONS COMPLETED
930	001070	000000		TSTNO:	0		;NUMBER OF TEST IN PROGRESS
931	001072	000000		PASCNT:	0		;NUMBER OF PASSES COMPLETED
932	001074	000000		ERRCNT:	0		;TOTAL NUMBER OF ERRORS
933	001076	000000		LSTERR:	0		;PC OF LAST ERROR CALL
934	001100	000000					
935							
936							
937							
938	001102	000000		TEMP1:	0		;TEMPORARY STORAGE
939	001104	000000		TEMP2:	0		;TEMPORARY STORAGE
940	001106	000000		TEMP3:	0		;TEMPORARY STORAGE
941	001110	000000		TEMP4:	0		;TEMPORARY STORAGE
942	001112	000000		TEMP5:	0		;TEMPORARY STORAGE
943	001114	000000		SAVR0:	0		;R0 STORAGE
944	001116	000000		SAVR1:	0		;R1 STORAGE
945	001120	000000		SAVR2:	0		;R2 STORAGE
946	001122	000000		SAVR3:	0		;R3 STORAGE
947	001124	000000		SAVR4:	0		;R4 STORAGE
948	001126	000000		SAVR5:	0		;R5 STORAGE
949	001130	000000		SAVSP:	0		;STACK POINTER STORAGE
950	001132	000000		SAVPC:	0		;PROGRAM COUNTER STORAGE
951	001134	000000		SAVSR1:	0		
952	001136	000000		TMPDAT:	0		
953	001140	000000		SLIM:	0		
954	001142	000000		BPC:	0		
955	001144	000000		TSYNC:	0		

J02

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 24
DZDPAB.SRC

956	001146	000000
957	001150	000000
958	001152	000000
959	001154	000000
960	001156	000000
961	001160	000000
962	001162	000000
963	001164	000000
964	001166	000000
965	001170	000000
966	001172	000000
967	001174	000000
968		

XLINEX:	0
CABLE:	00
TDATA:	00
RDATA:	00
CHLEN:	00
LIMIT:	00
SCNT:	00
SAVSR2:	00
TIME:	00
TP:	00
RP:	00
BACK:	0

```

969
970
971
972 001176 000
973 001177 000
974 001200 000
975 001201 000
976      000000
977
978
979
980
981
982
983
984 001202
985      104400
986 001202 015640
987      104401
988 001204 013342
989      104402
990 001206 013402
991      104403
992 001210 014300
993      104404
994 001212 012752
995      104405
996 001214 012670
997      104406
998 001216 016052
999      104407
1000 001220 016070
1001      104410
1002 001222 016676
1003      104411
1004 001224 016736
1005      104412
1006 001226 016476
1007      104413
1008 001230 016502
1009
1010
1011
1012      09100
1013      09200
1014      09300
1015      09400
1016 001232 000001
1017 001234 000001
1018 001236 000001
1019 001240 000001
1020 001242 000001
1021 001244 000001
1022
1023
1024

```

;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
SAVOS=TRAP+10 ;CALL TO REGISTER SAVE ROUTINE
      .SAVOS
RESOS=TRAP+11 ;CALL TO REGISTER RESTORE ROUTINE
      .RESOS
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

```

```

;*****

```

```

1025 001246 000001      10400
1026 001250 000001      10500
1027 001252 000001      10600
1028 001254 000001      10700
1029 001256 000300      10800
1030 001260 174770      10900
1031      11000
1032      11100
1033      11200
1034 001262 000005      11300
1035 001264 005037 001150 11400
1036 001270 012706 001050 11500
1037 001274 012737 000340 177776 11600
1038 001302 105737 177570 11700
1039 001306 100005 11800
1040 001310 004737 012462 11900
1041 001314 004737 012202 12000
1042 001320 000404 12100
1043 001322 004737 012462 12200
1044 001326 004737 012246 12300
1045 001332 005737 001150 12400
1046 001336 001402 12500
1047 001340 000137 007254 12600
1048 001344 012737 001352 001060 12700
1049      12800
1050      12900
1051      13000
1052      13100
1053
1054
1055
1056
1057
1058
1059 001352 012737 000001 001072 13300
1060 001360 012737 001464 001062 13400
1061 001366 012737 000340 177776 13500
1062 001374 005077 177640 13600
1063 001400 005077 177626 13700
1064 001404 012777 014500 177640 13800
1065 001412 012777 014504 177626 13900
1066 001420 012777 000240 177622 14000
1067 001426 012777 000240 177620 14100
1068 001434 112777 000026 177574 14200
1069 001442 052777 000004 177562 14300
1070      14400
1071 001450 032777 000004 177554 14500
1072 001456 001001 14600
1073 001460 104000 14700
1074 001462 104400 14800

```

```

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
STAR: TSTB SWR ;IS SWITCH SEVEN SET??
        BPL BGND ;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.
BGND: 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
        MOV #TST2,NEXT
        MOV #340,PS ;SET PROCESSOR STATUS TO 7
        CLR @DPTS ;CLEAR TRANSMITTER 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

```

M02

1075 14800
1076 14900
1077 15000
1078 15100
1079 15200

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

1080
1081
1082
1083
1084

*
: TEST 2
*

1085
1086 001464 012737 000002 001072
1087 001472 012737 001572 001062
1088 001500 005000 15400
1089 001502 013737 001232 013056 15500
1090 001510 004537 012510 15600
1091 001514 000001 15700
1092 001516 004537 012510 15800
1093 001522 000002 15900
1094 001524 004537 012510 16000
1095 001530 000004 16100
1096 001532 004537 012510 16200
1097 001536 000100 16300
1098 001540 004537 012510 16400
1099 001544 000200 16500
1100 001546 004537 012510 16600
1101 001552 000400 16700
1102 001554 004537 012510 16800
1103 001560 001000 16900
1104 001562 004537 012510 17000
1105 001566 002000 17100
1106 001570 104400 17200
1107 17300
1108 17400
1109 17500
1110 17600
1111 17700
1112 17800
1113 17900
1114 18000
1115 18100
1116 18200

TST2: MOV #2,TSTNO
MOV #TST3,NEXT
CLR RO
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

1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122

;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

1123
1124
1125
1126
1127
1128
1129
1130

*
: TEST 3
*

1123 001572 012737 000003 001072
1124 001600 012737 001722 001062
1125 001606 005000 18400
1126 001610 013737 001240 013056 18500
1127 001616 004537 012510 18600
1128 001622 000001 18700
1129 001624 042777 000004 177400 18800
1130 001632 004537 012510 18900

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

NO2

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 28
 DZDPAB.SRC

1131	001636	000002		19000
1132	001640	052777	000004 177364	19100
1133	001646	004537	012510	19200
1134	001652	000010		19300
1135	001654	004537	012510	19400
1136	001660	000040		19500
1137	001662	004537	012510	19600
1138	001666	000100		19700
1139	001670	004537	012510	19800
1140	001674	000200		19900
1141	001676	004537	012510	20000
1142	001702	020000		20100
1143	001704	004537	012510	20200
1144	001710	100000		20300
1145	001712	004537	012510	20400
1146	001716	040000		20500
1147	001720	104400		20600
1148				20700
1149				20800
1150				20900
1151				21000
1152				21100
1153				21200
1154				21300
1155				21400
1156				21500
1157				21600
1158				21700
1159				21800
1160				21900
1161				22000
1162				22100
1163				22200
1164				
1165				
1166				
1167				
1168				
1169				
1170	001722	012737	000004 001072	
1171	001730	012737	002002 001062	
1172	001736	005077	177276	22400
1173	001742	012777	000004 177262	22500
1174	001750	012777	143707 177254	22600
1175	001756	017701	177250	22700
1176	001762	042701	010000	22800
1177	001766	012700	003707	22900
1178	001772	020001		23000
1179	001774	001401		23100
1180	001776	104001		23200
1181	002000	104400		23300
1182				23400
1183				23500
1184				23600
1185				23700
1186				23800

```

BIT1
BIS #BIT2,JDPRS ;START CLOCK
JSR R5,BITST ;SECONDARY TRANSMIT R/W
BIT3
JSR R5,BITST ;STATUS INTERRUPT ENABLE R/W
BIT5
JSR R5,BITST ;TRANSMITTER INTERRUPT R/W
BIT6
JSR R5,BITST ;DONE
BIT7
JSR R5,BITST ;RING FLAG R/W
BIT13
JSR R5,BITST ;CARRIER DOWN
BIT15
JSR R5,BITST ;RECEIVER OVERUN FLAG R/W
BIT14
SCOPE
  
```

```

;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 JDPTS ;CLEAR TRANSMITTER STATUS
      MOV #BIT2,JDPRS ;MAINTENANCE MODE
      MOV #143707,JDPRS ;SET ALL RECEIVER STATUS BITS
      MOV JDPRS,R1 ;SAVE THE RX STATUS
      BIC #BIT12,R1 ;CLEAR THE PARITY BIT
      MOV #3707,RO ;SET RO FOR ERROR MESSAGE
      CMP RO,R1 ;IS THE STATUS WHAT I EXPECTED??
      BEQ +4 ;BR IF STATUS IS OK.
      HLT i ;ERROR RX STATUS NOT WHAT EXPECTED.
      SCOPE ;SCOPE THIS TEST.
  
```

```

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



```

1 002134 012737 000007 001072
1 002142 012737 002206 001062
1 002150 012777 000004 177054 27800
1 002156 012777 160353 177054 27900
1 002164 000005 28000
1 002166 005100 28100
1 002170 017701 177044 28200
1 002174 005701 28300
1 002176 001402 28400
1 002200 005000 28500
1 002202 104001 28600
1 002204 104400 28700
1 28800
1 28900
1 29000
1 29100
1 29200
1 002206 012737 000010 001072
1 002214 012737 002266 001062
1 002222 012777 160377 177010 29400
1 002230 000005 29500
1 002232 005100 29600
1 002234 017701 177000 29700
1 002240 105777 176774 29800
1 002244 100002 29900
1 002246 005000 30000
1 002250 104001 30100
1 002252 005777 176762 30200
1 002256 001402 30300
1 002260 005000 30400
1 002262 104001 30500
1 002264 104400 30600
1 30700
1 30800
1 30900
1 31000
1 31100
1 31200
1 31300
1 31400
1 31500
1 002266 012737 000011 001072

```

```

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

```

:TEST READY BIT CLEAR BEFORE READY CAN COME UP

```

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

```

: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

```

1299	002274	012737	002354	001062	
1300	002302	005000			31700
1301	002304	013737	001232	013056	31800
1302	002312	104405			31900
1303	002314	000001			32000
1304	002316	104405			32100
1305	002320	000002			32200
1306	002322	104405			32300
1307	002324	000004			32400
1308	002326	104405			32500
1309	002330	000100			32600
1310	002332	104405			32700
1311	002334	000200			32800
1312	002336	104405			32900
1313	002340	000400			33000
1314	002342	104405			33100
1315	002344	001000			33200
1316	002346	104405			33300
1317	002350	002000			33400
1318	002352	104400			33500
1319					33600
1320					33700
1321					33800
1322					
1323					
1324					
1325					
1326					
1327					
1328	002354	012737	000012	001072	
1329	002362	012737	002452	001062	
1330	002370	005000			34000
1331	002372	013737	001240	013056	34100
1332	002400	005077	176634		34200
1333	002404	104405			34300
1334	002406	000001			34400
1335	002410	104405			34500
1336	002412	000002			34600
1337	002414	104405			34700
1338	002416	000010			34800
1339	002420	104405			34900
1340	002422	000040			35000
1341	002424	104405			35100
1342	002426	000100			35200
1343	002430	104405			35300
1344	002432	000200			35400
1345	002434	104405			35500
1346	002436	020000			35600
1347	002440	104405			35700
1348	002442	040000			35800
1349	002444	104405			35900
1350	002446	100000			36000
1351	002450	104400			36100
1352					36200
1353					36300
1354					36400

```

MOV #TST12,NEXT
CLR RO
MOV DPRS,REG
VALID
BIT0
VALID
BIT1
VALID
BIT2
VALID
BIT6
VALID
BIT7
VALID
BIT8
VALID
BIT9
VALID
BIT10
SCOPE

```

```

;TEST RCV
;STRIP SYNC
;HALF DUPLEX
;MAINTENANCE MODE
;INTERRUPT ENABLE
;DONE
;BITS/CHAR
;" "
;" "

```

; TRANSMITTER STATUS BIT VALIDITY TEST

```

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

```

```

TST12: MOV #12,TSTNO
MOV #TST13,NEXT
CLR RO
MOV DPTS,REG
CLR DPTS
VALID
BIT0
VALID
BIT1
VALID
BIT3
VALID
BIT5
VALID
BIT6
VALID
BIT7
VALID
BIT13
VALID
BIT14
BIT15
SCOPE

```

```

;TEST XMIT STATUS
;CLEAR TRANSMITTER STATUS
;TERMINAL READY
;IDLE SYNC
;MISC TRANSMIT
;STATUS ENABLE
;TRANSMITTER ENABLE
;TRANSMITTER DONE
;RING FLAG
;RECEIVER OVERRUN
;CARRIER DOWN

```

```

1355      36500
1356      36600
1357      36700
1358      36800
1359      36900
1360
1361
1362
1363
1364
1365
1366      002452  012737  000013  001072
1367      002460  012737  002560  001062
1368      002466  112777  000026  176542  37100
1369      002474  013737  001232  013056  37200
1370      002502  012737  003707  001136  37300
1371      002510  012777  003707  176514  37400
1372      002516  104404  37500
1373      002520  000001  37600
1374      002522  104404  37700
1375      002524  000002  37800
1376      002526  104404  37900
1377      002530  000004  38000
1378      002532  104404  38100
1379      002534  000100  38200
1380      002536  104404  38300
1381      002540  000200  38400
1382      002542  104404  38500
1383      002544  000400  38600
1384      002546  104404  38700
1385      002550  001000  38800
1386      002552  104404  38900
1387      002554  002000  39000
1388      002556  104400  39100
1389      39200
1390      39300
1391      39400
1392      39500
1393      39600

```

;VERIFY BIT CLEAR ONLY CLEARS SPECIFIED BIT

;RECEIVER TEST SECTION

TEST 13

```

TST13:  MOV      #13,TSTNO
        MOV      #TST14,NEXT
        MOV      #26,SYNC      ;LOAD SYNC WITH ANYTHING
        MOV      DPRS,REG      ;TEST RCV STATUS
        MOV      #3707,TMPDAT  ;STORE STATUS IMAGE
        MOV      #3707,DPRES   ;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

```

1394          39800
1395
1396
1397
1398
1399
1400
1401 002560 012737 000014 001072
1402 002566 012737 003044 001062
1403 002574 012737 000340 177776 40000
1404 002602 012777 000004 176422 40100
1405 002610 012777 160353 176422 40200
1406 002616 012737 163353 001136 40300
1407 002624 012700 005050          40400
1408 002630 005300          40500
1409 002632 001376          40600
1410 002634 032777 001000 176376 40700
1411 002642 001001          40800
1412 002644 104000          40900
1413 002646 013737 001240 013056 41000
1414 002654 104404          41100
1415 002656 000001          41200
1416          41300
1417          41400
1418 002660 042737 003002 001136 41500
1419 002666 042777 000002 176344 41600
1420 002674 012700 007020          41700
1421 002700 005300          41800
1422 002702 001376          41900
1423 002704 013700 001136          42000
1424 002710 017701 176324          42100
1425 002714 023777 001136 176316 42200
1426 002722 001401          42300
1427 002724 104001          42400
1428 002726 052737 003002 001136 42500
1429 002734 052777 000002 176276 42600
1430 002742 104404          42700
1431 002744 000010          42800
1432 002746 104404          42900
1433 002750 000040          43000
1434 002752 104404          43100
1435 002754 000100          43200
1436 002756 042777 000004 176246 43300
1437 002764 042737 002000 001136 43400
1438 002772 104404          43500
1439 002774 000200          43600
1440 002776 052777 000004 176226 43700
1441 003004 052737 002000 001136 43800
1442 003012 042777 004004 176212 43900
1443 003020 042737 002000 001136 44000
1444 003026 104404          44100
1445 003030 020000          44200
1446 003032 104404          44300
1447 003034 040000          44400
1448 003036 104404          44500
1449 003040 100000          44600

```

```

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

```

```

1450 003042 104400 44700
1451 44800
1452 44900
1453 45000
1454 45100
1455 45200
1456 45300
1457 45400
1458 45500
1459
1460
1461
1462
1463
1464
1465 003044 012737 000015 001072
1466 003052 012737 003260 001062
1467 003060 012737 000240 177776 45700
1468 003066 012777 014500 176156 45800
1469 003074 012777 014504 176144 45900
1470 003102 052777 000004 176122 46000
1471 003110 052777 000040 176122 46100
1472 003116 052777 010000 176114 46200
1473 003124 042777 020040 176106 46300
1474 003132 052777 000040 176100 46400
1475 003140 052777 040000 176072 46500
1476 003146 042777 040040 176064 46600
1477 003154 052777 000040 176056 46700
1478 003162 052777 100000 176050 46800
1479 003170 042777 100040 176042 46900
1480 47000
1481 47100
1482 47200
1483 003176 052777 000100 176034 47300
1484 003204 052777 000001 176026 47400
1485 003212 052777 000002 176020 47500
1486 003220 052777 000004 176012 47600
1487 003226 052777 000010 176004 47700
1488 003234 052777 000020 175776 47800
1489 003242 042777 000037 175770 47900
1490 003250 042777 000100 175762 48000
1491 003256 104400 48100
1492 48200
1493 48300
1494 48400
1495 48500
1496 48600
1497 48700
1498
1499
1500
1501
1502
1503
1504 003260 012737 000016 001072
1505 003266 012737 003416 001062

```

SCOPE

```

;*****
;PRIORITY TESTS
;VERIFY THAT THERE ARE NO STATUS INTERRUPTS
;WHEN PS=5
;INTERRUPT WILL TRAP TO ERROR MESSAGE
;*****
;
; TEST 15
;
;*****
;*****
TST15: MOV #15,TSTNO
MOV #TST16,NEXT
MOV #240,PS ;PRIORITY=5
MOV #FTINT,ADPTIV ;SET FOR UNEXPECTED INTERUPT.
MOV #FRINT,ADPRIV ;SET FOR UNEXPECTED INTERUPT.
BIS #BIT2,ADPRS ;SET MAINTENANCE MODE
BIS #BIT5,ADPTS ;STATUS INTERRUPT ENABLE (SIE)
BIS #BIT12,ADPTS ;RING FLAG
BIC #20040,ADPTS ;CLEAR CSR
BIS #BIT5,ADPTS ;INT ENB (STATUS)
BIS #BIT14,ADPTS ;RECEIVER OVERRUN FLAG
BIC #40040,ADPTS ;CLEAR CSR
BIS #BIT5,ADPTS ;INT ENB (STATUS)
BIS #BIT15,ADPTS ;CARRIER DOWN FLAG
BIC #100040,ADPTS ;CLEAR CSR
;VERIFY NO TRANSMITTER READY INTERRUPTS
BIS #BIT6,ADPTS ;XMIT INT ENB
BIS #BIT0,ADPTS ;NOISE
BIS #BIT1,ADPTS ;MORE NOISE
BIS #BIT2,ADPTS ;MORE NOISE
BIS #BIT3,ADPTS ;MORE NOISE
BIS #BIT4,ADPTS ;MORE NOISE
BIC #37,ADPTS ;QUIET!
BIC #BIT6,ADPTS
SCOPE
;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
;
;*****
;*****
TST16: MOV #16,TSTNO
MOV #TST17,NEXT

```

H03

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 35
DZDPAB.SRC

```
1506 003274 012737 000200 177776 48900      MOV      #200,PS          ;PRIORITY=4
1507 003302 012777 014500 175742 49000      MOV      #FTINT,ADPTIV   ;SET FOR UNEXPECTED TX INTER.
1508 003310 012777 014504 175730 49100      MOV      #FRINT,ADPKIV   ;SET FOR UNEXPECTED RX INTER.
1509 003316 052777 020000 175714 49200      1$:     BIS      #BIT13,ADPTS ;RING FLAG
1510 003324 042777 020000 175706 49300      BIC      #BIT13,ADPTS   ;CLEAR
1511 003332 052777 040000 175700 49400      BIS      #BIT14,ADPTS   ;RECEIVER OVERRUN FLAG
1512 003340 042777 040000 175672 49500      BIC      #BIT14,ADPTS   ;CLEAR
1513 003346 052777 100000 175664 49600      BIS      #BIT15,ADPTS   ;CARRIER DOWN FLAG
1514 003354 042777 100000 175656 49700      BIC      #BIT15,ADPTS
1515 003362 052777 160000 175650 49800      BIS      #160000,ADPTS  ;SET ALL STATUS ERROR BITS
1516 003370 042777 160000 175642 49900      BIC      #160000,ADPTS  ;CLEAR
1517 003376 162737 000040 177776 50000      SUB      #40,PS        ;DECREASE PRIORITY LEVEL
1518 003404 032737 000340 177776 50100      BIT      #340,PS
1519 003412 001341 000340 177776 50200      BNE
1520 003414 104400 000340 177776 50300      SCOPE
1521 003414 104400 000340 177776 50400
1522 003414 104400 000340 177776 50500
1523 003414 104400 000340 177776 50600
1524 003414 104400 000340 177776 50700
1525 003414 104400 000340 177776 50800
1526 003414 104400 000340 177776 50900
1527 003414 104400 000340 177776 51000
1528 003414 104400 000340 177776 51100
1529
1530
1531
1532
1533
1534
1535 003416 012737 000017 001072 51300
1536 003424 012737 003602 001062 51400
1537 003432 012777 003466 175612 51500
1538 003440 012737 000200 177776 51600
1539 003446 052777 000040 175564 51700
1540 003454 052777 020000 175556 51800
1541 003462 000240 000240 175556 51900
1542 003464 104003 000240 175556 52000
1543 003466 042777 020000 175544 52100      1$:     HLT      3
1544 003474 022626 020000 175544 52200      BIC      #BIT13,ADPTS
1545 003476 012777 003524 175546 52300      POP.SP
1546 003504 042737 000040 177776 52400      MOV      #2$,ADPTIV
1547 003512 052777 040000 175520 52500      BIC      #BIT5,PS
1548 003520 000240 040000 175520 52600      BIS      #BIT14,ADPTS
1549 003522 104003 000240 175520 52700      NOP
1550 003524 042777 040000 175506 52800      HLT      3
1551 003532 022626 003524 175506 52900      2$:     BIC      #BIT14,ADPTS
1552 003534 012777 003562 175510 53000      POP.SP
1553 003542 042737 000040 177776 53100      MOV      #3$,ADPTIV
1554 003550 052777 100000 175462 53200      BIC      #BIT5,PS
1555 003556 000240 100000 175462 53300      BIS      #BIT15,ADPTS
1556 003560 104003 000240 175462 53400      NOP
1557 003562 042777 100000 175450 53500      HLT      3
1558 003570 022626 014500 175452 53600      3$:     BIC      #BIT15,ADPTS
1559 003572 012777 014500 175452 53700      POP.SP
1560 003600 104400 014500 175452 53800      MOV      #FTINT,ADPTIV
1561 003600 104400 014500 175452 53900      SCOPE
1562 003600 104400 014500 175452 54000
```

;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$,ADPTIV          ;SET UP INTERRUPT VECTOR=RTI
        MOV      #200,PS            ;PRIORITY=4
        BIS      #BIT5,ADPTS        ;INT ENB STATUS
        BIS      #BIT13,ADPTS       ;RING FLAG
        NOP                          ;SHOULD INTERRUPT AFTER NOP
        HLT      3                  ;REPORT ERROR
        BIC      #BIT13,ADPTS       ;CLEAR RING
        POP.SP                       ;ADJUST STACK
        MOV      #2$,ADPTIV        ;SET UP NEXT INTERRUPT VECTOR
        BIC      #BIT5,PS           ;SET PRIORITY TO 4
        BIS      #BIT14,ADPTS       ;RECEIVER OVERRUN FLAG
        NOP
        HLT      3                  ;REPORT ERROR
        BIC      #BIT14,ADPTS       ;CLEAR RCV O RUN
        POP.SP                       ;ADJUST STACK
        MOV      #3$,ADPTIV        ;SET UP NEXT INTERRUPT VECTOR
        BIC      #BIT5,PS           ;SET PRIORITY TO 4
        BIS      #BIT15,ADPTS       ;CARRIER DOWN FLAG
        NOP
        HLT      3                  ;REPORT ERROR
        BIC      #BIT15,ADPTS       ;CLEAR CARRIER DOWN FLAG
        POP.SP                       ;ADJUST STACK
        MOV      #FTINT,ADPTIV      ;FALSE INT TRAP
        SCOPE                       ;YES
```

```

1562          53800
1563          53900
1564          54000
1565
1566
1567
1568
1569
1570
1571 003602 012737 000020 001072
1572 003610 012737 003656 001062
1573 003616 005077 175410 54200
1574 003622 012777 000200 175410 54300
1575 003630 105777 175404 54400
1576 003634 100401 54500
1577 003636 104000 54600
1578 003640 010077 175376 54700
1579 003644 105777 175370 54800
1580 003650 100001 54900
1581 003652 104000 55000
1582 003654 104400 55100
1583          55200
1584          55300
1585          55400
1586          55500
1587
1588
1589
1590
1591
1592
1593 003656 012737 000021 001072
1594 003664 012737 003760 001062
1595 003672 012777 000004 175332 55700
1596 003700 005077 175334 55800
1597 003704 012777 003750 175340 55900
1598 003712 012737 000200 177776 56000
1599 003720 110077 175316 56100
1600 003724 052777 000100 175306 56200
1601 003732 105777 175302 56300
1602 003736 100001 56400
1603 003740 104000 56500
1604 003742 104403 004704 56600
1605 003746 104003 56700
1606 003750 005077 175264 56800
1607 003754 022626 56900
1608 003756 104400 57000
1609          57100
1610          57200
1611          57300
1612          57400
1613          57500
1614          57600
1615
1616
1617

```

; 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 3 ; TRANSMITTER FAILED TO INTERRUPT
1$: CLR @DPTS ; CLEAR INT ENB
   POP.SP ; ADJUST STACK
   SCOPE

```

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

```

*****
:
: TEST 22

```

J03

```

1618
1619
1620
1621 003760 012737 000022 001072
1622 003766 012737 004074 001062
1623 003774 012777 000004 175230 57800
1624 004002 005077 175232 57900
1625 004006 012777 004060 175236 58000
1626 004014 012737 000200 177776 58100
1627 004022 052777 000400 175202 58200
1628 004030 110077 175206 58300
1629 004034 052777 000100 175176 58400
1630 004042 105777 175172 58500
1631 004046 100001 58600
1632 004050 104000 58700
1633 004052 104403 004704 58800
1634 004056 104003 58900
1635 004060 005077 175146 59000
1636 004064 005077 175150 59100
1637 004070 022626 59200
1638 004072 104400 59300
1639 59400
1640 59500
1641 59600
1642 59700
1643
1644
1645
1646
1647
1648
1649 004074 012737 000023 001072
1650 004102 012737 004204 001062
1651 004110 012777 000004 175114 59900
1652 004116 012737 000200 177776 60000
1653 004124 005077 175110 60100
1654 004130 012777 004174 175114 60200
1655 004136 052777 001000 175066 60300
1656 004144 110077 175072 60400
1657 004150 052777 000100 175062 60500
1658 004156 105777 175056 60600
1659 004162 100001 60700
1660 004164 104000 60800
1661 004166 104403 004704 60900
1662 004172 104003 61000
1663 004174 005077 175032 61100
1664 004200 022626 61200
1665 004202 104400 61300
1666 61400
1667 61500
1668 61600
1669 61700
1670 61800
1671 61900
1672 62000
1673 62100

```

```

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

```

```

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

```

```

*****
*
*
TEST 23
*
*****

```

```

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

```

```

;TEST SYNC BUFFER IS READ/WRITE
;NOTE: SW09=1 WILL FREEZE ON CURRENT SYNC

```

K03

```

1674
1675
1676
1677
1678
1679
1680 004204 012737 000024 001072
1681 004212 012737 004326 001062
1682 004220 012737 004256 001064
1683 004226 005037 001136 62300
1684 004232 005077 174774 62400
1685 004235 005077 174776 62500
1686 004242 105077 174770 62600
1687 004246 105777 174764 62700
1688 004252 001401 62800
1689 004254 104000 62900
1690 004256 113777 001136 174752 63000
1691 004264 113700 001136 63100
1692 004270 117701 174742 63200
1693 004274 123777 001136 174734 63300
1694 004302 001401 63400
1695 004304 104001 63500
1696 004306 104406 63600
1697 004310 105237 001136 63700
1698 004314 001360 63800
1699 004316 112777 000026 174712 63900
1700 004324 104400 64000
1701 64100
1702 64200
1703 64300
1704 64400
1705
1706
1707
1708
1709
1710
1711 004326 012737 000025 001072
1712 004334 012737 004462 001062
1713 004342 032737 000400 001134 64600
1714 004350 001005 64700
1715 004352 013737 001062 001060 64800
1716 004360 000177 174474 64900
1717 004364 005037 001136 65000
1718 004370 112777 000017 174646 65100
1719 004376 113777 001136 174640 65200
1720 004404 117701 174634 65300
1721 004410 113700 001136 65400
1722 004414 127737 174624 001136 65500
1723 004422 001401 65600
1724 004424 104001 65700
1725 004426 104406 65800
1726 004430 005237 001136 65900
1727 004434 022737 000020 001136 66000
1728 004442 001355 66100
1729 004444 105077 174574 66200

```

```

*****
: 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 ;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
HLT ;REPORT ERROR
SCOP1
INC TMPDAT ;NEXT SYNC
CMP #20,TMPDAT ;HAVE ALL SYNC'S BEEN TESTED
BNE 2$ ;NO, CONTINUE TEST
CLRB @SEXT ;CLEAR SYNC EXT

```

```

1730 004450 105777 174570 66300
1731 004454 001401 66400
1732 004456 104000 66500
1733 004460 104400 66600
1734 66700
1735 66800
1736 66900
1737 67000
1738 67100
1739 67200
1740 67300
1741 67400
1742 67500
1743 67600
1744 67700
1745 67800
1746 67900
1747 68000
1748
1749
1750
1751
1752
1753
1754 004462 012737 000026 001072 68200
1755 004470 012737 005076 001062 68300
1756 004476 105077 174542 68400
1757 004502 005077 174532 68500
1758 004506 005077 174520 68600
1759 004512 012737 000200 177776 68700
1760 004520 012737 000377 001140 68800
1761 004526 012737 000400 001142 68900
1762 004534 052777 000004 174470 69000
1763 004542 012737 000001 001144 69100
1764 004550 012777 014504 174470 69200
1765 004556 104403 013560 174446 69300
1766 004562 113777 001144 174442 69400
1767 004570 117737 174442 001152 69500
1768 004576 113777 001144 174436 69600
1769 004604 105777 174430 69700
1770 004610 100375 69800
1771 004612 032777 004000 174412 69900
1772 004620 001401 67000
1773 004622 104000 70100
1774 004624 113777 001144 174410 70200
1775 004632 012777 004670 174406 70300
1776 004640 105777 174374 70400
1777 004644 100375 70500
1778 004646 013777 001144 174366 70600
1779 004654 052777 000100 174350 70700
1780 004662 104403 005670 70800
1781 004666 104004 70900
1782 004670 017737 174340 001136 71000
1783 004676 013701 001136 71100
1784 004702 013700 001144
1785 004706 023737 001136 001144

```

```

TSTB @SEXT ;TEST SYNC EXT
BEQ .+4 ;BRANCH IF SYNC CLEARED
HLT ;REPORT ERROR
SCOPE

```

***** SYNC TESTS *****

```

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

```

```

*****
; TEST 26
*****

```

```

TST26: MOV #26,TSTNO
MOV #TST27,NEXT
CLRB @SEXT ;CLEAR SYNC EXTENSION
CLR @DPTS ;CLEAR TRANSMITTER STATUS
CLR @DPRS ;CLEAR RECEIVER STATUS
MOV #200,PS ;PRIORITY=4
MOV #377,SLIM ;SYNC LIMIT FOR 8BITS/CHAR
MOV #400,BPC ;INDEX TO CHANGE BITS/CHAR
BIS #BIT2,@DPRS ;MAINT MODE
MOV #1,TSYNC ;FIRST SYNC = 1
1$: MOV #FRINT,@DPRIV ;SET UP RECEIVER INT VECTOR TO ERROR
2$: DELAY 6000. ;10 CHAR TIMES FOR ALL 1'S IN BUFF
MOVSB TSYNC,@SYNC ;LOAD SYNC BUFFER
MOVSB @SYNC,TDATA ;STORE SYNC
MOVSB TSYNC,@DPTB ;LOAD FIRST SYNC CHAR
TSTB @DPTS ;READY FOR NEXT SYNC
BPL -4 ;NO TEST AGAIN
BIT #BIT11,@DPRS ;TEST FOR PREMATURE ACTIVE
BEQ .+4 ;BRANCH IF NOT SET
HLT ;PREMATURE ACTIVE
MOVSB TSYNC,@DPTB ;LOAD SECOND SYNC BYTE
MOV #3,@DPRIV ;SET UP TEST VECTOR
TSTB @DPTS ;TRANSMITTER READY
BPL -4 ;NO
MOV TSYNC,@DPTB ;XMIT 3ED SYNC AS DATA
BIS #BIT6,@DPRS ;RCV INT ENB
DELAY 3000. ;STALL 10 CHARACTER TIMES
HLT 4 ;REPORT ERROR
3$: MOV @DPRB,TMPDAT ;SAVE DATA
MOV TMPDAT,R1
MOV TSYNC,R0
CMP TMPDAT,TSYNC ;CORRECT SYNC CHARACTER

```

M03

```

1786 004714 001404          71200
1787 004716 042777 000004 174306 71300
1788 004724 104001          71400
1789 004726 105777 174300          71500
1790 004732 100001          71600
1791 004734 104000          71700
1792 004736 042777 000100 174266 71800
1793 004744 032777 004000 174260 71900
1794 004752 001001          72000
1795 004754 104000          72100
1796 004756 112777 000026 174252 72200
1797 004764 042777 004000 174240 72300
1798 004772 032777 004000 174232 72400
1799 005000 001401          72500
1800 005002 104000          72600
1801 005004 022626          72700
1802 005006 042737 000040 177776 72800
1803 005014 032737 040000 177570 72900
1804 005022 001002          73000
1805 005024 105237 001144          73100
1806 005030 123737 001140 001144 73200
1807 005036 001247          73300
1808 005040 005037 001144          73400
1809 005044 053777 001142 174160 73500
1810 005052 006237 001140          73600
1811 005056 062737 000400 001142 73700
1812 005064 022737 001400 001142 73800
1813 005072 001226          73900
1814 005074 104400          74000
1815          74100
1816          74200
1817          74300
1818
1819
1820
1821
1822
1823
1824 005076 012737 000027 001072
1825 005104 012737 005510 001062
1826 005112 032737 000400 001134 74500
1827 005120 001005          74600
1828 005122 013737 001062 001060 74700
1829 005130 000177 173726          74800
1830 005134 012777 002104 174070 74900
1831 005142 012737 000400 001144 75000
1832 005150 105077 174062          75100
1833 005154 112777 000001 174062 75200
1834 005162 012737 007400 001140 75300
1835 005170 012737 002000 001142 75400
1836 005176 012777 014504 174042 75500
1837 005204 104403 006000          75600
1838          75700
1839 005210 105077 174022          75800
1840 005214 113777 001145 174022 75900
1841 005222 013777 001144 174012 76000

```

```

4$: BEQ 4$ ;YES
    BIC #BIT2, @DPRS ;NO SHUT OFF CLOCK
    HLT 1 ;REPORT ERROR
    TSTB @DPRS ;DONE CLEARED
    BPL .+4 ;YES
    HLT ;REPORT ERROR
    BIC #BIT6, @DPRS ;CLEAR REV INT ENB
    BIT #BIT11, @DPRS ;RECEIVER ACTIVE
    BNE .+4 ;YES
    HLT ;REPORT ERROR
    MOVB #26, @SYNC ;CHANGE SYNC
    BIC #BIT11, @DPRS ;CLEAR RECEIVER ACTIVE
    BIT #BIT11, @DPRS ;RCV ACTIVE CLEARED
    BEQ .+4 ;YES
    HLT ;REPORT ERROR
    POP, SP ;ADJUST STACK
    BIC #BIT5, PS ;SET PRIORITY TO 4
    BIT #BIT14, SWR ;TEST FOR SCOPE LOOP
    BNE 5$ ;BRANCH AROUND INC IF SCOPE
    INCB TSYNC ;NEXT SYNC
    CMPB SLIM, TSYNC ;HAVE ALL SYNC'S BEEN TESTED
    BNE 2$ ;NO
    CLR TSYNC ;YES
    BIS BPC, @DPRS ;DEC BITS/CHAR BY 1 BIT
    ASR SLIM ;DECREASE #BITS/CHAR
    ADD #400, BPC ;DEC BITS/CHAR BY 1 BIT
    CMP #1400, BPC ;HAVE ALL CHAR SEIZES BEEN TESTED
    BNE 1$ ;NO
    SCOPE

;*****SYNC EXTENSION TEST*****
;*****
; *
; TEST 27
; *
;*****
;*****
TST27: MOV #27, TSTNO
      MOV #TST30, NEXT
      BIT #BIT8, SAVSR1 ;DOES TWELVE BIT OPTION EXIST
      BNE 1$ ;BRANCH IF NOT
      MOV NEXT, RETURN
      JMP @NEXT
1$: MOV #2104, @DPRS ;SET STATUS TO 12 BITS/CHARACTER
   MOV #400, TSYNC ;FIRST SYNC CHARACTER
   CLRB @SYNC ;LOAD SYNC BUFFERS WITH 400
   MOVB #1, @SEXT ;LOAD SYNC BUFFERS WITH 400
   MOV #7400, SLIM ;SET UP SYNC LIMIT
   MOV #2000, BPC ;SET # BITS/CHAR TO 12
2$: MOV #FRINT, @DPRIV ;RCV INTERRUPT VECTOR = ERROR
   DELAY ,6000 ;WAIT FOR ALL 1'S
   CLRB @SYNC ;TO SHIFT INTO XMIT, RCV BUFS
   MOVB TSYNC+1, @SEXT ;CLEAR SYNC EXTENTION
   MOV TSYNC, @DPTB ;LOAD NEXT SYNC
   ;TRANSMIT FIRST SYNC

```



```

00100
00200
00300
00400
00500
00600
00700
00800
00900
005510 012737 000030 001072
005516 012737 006104 001062
005524 012737 000026 001144
005532 005077 173502
005536 005077 173470
005542 013703 001232
005546 005203
005550 052777 000004 173454
005556 012702 006052
104403 013560
012777 005640 173452
113777 001144 173434
113777 001145 173434
013777 001144 173424
052777 000002 173414
052777 000100 173400
104403 027340
104004
017737 173370 001136
000412
013700 001144
013701 001136
023737 001136 001144
001401
000415
105737 001137
001401
104000
013700 001144
013701 001136
123737 001136 001144
001401
104001
105777 173300
100001
104000
042777 004000 173266
032777 004000 173260 04900

```

```

:*****
: IDLE SYNC TEST
: RAISE "ACTIVE" BY IDLEING IN EACH AVAILABLE CHARACTER LENGTH
:*****
: TEST 30
:*****
:*****
ST30:  MOV    #30, TSTNO
      MOV    #TST31, NEXT
      MOV    #26, TSYNC
      CLR    @DPTS
      CLR    @DPRS
      MOV    DPRS, R3
      INC    R3
      BIS    #BIT2, @DPRS
      MOV    #6$, R2
1$:   DELAY  ,6000.
      MOV    #2$, @DPRIV
      MOVB   TSYNC, @SYNC
      MOVB   TSYNC+1, @SEXT
      MOV    TSYNC, @DPTB
      BIS    #BIT1, @DPTS
      BIS    #BIT6, @DPRS
      DELAY  ,12000.
      HLT    4
2$:   MOV    @DPRB, TMPDAT
3$:   BR     4$
      MOV    TSYNC, R0
      MOV    TMPDAT, R1
      CMP    TMPDAT, TSYNC
      BEQ    ,+4
      HLT    1
      BR     5$
4$:   TSTB  TMPDAT+1
      BEQ    ,+4
      HLT    5$
5$:   TSTB  @DPRS
      BPL    ,+4
      HLT
      BIC    #BIT11, @DPRS
      BIT    #BIT11, @DPRS

```

```

: LOAD TEST SYNC CHARACTER
: CLEAR STATUS REGISTERS
:
: FETCH DEVICE ADRS
: CHANGE ADRS TO HIGH BYTE OF STATUS
: START MAINTENANCE
: SET UP CHARACTER LENGTH SELECTOR
:
: WAIT FOR ALL 1'S TO SHIFT IN
: LOAD DP RCV INTERRUPT VECTOR
: LOAD LOW BYTE OF SYNC
: LOAD SYNC EXTENSION BITS
: LOAD XMIT BUFFER
: SET IDLE SYNC
: SET RCV INTERRUPT ENABLE
:
: WAIT FOR RCV INTERRUPT
: REPORT ERROR
: SAVE RCV DATA
: CONTROL WORD 12 BITS=BR .+2
:
: DOES SYNC CHECK
: YES
: REPORT ERROR
:
: VERIFY ONLY 8 BITS WERE TRANSMITTED
: BRANCH IF OK
: REPORT ERROR
:
: CHECK SYNC IN LOW BYTE
: BRANCH IF SYNC OK
: REPORT ERROR
: DID READING RCV BUFF CLR DONE
: YES
: REPORT ERROR
: CLEAR ACTIVE
: ACTIVE CLEARED?

```

```

1942 005752 001401 05000
1943 005754 104000 05100
1944 005756 042777 000002 173254 05200
1945 005764 042737 000040 177776 05300
1946 005772 022626 05400
1947 005774 142213 05500
1948 005776 152213 05600
1949 006000 020227 006066 05700
1950 006004 001266 05800
1951 006006 032737 000400 001134 05900
1952 006014 001424 06000
1953 006016 032777 002000 173206 06100
1954 006024 001020 06200
1955 006026 052777 002000 173176 06300
1956 006034 024242 06400
1957 006036 042777 001400 173156 06500
1958 006044 012737 000400 005646 06600
1959 006052 012737 001426 001144 06700
1960 006060 000640 06800
1961 006062 000403 06900
1962 006064 001003 07000
1963 006066 012737 000026 001144 07100
1964 006074 012737 000412 005646 07200
1965 006102 104400 07300
1966 07400
1967 07500
1968 07600
1969 07700
1970
1971
1972
1973
1974
1975
1976 006104 012737 000031 001072
1977 006112 012737 006316 001062
1978 006120 012737 006314 001174 07900
1979 006126 105077 173112 08000
1980 006132 005037 001154 08100
1981 006136 005037 001152 08200
1982 006142 005077 173072 08300
1983 006146 012777 000005 173056 08400
1984 006154 012737 000400 001156 08500
1985 006162 032737 000400 001134 08600
1986 006170 001414 08700
1987 006172 012737 010000 001160 08800
1988 006200 052777 002000 173024 08900
1989 006206 012737 000426 001144 09000
1990 006214 105277 173024 09100
1991 006220 000406 09200
1992 006222 012737 000400 001160 09300
1993 006230 012737 000026 001144 09400
1994 006236 012777 014510 173006 09500
1995 006244 012777 014616 172774 09600
1996 006252 012737 000200 177776 09700
1997 006260 012737 000003 001162 09800

```

```

      BEQ      .+4          ;YES
      HLT
      BIC      #BIT1,ADPTS ;REPORT ERROR
      BIC      #BITS,PS    ;CLEAR IDLE
      POP      SP          ;LOWER PRIORITY TO 4
      BICB    (R2)+,AR3    ;ADJUST STACK
      BICB    (R2)+,AR3    ;CLEAR CHAR LENGTH
      CMP      R2,#6$+4    ;SELECT NEXT CHAR LENGTH
      BNE     1$          ;END OF MODE?
      BIT      #BIT9,SAVSR1 ;NO
      BEQ     7$          ;TEST 12 BITS/CHARACTER
      BIT      #BIT10,ADPRS ;NO
      BNE     7$          ;END OF 12 BIT TEST
      BIS      #BIT10,ADPRS ;YES
      CMP      -(R2),-(R2) ;NO
      BIC      #1400,ADPRS ;ADJUST CHAR SELECTION
      MOV      #400,3$     ;CLEAR CHAR LENGTH LSB'S
      BR      1$          ;CHANCE CONTROL WORD
      BR      1$          ;CHANGE SYNC
6$:   403
7$:   MOV      #26,TSYNC  ;CHARACTER LENGTH SELECTION
      MOV      #412,3$   ;CHARACTER LENGTH SELECTION
      SCOPE              ;RESTORE TSYNC
                          ;RESTORE CONTROL WORD

```

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

```

:*****
:
:TEST 31
:
:*****
:*****
TST31: MOV      #31,TSTNO
      MOV      #TST32,NEXT
      MOV      #3$,BACK
      CLRB    DSEXT      ;CLEAR SYNC EXTENTION
      CLR     RDATA      ;INIT RCV DATA
      CLR     TDATA      ;INIT XMIT DATA
      CLR     ADPTS      ;TRANSMITTER STATUS
      MOV     #5,ADPRS    ;CLOCK ON + STRIP SYNC
      MOV     #400,CHLEN  ;CHAR LENGTH INDEX
      BIT     #BIT9,SAVSR1 ;TEST 12 BIT CHAR MODE
      BEQ    1$          ;NO
      MOV     #10000,LIMIT ;SELECT END OF DATA
      BIS     #BIT10,ADPRS ;SELECT 12 BITS/CHARACTER
      MOV     #426,TSYNC  ;SYNC FOR 12 BIT CHAR
      INCB   DSEXT      ;PLACE MSB OF SYNC IN SYNC EXT
      BR     2$
1$:   MOV     #400,LIMIT  ;TEMPORARY CHARACTER LIMIT
      MOV     #26,TSYNC  ;INIT SYNC STORAGE
2$:   MOV     #TV18,ADPTIV ;TRANSMITTER VECTOR
      MOV     #RV18,ADPRIV ;RECEIVER VECTOR
      MOV     #200,PS    ;PRIORITY=4
      MOV     #3,SCNT    ;SYNC COUNT=3

```

```

1998 006266 113777 001144 172742 09900
1999 006274 052777 000100 172730 10000
2000 006302 052777 000340 172730 10100
2001 10200
2002 10300
2003 006310 000001 10400
2004 006312 000776 10500
2005 006314 104400 10600
2006 10700
2007 10800
2008 10900
2009 11000
2010
2011
2012
2013
2014
2015
2016 006316 012737 000032 001072
2017 006324 012737 006442 001062
2018 006332 005037 001156 11200
2019 006336 012737 000026 001144 11300
2020 006344 113777 001144 172664 11400
2021 006352 004737 013060 11500
2022 11600
2023 11700
2024 11800
2025 006356 012737 000200 001156 11900
2026 006364 052777 000400 172640 12000
2027 006372 042777 004000 172632 12100
2028 006400 004737 013060 12200
2029 12300
2030 12400
2031 12500
2032 006404 012737 000300 001156 12600
2033 006412 042777 000400 172612 12700
2034 006420 052777 001000 172604 12800
2035 006426 042777 004000 172576 12900
2036 13000
2037 006434 004737 013060 13100
2038 006440 104400 13200

```

```

MOVW TSYNC,JSYNC ;LOAD SYNC
BIS #BIT6,JDPRS ;RCV INT ENB
BIS #340,JDPTS ;STATUS INT ENB
;TRANS INT ENB
;TRANS DONE
;WAIT FOR INTERRUPTS

3$: WAIT
BR -2
SCOPE

;*****RANDOM DATA, RANDOM STALL*****

:*****
: TEST 32
:*****
:*****
:*****
TST32: MOV #32,TSTNO
MOV #TST33,NEXT
CLR CHLEN ;SET CHAR LENGTH TO 8 BITS
MOV #26,TSYNC ;SYNC = 26
MOVW TSYNC,JSYNC ;LOAD SYNC BUFFER
JSR PC,AND ;EXECUTE DATA + STALL MODES

;REPEAT PREVIOUS TEST AT 7 BITS/CHAR

MOV #200,CHLEN ;SET CHAR LENGTH TO 7 BITS
BIS #BIT8,JDPRS ;7 BITS/CHAR
BIC #BIT11,JDPRS ;CLEAR ACTIVE
JSR PC,AND ;EXECUTE DATA + STALL MODES

;REPEAT PREVIOUS TEST AT 6 BITS/CHAR

MOV #300,CHLEN ;SET CHAR LENGTH TO 6 BITS
BIC #BIT8,JDPRS
BIS #BIT9,JDPRS ;SET MODE TO 6 BITS/CHAR
BIC #BIT11,JDPRS ;CLEAR ACTIVE
JSR PC,AND ;EXECUTE DATA & STALL MODES
SCOPE

```

E04

2039					13400
2040					13500
2041					13600
2042					13700
2043					
2044					
2045					
2046					
2047					
2048					
2049	006442	012737	000033	001072	
2050	006450	012737	006646	001062	
2051	006456	012737	006644	001174	13900
2052	006464	005077	172542		14000
2053	006470	005077	172544		14100
2054	006474	012737	001401	015422	14200
2055	006502	012737	000400	001160	14300
2056	006510	005037	001154		14400
2057	006514	005037	001152		14500
2058	006520	105077	172520		14600
2059	006524	012737	000026	001144	14700
2060	006532	113777	001144	172476	14800
2061	006540	012737	000003	001162	14900
2062	006546	005077	172460		15000
2063	006552	005077	172462		15100
2064	006556	032737	000400	001134	15200
2065	006564	001406			15300
2066	006566	012737	010000	001160	15400
2067	006574	052777	002000	172430	15500
2068	006602	012737	000200	177776	15600
2069	006610	012777	015266	172434	15700
2070	006616	012777	015356	172422	15800
2071	006624	052777	000105	172400	15900
2072	006632	052777	000300	172400	16000
2073	006640	000001			16100
2074	006642	000776			16200
2075	006644	104400			16300
2076					16400
2077					16500
2078					16600
2079					
2080					
2081					
2082					
2083					
2084					
2085	006646	012737	000034	001072	
2086	006654	012737	007104	001062	
2087	006662	005037	001136		16800
2088	006666	105077	172352		16900
2089	006672	112777	000026	172336	17000
2090	006700	032737	000400	001134	17100
2091	006706	001403			17200
2092	006710	012777	002000	172314	17300
2093	006716	012777	007040	172326	17400
2094	006724	052777	000004	172300	17500

```

;*****PARITY TEST*****
;VERIFY "PARITY" BIT=1 FOR ODD PARITY AND=0 FOR EVEN
;*****
; TEST 33
;*****
;*****
TST33:  MOV    #33,TSTNO
        MOV    #TST34,NEXT
        MOV    #2$,BACK
        CLR    @DPRS
        CLR    @DPTS
        MOV    #1401,RPRT1      ;LOAD RPRT2WITH BEQ .+4
        MOV    #400,LIMIT      ;SET UP CHARACTER LIMIT
        CLR    RDATA          ;CLR RCV DATA
        CLR    TDATA          ;CLR XMIT DATA
        CLRB   @SEXT          ;CLEAR SYNC EXTENTION
        MOV    #26,TSYNC       ;SET UP SYNC
        MOVB   TSYNC,@SYNCR   ;INIT SYNC
        MOV    #3,SCNT         ;3 SYNC'S
        CLR    @DPRS          ;CLR RECEIVER STATUS
        CLR    @DPTS          ;CLR TRANSMITTER STATUS
        BIT    #BIT8,SAVSR1    ;8/12 BITS/CHAR
        BEQ    1$             ;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
        CLRB   @SEXT          ;CLEAR SYNC EXTENTION
        MOVB   #26,@SYNCR     ;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

```

F04

2095	006732	104403	013560		17600
2096	006736	052777	000200	172274	17700
2097	006744	105777	172270		17800
2098	006750	100375			17900
2099	006752	012777	000026	172262	18000
2100	006760	105777	172254		18100
2101	006764	100375			18200
2102	006766	012777	000026	172246	18300
2103	006774	105777	172240		18400
2104	007000	100375			18500
2105	007002	013777	001136	172232	18600
2106	007010	105777	172224		18700
2107	007014	100375			18800
2108	007016	013777	001136	172216	18900
2109	007024	052777	000040	172206	19000
2110	007032	104403	005670		19100
2111	007036	104000			19200
2112	007040	032777	040000	172172	19300
2113	007046	001001			19400
2114	007050	104000			19500
2115	007052	005077	172162		19600
2116	007056	042777	004000	172146	19700
2117	007064	022626			19800
2118	007066	042737	000040	177776	19900
2119	007074	105237	001136		20000
2120	007100	001314			20100
2121	007102	104400			20200
2122					20300
2123					20400
2124					20500
2125					20600
2126					
2127					
2128					
2129					
2130					
2131					
2132	007104	012737	000035	001072	
2133	007112	012737	007306	001062	
2134	007120	005077	172114		20800
2135	007124	012777	000103	172100	20900
2136	007132	012777	014504	172106	21000
2137	007140	005037	001144		21100
2138	007144	012737	000200	177776	21200
2139	007152	113777	001144	172056	21300
2140	007160	052777	000002	172052	21400
2141	007166	104403	005670		21500
2142	007172	105237	001144		21600
2143	007176	001365			21700
2144	007200	104400			21800
2145					21900

```

2$: DELAY ,6000. ;WAIT FOR BCV TO CLEAR
    BIS #BIT7,ADPTS ;DONE
    TSTB ADPTS ;TRANSMIT FIRST SYNC
    BPL -4
    MOV #26,ADPTB
    TSTB ADPTS
    BPL -4
    MOV #26,ADPTB ;TRANSMIT SECOND SYNC
    TSTB ADPTS
    BPL -4
    MOV TMPDAT,ADPTB ;TRANSMIT DATA CHAR #1
    TSTB ADPTS
    BPL -4
    MOV TMPDAT,ADPTB ;TRANSMIT DATA CHAR #2
    BIS #BITS,ADPTS ;SET STATUS INT ENB
    DELAY ,3000. ;WAIT FOR O'RUN INTERRUPT
    HLT ;REPORT ERROR, NO O'RUN INT
3$: BIT #BIT14,ADPTS ;TEST FOR O'RUN
    BNE .+4 ;BRANCH IF O'RUN CAUSED INT
    HLT ;REPORT ERROR
    CLR ADPTS ;CLEAR XMIT STATUS
    BIC #BIT11,ADPRS ;CLEAR ACTIVE
    POP.SP ;ADJUST STACK
    BIC #BITS,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 ADPTS ;CLEAR TRANSMITTER STATUS
      MOV #103,ADPRS ;HALF DUPLEX,INT EN,TURN CLK ON
      MOV #FRINT,ADPRIV ;SETUP TEST VECTOR
      CLR TSYNC ;CLR TEST SYNC
      MOV #200,PS ;PRIORITY=4
1$: MOVB TSYNC,ASync ;LOAD SYNC BUFFER
    BIS #BIT1,ADPTS ;IDLE SYNC
    DELAY ,3000. ;DELAY 20.1 MS
    INCB TSYNC ;HAVE ALL SYNC BEEN TESTED
    BNE 1$ ;NO
    SCOPE

;NOTE END OF THIS TEST.

```

2146					22100
2147					22200
2148	007202	000005			22300
2149	007204	012737	177777	001150	22400
2150	007212	012706	001050		22500
2151	007216	012737	000340	177776	22600
2152	007224	105737	177570		22700
2153	007230	100005			22800
2154	007232	004737	012462		22900
2155					23000
2156					23100
2157	007236	004737	012202		23200
2158	007242	000404			23300
2159	007244	004737	012462		23400
2160	007250	004737	012246		23500
2161	007254	032737	000001	001134	23600
2162	007262	001405			23700
2163	007264	012737	011770	001060	23800
2164	007272	000177	171562		23900
2165	007276	012737	007254	001060	24000
2166	007304	000240			24100
2167					24200
2168					24300
2169					24400
2170					24500
2171					24600
2172					
2173					
2174					
2175					
2176					
2177					
2178	007306	012737	000036	001072	
2179	007314	012737	007436	001062	
2180	007322	005737	001150		24800
2181	007326	001013			24900
2182	007330	032737	000001	001134	25000
2183	007336	001405			25100
2184	007340	012737	011770	001060	25200
2185	007346	000177	171506		25300
2186	007352	000137	015750		25400
2187	007356	112777	000026	171652	25500
2188	007364	005077	171650		25600
2189	007370	105077	171650		25700
2190	007374	005077	171632		25800
2191	007400	012777	014500	171644	25900
2192	007406	012777	014504	171632	26000
2193	007414	012777	000240	171626	26100
2194	007422	012777	000240	171624	26200
2195	007430	104401			26300
2196	007432	000030			26400
2197	007434	104400			26500
2198					26600
2199					26700
2200					26800
2201					26900

```

:*****PART2 DB255 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     BGNOA           ;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
BGNOA: JSR     PC,CLRVEC
PART2: JSR     PC,LINE.X
      BIT     #BIT0,SAVSR1
      BEQ     IS
      MOV     #PART3,RETURN
      JMP     @RETURN
IS:    MOV     #PART2,RETURN
      NOP

:*****TEST 1: CABLE TESTS
:*****
:TEST 36
:*****
:*****
TST36: MOV     #36,TSTNO
      MOV     #TST37,NEXT
      TST    CABLE
      BNE    IS
      BIT    #BIT0,SAVSR1
      BEQ    .+14
      MOV    #PART3,RETURN
      JMP    @RETURN
IS:    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

```

H04

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 48
DZDPAB.SRC

Address	Op1	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10
2202					27000					
2203										
2204										
2205										
2206										
2207										
2208										
2209	007436	012737	000037	001072						
2210	007444	012737	007726	001062						
2211	007452	004537	013472		27200					
2212	007456	012737	000376	001160	27300					
2213	007464	000403			27400					
2214					27500					
2215	007466	012737	007776	001160	27600					
2216	007474	012737	007656	001174	27700					
2217	007502	012777	014510	171542	27800					
2218	007510	012737	000003	001162	27900					
2219					28000					
2220	007516	104401			28100					
2221	007520	000030			28200					
2222	007522	113777	001144	171506	28300					
2223	007530	113777	001145	171506	28400					
2224	007536	052777	000100	171466	28500					
2225	007544	052777	000300	171466	28600					
2226	007552	104402			28700					
2227	007554	000003			28800					
2228	007556	032777	001000	171454	28900					
2229	007564	001001			29000					
2230	007566	104000			29100					
2231	007570	104401			29200					
2232	007572	000010			29300					
2233	007574	032777	004000	171430	29400					
2234	007602	001401			29500					
2235	007604	104000			29600					
2236	007606	104401			29700					
2237	007610	000007			29800					
2238	007612	032777	004000	171420	29900					
2239	007620	001401			30000					
2240	007622	104000			30100					
2241	007624	104402			30200					
2242	007626	000001			30300					
2243	007630	032777	004000	171374	30400					
2244	007636	001001			30500					
2245	007640	104000			30600					
2246	007642	012777	015570	171376	30700					
2247	007650	104401			30800					
2248	007652	000010			30900					
2249	007654	104000			31000					
2250	007656	042777	004000	171346	31100					
2251	007664	042777	000100	171346	31200					
2252	007672	032737	040000	177570	31300					
2253	007700	001300			31400					
2254	007702	112777	000026	171326	31500					
2255	007710	005237	001144		31600					
2256	007714	023737	001160	001144	31700					
2257	007722	001267			31800					

```
*****
: TEST 37
*****
*****
†ST37:  MOV #37,TSTNO
        MOV #TST40,NEXT
        JSR RS,REE
        MOV #376,LIMIT
        BR .+10
:REE WILL ENER HERE IF 12 BITS/CHAR
        MOV #7776,LIMIT
        MOV #2$,BACK
1$:     MOV #TV18,ADPTIV
        MOV #3,SCNT
        CLOCK
        30
        MOVB TSYNC,ASync
        MOVB TSYNC+1,ASEXT
        BIS #BIT6,ADPRS
        BIS #300,ADPTS
        RXCLK
        3
        BIT #BIT9,ADPTS
        BNE .+4
        HLT
        CLOCK
        10
        BIT #BIT11,ADPRS
        BEQ .+4
        HLT
        CLOCK
        7
        BIT #BIT11,ADPTS
        BEQ .+4
        HLT
        RXCLK
        1
        BIT #BIT11,ADPRS
        BNE .+4
        HLT
        MOV #SRV5,ADPRIV
        CLOCK
        10
        HLT
2$:     BIC #BIT11,ADPRS
        BIC #BIT6,ADPTS
        BIT #BIT14,SWR
        BNE 1$
        MOVB #26,ASync
        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 RCY 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
```

2258	007724	104400			31900
2259					32000
2260					32100
2261					32200
2262					32300
2263					32400
2264					
2265					
2266					
2267					
2268					
2269					
2270	007726	012737	000040	001072	
2271	007734	012737	010224	001062	
2272	007742	004537	013472		32600
2273					32700
2274	007746	012737	000176	001160	32800
2275	007754	000403			32900
2276					33000
2277	007756	012737	003776	001160	33100
2278	007764	052777	000400	171240	33200
2279	007772	012737	010154	001174	33300
2280	010000	012777	014504	171240	33400
2281	010006	012777	014510	171236	33500
2282	010014	012737	000003	001162	33600
2283	010022	104401			33700
2284	010024	000024			33800
2285	010026	113777	001144	171202	33900
2286	010034	052777	000300	171176	34000
2287	010042	052777	000100	171162	34100
2288	010050	104402			34200
2289	010052	000003			34300
2290	010054	032777	001000	171156	34400
2291	010062	001001			34500
2292	010064	104000			34600
2293	010066	104401			34700
2294	010070	000007			34800
2295	010072	032777	004000	171132	34900
2296	010100	001401			35000
2297	010102	104000			35100
2298	010104	104401			35200
2299	010106	000006			35300
2300	010110	032777	004000	171122	35400
2301	010116	001401			35500
2302	010120	104000			35600
2303	010122	104402			35700
2304	010124	000001			35800
2305	010126	032777	004000	171076	35900
2306	010134	001001			36000
2307	010136	104000			36100
2308	010140	012777	015570	171100	36200
2309	010146	104401			36300
2310	010150	000007			36400
2311	010152	104000			36500
2312	010154	042777	004000	171050	36600
2313	010162	042777	000100	171050	36700

SCOPE

```

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

:*****
:
: TEST 40
:
:*****
:*****
TST40:  MOV    #40,TSTNO
        MOV    #TST41,NEXT
        JSR    R5,REE                ;REINIT DP11
;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    #25,BACK             ;SET UP RCV SERVICE RETURN
        MOV    #FRINT,@DPRIV        ;FALSE INT TEST VECTOR
1$:     MOV    #TV18,@DPTIV         ;SET XMIT INT VECTOR TO SYNC
        MOV    #3,SCNT              ;XMIT 3 SYNCs
        CLOCK
        24                          ;RUN CLOCK
        MOVB   TSYNC,@SYNC          ;LOAD SYNC BUFFER
        BIS    #300,@DPTS           ;XMIT INT ENB
        BIS    #BIT6,@DPRS         ;RCINT ENB
        RXCLK
        3
        BIT    #BIT9,@DPTS         ;SEND REQUEST UP?
        BNE    .+4
        HLT
        CLOCK                       ;REPORT ERROR
        7                          ;RUN CLOCK
        BIT    #BIT11,@DPRS        ;RECEIVER ACTIVE
        BEQ    .+4                  ;NO
        HLT                          ;REPORT ERROR
        CLOCK                       ;RUN CLOCK
        6
        BIT    #BIT11,@DPTS        ;RCV ACTIVE?
        BEQ    .+4                  ;NO
        HLT                          ;REPORT ERROR
        RXCLK
        1
        BIT    #BIT11,@DPRS        ;RCV ACTIVE?
        BNE    .+4                  ;YES
        HLT                          ;REPORT ERROR
        MOV    #SRVS,@DPRIV        ;TEST PASS VECTOR
        CLOCK                       ;RUN CLOCK
        7
        HLT                          ;REPORT ERROR
2$:     BIC    #BIT11,@DPRS        ;CLEAR RCV ACTIVE
        BIC    #BIT6,@DPTS         ;CLEAR XMIT INT ENB
    
```

2314	010170	112777	000026	171040	36800
2315	010176	032737	040000	177570	36900
2316	010204	001300			37000
2317	010206	005237	001144		37100
2318	010212	023737	001160	001144	37200
2319	010220	001272			37300
2320	010222	104400			37400
2321					37500
2322					37600
2323					37700
2324					37800
2325					37900
2326					
2327					
2328					
2329					
2330					
2331					
2332	010224	012737	000041	001072	
2333	010232	012737	010530	001062	
2334	010240	004537	013472		38100
2335	010244	012737	000075	001160	38200
2336	010252	000403			38300
2337					38400
2338	010254	012737	001775	001160	38500
2339	010262	052777	001000	170742	38600
2340	010270	012737	010466	001174	38700
2341	010276	012777	014510	170746	38800
2342	010304	012737	000003	001162	38900
2343	010312	042777	000100	170720	39000
2344	010320	042777	004000	170704	39100
2345	010326	112777	000026	170702	39200
2346	010334	104401			39300
2347	010336	000021			39400
2348	010340	113777	001144	170670	39500
2349	010346	052777	000100	170656	39600
2350	010354	052777	000300	170656	39700
2351	010362	104402			39800
2352	010364	000003			39900
2353	010366	032777	001000	170644	40000
2354	010374	001001			40100
2355	010376	104000			40200
2356	010400	104401			40300
2357	010402	000006			40400
2358	010404	032777	004000	170620	40500
2359	010412	001401			40600
2360	010414	104000			40700
2361	010416	104401			40800
2362	010420	000005			40900
2363	010422	032777	004000	170610	41000
2364	010430	001401			41100
2365	010432	104000			41200
2366	010434	104402			41300
2367	010436	000001			41400
2368	010440	032777	004000	170564	41500
2369	010446	001001			41600

```

MOV#B #26, @SYNC ; DUMMY 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

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

; *****
; TEST 41
; *****
; *****
TST41: MOV #41, TSTNO
MOV #TST42, NEXT
JSR R5, REE ; REINIT DP11
MOV #75, LIMIT ; 6 BIT LIMIT
BR .+10 ; BRANCH AROUND 10 BIT LIMIT
; RETURN HERE IF 10 BITS/CHAR
MOV #1775, LIMIT ; SET UP 10 BITS/CHAR
BIS #BIT9, @DPRS ; 6/10 BITS/CHAR
MOV #25, BACK ; SET UP RCV SERVICE RETURN
1$: MOV #TV18, @DPTIV ; SET XMIT INT VECTOR TO SYNC
MOV #3, SCNT ; XMIT 3 SYNC
BIC #BIT6, @DPTS ; CLEAR XMIT INT ENB
BIC #BIT11, @DPRS ; CLEAR ACTIVE
MOV#B #26, @SYNC ; CHALGE SYNC
CLOCK ; RUN CLOCK
21
MOV#B TSYNC, @SYNC ; LOAD SYNC BUFFER
BIS #BIT6, @DPRS ; RCVINT ENB
BIS #300, @DPTS ; XMIT INT ENB
RXCLK
3
BIT #BIT9, @DPTS ; SEND REQUEST UP?
BNE .+4
HLT ; REPORT ERROR
CLOCK ; RUN CLOCK
6
BIT #BIT11, @DPRS ; RECEIVER ACTIVE
BEQ .+4 ; NO
HLT ; REPORT ERROR
CLOCK ; RUN CLOCK
5
BIT #BIT11, @DPTS ; RCV ACTIVE?
BEQ .+4 ; NO
HLT ; REPORT ERROR
RXCLK
1
BIT #BIT11, @DPRS ; RCV ACTIVE?
BNE .+4 ; YES

```

K04

```

2370 010450 104000 41700
2371 010452 012777 015570 170566 41800
2372 010460 104401 41900
2373 010462 000006 42000
2374 010464 104000 42100
2375 42200
2376 010466 042777 004000 170536 42300
2377 010474 112777 000026 170534 42400
2378 010502 032737 040000 177570 42500
2379 010510 001272 42600
2380 010512 005237 001144 42700
2381 010516 023737 001160 001144 42800
2382 010524 001264 42900
2383 010526 104400 43000
2384 43100
2385 43200
2386 43300
2387
2388
2389
2390
2391
2392
2393 010530 012737 000042 001072 43500
2394 010536 012737 010552 001062 43600
2395 010544 004737 014000 43700
2396 010550 104400 43800
2397 43900
2398 44000
2399 44100
2400 44200
2401
2402
2403
2404
2405
2406
2407
2408
2409 010552 012737 000043 001072 44400
2410 010560 012737 010730 001062 44500
2411 010566 005077 170446 44600
2412 010572 005077 170434 44700
2413 010576 105077 170442 44800
2414 010602 012737 177400 001156 44900
2415 010610 012737 000026 001144 45000
2416 010616 113777 001144 170412 45100
2417 010624 012701 015371 45200
2418 010630 012702 072414 45300
2419 010634 012703 004036 45400
2420 010640 004737 013562 45500
2421 45600
2422 45700
2423 45800
2424
2425 010644 012737 177600 001156 45800

```

```

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 ;DO 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 #7214, R2
MOV #4036, R3
JSR PC, A2ND

;REPEAT PREVIOUS TEST AT 7 BITS/CHAR

MOV #177600, CHLEN ;SET CHAR LENGHT TO 7 BITS

```

2426	010652	052777	000400	170352	45900	BIS	#BIT8,ADPRS	:7 BITS/CHAR
2427	010660	042777	004000	170344	46000	BIC	#BIT11,ADPRS	:CLEAR ACTIVE
2428	010666	004737	013562		46100	JSR	PC,A2ND	
2429					46200			
2430					46300			
2431					46400			
2432	010672	012737	177700	001156	46500	MOV	#177700,CHLEN	:SET CHAR LENGTH TO 6 BITS
2433	010700	042777	000400	170324	46600	BIC	#BIT8,ADPRS	
2434	010706	052777	001000	170316	46700	BIS	#BIT9,ADPRS	:SET MODE TO 6 BITS/CHAR
2435	010714	042777	004000	170310	46800	BIC	#BIT11,ADPRS	:CLEAR ACTIVE
2436	010722	004737	013562		46900	JSR	PC,A2ND	
2437	010726	104400			47000	SCOPE		
2438								

;REPEAT PREVIOUS TEST AT 6 BITS/CHAR

```

2439      00100
2440      00200
2441      00300
2442      00400
2443      00500
2444      00600
2445      00700
2446      00800
2447      00900
2448
2449
2450
2451
2452
2453
2454 010730 012737 000044 001072
2455 010736 012737 011124 001062
2456 010744 012737 000240 177776 01100
2457 010752 005077 170262 01200
2458 010756 005077 170250 01300
2459 010762 104401 01400
2460 010764 000010 01500
2461 010766 012777 014500 170256 01600
2462 010774 012777 014504 170244 01700
2463 011002 052777 000040 170230 01800
2464 011010 013777 001152 170224 01900
2465 011016 104401 02000
2466 011020 000003 02100
2467 011022 032777 001000 170210 02200
2468 011030 001001 02300
2469 011032 104000 02400
2470 011034 032777 002000 170176 02500
2471 011042 001001 02600
2472 011044 104000 02700
2473 011046 032777 010000 170164 02800
2474 011054 001001 02900
2475 011056 104000 03000
2476 011060 104401 03100
2477 011062 000012 03200
2478 011064 032777 001000 170146 03300
2479 011072 001401 03400
2480 011074 104000 03500
2481 011076 032777 002000 170134 03600
2482 011104 001401 03700
2483 011106 104000 03800
2484 011110 032777 010000 170122 03900
2485 011116 001401 04000
2486 011120 104000 04100
2487 011122 104400 04200

```

;DB255 TEST CONNECTOR DISCRETE EVENTS TEST

;VERIFY "SEND REQUEST" RAISES "CLEAR-TO-SEND" AND "MODEM READY"
;WHEN TRANSMITTER BUFFER IS LOADED

;*****
;*

; TEST 44

;*****
;*

;*****

```

†ST44:  MOV      #44,TSTNO
        MOV      #TST45,NEXT
        MOV      #240,PS          ;PRIORITY=5
        CLR      @DPTS            ;CLR XMIT STATUS
        CLR      @DPRS            ;CLR RCV STATUS
        CLOCK    10              ;RUN CLOCK
        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    3              ;RUN CLOCK
        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    12              ;RUN CLOCK
        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

```

```

2488
2489
2490
2491
2492
2493
2494
2495
2496 011124 012737 000045 001072 04400
2497 011132 012737 011472 001062 04500
2498 011140 012737 000240 177776 04700
2499 011146 005077 170066 04800
2500 011152 005077 170054 04900
2501 011156 012777 014500 170066 05000
2502 011164 012777 014504 170054 05100
2503 011172 052777 000400 170032 05200
2504 011200 052777 000040 170032 05300
2505 011206 013777 001152 170026 05400
2506 011214 104401 05500
2507 011216 000003 05600
2508 011220 032777 001000 170012 05700
2509 011226 001001 05800
2510 011230 104000 05900
2511 011232 032777 002000 170000 06000
2512 011240 001001 06100
2513 011242 104000 06200
2514 011244 032777 010000 167766 06300
2515 011252 001001 06400
2516 011254 104000 06500
2517 06600
2518 011256 104401 06700
2519 011260 000011 06800
2520 011262 032777 001000 167750 06900
2521 011270 001401 07000
2522 011272 104000 07100
2523 011274 032777 002000 167736 07200
2524 011302 001401 07300
2525 011304 104000 07400
2526 011306 032777 010000 167724 07500
2527 011314 001401 07600
2528 011316 104000 07700
2529 011320 104400 07800
2530 07900
2531 08000
2532 08100
2533 011322 012737 000240 177776 08200
2534 011330 005077 167704 08300
2535 011334 005077 167672 08400
2536 011340 012777 014500 167704 08500
2537 011346 012777 014504 167672 08600
2538 011354 052777 001000 167650 08700
2539 011362 052777 000040 167650 08800
2540 011370 013777 001152 167644 08900
2541 011376 104401 09000
2542 011400 000003 09100
2543 011402 032777 001000 167630 09200

```

```

;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

```

```

011410 001001 09300
011411 104000 09400
011414 032777 010000 167616 09500
011415 001001 09600
011416 104000 09700
011417 104401 09800
011420 000010 09900
011423 032777 001000 167600 10000
011424 001401 10100
011425 104000 10200
011426 032777 002000 167566 10300
011427 001401 10400
011428 104000 10500
011431 032777 010000 167554 10600
011432 001401 10700
011433 104000 10800
011434 104400 10900
011470 104400 11000
011471 11100
011472 11200
011473 11300
011474 11400
011475 11500
011476 11600
011477 11700
011478 11700
011479 11700
011480 11700
011481 11700
011482 11700
011483 11700
011484 11700
011485 11700
011486 11700
011487 11700
011488 11700
011489 11700
011490 11700
011491 11700
011492 11700
011493 11700
011494 11700
011495 11700
011496 11700
011497 11700
011498 11700
011499 11700
011500 012737 000046 001072
011501 012737 011770 001062
011502 012737 000240 177776 11900
011503 005077 167520 12000
011504 012777 014500 167512 12100
011505 012777 000001 167504 12200
011506 104403 001500 12300
011507 032777 000001 167472 12400
011508 001001 12500
011509 104000 12600
011510 032777 004000 167460 12700
011511 001001 12800
011512 104000 12900
011513 032777 020000 167446 13000
011514 001001 13100
011515 104000 13200
011516 104000 13300
011517 104000 13400
011518 104000 13500
011519 042777 000001 167434 13600
011520 104403 001500 13700
011521 032777 000001 167422 13800
011522 001401 13900
011523 104000 14000
011524 032777 004000 167410 14100

```

```

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

:DB255 TEST CONNECT CONNECTOR TEST

:TERMINAL READY

:TERM RDY-RAISE "CARRIER" AND "RING FLAG", NO INT ENB
:*****
: *
: TEST 46
: *
:*****
:*****
:ST46: MOV #46,TSTNO
MOV #TST47,NEXT
MOV #240,PS :PRIORITY = 5
CLR JDPTS :CLR XMIT STATUS
MOV #FTINT,JDPTS :ERROR TRAP VECTOR
MOV #BIT0,JDPTS :TERMINAL READY
DELAY .1500 :15MS PROPOGATION DELAY
BIT #BIT0,JDPTS :TERMINAL READY
BNE .+4 :YES
HLT :REPORT ERROR
BIT #BIT11,JDPTS :CARRIER
BNE .+4 :YES
HLT :REPORT ERROR
BIT #BIT13,JDPTS :RING FLAG UP
BNE .+4 :YES
HLT :REPORT ERROR

:CLEAR TERMINAL READY

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

```

2600	011630	001401			14200
2601	011632	104000			14300
2602	011634	032777	100000	167376	14400
2603	011642	001001			14500
2604	011644	104000			14600
2605	011646	042777	100000	167364	14700
2606					14800
2607					14900
2608	011654	012777	000240	167372	15000
2609	011662	005077	167352		15100
2610	011666	005077	167340		15200
2611	011672	012777	014354	167352	15300
2612	011700	012737	000200	177776	15400
2613	011706	012777	000041	167324	15500
2614	011714	104403	001500		15600
2615	011720	104000			15700
2616					15800
2617					15900
2618	011722	012777	014426	167322	16000
2619	011730	012737	011766	001174	16100
2620	011736	012737	000200	177776	16200
2621	011744	042777	000001	167266	16300
2622	011752	104403	001500		16400
2623	011756	104000			16500
2624	011760	042777	100000	167252	16600
2625	011766	104400			16700
2626					16800
2627					16900

```

BEQ      .+4          ; YES
HLT      ; REPORT ERROR
BIT      #BIT15, @DPTS ; "CARRIER DOWN" FLAG UP
BNE      .+4          ; YES
HLT      ; REPORT ERROR
BIC      #BIT15, @DPTS ; CLEAR DOWN FLAG

; VERIFY THAT "RING" AND "CARRIER DOWN" INTERRUPT
RCD:     MOV      #240, @DPTP ; INTERRUPT PRIORITY 5.
CLR      @DPTS ; CLEAR XIMT STATUS
CLR      @DPRS ; CLEAR RCV STATUS
MOV      #TV24, @DPTIV ; TEST PASS VECTOR
MOV      #200, PS ; PRIORITY = 4
MOV      #BITS+BIT0, @DPTS ; STATUS INTERRUPT ENABLE
DELAY    ,1500 ; 15.75 MS DELAY
HLT      ; REPORT ERROR

; VERIFY "CARRIER DOWN" RAISES INTERRUPT
RCD1:    MOV      #TV25, @DPTIV ; NEXT TEST VECTOR
MOV      #1$, BACK
MOV      #200, PS
BIC      #BIT0, @DPTS ; CLEAR TERM. RDY. SHOULD SET "CARRIER DOWN".
DELAY    ,1500 ; 15.75MS DELAY
HLT      ; REPORT ERROR
BIC      #BIT15, @DPTS ; CLEAR "CARRIER DOWN"
IS:      SCOPE
  
```

```

2628      17100
2629      17200
2630      17300
2631      17400
2632      17500
2633      17600
2634      17700
2635      17800
2636      17900
2637      18000
2638      18100
2639      18200
2640
2641
2642
2643
2644
2645
2646
2647      011770 012737 000047 001072
2648      011776 012737 015750 001062
2649      012004 032737 000001 001134 18400
2650      012012 001002          18500
2651      012014 000137 015750 18600
2652      012020 005077 167214 18700
2653      012024 005077 167202 18800
2654      012030 005702          18900
2655      012032 012700 001300 19000
2656      012036 010277 167200 19100
2657      012042 017701 167172 19200
2658      012046 042701 164777 19300
2659      012052 022701 013000 19400
2660      012056 001403          19500
2661      012060 005202          19600
2662      012062 001367          19700
2663      012064 104001          19800
2664      012066 005002          19900
2665      012070 005202          20000
2666      012072 001376          20100
2667      012074 032777 013000 167136 20200
2668      012102 001401          20300
2669      012104 104000          20400
2670          20500
2671          20600
2672          20700
2673          20800
2674          20900
2675          21000
2676      012106 012777 000001 167124 21100
2677      012114 104403 001500          21200
2678      012120 017701 167114          21300
2679      012124 042701 153776          21400
2680      012130 022701 024001          21500
2681      012134 001401          21600
2682      012136 104001          21700
2683      012140 042777 000001 167072 21800

```

```

: 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

```

2694	012146	104403	001500	22000
2695	012152	022777	120000 167060	22100
2696	012160	001401		22200
2697	012162	104000		22300
2698				22400
2699				22500
2700				22600
2701				22700
2702				22800
2703	012164	005037	001102	22900
2704	012170	005037	001104	23000
2705	012174	004737	014000	23100
2706	012200	104400		23200
2707				

```

DELAY      ,1500
CMP        #120000,0DPTS
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

```

F05

2698									
2699				00100					
2700				00200					;LINE.N SUBROUTINE TO FETCH THE LINE
2701				00300					;NUMBER AND FIRST DP11 VECTOR ADDRESS FROM
2702				00400					;THE CONSOL SWITCHES
2703				00500					;SW0-SW8=VECTOR ADDRESS OF FIRST DP11
2704				00600					;SW9-SW15=LINE NUMBER OF DP11 SELECTED FOR TEST
2705	012202	005037	001076	00700					
2706	012206	013737	177570	00800	001134				LINE.N: CLR ERRCNT ;CLEAR ERROR COUNT
2707	012214	013700	177570	00900					MOV SWR,SAVSR1 ;SAVE CONSOL SWITCH SETTINGS
2708	012220	000000		01000					MOV SWR,RO
2709				01100					HALT ;SET SWR TO LINE NUMBER
2710				01200					;LOW BYTE = FIRST DP VECTOR
2711	012222	013737	177570	01300	001164				;HIGH BYTE = LINE NUMBER(8)
2712	012230	013700	177570	01400					MOV SWR,SAVSR2 ;SAVE CONSOL SWITCHES
2713	012234	000000		01500					MOV SWR,RO
2714	012236	005001		01600					HALT
2715	012240	113701	001165	01700					CLR R1
2716	012244	000405		01800					MOV SBVSR2+1,R1 ;SAVE LINE NUMBER
2717	012246	013701	001146	01900					BR XTAG
2718	012252	005037	001134	02000					LINE.X: MOV XLINEX,R1
2719	012256	006301		02100					CLR SAVSR1
2720	012260	006201		02200					ASL R1
2721	012262	010137	001146	02300					XTAG: ASR R1 ;CLEAR LSB
2722	012266	006301		02400					MOV R1,XLINEX
2723	012270	006301		02500					ASL R1 ;SCALE LINE NUMBER TO ADDRESS
2724	012272	006301		02600					ASL R1 ;MODULO 10(8)
2725	012274	005777	166760	02700					ASL R1
2726	012300	000240		02800					TST @BASCSR ;DOES LINE 0 REALLY EXIST??
2727	012302	105737	001134	02900					NOP
2728	012306	100403		03000					TSTB SAVSR1
2729	012310	012737	012422	03100	000004				BMI .+10
2730	012316	013702	001260	03200					MOV #3\$,@#4
2731	012322	160102		03300					MOV BASCSR,R2 ;SET R2 = LINE 0 ADDRESS
2732	012324	012703	001232	03400					SUB R1,R2 ;MANUFACTURE DEVICE ADDRESS
2733	012330	010223		03500					MOV #DPRS,R3 ;R3 = ADDRESS OF RCV STATUS ADRS
2734	012332	005722		03600					MOV R2,(R3)+ ;LOAD RCV STATUS ADRS
2735	012334	010223		03700					TST (R2)+ ;INC TO RCV BUFFER ADRS
2736	012336	005202		03800					MOV R2,(R3)+ ;LOAD RCV BUFFER ADRS
2737	012340	010223		03900					INC R2 ;INC TO SYNC BUFFER ADRS
2738	012342	005202		04000					MOV R2,(R3)+ ;LOAD SYNC ADRS
2739	012344	010223		04100					INC R2 ;INC TO XMIT STATUS ADRS
2740	012346	005722		04200					MOV R2,(R3)+ ;LOAD TRANSMITTER STATUS ADRS
2741	012350	010223		04300					TST (R2)+ ;INC TO XMIT BUFFER
2742	012352	005202		04400					MOV R2,(R3)+ ;LOAD XMIT BUFFER ADRS
2743	012354	010223		04500					INC R2 ;INC TO SYNC EXTENTION
2744	012356	013702	001164	04600					MOV R2,(R3)+ ;LOAD SYNC EXTENSION ADRS
2745	012362	042702	177000	04700					MOV SAVSR2,R2 ;SET UP VECTOR ADDRESS
2746	012366	105737	001134	04800					BIC #177000,R2 ;CLEAR LINE NUMBER FROM VEC ADRS
2747	012372	100402		04900					TSTB SAVSR1
2748	012374	013702	001256	05000					BMI 2\$
2749	012400	060102		05100					MOV BASVEC,R2
2750	012402	010223		05200					2\$: ADD R1,R2 ;SET VECTOR ADDRESS TO LINE NUMBER
2751	012404	005722		05300					MOV R2,(R3)+ ;LOAD RCV VECTOR ADRS
2752	012406	010223		05400					TST (R2)+ ;INC TO NEXT VECTOR
2753	012410	005722		05500					MOV R2,(R3)+ ;LOAD RCV PRIORITY ADRS
				05600					TST (R2)+ ;INC TO NEXT VECTOR

G05

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 60
DZDPAB.SRC

2754	012412	010223		05700		MOV R2,(R3)+	;LOAD XMIT VECTOR ADRS
2755	012414	005722		05800		TST (R2)+	;INC TO NEXT VECTOR
2756	012416	010213		05900		MOV R2,(R3)	;LOAD XMIT PRIORITY ADRS
2757	012420	000414		06000		BR 5\$	
2758	012422	005037	001146	06100	3\$:	CLR XLINEX	
2759	012426	013701	000042	06200		MOV @#42,R1	
2760	012432	001405		06300		BEQ 4\$	
2761	012434	000005		06400		RESET	
2762		012436		06550		LOGICAL=	
2763	012436	004711		06500		JSR PC,(R1)	
2764	012440	000240		06600		NOP	
2765	012442	000240		06700		NOP	
2766	012444	000240		06800		NOP	
2767	012446	022626		06900	4\$:	POP.SP	
2768	012450	000676		07000		BR LINE.X	
2769	012452	012737	000006 000004	07100	5\$:	MOV #6,@#4	
2770	012460	000207		07200		RTS PC	
2771				07300		;*****CLRVEC*****	
2772				07400			
2773				07500		;CLRVEC,ROUTINE TO FILL COMMUNICATION VECTOR AREA WITH .+2,HALT	
2774				07600			
2775	012462	012702	000300	07700	CLRVEC:	MOV #300,R2	;R2 COMM VECTOR AREA ADRS
2776	012466	012701	000302	07800		MOV #302,R1	;INIT R1 WITH ADRS OF HALT
2777	012472	010122		07900	1\$:	MOV R1,(R2)+	;MOV .+2 TO PC
2778	012474	005022		08000		CLR (R2)+	;MOV HALT TO PC
2779	012476	022121		08100		CMP (R1)+,(R1)+	;INC TO NEXT VECTOR AREA
2780	012500	022701	000776	08200		CMP #776,R1	;END OF VECTOR AREA
2781	012504	001372		08300		BNE 1\$;NO
2782	012506	000207		08400		RTS PC	;RETURN
2783				08500			
2784				08600			
2785				08700			
2786				08800		;BITSR,ROUTINE TO TEST READ WRITE BITS OF STATUS	
2787				08900		;THIS ROUTINE VERIFIES THAT EACH READ/WRITE BIT	
2788				09000		;CAN BE SET AND CLEARED	
2789				09100		;EXAMINE LOCATIONS	
2790				09200		;BITS: FOR BIT UNDER TEST	
2791				09300		;REG: FOR REGISTER UNDER TEST	
2792				09400			
2793	012510	010537	001110	09500	BITST:	MOV R5,TEMP4	
2794	012514	012537	013054	09600		MOV (R5)+,BITS	;SAVE BIT NUMBER
2795	012520	053777	013054 000330	09700		BIS BITS,@REG	;SET BIT
2796	012526	053700	013054	09800		BIS BITS,R0	
2797	012532	017701	000320	09900		MOV @REG,R1	
2798	012536	033777	013054 000312	10000		BIT BITS,@REG	;IS BIT SET?
2799	012544	001001		10100		BNE .+4	;YES
2800	012546	104002		10200		HLT 2	;REPORT ERROR
2801	012550	043777	013054 000300	10300		BIC BITS,@REG	;CLEAR BIT
2802	012556	043700	013054	10400		BIC BITS,R0	
2803	012562	017701	000270	10500		MOV @REG,R1	
2804	012566	033777	013054 000262	10600		BIT BITS,@REG	;IS BIT CLEARED
2805	012574	001401		10700		BEQ .+4	;YES
2806	012576	104002		10800		HLT 2	;REPORT ERROR
2807	012600	053777	013054 000250	10900		BIS BITS,@REG	;SET BIT
2808	012606	053700	013054	11000		BIS BITS,R0	
2809	012612	017701	000240	11100		MOV @REG,R1	

H05

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 61
 DZDPAB.SRC

2810	012616	033777	013054	000232	11200
2811	012624	001001			11300
2812	012626	104002			11400
2813	012630	005077	000222		11500
2814	012634	005000			11600
2815	012636	017701	000214		11700
2816	012642	033777	013054	000206	11800
2817	012650	001401			11900
2818	012652	104002			12000
2819	012654	052777	000004	166350	12100
2820	012662	052700	000004		12200
2821	012666	000205			12300
2822					12400
2823					12500
2824					12600
2825					12700
2826					12800
2827					12900
2828					13000
2829	012670	011637	001110		13100
2830	012674	017637	000000	013054	13200
2831	012702	062716	000002		13300
2832	012706	053777	013054	000142	13400
2833	012714	053700	013054		13500
2834	012720	017701	000132		13600
2835	012724	023777	013054	000124	13700
2836	012732	001401			13800
2837	012734	104002			13900
2838	012736	043777	013054	000112	14000
2839	012744	053700	013054		14100
2840	012750	000002			14200
2841					14300
2842					14400
2843					14500
2844	012752	011637	001110		14600
2845	012756	017637	000000	013054	14700
2846	012764	062716	000002		14800
2847	012770	043777	013054	000060	14900
2848	012776	043737	013054	001136	15000
2849	013004	012700	013000		15100
2850	013010	005300			15200
2851	013012	001376			15300
2852	013014	013700	001136		15400
2853	013020	017701	000032		15500
2854	013024	023777	001136	000024	15600
2855	013032	001401			15700
2856	013034	104002			15800
2857	013036	053777	013054	000012	15900
2858	013044	053737	013054	001136	16000
2859	013052	000002			16100
2860					16200
2861	013054	000000			16300
2862	013056	000000			16400
2863					16500
2864					16600
2865					16700

```

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

; 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

```

```

BITS:    0
REG:     0
; COMMON DATA AND IDLE SUBROUTINE

```

2866	013060	012701	015371	16800	AND:	MOV	#15371,R1	;PRIME RANDOM # GEN
2867	013064	012702	072414	16900		MOV	#72414,R2	;;
2868	013070	012703	004036	17000		MOV	#4036,R3	;;
2869	013074	042777	004300	166130		BIC	#4300,ADPRS	;RCV INT ENB, RCV ACTIVE
2870	013102	042777	160342	166130		BIC	#160342,ADPTS	;INT ENBS IDLE SYNC, ERRORS
2871	013110	012737	015556	001172		MOV	#BOTTOM,RP	;SET UP RCV POINTER
2872	013116	013737	001172	001170		MOV	RP,TP	;SET UP XMIT POINTER
2873	013124	012777	015172	166114		MOV	#RRRR,ADPRIV	;RCV INT VECTOR
2874	013132	012777	015044	166112		MOV	#RRRT,ADPTIV	;XMIT INT VECTOR
2875	013140	012737	000002	001162		MOV	#2,SCNT	;SYNC COUNT = 2
2876	013146	110237	001152	17800		MOVB	R2,TDATA	;RANDOM DATA
2877	013152	052777	000100	166052		BIS	#100,ADPRS	;RCV INT ENB
2878	013160	013777	001144	166054	15:	MOV	TSYNC,ADPTB	;LOAD BUFFER
2879	013166	052777	000340	166044		BIS	#340,ADPTS	;XMIT DONE, INT ENB, STATUS ENB
2880	013174	010137	001166	18200		MOV	R1,TIME	; "ON" STALL
2881	013200	005337	001166	18300	25:	DEC	TIME	;0.6 SEC MAX
2882	013204	001375		18400		BNE	25	
2883	013206	042777	000140	166024		BIC	#140,ADPTS	;TURN OFF INT ENB
2884	013214	052777	000002	166016		BIS	#BIT1,ADPTS	;IDLE SYNC
2885	013222	004537	013310	18700		JSR	R5,RNUM	;GENERATE "STALL" TIME
2886	013226	010137	001166	18800		MOV	R1,TIME	;FETCH RANDOM STALL TIME
2887	013232	005337	001166	18900	35:	DEC	TIME	;COUNT IDLE TIME
2888	013236	001375		19000		BNE	35	;TIME OUT?
2889	013240	004537	013310	19100		JSR	R5,RNUM	;GENERATE "ON" TIME + SYNC
2890	013244	042777	000002	165766		BIC	#BIT1,ADPTS	;CLEAR IDLE
2891	013252	000240		19300		NOP		
2892	013254	042777	000100	165750		BIC	#100,ADPRS	;CLEAR RCV INT ENB
2893	013262	000207		19500		RTS	PC	
2894				19600				
2895				19700				
2896				19800				
2897				19900				
2898	013264	112777	000026	165744		MOVB	#26,ASync	;SET UP FILLER SYNC
2899	013272	104403	005670	20100	SGEN:	DELAY	3000.	;DELAY 10 CHAR
2900	013276	004537	013310	20200		JSR	R5,RNUM	;RANDOM #
2901	013302	110137	001144	20300		MOVB	R1,TSync	;LOAD SYNC
2902	013306	000207		20400		RTS	PC	
2903				20500				
2904				20600				
2905				20700				
2906				20800				
2907				20900				
2908	013310	032737	040000	177570		BIT	#BIT14,SWR	;TEST FOR SCOPE LOOP
2909	013316	001010		21100	RNUM:	BNE	15	;EXIT IF SCOPE
2910	013320	060201		21200		ADD	R2,R1	
2911	013322	005501		21300		ADC	R1	
2912	013324	060102		21400		ADD	R1,R2	
2913	013326	005502		21500		ADC	R2	
2914	013330	060302		21600		ADD	R3,R2	
2915	013332	005502		21700		ADC	R2	
2916	013334	060203		21800		ADD	R2,R3	
2917	013336	005503		21900		ADC	R3	
2918	013340	000205		22000	15:	RTS	R5	
2919				22100				
2920				22200				
2921				22300				

;SGEN,ROUTINE TO GENERATE A UNIQUE SYNC CHARACTER

;RNUM, PSEUDO RANDOM NUMBER GENERATOR

```

2922      22400 ;CLOCK SUBROUTINE TO RUN SOFTWARE CLOCK
2923      22500 ;NUMBER OF CYCLES IS FETCHED CALL
2924      22600
2925      013342 005037 001102      22700 .CLOCK: CLR      TEMP1
2926      013346 032737 000400 001134 22800      BIT      #BIT8,SAVSRI      ;12/8 BITS/CHAR
2927      013354 001412      22900      BEQ      .RXCLK      ;BRANCH IF 9 BITS/CHAR
2928      013356 052777 002000 165646 23000      BIS      #BIT10,ADPRS      ;SELECT 12 BIT MODE
2929      013364 052737 000400 001160 23100      BIS      #BIT8,LIMIT      ;9 BIT SYNC
2930      013372 062737 000004 001102 23200      ADD      #4,TEMP1      ;INCREASE CLOCK COUNT
2931      013400 000402      23300      BR      .+6
2932      013402 005037 001102      23400 .RXCLK: CLR      TEMP1
2933      013406 005037 177776      23500      CLR      PS
2934      013412 067637 000000 001102 23600      ADD      @ (SP),TEMP1
2935      013420 062716 000002      23700      ADD      #2,(SP)
2936      013424 052777 000010 165606 23800 1$:      BIS      #BIT3,ADPTS      ;SET CLOCK HIGH
2937      013432 013705 013470      23900      MOV      FREQ,R5      ;SET UP DELAY COUNT
2938      013436 005305      24000      DEL=.
2939      013436 005305      24100      DEC      R5      ;DECREMENT COUNT
2940      013440 001376      24200      BNE      DEL      ;BRANCH IF NO TIMEOUT
2941      013442 042777 000010 165570 24300      BIC      #BIT3,ADPTS      ;SET CLOCK LOW
2942      013450 013705 013470      24400      MOV      FREQ,R5      ;SET UP DELAY COUNT
2943      013454 005305      24500      DEL=.
2944      013454 005305      24600      DEC      R5      ;DEC COUNT
2945      013456 001376      24700      BNE      DEL      ;BRANCH IF NO TIMEOUT
2946      013460 005337 001102      24800      DEC      TEMP1
2947      013464 001357      24900      BNE      1$
2948      013466 000002      25000      RTI
2949      013470 000030      25100      FREQ:      30      ;NORMAL 12.8 US DELAY
2950      25200      ;PATCH FOR 50 FT CABEL
2951      25300
2952      25400
2953      25500 ;REE, SUBROUTINE TO REINITIALIZE DP11 FOR NEXT TEST
2954      25600
2955      013472 012737 000200 177776 25700 REE:      MOV      #200,PS      ;SET PRIORITY TO 4
2956      013500 005077 165534      25800      CLR      ADPTS      ;CLEAR XMIT STATUS
2957      013504 005077 165522      25900      CLR      ADPRS      ;CLEAR RCY STATUS
2958      013510 105077 165530      26000      CLRB     @SEXT      ;CLEAR SYNC EXTENTION
2959      013514 012737 000001 001144 26100      MOV      #1,TSYNC      ;INIT TEST SYNC
2960      013522 012777 014504 165516 26200      MOV      #FRINT,ADPRIV      ;SET UP RVI INT VECTOR
2961      013530 012777 014510 165514 26300      MOV      #TV18,ADPTIV      ;SET XMIT INT VECTOR TO SYNC
2962      013536 032737 000400 001134 26400      BIT      #BIT8,SAVSRI      ;TEST FOR 8/12 BITS/CHAR
2963      013544 001405      26500      BEQ      1$      ;EXIT IF 8 BITS
2964      013546 062705 000010      26600      ADD      #10,R5      ;SET RETURN ADRS FOR 12 BIT LIMIT
2965      013552 052777 002000 165452 26700      BIS      #BIT10,ADPRS      ;SET 12 BIT/CHAR MODE
2966      013560 000205      26800      1$:      RTS      R5      ;RETURN
2967      26900
2968      27000 ;COMMON DATA AND IDLE SUBROUTINE
2969      27100
2970      013562 042777 004300 165442 27200 A2ND:     BIC      #4300,ADPRS      ;RCV INT ENB, RCV ACTIVE
2971      013570 042777 160342 165442 27300      BIC      #160342,ADPTS      ;INT ENBS IDLE SYNC, ERRORS
2972      013576 012737 015556 001172 27400      MOV      #BOTTOM,RP      ;RP = BOTTOM OF TUMBLE TABLE
2973      013604 013737 001172 001170 27500      MOV      RP,TP      ;SET UP TRANSMIT POINTER
2974      013612 012777 015172 165426 27600      MOV      #RRRR,ADPRIV      ;RCV INT VECTOR
2975      013620 012777 015044 165424 27700      MOV      #RRRT,ADPTIV      ;XMIT INT VECTOR
2976      013626 012737 000002 001162 27800      MOV      #2,SCNT      ;SYNC COUNT = 2
2977      013634 110237 001152      27900      MOVB     R2,TDATA      ;RANDOM DATA
  
```

K05

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 64
 DZDPAB.SRC

2978	013640	013777	001144	165374	28000	MOV	TSYNC, @DPTB	;LOAD BUFFER
2979	013646	052777	000100	165356	28100	BIS	#100, @DPRS	;RCV INT ENB
2980	013654	052777	000140	165356	28200	BIS	#140, @DPTS	;XMIT INT ENB
2981	013662	010137	001166		28300	MOV	R1, TIME	; "ON" STALL
2982	013666	104402			28400	1\$: RXCLK		
2983	013670	000001			28500	1		
2984	013672	005337	001166		28600	DEC	TIME	;0.6 SEC AVERAGE
2985	013676	001373			28700	BNE	1\$	
2986	013700	042777	000140	165332	28800	BIC	#140, @DPTS	;TURN OFF INT ENB
2987	013706	052777	000002	165324	28900	BIS	#BIT1, @DPTS	;IDLE SYNC
2988	013714	004537	013310		29000	JSR	R5, RNUM	;GENERATE "STALL" TIME
2989	013720	010137	001166		29100	MOV	R1, TIME	
2990	013724	104402			29200	2\$: RXCLK		
2991	013726	000001			29300	1		
2992	013730	005337	001166		29400	DEC	TIME	;COUNT IDLE TIME
2993	013734	001373			29500	BNE	2\$;TIME OUT?
2994	013736	004537	013310		29600	JSR	R5, RNUM	;GENERATE "ON" TIME + SYNC
2995	013742	000207			29700	RTS	PC	
2996					29800			
2997					29900			
2998					30000			
2999					30100			
3000	013744	012737	013754	000024	30200	.PFAIL: MOV	#PWRUP, 24	;LOAD PFAIL VECTOR FOR POWER UP
3001	013752	000000			30300	HALT		
3002	013754	000005			30400	PWRUP: RESET		;WAIT TTY TO COME UP
3003	013756	012706	001050		30500	MOV	#STACK, SP	;REINIT STACK POINTER
3004	013762	012737	013744	000024	30600	MOV	#.PFAIL, 24	;LOAD PFAIL VECTOR FOR POWER DOWN
3005	013770	104407			30700	TYPE		
3006	013772	016770			30800	MPOWER		
3007	013774	000177	165060		30900	JMP	@RETURN	
3008					31000			
3009					31100			
3010	014000				31200	SEQ.DATA:		
3011	014000	011637	001174		31300	MOV	(SP), BACK	
3012	014004	105077	165234		31400	CLRB	@SEXT	;CLEAR SYNC EXTENTION
3013	014010	005037	001154		31500	CLR	RDATA	;RECEIVER DATA
3014	014014	005037	001152		31600	CLR	TDATA	;TRANSMITTER DATA
3015	014020	005077	165206		31700	CLR	@DPRS	;RECEIVER STATUS
3016	014024	005077	165210		31800	CLR	@DPTS	;TRANSMITTER STATUS
3017	014030	052777	000001	165174	31900	BIS	#BIT0, @DPRS	;STRIP SYNC
3018	014036	012737	000400	001156	32000	MOV	#400, CHLEN	;CHAR LENGTH INDEX
3019	014044	032737	000400	001134	32100	BIT	#BIT8, SAVSR1	;TEST 12 BIT CHAR MODE
3020	014052	001414			32200	BEQ	1\$;NO
3021	014054	012737	010000	001160	32300	MOV	#10000, LIMIT	;SELECT END OF DATA
3022	014062	052777	002000	165142	32400	BIS	#BIT10, @DPRS	;SELECT 12 BITS/CHARACTER
3023	014070	012737	000426	001144	32500	MOV	#426, TSYNC	;SYNC FOR 12 BIT CHAR
3024	014076	105277	165142		32600	INCB	@SEXT	;PLACE MSB OF SYNC IN SYNC EXT
3025	014102	000406			32700	BR	2\$	
3026	014104	012737	000400	001160	32800	1\$: MOV	#400, LIMIT	;TEMPORARY CHARACTER LIMIT
3027	014112	012737	000026	001144	32900	MOV	#26, TSYNC	;INIT SYNC STORAGE
3028	014120	012777	014510	165124	33000	2\$: MOV	#TV18, @DPTIV	;TRANSMITTER VECTOR
3029	014126	012777	014616	165112	33100	MOV	#RV18, @DPRIV	;RECEIVER VECTOR
3030	014134	012737	000200	177776	33200	MOV	#200, PS	;PRIORITY=4
3031	014142	012737	000004	001162	33300	MOV	#4, SCNT	;SYNC COUNT=4
3032	014150	113777	001144	165060	33400	MOVB	TSYNC, @SYNC	;LOAD SYNC
3033	014156	052777	000100	165046	33500	BIS	#BIT6, @DPRS	;RCV INT ENB


```

3076      00100
3077      00200
3078      00300
3079      00400      ; INTERRUPT SERVICE ROUTINES
3080      00500
3081      00600      ; THESE ROUTINES MAY FUNCTION AS:
3082      00700      ; 1. ERROR TRAPS FOR FALSE INTERRUPTS
3083      00800      ; 2. POINTERS BACK TO THE MAIN LINE FOR VALID INTERRUPTS
3084      00900      ; 3. FUNCTIONAL TEST WHICH ARE INTERRUPT DRIVEN
3085      01000
3086      01100      ; VERIFY THAT INTERRUPT RESULTED FROM RING.
3087      014354 032777 020000 164656 01200 TV24: BIT      #BIT13,ADPTS      ; TEST FOR RING
3088      014362 001001      01300      BNE      1$      ; BRANCH IF SET.
3089      014364 104000      01400      HLT      ; REPORT ERROR
3090      01500
3091      014366 042777 020000 164644 01600 1$: BIC      #BIT13,ADPTS      ; CLEAR RING FLAG.
3092      014374 032777 020000 164636 01700      BIT      #BIT13,ADPTS      ; TEST IT
3093      014402 001401      01800      BEQ      2$      ; BRANCH IF CLEAR.
3094      014404 104000      01900      HLT      ; REPORT ERROR
3095      02000
3096      014406 032777 140200 164624 02100 2$: BIT      #140200,ADPTS      ; NO OTHER STATUS FLAG ON?
3097      014414 001401      02200      BEQ      3$
3098      014416 104000      02300      HLT      ; REPORT ERROR
3099      014420 022626      02400 3$: POP.SP
3100      014422 000137 011722 02500      JMP      RCD1
3101      02600
3102      02700      ; VERIFY THAT INTERRUPT RESULTED FROM 'CARRIER DOWN' FLAG
3103      014426 032777 100000 164604 02800 TV25: BIT      #BIT15,ADPTS      ; TEST FOR 'CARRIER DOWN' FLAG
3104      014434 001001      02900      BNE      1$      ; BRANCH IF SET
3105      014436 104000      03000      HLT      ; REPORT ERROR
3106      03100
3107      014440 042777 100000 164572 03200 1$: BIC      #BIT15,ADPTS      ; CLEAR 'CARRIER DOWN' FLAG.
3108      014446 032777 100000 164564 03300      BIT      #BIT15,ADPTS      ; TEST IT
3109      014454 001401      03400      BEQ      2$      ; BRANCH IF CLEAR.
3110      014456 104000      03500      HLT      ; REPORT ERROR
3111      03600
3112      014460 032777 060200 164552 03700 2$: BIT      #060200,ADPTS      ; NO OTHER FLAGS ON?
3113      014466 001401      03800      BEQ      3$
3114      014470 104000      03900      HLT      ; REPORT ERROR
3115      04000
3116      014472 013716 001174 04100 3$: MOV      BACK,(SP)
3117      014476 000002      04200      RTI
3118      04300
3119      04400
3120      04500
3121      04600
3122      014500 104005      04700 FTINT: HLT      5      ; ERROR ROUTINE SHOULD RETURN TO MAINLINE
3123      014502 000002      04800      RTI
3124      04900
3125      014504 104006      05000 FRINT: HLT      6      ; ERROR ROUTINE SHOULD RETURN TO MAINLINE
3126      014506 000002      05100      RTI
3127      05200
3128      05300
3129      05400
3130      014510 013777 001144 164524 05500 TV18: MOV      TSYNC,ADPTB      ; XMIT SYNC
3131      014516 113777 001145 164520 05600      MOVB    TSYNC+1,ASEXT      ; LOAD SYNC EXT
  
```

N05.

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 67
 DZDPAB.SRC

3132	014524	005337	001162	05700	DEC	SCNT	;HAVE 2 SYNC'S BEEN XMITED
3133	014530	001003		05800	BNE	1\$;NO
3134	014532	012777	014542	164512	MOV	#TV19,ADPTIV	;YES CHANGE VECTOR
3135	014540	000002		06000	1\$: RTI		
3136				06100			
3137				06200			;SEQUENTIAL DATA TRANSMISSION ROUTINE
3138				06300			
3139				06400			
3140	014542	032777	140000	164470	TV19: BIT	#140000,ADPTS	;ANY STATUS ERRORS
3141	014550	001401		06600	BEQ	+.4	;NO
3142	014552	104000		06700	HLT		;REPORT ERROR
3143	014554	105777	164460	06800	TSTB	ADPTS	;TRANSMITTER READY
3144	014560	100401		06900	BMI	+.4	;YES
3145	014562	104000		07000	HLT		;REPORT ERROR
3146	014564	013777	001152	164450	MOV	TDATA,ADPTB	;LOAD BUFFER
3147	014572	005237	001152	07200	INC	TDATA	;NEXT CHARACTER
3148	014576	023737	001160	001152	CMP	LIMIT,TDATA	;HAVE ALL CHARACTERS
3149	014604	001003		07400	BNE	1\$;NO
3150	014606	042777	000140	164424	BIC	#140,ADPTS	;YES, CLEAR INTERRUPTS
3151	014614	000002		07600	1\$: RTI		
3152				07700			
3153				07800			;RECEIVE SEQUENTIAL DATA
3154				07900			
3155	014616	105777	164410	08000	RV18: TSTB	ADPRS	;RECEIVER DONE??
3156	014622	100401		08100	BMI	+.4	;YES
3157	014624	104000		08200	HLT		;REPORT ERROR
3158	014626	013700	001154	08300	MOV	RDATA,R0	
3159	014632	017701	164376	08400	MOV	ADPRB,R1	
3160	014636	023777	001154	164370	CMP	RDATA,ADPRB	;CORRECT DATA
3161	014644	001404		08600	BEQ	1\$	
3162	014646	017737	164362	001136	MOV	ADPRB,TMPDAT	;STORE DATA
3163	014654	104001		08800	HLT	1	;REPORT ERROR
3164	014656	042777	000001	164346	1\$: BIC	#BIT0,ADPRS	;CLEAR STRIP SYNC
3165	014664	005237	001154	09000	INC	RDATA	;NEXT CHARACTER
3166	014670	023737	001160	001154	CMP	LIMIT,RDATA	
3167	014676	001047		09200	BNE	3\$	
3168	014700	005037	001154	09300	CLR	RDATA	
3169	014704	005037	001152	09400	CLR	TDATA	
3170	014710	006237	001160	09500	ASR	LIMIT	;DECREASE LIMIT TO 7 BITS
3171	014714	012777	014510	164330	MOV	#TV18,ADPTIV	;SET UP SYNC TRANSMISSION
3172	014722	012737	000004	001162	MOV	#4,SCNT	;SYNC COUNT =4
3173	014730	052777	000001	164274	BIS	#BIT0,ADPRS	;STRIP SYNC
3174	014736	042777	004000	164266	BIC	#BIT11,ADPRS	;CLEAR RCV ACTIVE
3175	014744	032737	000001	001134	BIT	#BIT0,SAVSR1	
3176	014752	001004		10100	BNE	+.12	
3177	014754	052777	000340	164256	BIS	#340,ADPTS	;INT ENB + DONE
3178	014762	000403		10300	BR	+.10	
3179	014764	052777	000301	164246	BIS	#301,ADPTS	
3180	014772	053777	001156	164232	BIS	CHLEN,ADPRS	;CHANGE CHAR LENGTH
3181	015000	062737	000400	001156	ADD	#400,CHLEN	;DECREASE CHAR LENGTH
3182	015006	022737	001400	001156	2\$: CMP	#1400,CHLEN	;HAVE ALL LENGTHS BEEN TESTED
3183	015014	001401		10800	BEQ	4\$;YES
3184	015016	000002		10900	3\$: RTI		;NO
3185	015020	005077	164214	11000	4\$: CLR	ADPTS	;CLR TRANSMITTER STATUS
3186	015024	005077	164202	11100	CLR	ADPRS	;CLR RECEIVER STATUS
3187	015030	005037	177776	11200	CLR	PS	

```

015034 012706 001050 11300
015040 000177 164130 11400
11500
11600
11700
11800
11900
12000
015044 105777 164170 12100
015044 100401 12200
015050 104000 12300
015054 013777 001144 164160 12400
015062 005337 001162 12500
015066 001006 12600
015070 012777 015106 164154 12700
015076 042777 160000 164134 12800
015104 12900
015104 000002 13000
13100
13200
13300
13400
13500
015106 105777 164126 13600
015106 100401 13700
015112 104000 13800
015116 004537 013310 13900
015122 010137 001152 14000
015126 043737 001156 001152 14100
015134 013777 001152 164100 14200
015142 013704 001173 14300
015146 013724 001152 14400
015152 020427 015566 14500
015156 001002 14600
015160 012704 015556 14700
015164 010437 001170 14800
015170 000002 14900
15000
15100
015172 011637 001110 15200
015176 017737 164032 001154 15300
015204 013702 001172 15400
015210 013701 001154 15500
015214 011200 15600
015216 023722 001154 15700
015222 001406 15800
015224 023737 001154 001144 15900
015232 001401 16000
015234 104002 16100
015236 005742 16200
015240 022702 015566 16300
015244 001002 16400
015246 012702 015556 16500
015252 010237 001172 16600
015256 042777 000001 163746 16700
015264 000002 16800
  
```

```

MOV #STACK, SP
JMP @BACK
; SYNC ROUTINE FOR RANDOM DATA TEST
; SYNC ROUTINE FOR RANDOM DATA TEST
RRRT:
TSTB @DPTS :READY
BMI .+4 :YES
HLT :REPORT ERROR
MOV TSYNC, @DPTB :TRANSMIT SYNC
DEC SCNT :2 SYNC'S
BNE IS :NO
MOV @RRT1, @DPTIV :YES, SET UP DATA TRANSMIT VECTOR
BIC #160000, @DPTS :CLEAR ERROR BITS
IS:
RTI
;RRRT, RANDOM DATA, SYNC, RANDOM STALL
;TRANSMITTER SERVICE ROUTINE
RRT1:
TSTB @DPTS :TRANSMITTER READY
BMI .+4 :YES
HLT :REPORT ERROR
JSR R5, RNUM :GENERATE NEXT CHARACTER
MOV R1, TDATA
BIC CHLEN, TDATA :REDUCE DATA TO # BITS/CHAR
MOV TDATA, @DPTB :TRANSMIT CHARACTER
MOV TP, R4 :SET UP TRANSMITTER POINTER
MOV TDATA, (R4)+ :MOV CHARACTER TO TUMBLE TABLE
CMP R4, #TOP :END OF TUMBLE TABLE
BNE IS
MOV #BOTTOM, R4
IS:
MOV R4, TP :SAVE TRANSMITTER POINTER
RTI
;RRRR, RANDOM DATA, RANDOM SYNC, RANDOM STALL, RECEIVER SERVICE
RRRR:
MOV (SP), TEMP4
MOV @DPRB, RDATA :SAVE RECEIVED DATA
MOV RP, R2 :SET UP RECEIVER POINTER
MOV RDATA, R1
MOV (R2), R0
CMP RDATA, (R2)+ :IS DATA CORRECT
BEQ IS :YES
CMP RDATA, TSYNC :IF NOT DATA IS IT SYNC
BEQ .+4 :YES
HLT 2 :REPORT ERROR
TST -(R2) :ADJUST TUMBLE TABLE
IS:
CMP #TOP, R2 :TOP OF TUMBLE TABLE
BNE 2$ :NO
MOV #BOTTOM, R2 :YES, RAP AROUND
2$:
MOV R2, RP :SAVE RECEIVER POINTER
BIC #BIT0, @DPRS :CLEAR STRIP SYNC
RTI
  
```

```

16900
17000
17100
17200
17300
17400
17500
015266
015266 013777 001144 163746
015274 005337 001162
015300 001003
015302 012777 015312 163742
015310 000002
18000
18100
015312 032777 160000 163720
015320 001401
015322 104000
015324 013777 001152 163710
015332 005237 001152
015336 023737 001152 001160
015344 001401
015346 000002
015350 005077 163664
015354 000002
19000
19100
19200
19300
19400
000000
19500
19600
015356 017727 163650 000000
015364 017737 163644 001136
015372 013700 001154
015376 013701 001136
015402 023737 001136 001154
015410 001401
015412 104001
015414 032737 010000 015362
015422 001401
015424 104000
20000
20100
20200
20300
20400
20500
20600
20700
20800
20900
21000
015426 005237 001154
015432 005037 001104
015436 012737 000020 001106
015444 000241
015446 006137 001154
015452 103002
015454 005137 001104
015460 005337 001106
015464 001370
21100
21200
21300
21400
21500
21600
21700
21800
21900
22000
22100
22200
22300
015466 006137 001154
22400
  
```

; TRANSMITTER SERVICE ROUTINES FOR PARITY TEST

```

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

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

; RECEIVER SERVICE ROUTINE FOR PARITY TEST

```

HERE=0
RPRTY:  MOV    DPPTS, #HERE  ;SAVE RCV STATUS HERE
        MOV    DPPTB, 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
  
```


E06

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 71
 DZDPAB.SRC

```

3343
3344                                     ;SCOPE LOOP AND INTERATION HANDLER
3345
3346 015640 032737 040000 177570      .SCOPE: BIT      #BIT14,SWR
3347 015646 001407      TTST: BEQ      1$
3348 015650 000432      BR      3$
3349 015652 105777 163172      TSTB     @TKCSR
3350 015656 100027      BPL      3$
3351 015660 017700 163166      MOV      @TKDBR,RO
3352 015664 000412      BR      2$
3353 015666 032737 004000 177570      1$: BIT      #SW11,SWR
3354 015674 001006      BNE      2$
3355 015676 005237 001070      INC      LPCNT
3356 015702 023737 001070 001066      CMP      LPCNT,ICOUNT
3357 015710 001012      BNE      3$
3358 015712 105037 001200      2$: CLRB     ERRFLG
3359 015716 005037 001070      CLR      LPCNT
3360 015722 012737 000012 001066      MOV      #10,ICOUNT
3361 015730 013737 001062 001060      MOV      NEXT,RETURN
3362 015736 013716 001060      3$: MOV      RETURN,(SP)
3363 015742 000002      RTI
3364 015744 001407      BRW: 1407
3365 015746 000432      BRX: 432
3366
3367                                     ;END OF PASS
3368                                     ;TYPE "END OF PASS CSR: XXXXXX"
3369                                     ;UPDATE PASS COUNT
3370                                     ;UPDATE LINE NUMBER
3371                                     ;IF IN CYCLE MODE
3372                                     ;RESTART TEST
3373
3374 015750 005037 001100      .EOP: CLR      LSTERR      ;CLEAR LAST ERROR PC
3375 015754 005037 001200      CLR      ERRFLG      ;CLEAR ERROR FLAG
3376 015760 005237 001074      INC      PASCNT      ;UPDATE PASS COUNT
3377 015764 104407      TYPE
3378 015766 017005      MEPASS
3379 015770 104413      CNVRT
3380 015772 016044      XCSR
3381 015774 105737 001134      TSTB     SAVSR1
3382 016000 100402      BMI      .+6
3383 016002 005237 001146      INC      XLINEX
3384 016006 013737 001074 177570      MOV      PASCNT,LIGHTS      ;DISPLAY PASS COUNT
3385 016014 012737 001332 001060      RESTRT: MOV      #PART1,RETURN
3386 016022 012706 001050      MOV      #STACK,SP
3387 016026 105737 001134      TSTB     SAVSR1
3388 016032 100002      BPL      .+6
3389 016034 000137 001332      JMP      PART1
3390 016040 000137 001322      JMP      BGNO
3391 016044 000001      XCSR: 1
3392 016046 006      .BYTE 6,2
3393 016050 001232      DPRS
3394
3395                                     ;CHECK FOR FREEZE ON CURRENT DATA
3396
3397 016052 032737 001000 177570      .SCOPI: BIT      #SW09,SWR
3398 016060 001402      BEQ      1$

```

3399	016062	013716	001064			MOV	LOCK, (SP)
3400	016066	000002			1\$:	RTI	
3401							
3402							; TELETYPE OUTPUT ROUTINE
3403							
3404	016070	017605	000000		.TYPE:	MOV	@(SP), R5
3405	016074	062716	000002			ADD	#2, (SP)
3406	016100	032737	010000	177570	1\$:	BIT	#SW12, SWR
3407	016106	001010				BNE	3\$
3408	016110	105715				TSTB	(R5)
3409	016112	001406				BEQ	3\$
3410	016114	105777	162734		2\$:	TSTB	@TPCSR
3411	016120	100375				BPL	2\$
3412	016122	112577	162730			MOVB	(R5)+, @TPDBR
3413	016126	000764				BR	1\$
3414	016130	000002			3\$:	RTI	
3415							
3416							; ERROR HANDLER
3417							
3418	016132	032737	010000	177570	.HLT:	BIT	#SW12, SWR
3419	016140	001406				BEQ	XBX
3420	016142	105777	162706			TSTB	@TPCSR
3421	016146	100003				BPL	XBX
3422	016150	112777	000207	162700		MOVB	#207, @TPDBR
3423	016156	032737	020000	177570	XBX:	BIT	#SW13, SWR
3424	016164	001075				BNE	HALTS
3425	016166	021637	001100			CMP	(SP), LSTERR
3426	016172	001404				BEQ	1\$
3427	016174	011637	001100			MOV	(SP), LSTERR
3428	016200	105037	001200			CLRB	ERRFLG
3429	016204	104410			1\$:	SAVOS	
3430	016206	011605				MOV	(SP), R5
3431	016210	162705	000002			SUB	#2, R5
3432	016214	011504				MOV	(R5), R4
3433	016216	006304				ASL	R4
3434	016220	061504				ADD	(R5), R4
3435	016222	006304				ASL	R4
3436	016224	042704	177001			BIC	#177001, R4
3437	016230	062704	017246			ADD	#.ERRTAB, R4
3438	016234	012437	016330			MOV	(R4)+, ERRMSG
3439	016240	012437	016342			MOV	(R4)+, DATAHD
3440	016244	011437	016354			MOV	(R4), DATABP
3441	016250	105737	001200			TSTB	ERRFLG
3442	016254	001403				BEQ	TYPMSG
3443	016256	005737	016354			TST	DATABP
3444	016262	001030				BNE	TYPDAT
3445	016264	104407			TYPMSG:	TYPE	
3446	016266	017030				MTSTN	
3447	016270	104413				CNVRT	
3448	016272	016462				XTSTN	
3449	016274	104407				TYPE	
3450	016276	017044				MLINE	
3451	016300	104413				CNVRT	
3452	016302	016470				ZLINE	
3453	016304	104407				TYPE	
3454	016306	017053				MERRPC	

H06

3511	016550	012700	017350
3512	016554	010403	
3513	016556	042703	177770
3514	016552	062703	000260
3515	016556	110320	
3516	016570	000241	
3517	016572	006004	
3518	016574	000241	
3519	016576	006004	
3520	016600	000241	
3521	016602	006004	
3522	016604	005305	
3523	016606	001362	
3524	016610	012703	017412
3525	016614	114023	
3526	016616	105337	016672
3527	016622	001374	
3528	016624	105737	016673
3529	016630	001405	
3530	016632	112723	000240
3531	016636	105337	016673
3532	016642	001373	
3533	016644	105013	
3534	016646	104407	
3535	016650	017412	
3536	016652	005337	016670
3537	016656	001322	
3538	016660	013737	001106 017350
3539	016666	000002	
3540	016670	000000	
3541	016672	000000	
3542		016673	
3543	016674	000000	
3544			
3545			
3546			
3547	016676	016637	000004 001132
3548			
3549			
3550			
3551	016704	010537	001126
3552	016710	010437	001124
3553	016714	010337	001122
3554	016720	010237	001120
3555	016724	010137	001116
3556	016730	010037	001114
3557	016734	000002	
3558			
3559			
3560			
3561	016736	013700	001114
3562	016742	013701	001116
3563	016746	013702	001120
3564	016752	013703	001122
3565	016756	013704	001124
3566	016762	013705	001126

```

3$: MOV #TEMP, R0
   MOV R4, R3
   BIC #177770, R3
   ADD #260, R3
   MOV R3, (R0)+
   CLC
   RCR R4
   CLC
   ROR R4
   CLC
   ROR R4
   DEC R5
   BNE 3$
4$: MOV #MDATA, R3
   MOVB -(R0), (R3)+
   DECB CHRCNT
   BNE 4$
   TSTB SPACNT
   BEQ 6$
5$: MOVB #240, (R3)+
   DECB SPACNT
   BNE 5$
6$: CLRB (R3)
   TYPE
   MDATA
   DEC WRDCNT
   BNE 1$
   MOV TEMP3, TEMP
   RTI

WRDCNT: 0
CHRCNT: 0
SPACNT=CHRCNT+1
BINWRD: 0

;SAVE PC OF TEST THAT FAILED AND R0-R5
.SAV05: MOV 4(SP), SAVPC

;SAVE R0-R5
SV05: MOV R5, SAVR5
      MOV R4, SAVR4
      MOV R3, SAVR3
      MOV R2, SAVR2
      MOV R1, SAVR1
      MOV R0, SAVR0
      RTI

;RESTORE R0-R5
.RES05: MOV SAVR0, R0
        MOV SAVR1, R1
        MOV SAVR2, R2
        MOV SAVR3, R3
        MOV SAVR4, R4
        MOV SAVR5, R5
  
```

3567	016756	000002							RTI
3568	016770	005015	053520	020122	27600	MPOWER:	.ASCIZ	<15><12>/PWR FAILED/	
3569	016776	040506	046111	042105					
3570	017004	000							
3571	017005	007	005407	042412	27700	MEPASS:	.ASCIZ	<7><7><15><12>/END PASS CSR: /	
3572	017012	042116	050040	051501					
3573	017020	020123	051503	035122					
3574	017026	000040							
3575	017030	005015	042524	052123	27800	MTSTN:	.ASCIZ	<15><12>/TEST NO. /	
3576	017036	047040	027117	000040					
3577	017044	044514	042516	020072	27900	MLINE:	.ASCIZ	/LINE: /	
3578	017052	000							
3579	017053	015	050012	035103	28000	MERRPC:	.ASCIZ	<15><12>/PC: /	
3580	017060	000040							
3581	017062	005015	000		28100	MCRLF:	.ASCIZ	<15><12>	
3582	017065	015	042412	052116	28200	EM1:	.ASCIZ	<15><12>/ENTERED FROM /	
3583	017072	051105	042105	043040					
3584	017100	047522	020115	000040					
3585	017106	005015	051124	047101	28300	EM2:	.ASCIZ	<15><12>/TRANSMITTER /	
3586	017114	046523	052111	042524					
3587	017122	020122	000040						
3588	017126	005015	042522	042503	28400	EM3:	.ASCIZ	<15><12>/RECEIVER /	
3589	017134	053111	051105	020040					
3590	017142	000							
3591	017143	105	050130	041505	28500	DH0:	.ASCIZ	/EXPECTED RECEIVED /	
3592	017150	042524	020104	051040					
3593	017156	041505	044505	042526					
3594	017164	020104	000						
3595	017167	106	044501	042514	28600	DH1:	.ASCIZ	/FAILED TO INTERRUPT. /	
3596	017174	020104	047524	044440					
3597	017202	052116	051105	050125					
3598	017210	027124	000040						
3599	017214	047111	042524	052522	28700	DH2:	.ASCIZ	/INTERRUPTED UNEXPECTEDLY. /	
3600	017222	052120	042105	052440					
3601	017230	042516	050130	041505					
3602	017236	042524	046104	027131					
3603	017244	000							
3604					28800				
3605		017246			28900				
3606	017246				29000	.EVEN			
3607	017246	000000			29100	.ERRTAB:			
3608	017250	000000			29200		0		
3609	017252	000000			29300		0		
3610	017254	017062			29400		MCRLF		
3611	017256	017143			29500		DH0	;HALT 1	
3612	017260	017320			29600		DT0		
3613					29700				
3614	017262	017065			29800		EM1		
3615	017264	017143			29900		DH0	;HALT 2	
3616	017266	017332			30000		DT1		
3617					30100				
3618	017270	017106			30200		EM2		
3619	017272	017167			30300		DH1	;HALT 3	
3620	017274	000000			30400		0		
3621					30500				
3622	017276	017126			30600		EM3		

3623	017300	017167		30700		DH1	;HALT 4
3624	017302	000000		30800		0	
3625				30900			
3626	017304	017106		31000		EM2	
3627	017306	017214		31100		DH2	;HALT 5
3628	017310	000000		31200		0	
3629				31300			
3630	017312	017126		31400		EM3	
3631	017314	017214		31500		DH2	;HALT 6
3632	017316	000000		31600		0	
3633				31700			
3634	017320	000002		31800	DT0:	2	
3635	017322	006	004	31900		.BYTE	6,4
3636	017324	001114		32000		SAVR0	
3637	017326	006	002	32100		.BYTE	6,2
3638	017330	001116		32200		SAVR1	
3639				32300			
3640	017332	000003		32400	DT1:	3	
3641	017334	006	010	32500		.BYTE	6,8.
3642	017336	001110		32600		TEMP4	
3643	017340	006	004	32700		.BYTE	6,4
3644	017342	001114		32800		SAVR0	
3645	017344	006	002	32900		.BYTE	6,2
3646	017346	001116		33000		SAVR1	
3647				33100			
3648	017350	000000		33200	TEMP:	0	
3649		017412		33300	.=. +40		
3650	017412	000000		33400	MDATA:	0	
3651		017454		33500	.=. +40		
3652		000001		33600	.END		

RESOS = 104411	1003#	3470												
RETURN 001060	926#	1048*	1715*	1716	1828*	2163*	2164	2165*	2184*	2185	3007	3361*	3362	
RNUM 013310	3385*	3484*	3486											
RP 001172	2885	2889	2900	2908#	2988	2994	3214							
RPRTY 015356	966#	2871*	2872	2972*	2973	3229	3241*							
RPRT1 015422	2070	3272#	3279											
RPRT2 015446	2054*	3280#	3302*	3304*										
RPRT3 015460	3290#	3294												
RPRT4 015510	3291	3293#												
RPRT5 015516	3301	3304#												
RPRT6 015536	3303	3305#												
RRRR 015172	3307	3309#												
RRRT 015044	2873	2974	3227#											
RRT1 015106	2874	2975	3195#											
RV18 014616	3202	3210#												
RXCLK = 104402	1995	3029	3155#											
RO =%000000	989#	2226	2241	2288	2303	2351	2366	2982	2990	3050				
	575#	1088*	1125*	1177*	1178	1198*	1199*	1202*	1203	1224*	1228*	1250*	1254*	
	1272*	1276*	1280*	1300*	1330*	1407*	1408*	1420*	1421*	1423*	1578	1599	1628	
	1656	1691*	1721*	1784*	1861*	1923*	1932*	2654*	2707*	2712*	2796*	2802*	2808*	
	2814*	2820*	2833*	2839*	2849*	2850*	2852*	3158*	3231*	3274*	3333*	3351*	3474*	
	3511*	3515*	3525	3556	3561*									
R1 =%000001	576#	1175*	1176*	1178	1201*	1203	1225*	1226	1251*	1252	1273*	1424*	1692*	
	1720*	1783*	1862*	1924*	1933*	2417*	2656*	2657*	2658	2678*	2679*	2680	2714*	
	2715*	2717*	2719*	2720*	2721	2722*	2723*	2724*	2731	2749	2759*	2763	2776*	
	2777	2779	2780	2797*	2803*	2809*	2815*	2834*	2853*	2866*	2880	2886	2901	
	2910*	2911*	2912	2981	2989	3159*	3215	3230*	3275*	3334*	3502*	3505	3506	
	3507	3508	3555	3562*										
R2 =%000002	577#	1909*	1947	1948	1949	1956	2418*	2653*	2655	2660*	2664*	2665*	2730*	
	2731*	2733	2734	2735	2736*	2737	2738*	2739	2740	2741	2742*	2743	2744*	
	2745*	2748*	2749*	2750	2751	2752	2753	2754	2755	2756	2775*	2777*	2778*	
	2867*	2876	2910	2912*	2913*	2914*	2915*	2916	2977	3229*	3231	3232	3237	
	3238	3240*	3241	3554	3563*									
R3 =%000003	578#	1906*	1907*	1947*	1948*	2419*	2732*	2733*	2735*	2737*	2739*	2741*	2743*	
	2750*	2752*	2754*	2756*	2868*	2914	2916*	2917*	3512*	3513*	3514*	3515	3524*	
	3525*	3530*	3533*	3553	3564*									
R4 =%000004	579#	3218*	3219*	3220	3222*	3223	3432*	3433*	3434*	3435*	3436*	3437*	3438	
	3439	3440	3509*	3512	3517*	3519*	3521*	3552	3565*					
R5 =%000005	580#	1090*	1092*	1094*	1096*	1098*	1100*	1102*	1104*	1127*	1130*	1133*	1135*	
	1137*	1139*	1141*	1143*	1145*	2211*	2272*	2334*	2793	2794	2821*	2885*	2889*	
	2900*	2918*	2937*	2939*	2942*	2944*	2964*	2966*	2988*	2994*	3214*	3404*	3408	
	3412	3430*	3431*	3432	3434	3510*	3522*	3551	3566*					
SAVPC 001132	950#	3490	3547*											
SAVR0 001114	943#	3556*	3561	3636	3644									
SAVR1 001116	944#	3555*	3562	3638	3646									
SAVR2 001120	945#	3554*	3563											
SAVR3 001122	946#	3553*	3564											
SAVR4 001124	947#	3552*	3565											
SAVR5 001126	948#	3551*	3566											
SAVSP 001130	949#													
SAVSR1 001134	951#	1713	1826	1951	1985	2064	2090	2161	2182	2648	2706*	2718*	2727	
	2746	2926	2962	3019	3034	3041	3175	3381	3387	3478				
SAVSR2 001164	963#	2711*	2715	2744										
SAVOS = 104410	1001#	3429												
SCNT 001162	962#	1997*	2061*	2218*	2282*	2342*	2875*	2976*	3031*	3132*	3172*	3200*	3252*	
SCOPE = 104400	985#	1074	1106	1147	1181	1206	1230	1256	1282	1318	1351	1388	1450	

		1491	1520	1560	1582	1608	1638	1665	1700	1733	1814	1884	1965	2005
		2075	2075	2121	2144	2197	2258	2320	2383	2396	2437	2487	2529	2560
SCOPE =	104406	2696	1696	1725										
SYD =	014000	2695	3010*											
SYT =	001244	1718*	1719*	1720	1722	1729*	1730	1756*	1833*	1840*	1914*	1979*	1990*	
		2068*	2189*	2223*	2413*	2958*	3012*	3024*	3131*					
SGEN	013264													
SLIM	001140	1760*	1806	1810*	1834*	1873	1978*							
SP =	000006	897*	898*	899*	900*	901*	902*	903*	904	1035*	2150*	2829	2830	
		2844	2845	2846*	2934	2935*	3003*	3011	3060	3061*	3062	3116*	3188*	
		3311*	3362*	3386*	3399*	3404	3405*	3425	3427	3430	3474	3485*	3502	
		3547												
SPACNT =	016673	3528	3531*	3542*										
SPV =	015570	2308	2371	3330*										
STACK =	001090	1036	2150	3003	3189	3386	3485							
STARR	001302													
STRT1	000200													
STRT2	001177													
SYV =	016704													
SWR =	177570	1038	1803	1870	2152	2252	2315	2378	2706	2707	2711	2712	2908	
		3353	3397	3406	3418	3423	3471	3480	3492					
SW00	000001													
SW01	000002													
SW02	000004													
SW03	000010													
SW04	000020													
SW05	000040													
SW06	000100													
SW07	000200													
SW08	000400													
SW09	001000	3480												
SW10	002000	3397												
SW11	004000	3482												
SW12	010000	3353												
SW13	020000	3406	3418											
SW14	040000	3423												
SW15	100000													
SYNC	001236	1018*	1068*	1268*	1686*	1687	1690*	1692	1693	1699*	1766*	1767	1796*	1832*
		1829*	1875*	1913*	1998*	2020*	2060*	2089*	2139*	2187*	2222*	2254*	2285*	2314*
		2345*	2348*	2377*	2416*	2898*	3032*							
TDATA	001152	958*	1767*	1981*	2057*	2464	2505	2540	2876*	2977*	3014*	3146	3147*	3148
		3169*	3215*	3216*	3217	3219	3260	3261*	3262					
TEMP	017350	3503	3511	3538*	3648*									
TEMP1	001102	938*	2693*	2925*	2930*	2932*	2934*	2946*	3053*					
TEMP2	001104	939*	2694*	3055*	3287*	3292*	3300							
TEMP3	001106	940*	3060*	3065*	3068*	3072*	3288*	3293*	3503*	3538				
TEMP4	001110	941*	2793*	2829*	2844*	3062*	3074	3227*	3642					
TEMP5	001112	942*												
TIME	001166	964*	2980*	2881*	2886*	2887*	2981*	2984*	2989*	2992*				
TKCSR	001050	919*	3349											
TKDER	001052	920*	3351											
TLAST =	011770	3367*												
TMPDAT	001136	952*	1370*	1406*	1418*	1423	1425	1428*	1437*	1441*	1443*	1683*	1690	1691
		1693	1697*	1717*	1719	1721	1722	1726*	1727	1782*	1783	1785	1860*	1861
		1863	1921*	1924	1925	1929	1933	1934	2087*	2105	2108	2119*	2248*	2250

TSP	015566	32854	2858*	3162*	3273*	3275	3276													
TP	001170	3220	3238	3317*																
TPCOS	001054	965*	2872*	3297*	3218	3223*														
TPDOR	001056	921*	3410	3420																
TPRTY	015266	923*	3412*	3422*																
TPROK	000070	2069	3250																	
TSTNO	001072	931*																		
		1465*	1059*	1086*	1123*	1170*	1194*	1219*	1245*	1268*	1298*	1328*	1366*	1401*						
		2016*	1504*	1535*	1571*	1593*	1621*	1649*	1680*	1711*	1754*	1824*	1901*	1976*						
		2646*	2049*	2085*	2132*	2178*	2209*	2270*	2332*	2393*	2403*	2454*	2496*	2575*						
TST1	001352	1048	1059																	
TST10	002206	1246	1268																	
TST11	002266	1269	1298																	
TST12	002354	1329	1328																	
TST13	002452	1367	1366																	
TST14	002560	1402	1401																	
TST15	003044	1465	1465																	
TST16	003260	1466	1504																	
TST17	003416	1505	1535																	
TST18	001464	1060	1086																	
TST19	003602	1526	1571																	
TST20	003652	1572	1593																	
TST21	003760	1594	1621																	
TST22	004074	1622	1649																	
TST23	004204	1650	1680																	
TST24	004326	1681	1711																	
TST25	004462	1712	1754																	
TST26	005076	1755	1824																	
TST27	001572	1087	1123																	
TST28	005510	1825	1901																	
TST29	006104	1902	1976																	
TST30	006316	1977	2016																	
TST31	006442	2017	2049																	
TST32	006646	2050	2085																	
TST33	007104	2086	2132																	
TST34	007306	2133	2178																	
TST35	007436	2179	2209																	
TST36	001722	1124	1170																	
TST37	007726	2210	2270																	
TST4	010224	2271	2332																	
TST40	010520	2333	2393																	
TST41	010520	2333	2393																	
TST42	010520	2333	2393																	
TST43	010730	2410	2454																	
TST44	011124	2455	2496																	
TST45	011472	2497	2575																	
TST46	011770	2576	2646	3367																
TST47	002002	1171	1194																	
TST50	*****	2647																		
TST6	002062	1195	1219																	
TST7	002134	1220	1245																	
TSYNC	001144	955*	1763*	1766	1768	1774	1778	1784	1785	1805*	1806	1808*	1831*	1840						
		1841	1847	1852	1861	1863	1872*	1873	1877*	1903*	1913	1914	1915	1923						
		1925	1932	1934	1959*	1963*	1989*	1993*	1998	2019*	2020	2059*	2060	2137*						
		2139	2142*	2222	2223	2255*	2256	2285	2317*	2318	2348	2380*	2381	2415*						
		2416	2878	2901*	2959*	2978	3023*	3027*	3032	3130	3131	3199	3234	3251						

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 85
 DZDPAB.SRC CROSS REFERENCE TABLE -- USER SYMBOLS

.CLOCK	013342	998	2925#		
.CNVRT	016502	1008	3502#		
.CONVR	016476	1006	3500#		
.DELAY	014300	992	3060#		
.EOP	015750	2186	2647	2650	3374#
.ERRTA	017246	3437	3606#		
.HLT	016132	882	3418#		
.PTAIL	013744	880	3000#	3004	
.RESOS	016736	1004	3561#		
.RXCLK	013402	990	2927	2932#	
.SAVOS	016676	1002	3547#		
.SCOPE	015640	986	3346#		
.SCOPI	016052	998	3397#		
.TRPSR	000056	884	897#		
.TRPTA	001202	902	984#		
.TYPE	016070	1000	3404#		
.VALID	012670	996	2829#		

CMPB	1693	1722	1906	1873	1934	3335									
COM	1224	1250	1272	3292											
DEC	1408	1421	2850	2881	2887	2939	2944	2946	2984	2992	3055	3132	3200	3252	3293
DECB	3522	3536													
EMT	3526	3531													
HALT	599														
	623	625	627	629	631	633	635	637	639	641	643	645	647	649	651
	653	655	657	659	661	663	665	667	669	671	673	675	677	679	681
	683	685	687	689	691	693	695	697	699	701	703	705	707	709	711
	713	715	717	719	721	723	725	727	729	731	733	735	737	739	741
	743	745	747	749	751	753	755	757	759	761	763	765	767	769	771
	773	775	777	779	781	783	785	787	789	791	793	795	797	799	801
	803	805	807	809	811	813	815	817	819	821	823	825	827	829	831
	833	835	837	839	841	843	845	847	849	851	853	855	857	859	861
	863	865	867	869	871	873	875	877	2708	2713	3001	3048	3475	859	861
INC	1726	1907	2255	2317	2380	2660	2665	2736	2738	2742	3053	3147	3165	3261	3286
	3355	3376	3383	3477											
INCB	1697	1805	1872	1990	2119	2142	3024								
JMP	904	908	912	1047	1716	1829	2164	2185	2186	2650	3007	3074	3100	3189	3341
	3389	3390	3486												
JSR	1040	1041	1043	1044	1090	1092	1094	1096	1098	1100	1102	1104	1127	1130	1133
	1135	1137	1139	1141	1143	1145	2021	2028	2037	2154	2157	2159	2160	2211	2272
	2334	2395	2420	2428	2436	2695	2763	2885	2889	2900	2988	2994	3214		
MOV	897	899	903	1036	1037	1048	1059	1060	1061	1064	1065	1066	1067	1086	1087
	1089	1123	1124	1126	1170	1171	1173	1174	1175	1177	1194	1195	1196	1197	1201
	1202	1219	1220	1221	1225	1245	1246	1247	1248	1251	1268	1269	1270	1273	1298
	1299	1301	1328	1329	1331	1366	1367	1369	1370	1371	1401	1402	1403	1404	1405
	1406	1407	1413	1420	1423	1424	1465	1466	1467	1468	1469	1504	1505	1506	1507
	1508	1535	1536	1537	1538	1545	1552	1559	1571	1572	1574	1578	1593	1594	1595
	1597	1598	1621	1622	1623	1625	1626	1649	1650	1651	1652	1654	1680	1681	1682
	1711	1712	1715	1754	1755	1759	1760	1761	1763	1764	1775	1778	1782	1783	1784
	1824	1825	1828	1830	1831	1834	1835	1836	1841	1847	1850	1852	1860	1861	1862
	1901	1902	1903	1906	1909	1912	1915	1921	1923	1924	1932	1933	1958	1959	1963
	1964	1976	1977	1978	1983	1984	1987	1989	1992	1993	1994	1995	1996	1997	2016
	2017	2019	2025	2032	2049	2050	2051	2054	2055	2059	2061	2066	2068	2069	2070
	2085	2086	2092	2093	2099	2102	2105	2108	2132	2133	2135	2136	2138	2149	2150
	2151	2163	2165	2178	2179	2184	2191	2192	2193	2194	2209	2210	2212	2215	2216
	2217	2218	2246	2270	2271	2274	2277	2279	2280	2281	2282	2308	2332	2333	2335
	2338	2340	2341	2342	2371	2393	2394	2409	2410	2414	2415	2417	2418	2419	2425
	2432	2454	2455	2456	2461	2462	2464	2496	2497	2498	2501	2502	2505	2533	2536
	2537	2540	2575	2576	2577	2579	2580	2608	2611	2612	2613	2618	2619	2620	2646
	2647	2654	2655	2656	2676	2678	2706	2707	2711	2712	2717	2721	2729	2730	2732
	2733	2735	2737	2739	2741	2743	2744	2748	2750	2752	2754	2756	2759	2769	2775
	2776	2777	2793	2794	2797	2803	2809	2815	2829	2830	2834	2844	2845	2849	2852
	2853	2866	2867	2868	2871	2872	2873	2874	2875	2878	2880	2886	2937	2942	2955
	2959	2960	2961	2972	2973	2974	2975	2976	2978	2981	2989	3000	3003	3004	3011
	3013	3021	3023	3026	3027	3028	3029	3030	3031	3060	3062	3116	3130	3134	3146
	3158	3159	3162	3171	3172	3188	3199	3202	3215	3217	3218	3219	3222	3223	3227
	3229	3229	3230	3231	3240	3241	3251	3254	3260	3272	3273	3274	3275	3288	3333
	3334	3351	3360	3361	3362	3384	3385	3386	3399	3404	3427	3430	3432	3438	3439
	3440	3474	3484	3485	3502	3503	3505	3508	3509	3511	3512	3524	3538	3547	3551
	3552	3553	3554	3555	3556	3561	3562	3563	3564	3565	3566				
MOVE	1068	1368	1599	1628	1656	1690	1691	1692	1699	1718	1719	1720	1721	1766	1767
	1768	1774	1796	1833	1840	1875	1913	1914	1998	2020	2060	2089	2139	2187	2222
	2223	2254	2285	2314	2345	2348	2377	2416	2715	2876	2898	2901	2977	3032	3131
	3412	3422	3459	3506	3507	3510	3515	3525	3530						

NOP	1541	1548	1555	2166	2726	2764	2765	2766	2891						
RESET	1034	1223	1249	1271	2148	2761	3002								
ROL	3290	3299													
ROR	3517	3519	3521												
RTI	2840	2859	2948	3117	3123	3126	3135	3151	3184	3205	3224	3243	3255	3264	3266
	3308	3312	3363	3400	3414	3487	3539	3557	3567						
RTS	2770	2782	2821	2893	2902	2918	2966	2995	3058						
SUB	898	1517	2731	3065	3072	3431									
TRAP	985	987	989	991	993	995	997	999	1001	1003	1005	1007			
TST	1045	1226	1252	1278	2180	2725	2734	2740	2751	2753	2755	3237	3300	3443	3462
	3466	3471													
TSTB	1039	1274	1575	1579	1601	1630	1658	1687	1730	1769	1776	1789	1842	1848	1853
	1929	1937	2097	2100	2103	2106	2152	2727	2746	3143	3155	3196	3211	3330	3349
	3381	3387	3408	3410	3420	3441	3528								
WAIT	2003	2073													
.ASCIZ	3568	3571	3575	3577	3579	3581	3582	3585	3588	3591	3595	3599			
.BLKW	1016	1017	1018	1019	1020	1021	1025	1026	1027	1028					
.BYTE	972	973	974	975	3392	3489	3492	3495	3635	3637	3641	3643	3645		
.ENABL	530														
.END	3652														
.ENDC	1061	1088	1125	1172	1196	1221	1247	1270	1300	1330	1368	1403	1467	1506	1537
	1573	1595	1623	1651	1682	1713	1756	1826	1903	1978	2018	2051	2087	2134	2180
	2211	2272	2334	2395	2411	2456	2498	2577	2648						
.EQUIV	599														
.EVEN	3605														
.IF	1060	1087	1124	1171	1195	1220	1246	1269	1299	1329	1367	1402	1466	1505	1536
	1572	1594	1622	1650	1681	1712	1755	1825	1902	1977	2017	2050	2086	2133	2179
	2210	2271	2333	2394	2410	2455	2497	2576	2647						
.IFF	1060	1061	1087	1088	1124	1125	1171	1172	1195	1196	1220	1221	1246	1247	1269
	1270	1299	1300	1329	1330	1367	1368	1402	1403	1466	1467	1505	1506	1536	1537
	1572	1573	1594	1595	1622	1623	1650	1651	1681	1682	1712	1713	1755	1756	1825
	1826	1902	1903	1977	1978	2017	2018	2050	2051	2086	2087	2133	2134	2179	2180
	2210	2211	2271	2272	2333	2334	2394	2395	2410	2411	2455	2456	2497	2498	2576
	2577	2647													
.IIF	1060	1061	1087	1088	1124	1125	1171	1172	1195	1196	1220	1221	1246	1247	1269
	1270	1299	1300	1329	1330	1367	1368	1402	1403	1466	1467	1505	1506	1536	1537
	1572	1573	1594	1595	1622	1623	1650	1651	1681	1682	1712	1713	1755	1756	1825
	1826	1902	1903	1977	1978	2017	2018	2050	2051	2086	2087	2133	2134	2179	2180
	2210	2211	2271	2272	2333	2334	2394	2395	2410	2411	2455	2456	2497	2498	2576
	2577	2647	2648												
.IRP	985	987	989	991	993	995	997	999	1001	1003	1005	1007	1053	1060	1080
	1087	1117	1124	1164	1171	1188	1195	1213	1220	1239	1246	1262	1269	1292	1299
	1322	1329	1360	1367	1395	1402	1459	1466	1498	1505	1529	1536	1565	1572	1587
	1594	1615	1622	1643	1650	1674	1681	1705	1712	1748	1755	1818	1825	1895	1902
	1970	1977	2010	2017	2043	2050	2079	2086	2126	2133	2172	2179	2203	2210	2264
	2271	2326	2333	2387	2394	2403	2410	2448	2455	2490	2497	2569	2576	2640	2647
	3367														
.LIST	522	530	531	552	879	987	989	991	993	995	997	999	1001	1003	1005
	1007	1009	1061	1088	1125	1172	1196	1221	1247	1270	1300	1330	1368	1403	1467
	1506	1537	1573	1595	1623	1651	1683	1713	1756	1826	1903	1978	2018	2051	2087
	2134	2180	2211	2272	2334	2395	2411	2456	2498	2577	2648	3367			
.MACR	531														
.MACRO	1	522	523												
.NLIST	522	530	531	552	879	987	989	991	993	995	997	999	1001	1003	1005
	1007	1009	1061	1088	1125	1172	1196	1221	1247	1270	1300	1330	1368	1403	1467
	1506	1537	1573	1595	1623	1651	1683	1713	1756	1826	1903	1978	2018	2051	2087

J07

DZDP MACY11 27(732) 20-APR-76 14:14 PAGE 92
DZDPAB.SRC CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

.PAGE	2134	2190	2211	2272	2334	2395	2411	2456	2498	2577	2648	3367
.REPT	571	620	969	1075	1394	2039	2146	2488	2628	3343		
.TITLE	622											
	552											

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

*DZDPAB.DZDPAB.SEQ/SOL/CRF=DZDPAB.DOC,DZDPAB.MCL,DZDPAB.SRC
RUN-TIME: 15 29 4 SECONDS
RUN-TIME RATIO: 223/50=4.4
CORE USED: 15K (29 PAGES)

