DUP11

BASIC DUP11 OFFLINE SDLC

MD-11-DZDPB-B

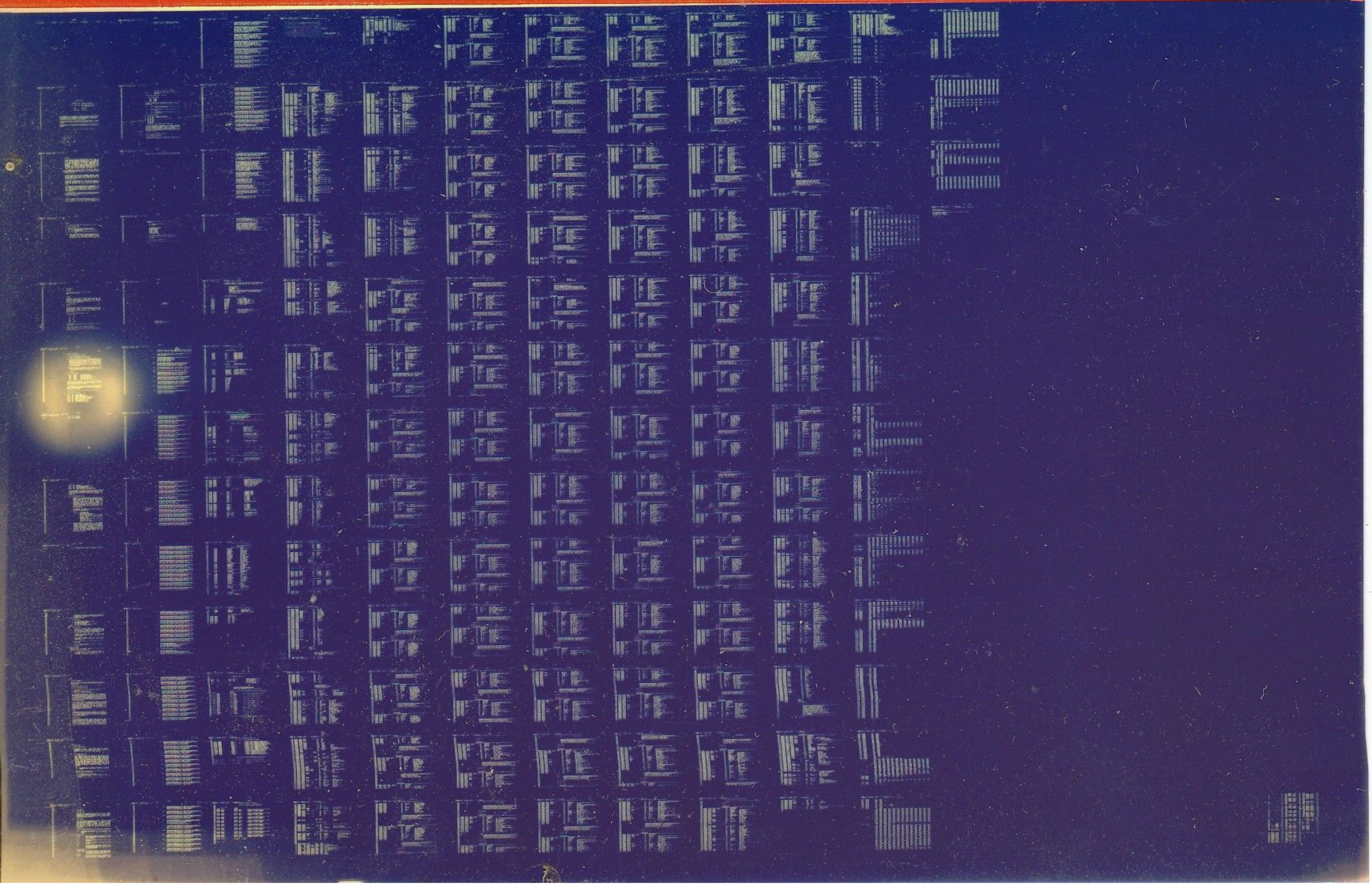
EP-DZDPB-B-DL-A

FICHE 1 OF 1

AUG 1977

digital

MADE IN USA



E3F1DZDM4BSEQ 00010000 770712 PDP10 411 1HDR1DZDP88SEQ 00010000 770712

.REM @

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDPB-B-D

PRODUCT NAME: BASIC DUP11 AND OFFLINE SDLC TRANSMITTER TESTS

DATE: JUNE 1977

MAINTAINER: DIAGNOSTICS

1.0 ABSTRACT

THE FUNCTION OF THE DUP11 DIAGNOSTICS IS TO VERIFY THAT THE OPTION OPERATES ACCORDING TO SPECIFICATIONS. THE DIAGNOSTICS VERFIY THAT THERE ARE NO MALFUNCTIONS AND THAT ALL OPERATIONS OF THE DUP11 ARE CORRECT IN ITS ENVIROMENT. PARAMETERS MAY BE SET TO ALERT DIAGNOSTICS AS TO THE DUP11 CONFIGURATION BY ANSWERING THE PARAMETER DIALOG (LOAD ADDRESS=200, START ADDRESS=1). ALL QUESTIONS SHOULD BE ANSWERED AND THEN EACH DIAGNOSTIC WILL "OVERLAY" THESE PARAMETERS WHICH ARE STORED IN THE "STATUS TABLE" (SEE SECTION 8.4). THE ALTERNATIVE TO THE PARAMETER DIALOG IS DEFAULT PARAMETERS (SEE SECTION 8.5).

THE DIAGNOSTICS WILL RUN UP TO EIGHT CONSECUTIVELY ADDRESSED AND CONSECUTIVELY VECTORED DUP11'S IN A CHAIN MODE, I.E., RUNNING THE DIAGNOSTIC COMPLETELY FOR ONE DEVICE BEFORE STARTING THE NEXT.

DZDPB DOES READ/WRITE CHECKING, BUS ADDRESS CHECKING, DEVICE AND BUS RESET TESTS ON THE CONTROL AND STATUS REGISTERS. TESTS ARE MADE TO PROVE THERE IS NO INTERACTION WITHIN AND BETWEEN REGISTERS. THE REGISTERS ARE CHECKED BOTH A BIT AT A TIME AND ALL AT ONCE.

IN ADDITION, THE TRANSMITTER SDLC FUNCTIONS ARE CHECKED IN MAINTENANCE INTERNAL MODE, I.E., CLOCKING OF THE DEVICE IS DONE BY THE PROGRAM.THE DEVICE IS SET UP, A SPECIFIC NUMBER OF HALF-CLOCKS ARE DONE, AND A TEST IS MADE FOR A SIGNIFICANT EVENT.

IN CHECKING DATA, THE SOFTWARE EMULATES THE HARDWARE AND USES THE PROCESSOR CARRY BIT AFTER A ROTATE TO DETERMINE WHAT THE OUTPUT SHOULD BE AT THE TRANSMITTER BIT WINDOW A BIT AT A TIME. THE PROGRAM THEN COMPARES THE EMULATED SOFTWARE BIT TC THE HARDWARE BIT OUTPUT. THE PROCESS IS REPEATED UNTIL ALL DATA IS CHECKED.

THE TRANSMITTER BCC IS CHECKED USING THE CRC.CCITT POLYNOMIAL IN THE SAME WAY AS DATA, WITH ONE EXCEPTION--THE BCC IS CALCULATED FIRST BY THE PROGRAM AND THEN COMPARED TO THE OUTPUT OF THE TRANSMITER.

CURRENTLY THERE ARE THREE OFF-LINE DIAGNOSTICS THAT ARE TO BE RUN IN SEQUENCE TO ENSURE THAT IF AN ERROR SHOULD OCCUR IT WILL BE DETECTED AT AN EARLY STAGE AND ESTABLISH THAT DIAGNOSIS OF THE ERROR WILL BE IMMEDIATE TO DISCOVERING THE

PROBLEM.

NOTE: ADDITIONAL DIAGNOSTICS MAY BE ADDED IN THE FUTURE.

THE THREE DIAGNOSTICS ARE:

1. DZDPB [REV] BASIC AND OFFLINE TRANSMITTER TESTS
2. DZDPC [REV] OFFLINE RECEIVER AND MODEM CONTROL AND
                INTERRUPT TESTS
3. DZDPD [REV] OFFLINE SDLC AND DECMODE DATA AND FUNCTION
                TESTS

NOTE: THERE IS A FOURTH MAINDEC, TAPE DZDPE [REV] WHICH IS A
       QUICK-VERIFY TAPE THAT REQUIRES ANSWERING A DIALOG.
       ITS FUNCTION IS TO ENABLE THE OPERATOR TO QUICKLY
       DETERMINE IF THERE IS A PROBLEM WITH THE DEVICE. SEE
       THE DOCUMENTATION IN THAT LISTING FOR MORE INFORMATION.

2.0 REQUIREMENTS

2.1 EQUIPMENT

ANY PDP11 FAMILY CPU (WITH MINIMUM 8K MEMORY)
ASR 33 (OR EQUILIVALENT)
DUP11

2.2 STORAGE

PROGRAM WILL USE ALL 8K OF MEMORY EXCEPT WHERE ABS AND BOOTSTRAP LOADER RESIDE. LOCATION 1500 THRU 1560 ARE ESPECIALLY TO BE NOTED AND LEFT UNTOUCHED BY THE OPERATOR AFTER THE DUP11 PARAMETER DIALOG HAS BEEN EXECUTED OR AFTER THE DEFAULT SETUP HAS BEEN DONE.

3.0 LOADING PROCEDURE

3.1 METHOD

ALL PROGRAMS ARE IN ABSOLUTE FORMAT AND ARE LOADED USING THE ABSOLUTE LOADER. NOTE: IF THE DIAGNOSTICS ARE ON A MEDIA SUCH AS DISK, MAGTAPE, DECTAPE, OR CASSETTE FOLLOW INSTRUCTIONS FOR THE MONITOR WHICH HAS BEEN PROVIDED ON THAT SPECIFIC MEDIA.

ABSOLUTE LOADER STARTING ADDRESS = #+500

MEMORY	SIZE (#)=
8K	37
12k	57
16k	77
20K	117
24K	137
28K	157

3.1.1 PLACE ADDRESS OF ABS LOADER INTO SWITCH REGISTER. (ALSO PLACE 'HALT' SW UP)

3.1.2 DEPRESS 'LOAD ADDRESS' KEY ON CONSOLE AND RELEASE.

3.1.3 DEPRESS 'START KEY' ON CONSOLE AND RELEASE (PROGRAM SHOULD NOW BE LOADING INTO CPU)

4.0 STARTING PROCEEDURE

A. SET SWITCH REGISTER TO 000200

B. DEPRESS 'LOAD ADDRESS' KEY AND RELEASE

C. SET SWR TO ZERO FOR DEFAULT PARAMETERS ESTABLISHED IN THE TAPE (SEE SECTION 8.5.3 FOR FULL EXPLANATION OF DEFAULT PARAMETERS) OR LEAVE SWR BIT 7=1 TO USE EXISTING PARAMETERS PREVIOUSLY SET UP BY THE DUP11 PARAMETER DIALOG OR A PREVIOUSLY RUN DUP11 DIAGNOSTIC. SET SWR=1 TO GO THROUGH THE PARAMETER DIALOG. (IT IS NOT NECESSARY TO INPUT NEW PARAMETERS FOR EACH TAPE.) (SECTION 7.2, 8.4 AND 8.5 MAY BE HELPFUL)

D. DEPRESS 'START KEY' AND RELEASE. THE PROGRAM WILL TYPE MAINDEC NAME AND PROGRAM NAME (IF THIS WAS THE FIRST START UP OF THE PROGRAM) AND ALSO THE FOLLOWING:

'EXAMPLE'

'MAP OF DUP11 STATUS'

1500	160050	CSR OF FIRST DUP11
1502	000300	VECTOR OF FIRST DUP11
1504	140026	STATUS AND SYNC FOR FIRST DUP11
1506	160060	CSR OF SECOND DUP11
1510	000310	VECTOR OF SECOND DUP11
1512	140026	STATUS AND SYNC FOR SECOND DUP11

THE ABOVE IS ONLY AN EXAMPLE! THIS WOULD INDICATE THE STATUS TABLE STARTING AT ADDRESS 1500 IN THE PROGRAM. THE STATUS TABLE MUST BE VERIFIED BY THE USER. FOR INFORMATION ON THE STATUS TABLE SEE SECTION 8.4 FOR HELP.

IT IS POSSIBLE FOR THE OPERATOR TO MANUALLY CHANGE (TOGGLE IN) THE INFORMATION IN THE MAP TO SUIT A SPECIFIC CONFIGURATION OF DEVICES, BUT THE RESPONSIBILITY FOR VERIFYING THAT INFORMATION RESTS WITH THE OPERATOR.

THE PROGRAM WILL TYPE 'R' AND PROCEED TO RUN THE DIAGNOSTIC

4.1 CONTROL SWITCH SETTINGS

SW 15	SET:	HALT ON ERROR
SW 14	SET:	LOOP ON CURRENT TEST
SW 13	SET:	INHIBIT ERROR PRINT OUT
SW 12	SET:	INHIBIT TYPE OUT/BELL ON ERROR.
SW 11	SET:	INHIBIT ITERATIONS. (QUICk PASS.
SW 10	SET:	ESCAPE TO NEXT TEST ON ERROR
SW 09	SET:	LOOP WITH CURRENT DATA
SW 08	SET:	CATCH ERROR AND LOOP ON IT
SW 07	SET:	USE PREVIOUS STATUS TABLE.

```
   222                                          SW 06   SET:  RESERVED
   223                                          SW 05   SET:  RESERVED
```

SW 04 SET: RESERVED
SW 03 SET: SELECT DUP11'S DESIRED ACTIVE
SW 02 SET: LOCK ON SELECTED TEST
SW 01 SET: RESTART PROGRAM AT SELECTED TEST
SW 00 SET: ENTER PARAMETERS USING MANUAL DIALOG

SWITCHES 8 THROUGH 15 ARE DYNAMIC AND SHOULD BE USED AS NEEDED IN THE DIAGNOSTIC. SWITCHES 0 THROUGH 3 ARE STATIC (ONLY ARE OPERABLE WHEN THE MONITOR PORTION OF THE TAPE IS RUNNING) AND SHOULD BE SET UP PRIOR TO STARTING OR RESTARTING THE DIAGNOSTIC.

4.1.2 SWITCH REGISTER RESTRICTIONS

SW 03 RESELECT DUP11'S DESIRED ACTIVE. PLEASE NOTE THAT A MESSAGE IS TYPED OUT FOR SETTING THE SWITCH REGISTER EQUAL TO DUP11'S ACTIVE. THIS MEANS IF THE SYSTEM HAS THREE DUP11S BITS 00, 01, 02 WILL BE SET IN LOC 'DUPACTV' FROM THE SWITCH' REGISTER. USING THIS SWITCH(SW03) ALTERS THAT LOCATION. THEREFORE, IF THREE DUP11S ARE IN THE SYSTEM ***DO NOT*** SET SWITCHES GREATER THAN SW 02 IN THE UP POSITION. THIS WOULD BE A FATAL ERROR. DO NOT SELECT MORE ACTIVE DUP11S THAN HAS BEEN GIVEN INFORMATION ABOUT IN THE PARAMETER PROGRAM.

AS EXPLAINED IN SECTION 1.0, DEVICES SHOULD BE CONSECUTIVELY ADDRESSED, AND CAN BE SELECTED OR DESELECTED USING THIS SWITCH.

METHOD:
A. LOAD ADDRESS 200
B. START WITH SW 03=1
C. PROGRAM WILL TYPE MESSAGE
D. SET THE BINARY NUMBER OF DUP11S DESIRED ACTIVE. EXAMPLE: 1=1 DUP11; 3=2 DUP11; 7=3 DUP11; 17=4 DUP11 37=5 DUP11 ETC. PRESS CONTINUE.
E. NUMBER (IF VALID) WILL BE IN DATA LIGHTS (EXCLUDING 11/05)
F. SET WITH ANY OTHER SWITCH SETTINGS DESIRED. PRESS CONTINUE.

SW 01 RESTART PROGRAM AT SELECTED TEST. IT IS STRONGLY SUGGESTED THAT AT LEAST ONE PASS HAS BEEN MADE BEFORE TRYING TO SELECT A TEST THAT IS NOT IN THE ORDER OF SEQUENCE. THE REASON FOR THIS IS THAT THE PROGRAM HAS TO CLEAR AREAS AND SET UP PARAMETERS IN THE MONITOR PORTION OF THE PROGRAM. IT IS POSSIBLE TO LD200, AND RAISE SW01, THEN START, PROVIDED PARAMETERS HAVE BEEN PREVIOUSLY SET UP AS DESCRIBED IN SECTION 4.0. ALSO, WHEN A TEST IS SELECTED, ALWAYS START AT THE VERY BEGINNING OF THAT TEST.

```
280
281                                     ;W 09   LOOP ON CURRENT DATA.  THIS SWITCH WILL ONLY  WORK  IF
                                                CALL 'SCOPI' IS IN THAT TEST.  THE REASON IS THAT MOST
```

TESTS DEAL WITH BLOCKS OF DIFFERENT DATA TO BE SENT OR RECEIVED ALL AT ONCE, THUS KNOWN AS BLOCK DATA--ONE PATTERN CAN'T BE SINGLED OUT. (SEE SECTION 4.1.3.B.1)

4.1.3 SWITCH REGISTER PRIORITIES

A) ERROR SWITCHES

1. SW 12 DELETE PRINT OUT/BELL ON ERROR.
2. SW 13 DELETE ERROR PRINTOUT.
3. SW 15 HALT ON THE ERROR.
4. SW 08 GOTO BEGINNING OF THE TEST(ON ERROR).
5. SW 10 GOTO NEXT TEST(ON ERROR).

B) SCOPE SWITCHES

1. SW 09 - (IF ENABLED BY 'SCOP1') ON AN ERROR.IF AN ASTERISK '*' IS PRINTED IN FRONT OF THE TEST NUMBER (EX. *TEST NO. 10), SW09 IS INCORPORATED IN THAT TEST AND THEREFORE SW09 IS USUALLY THE BEST SWITCH FOR THE SCOPE LOOP (SW14=0, SW10=0, SW09=1, SW08=0).

   IF SW09 IS NOT ENABELED AND THERE IS A *HARD* ERROR (CONSTANT ERROR) SW08 IS BEST. (SW14=0, SW10=0, SW09=0, SW08=1).

   FOR INTERMITTENT ERRORS, SW14=1 WILL LOOP ON TEST REGARDLESS OF ERROR OR NO ERROR. (SW14=1, SW10=0, SW09=0, SW08=1,0)

2. SW 14 - LOOP ON TEST. WILL LOOP ON TEST UNTIL SWITCH IS LOWERED.

3. SW 11 - INHIBIT ITERATIONS (QUICK PASS). ALLOWS ONLY ONE PASS THROUGH A TEST.

4.2 STARTING ADDRESS

STARTING ADDRESS IS AT 000200. THERE ARE NO OTHER STARTING ADDRESSES FOR THE DUP11 DIAGNOSTICS.

NOTE: IF ADDRESS 000042 IS NON-ZERO THE PROGRAM ASSUMES IT IS UNDER ACT11 OR XXDP CONTROL AND WILL ACT ACCORDINGLY. AFTER *ALL* AVAILABLE DUP11'S ARE TESTED THE PROGRAM WILL RETURN TO 'XXDP' OR 'ACT-11'.

5.0 OPERATING PROCEDURE

WHEN THE PROGRAM IS INITIALLY STARTED MESSAGES AS DESCRIBED IN SECTION FOUR WILL BE PRINTED AND PROGRAM WILL BEGIN RUNNING THE DIAGNOSTIC.

5.1 PROGRAM AND/OR OPERATOR ACTION

THE TYPICAL APPROACH SHOULD BE

1. HALT ON ERROR (VIA SW 15=1) WHENEVER AN ERROR OCCURS.
2. CLEAR SW 15.
3. SET SW 14: (LOOP ON THIS TEST)
4. SET SW 13: (INHIBIT ERROR PRINT OUT)

THE TEST NUMBER AND PC WILL BE TYPED OUT AND POSSIBILY AN ERROR MESSAGE (THIS DEPENDS ON THE TEST), TO GIVE THE OPERATOR AN IDEA AS TO THE SOURCE OF THE PROBLEM. IF IT IS NECESSARY TO KNOW MORE INFORMATION CONCERNING THE ERROR REPORT, LOOK IN THE LISTING FOR THAT TEST NUMBER WHICH WAS TYPED OUT AND THEN NOTE THE PC OF THE ERROR REPORT. IN THIS WAY THE EXACT FUNCTIONING OF THE TEST CAN BE INTERPRETED SINCE THE ERROR PC IS THE HLT+2 LOCATION.

IN SOME TESTS, THERE IS A SUBROUTINE CALL THROUGH A REGISTER (E.G., JSR R1,FLAG). THE SUBROUTINE DOES THE DATA CHECKING FOR THE TEST AND WILL REPORT AN ERROR IF ONE OCCURS. THIS MEANS THAT THE FAILING TEST COULD BE IN ONE PART OF THE LISTING WHILE THE SUBROUTINE THAT FOUND THE ERROR IS IN ANOTHER PART. TO DETERMINE THE PC OF THE FAILING TEST, CHECK THE REGISTER USED BY THE SUBROUTINE. IT WILL CONTAIN THE RETURN ADDRESS OF THE FAILING TEST.

6.0 ERRORS

AS DESCRIBED PREVIOUSLY THERE WILL ALWAYS BE A TEST NUMBER AND PC TYPED OUT AT THE TIME OF AN ERROR (PROVIDING SW 13=0 AND SW 12=0). IN MOST CASES ADDITIONAL INFORMATION WILL BE SUPPLIED TO THE ERROR MESSAGE WHICH IS TO GIVE THE OPERATOR AN INDICATION OF THE ERROR.

6.1 ERROR RECOVERY

IF FOR SOME REASON THE DUP11 SHOULD 'HANG THE BUS' (GAIN CONTROL OF BUS SO THAT CONSOLE MANUAL FUNCTIONS ARE INHIBITED) AN INIT OR POWER DOWN/UP IS NECESSARY FOR OPERATOR TO REGAIN CONTROL OF CPU. IF THIS SHOULD HAPPEN LOOK IN LOCATION 'TSTNO' FOR THE NUMBER OF THE TEST THAT WAS RUNNING AT THE TIME OF THE CATASTROPHIC ERROR. THIS GIVES THE OPERATOR SOME IDEA AS TO WHAT THE DUP11 WAS DOING AT THE TIME OF THE ERROR.

7.0 RESTRICTIONS

7.1 STARTING RESTRICTIONS

SEE SECTION 4 (PLEASE). STATUS TABLE SHOULD BE VERIFIED REGARDLESS OF HOW THE PROGRAM WAS STARTED. ALSO, IT IS IMPORTANT TO USE THE LISTING ALONG WITH THE INFORMATION PRINTED ON THE TTY TO COMPLETLY ISOLATE PROBLEMS.

7.2 OPERATING RESTRICTIONS

DUP11 "PARAMETER DIALOG" MUST BE RUN ONLY ONCE PRIOR TO THE FIRST RUNNING OF ANY DUP11 DIAGNOSTIC IF "DEFAULT PARAMETERS" ARE NOT USED. IF ONLY DUP11 DIAGNOSTICS WERE LOADED AFTER DUP11 PARAMETER SETUP, AND IF CORE MEMORY HAS NOT BEEN CHANGED, I.E. USE OF DIAGNOSTICS OTHER THAN DUP11 DIAGNOSTICS, AND IF THERE WERE NO DUP11 CONFIGURATION CHANGES, THE DUP11 PARAMETER SETUP NEED NEVER BE RUN AGAIN. HOWEVER, IF ANY OF THE ABOVE HAVE BEEN VIOLATED THE DUP11 PARAMETER SETUP MUST BE RUN AGAIN BEFORE RUNNING THE DIAGNOSTICS. UNDER NORMAL OPERATING CONDITIONS IT SHOULD NOT BE NECESSARY TO INPUT NEW PARAMETERS TO SUBSEQUENT DIAGNOSTICS, UNLESS A CHANGE IS REQUIRED.

NOTE: AN ALTERNATIVE TO THE ABOVE IS ATTEMPTING THE DEFALLT PARAMETERS WHEN THE PROGRAM IS INITIALLY STARTED WITH SWR=0.

7.3 HARDWARE CONFIGURATION RESTRICTIONS FOR THE PURPOSE OF RUNNING MULTIPLE DUP11'S IN CHAIN MODE.

1. CSR ADDRESSES MUST BE CONSECUTIVE.
2. VECTORS ARE CONSECUTIVE IF PARAMETER PROGRAM IS USED.
3. ALL JUMPERS ARE ASSUMED TO BE AS SETUP IN PARAMETER DIALOG.
4. PRIORITY LEVEL MUST BE THE SAME FOR ALL DEVICES.

8.0 MISCELLANEOUS

8.1 EXECUTION TIME

ALL DUP11 DEVICE DIAGNOSTICS WILL GIVE AN 'END PASS' MESSAGE (PROVIDING NO ERRORS AND SW12=0) WITHIN 4 MINS. THIS IS ASSUMING SW11=1 (DELETE ITERATIONS) IS SET TO GIVE THE FASTEST POSSIBLE EXECUTION. THE ACTUAL EXECUTION TIME DEPENDS GREATLY ON THE PDP11 CPU CONFIGURATION.

8.2 PASS COMPLETE

NOTE: *EVERY* TIME THE PROGRAM IS STARTED, THE TESTS WILL RUN AS IF SW11 (DELETE ITERATIONS) WAS UP (=1). THIS IS TO VERIFY NO *HARD* ERRORS AS SOON AS POSSIBLE. THEREFORE THE FIRST PASS--EACH TIME PROGRAM IS STARTED--WILL BE A 'QUICK PASS' UNTIL ALL DUP11'S IN SYSTEM ARE TESTED. WHEN THE DIAGNOSTIC HAS COMPLETED A PASS WITH THE NORMAL ITERATION COUNT (ICOUNT=50), THE FOLLOWING IS AN EXAMPLE OF THE PRINT OUT TO BE EXPECTED.

END PASS DZDPBB CSR:160050 VEC:300 PASSES:000001 ERRORS:000000

NOTE: THE NUMBERS FOR CSR AND VEC ARE NOT NECESSARILY THE VALUES FOR THE DEVICE. THEY ARE ONLY FOR THIS EXAMPLE.

8.3 KEY LOCATIONS

RETURN	CONTAINS THE ADDRESS WHERE PROGRAM WILL RETURN WHEN ITERATION COUNT IS REACHED OR IF LOOP ON TEST IS ASSERTED.
NEXT	CONTAINS THE ADDRESS OF THE NEXT TEST TO BE PERFORMED.
TSTNO	CONTAINS THE NUMBER OF THE TEST NOW BEING PEFORMED.
RUN	THE BIT IN 'RUN' ALWAYS POINTS ONE PAST THE DUP11 CURRENTLY BEING TESTED. EXAMPLE: (RUN) /0000000001000000 MEANS THAT DUP11 NO.05 IS THE DUP11 NOW RUNNING.
DUPCR00-DUPCR07 (1500)-(1560)	THESE LOCATIONS CONTAIN THE INFORMATION NEEDED TO TEST UP TO 8 (DECIMAL) DUP11S SEQUENTIALY. THEY CONTAIN THE CSR, VECTOR AND STATUS CONCERNING THE CONFIGURATION OF EACH DUP11.
DUPACTV	EACH BIT SET IN THIS LOCATION INDICATES THAT THE ASSOCIATED DUP11 WILL BE TESTED IN TURN. EXAMPLE: (DUPACTV) /0000000000011111 MEANS

```
478
479
```

THAT DUP11 NO. 00,01,02,03,04 WILL BE TESTED.
EXAMPLE: (DUPACTV) /0000000000C10001 MEANS

THAT DUP11 NO. 00,04 WILL BE TESTED.

RXCSR CONTAINS THE RECEIVER CSR OF THE CURRENT DUP11 UNDER TEST.

8.4 MORE ON THAT 'STATUS TABLE' (1500-1560)

'MAP OF DUP11 STATUS'

1500	160050
1502	000300
1504	140000

THE ABOVE INFORMATION WILL BE REPEATED FOR EACH OF UP TO 8 DUP11'S IN THE SYSTEM (THESE WILL FOLLOW UNDER THIS TABLE).
EXPLANATION:

1500	160050	THIS IS THE SYSTEM CONTROL REGISTER FOR THE 1ST DUP11 IN THE SYSTEM.
1502	000300	THIS IS VECTOR 'A' FOR THE FIRST DUP11 IN THE SYSTEM.
1504	140026	THIS REPRESENTS SYNC AND SOFTWARE STATUS FOR THE FIRST DUP.

THE BITS ARE AS FOLLOWS:

BIT 15 SET:	OPTIONAL CLEAR JUMPER IN
BIT 14 SET:	TURNAROUND CONNECTOR ON
BIT 13 SET:	
BIT 12 SET:	
BIT 11 SET:	
BIT 10 SET:	
BIT 09 SET:	
BIT 08 SET:	
BIT 07-00	SYNC CHARACTER FOR DECMODE TESTS.

THE ABOVE IS REPEATED FOR EACH DUP11 IN THE SYSTEM. THE TABLE IS FILLED BY DEFAULT PARAMETERS OR BY THE MANUAL PARAMETER INPUT AS DESCRIBED PREVIOUSLY. ALSO, IF DESIRED BY THE USER - THE LOCATIONS MAY BE ALTERED BY HAND (TOGGLED IN) TO SUIT THE SPECIFIC CONFIGURATION, THUS MAKING EACH DEVICE MAP DIFFERENT. IT IS THE RESPONSIBILITY OF THE OPERATOR TO VERIFY THE DATA IN THE MAP.

8.5 METHOD OF DEVELOPING DEFAULT PARAMETERS

8.5.1 DEFAULT PARAMETER ASSUMPTIONS

TOO MUCH HARDWARE WOULD HAVE TO BE ANALYZED TO SIZE THE THE PARAMETERS. THE PROGRAM MUST ASSUME THE VARIATIONS. THE

```
536
537
```

RESULT, IF NOT TO YOJR SPECIFIC CONFIGURATION, MAY BE ALTERED
BY HAND (TCGGLE IN) AS DESIRED. IN THIS WAY 95% OF THE

PARAMETER SETUP WAS DONE BY THE PROGRAM AND 5% BY YOU.
THEREFORE:

1) ALL JUMPERS ARE ASSUMED TO BE IN THE FOLLOWING CONFIGURATION.

	IN	OUT
W1=SECONDARY REC ENABLE	X	
W2=SEC REC DISABLE		X
W3=CLEAR OPTION	X	
W4=SEC TX ENABLE	X	
W5=DSC A CONTROL		X
W6=A+B DS CONTROL	X	
W7=BUS GRANT CONTROL	X	

2) THE H325 TURN AROUND CONNECTOR IS ASSUMED TO BE ON.

3) THE MANUFACTURING OPTION CSR OF 160050 AND VECTOR OF 770 ARE USED.

4) THE BR LEVEL IS ASSUMED TO BE 5.

IN ALL ADJUSTMENTS PLEASE REFER TO SECTION 8.4 FOR GREATER DETAIL.

DOCUMENT
****
****

```
                    1       MAINDEC-11-DZDPB-B ⁄(377)⁄BASIC DUP11 AND OFFLINE SDLC TRANSMITT
                            COPYRIGHT(C) 1975,1977, DIGITAL EQUIPMENT CORP., MAYNARD, MASS.
                            ------------------------------------------------------------

                    1197    ************************** TEST 1 ****************************
                            VERIFY THAT ADDRESSING DEVICE DOES *NOT* CAUSE
                            A TIME-OUT TRAP.

                    1227    ************************** TEST 2 ****************************
                            PRIMARY REGISTER ADDRESSING TEST
                            LOAD EACH PRIMARY REGISTER WITH A
                            DIFFERENT NUMBER AND VERIFY EACH
                            WAS INDIVIDUALLY ADDRESSED

                    1282    ************************** TEST 3 ****************************
                            RECEIVER CONTROL REGISTER RESET TEST.  TEST THAT AFTER
                            RECEIVER CONTROL REGISTER IS WRITTEN AND A BUS RESET IS
                            DONE THAT RECEIVER CONTROL REGISTER IS CLEARED.

                    1307    ************************** TEST 4 ****************************
                            RECEIVER BUFFER REGISTER RESET TEST.  TEST THAT AFTER A BUS

                    1309    RESET IS DONE THAT RECEIVER BUFFER REGISTER IS CLEARED.

                    1330    ************************** TEST 5 ****************************
                            PARAMETER STATUS REGISTER RESET TEST.  TEST THAT AFTER
                            PARAMETER STATUS REGISTER IS WRITTEN AND A BUS RESET IS
                            DONE THAT PARAMETER STATUS REGISTER IS CLEARED.

                    1355    ************************** TEST 6 ****************************
                            TRANSMITTER CONTROL REGISTER RESET TEST.  TEST THAT AFTER
                            TRANSMITTER CONTROL REGISTER IS WRITTEN AND A BUS RESET IS
                            DONE THAT TRANSMITTER CONTROL REGISTER IS CLEARED.

                    1381    ************************** TEST 7 ****************************
                            TRANSMITTER BUFFER REGISTER RESET TEST.  TEST THAT AFTER
                            TRANSMITTER BUFFER REGISTER IS WRITTEN AND A BUS RESET IS
                            DONE THAT TRANSMITTER BUFFER REGISTER IS CLEARED.

                    1406    ************************** TEST 10 ***************************
                            TRANSMITTER CONTROL REGISTER READ⁄WRITE BIT TEST.
                            SET BIT3, VERIFY BIT3 WAS SET.
                            CLEAR BIT3, VERIFY BIT3 WAS CLEARED.

                    1438    ************************** TEST 11 ***************************
                            TRANSMITTER CONTROL REGISTER READ⁄WRITE BIT TEST.
                            SET BIT4, VERIFY BIT4 WAS SET.
                            CLEAR BIT4, VERIFY BIT4 WAS CLEARED.
```

```
                                1470    ************************** TEST 12 ****************************
                                        TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
                                        SET BIT10, VERIFY BIT10 WAS SET.
                                        CLEAR BIT10, VERIFY BIT10 WAS CLEARED.

                                1502    ************************** TEST 13 ****************************
                                        TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
                                        SET BIT11, VERIFY BIT11 WAS SET.
                                        CLEAR BIT11, VERIFY BIT11 WAS CLEARED.

                                1534    ************************** TEST 14 ****************************
                                        TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
                                        SET BIT12, VERIFY BIT12 WAS SET.
                                        CLEAR BIT12, VERIFY BIT12 WAS CLEARED.

                                1566    ************************** TEST 15 ****************************
                                        TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
                                        SET BIT13, VERIFY BIT13 WAS SET.
                                        CLEAR BIT13, VERIFY BIT13 WAS CLEARED.

                                1598    ************************** TEST 16 ****************************
                                        TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
                                        SET BIT0, VERIFY BIT0 WAS SET.
                                        CLEAR BIT0, VERIFY BIT0 WAS CLEARED.

                                1628    ************************** TEST 17 ****************************
                                        TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
                                        SET BIT1, VERIFY BIT1 WAS SET.
                                        CLEAR BIT1, VERIFY BIT1 WAS CLEARED.

                                1658    ************************** TEST 20 ****************************
                                        TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
                                        SET BIT2, VERIFY BIT2 WAS SET.
                                        CLEAR BIT2, VERIFY BIT2 WAS CLEARED.

                                1688    ************************** TEST 21 ****************************
                                        TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
                                        SET BIT3, VERIFY BIT3 WAS SET.
                                        CLEAR BIT3, VERIFY BIT3 WAS CLEARED.

                                1718    ************************** TEST 22 ****************************
                                        TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
                                        SET BIT4, VERIFY BIT4 WAS SET.
                                        CLEAR BIT4, VERIFY BIT4 WAS CLEARED.

                                1748    ************************** TEST 23 ****************************
                                        TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
                                        SET BIT5, VERIFY BIT5 WAS SET.
                                        CLEAR BIT5, VERIFY BIT5 WAS CLEARED.
```

```
1778    ************************** TEST 24 ****************************
        TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
        SET BIT6, VERIFY BIT6 WAS SET.
        CLEAR BIT6, VERIFY BIT6 WAS CLEARED.

1808    ************************** TEST 25 ****************************
        TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
        SET BIT7, VERIFY BIT7 WAS SET.
        CLEAR BIT7, VERIFY BIT7 WAS CLEARED.

1838    ************************** TEST 26 ****************************
        TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
        SET BIT8, VERIFY BIT8 WAS SET.
        CLEAR BIT8, VERIFY BIT8 WAS CLEARED.

1868    ************************** TEST 27 ****************************

1869    TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
        SET BIT9, VERIFY BIT9 WAS SET.
        CLEAR BIT9, VERIFY BIT9 WAS CLEARED.

1898    ************************** TEST 30 ****************************
        TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
        SET BIT10, VERIFY BIT10 WAS SET.
        CLEAR BIT10, VERIFY BIT10 WAS CLEARED.

1928    ************************** TEST 31 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT 1 RESET AND CLEAR TEST
        WRITE BIT 1,AND TEST THAT IT WILL BE CLEARED AFTER A
        DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION

1960    ************************** TEST 32 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT 2 RESET AND CLEAR TEST
        WRITE BIT 2,AND TEST THAT IT WILL BE CLEARED AFTER A
        DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION

1992    ************************** TEST 33 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT 3 RESET AND CLEAR TEST
        WRITE BIT 3,AND TEST THAT IT WILL BE CLEARED AFTER A
        DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION

2024    ************************** TEST 34 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT 4 RESET AND CLEAR TEST
        WRITE BIT 4,AND TEST THAT IT WILL BE CLEARED AFTER A
        DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION

2056    ************************** TEST 35 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT 8 RESET AND CLEAR TEST
        WRITE BIT 8,AND TEST THAT IT WILL BE CLEARED AFTER A
        DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION
```

```
                                2088    ************************** TEST 36 ****************************
                                        TRANSMITTER CONTROL REGISTER READ/WRITE BIT 3 RESET AND CLEAR TE
                                        WRITE BIT 3,AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION

                                2120    ************************** TEST 37 ****************************
                                        TRANSMITTER CONTROL REGISTER READ/WRITE BIT 4 RESET AND CLEAR TE
                                        WRITE BIT 4,AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION

                                2152    ************************** TEST 40 ****************************
                                        TRANSMITTER CONTROL REGISTER READ/WRITE BIT 10 RESET AND CLEAR T
                                        WRITE BIT 10,AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION

                                2184    ************************** TEST 41 ****************************
                                        TRANSMITTER CONTROL REGISTER READ/WRITE BIT 11 RESET AND CLEAR T
                                        WRITE BIT 11,AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION

                                2216    ************************** TEST 42 ****************************
                                        TRANSMITTER CONTROL REGISTER READ/WRITE BIT 12 RESET AND CLEAR T
                                        WRITE BIT 12,AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION

                                2248    ************************** TEST 43 ****************************
                                        TRANSMITTER CONTROL REGISTER READ/WRITE BIT 13 RESET AND CLEAR T
                                        WRITE BIT 13,AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION

                                2280    ************************** TEST 44 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 0 RESET AND CLEAR TES
                                        WRITE BIT 0,AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION

                                2312    ************************** TEST 45 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 1 RESET AND CLEAR TES
                                        WRITE BIT 1,AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION

                                2344    ************************** TEST 46 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 2 RESET AND CLEAR TES
                                        WRITE BIT 2,AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION

                                2376    ************************** TEST 47 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 3 RESET AND CLEAR TES
                                        WRITE BIT 3,AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
```

```
                                2408    ************************** TEST 50 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 4 RESET AND CLEAR TES
                                        WRITE BIT 4 AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION

                                2440    ************************** TEST 51 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 5 RESET AND CLEAR TES
                                        WRITE BIT 5 AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION

                                2472    ************************** TEST 52 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 6 RESET AND CLEAR TES
                                        WRITE BIT 6 AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION

                                2504    ************************** TEST 53 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 7 RESET AND CLEAR TES
                                        WRITE BIT 7 AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION

                                2536    ************************** TEST 54 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 8 RESET AND CLEAR TES
                                        WRITE BIT 8 AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION

                                2568    ************************** TEST 55 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 9 RESET AND CLEAR TES
                                        WRITE BIT 9 AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION

                                2600    ************************** TEST 56 ****************************
                                        TRANSMITTER BUFFER REGISTER READ/WRITE BIT 10 RESET AND CLEAR TE
                                        WRITE BIT 10 AND TEST THAT IT WILL BE CLEARED AFTER A
                                        DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION

                                2632    ************************** TEST 57 ****************************
                                        RECEIVER BUFFER REGISTER TEST
                                        TEST THAT RECEIVER BUFFER REGISTER CANNOT BE WRITTEN.
                                        WRITE RECEIVER BUFFER REGISTER WITH ALL 1'S
                                        AND VERIFY THAT ALL 0'S ARE READ BACK.

                                2658    ************************** TEST 60 ****************************

                                2660    PARAMETER STATUS REGISTER TEST
                                        TEST THAT PARAMETER STATUS REGISTER CANNOT BE WRITTEN
                                        READ THE PARAMETER STATUS REGISTER AND STORE THE DATA;
                                        COMPLEMENT THE DATA AND  WRITE THE PARAMETER STATUS REGISTER
                                        VERIFING THAT THE PARAMETER STATUS REGISTER DID NOT CHANGE.
```

```
                                2688    ********** TEST 61 **********
                                        RECEIVER CONTROL REGISTER BIT 0 READ ONLY DEVICE RESET AND CLEAR
                                        WRITE BIT 0 A ONE AND VERIFY A ZERO IS READ BACK
                                        REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                                2728    ********** TEST 62 **********
                                        RECEIVER CONTROL REGISTER BIT 7 READ ONLY DEVICE RESET AND CLEAR
                                        WRITE BIT 7 A ONE AND VERIFY A ZERO IS READ BACK
                                        REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                                2768    ********** TEST 63 **********
                                        RECEIVER CONTROL REGISTER BIT 9 READ ONLY DEVICE RESET AND CLEAR
                                        WRITE BIT 9 A ONE AND VERIFY A ZERO IS READ BACK
                                        REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                                2808    ********** TEST 64 **********
                                        RECEIVER CONTROL REGISTER BIT 10 READ ONLY DEVICE RESET AND CLEA
                                        WRITE BIT 10 A ONE AND VERIFY A ZERO IS READ BACK
                                        REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                                2848    ********** TEST 65 **********
                                        RECEIVER CONTROL REGISTER BIT 11 READ ONLY DEVICE RESET AND CLEA
                                        WRITE BIT 11 A ONE AND VERIFY A ZERO IS READ BACK
                                        REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                                2888    ********** TEST 66 **********
                                        RECEIVER CONTROL REGISTER BIT 12 READ ONLY DEVICE RESET AND CLEA
                                        WRITE BIT 12 A ONE AND VERIFY A ZERO IS READ BACK
                                        REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                                2928    ********** TEST 67 **********
                                        RECEIVER CONTROL REGISTER BIT 13 READ ONLY DEVICE RESET AND CLEA
                                        WRITE BIT 13 A ONE AND VERIFY A ZERO IS READ BACK
                                        REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                                2968    ********** TEST 70 **********
                                        RECEIVER CONTROL REGISTER BIT 14 READ ONLY DEVICE RESET AND CLEA
                                        WRITE BIT 14 A ONE AND VERIFY A ZERO IS READ BACK
                                        REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                                3008    ********** TEST 71 **********
                                        RECEIVER CONTROL REGISTER BIT 15 READ ONLY DEVICE RESET AND CLEA
                                        WRITE BIT 15 A ONE AND VERIFY A ZERO IS READ BACK
                                        REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                                3048    ********** TEST 72 **********
                                        RECEIVER BUFFER REGISTER BIT 0 READ ONLY DEVICE RESET AND CLEAR
                                        WRITE BIT 0 A ONE AND VERIFY A ZERO IS READ BACK
                                        REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
```

```
915
916                     3088    ********** TEST 73 **********
917                             RECEIVER BUFFER REGISTER BIT 1 READ ONLY DEVICE RESET AND CLEAR
918                             WRITE BIT 1 A ONE AND VERIFY A ZERO IS READ BACK
919                             REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
920
921                     3128    ********** TEST 74 **********
922                             RECEIVER BUFFER REGISTER BIT 2 READ ONLY DEVICE RESET AND CLEAR
923                             WRITE BIT 2 A ONE AND VERIFY A ZERO IS READ BACK
924                             REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
925
926                     3168    ********** TEST 75 **********
927                             RECEIVER BUFFER REGISTER BIT 3 READ ONLY DEVICE RESET AND CLEAR
928                             WRITE BIT 3 A ONE AND VERIFY A ZERO IS READ BACK
929                             REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
930
931                     3208    ********** TEST 76 **********
932                             RECEIVER BUFFER REGISTER BIT 4 READ ONLY DEVICE RESET AND CLEAR
933                             WRITE BIT 4 A ONE AND VERIFY A ZERO IS READ BACK
934                             REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
935
936                     3248    ********** TEST 77 **********
937                             RECEIVER BUFFER REGISTER BIT 5 READ ONLY DEVICE RESET AND CLEAR
938                             WRITE BIT 5 A ONE AND VERIFY A ZERO IS READ BACK
939                             REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
940
941                     3288    ********** TEST 100 **********
942                             RECEIVER BUFFER REGISTER BIT 6 READ ONLY DEVICE RESET AND CLEAR
943                             WRITE BIT 6 A ONE AND VERIFY A ZERO IS READ BACK
944                             REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
945
946                     3328    ********** TEST 101 **********
947                             RECEIVER BUFFER REGISTER BIT 7 READ ONLY DEVICE RESET AND CLEAR
948                             WRITE BIT 7 A ONE AND VERIFY A ZERO IS READ BACK
949                             REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
950
951                     3368    ********** TEST 102 **********
952                             RECEIVER BUFFER REGISTER BIT 8 READ ONLY DEVICE RESET AND CLEAR
953                             WRITE BIT 8 A ONE AND VERIFY A ZERO IS READ BACK
954                             REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
955
956                     3408    ********** TEST 103 **********
957                             RECEIVER BUFFER REGISTER BIT 9 READ ONLY DEVICE RESET AND CLEAR
958                             WRITE BIT 9 A ONE AND VERIFY A ZERO IS READ BACK
959                             REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
960
961                     3448    ********** TEST 104 **********
962                             RECEIVER BUFFER REGISTER BIT 10 READ ONLY DEVICE RESET AND CLEAR
963                             WRITE BIT 10 A ONE AND VERIFY A ZERO IS READ BACK
964                             REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
965
```

```
                        3488    ;************************* TEST 105 ****************************
                                ;RECEIVER BUFFER REGISTER BIT 12 READ ONLY DEVICE RESET AND CLEAR
                                ;WRITE BIT 12 A ONE AND VERIFY A ZERO IS READ BACK
                                ;REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                        3528    ;************************* TEST 106 ****************************
                                ;RECEIVER BUFFER REGISTER BIT 14 READ ONLY DEVICE RESET AND CLEAR
                                ;WRITE BIT 14 A ONE AND VERIFY A ZERO IS READ BACK
                                ;REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                        3568    ;************************* TEST 107 ****************************
                                ;RECEIVER BUFFER REGISTER BIT 15 READ ONLY DEVICE RESET AND CLEAR
                                ;WRITE BIT 15 A ONE AND VERIFY A ZERO IS READ BACK
                                ;REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                        3607    ;************************* TEST 110 ****************************
                                ;THIS TEST VERIFIES BIT7 OF THE  TRANSMITTER CONTROL REGISTER
                                ;TEST THAT BIT 7 SETS AFTER A RESET AND MRESET
                                ;VERIFY THAT WRITING THE LOW BYTE OF
                                ;THE TRANSMITTER BUFFER REGISTER CLEARS BIT 7

                        3647    ;************************* TEST 111 ****************************
                                ;TRANSMITTER CONTROL REGISTER BIT 9 READ ONLY DEVICE RESET AND CL
                                ;WRITE BIT 9 A ONE AND VERIFY A ZERO IS READ BACK
                                ;REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                        3689    ;************************* TEST 112 ****************************
                                ;TRANSMITTER CONTROL REGISTER BIT 14 READ ONLY DEVICE RESET AND C
                                ;WRITE BIT 14 A ONE AND VERIFY A ZERO IS READ BACK
                                ;REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                        3731    ;************************* TEST 113 ****************************
                                ;TRANSMITTER CONTROL REGISTER BIT 15 READ ONLY DEVICE RESET AND C
                                ;WRITE BIT 15 A ONE AND VERIFY A ZERO IS READ BACK
                                ;REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                        3773    ;************************* TEST 114 ****************************
                                ;TRANSMITTER BUFFER REGISTER BIT 12 READ ONLY DEVICE RESET AND CL
                                ;WRITE BIT 12 A ONE AND VERIFY A ZERO IS READ BACK
                                ;REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                        3813    ;************************* TEST 115 ****************************
                                ;TRANSMITTER BUFFER REGISTER BIT 13 READ ONLY DEVICE RESET AND CL
                                ;WRITE BIT 13 A ONE AND VERIFY A ZERO IS READ BACK
                                ;REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.

                        3853    ;************************* TEST 116 ****************************
                                ;TRANSMITTER BUFFER REGISTER BIT 14 READ ONLY DEVICE RESET AND CL
                                ;WRITE BIT 14 A ONE AND VERIFY A ZERO IS READ BACK
                                ;REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
```

```
1018
1019                    3893    ********************** TEST 117 **************************
1020                            TRANSMITTER BUFFER REGISTER BIT 15 READ ONLY DEVICE RESET AND CL
1021                            WRITE BIT 15 A ONE AND VERIFY A ZERO IS READ BACK
1022                            REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
1023
1024                    3933    ********************** TEST 120 **************************
1025                            PARAMETER STATUS REGISTER BIT 9 WRITE ONLY DEVICE RESET AND CLEA
1026                            TEST THAT BIT 9 CANNOT BE WRITTEN AND READ
1027                            BACK THE SAME.  READ BIT 9,COMPLEMENT IT AND WRITE IT
1028                            VERIFYING THAT IT  DID NOT CHANGE. WRITE BIT 9,
1029                            DO A DEVICE RESET AND VERIFY THE BIT WAS  CLEARED.
1030                            REPEAT FOR A CLR INSTRUCTION.
1031
1032                    3981    ********************** TEST 121 **************************
1033                            PARAMETER STATUS REGISTER BIT 12 WRITE ONLY DEVICE RESET AND CLE
1034                            TEST THAT BIT 12 CANNOT BE WRITTEN AND READ
1035                            BACK THE SAME.  READ BIT 12,COMPLEMENT IT AND WRITE IT
1036                            VERIFYING THAT IT  DID NOT CHANGE. WRITE BIT 12,
1037                            DO A DEVICE RESET AND VERIFY THE BIT WAS  CLEARED.
1038                            REPEAT FOR A CLR INSTRUCTION.
1039
1040                    4029    ********************** TEST 122 **************************
1041                            PARAMETER STATUS REGISTER BIT 15 WRITE ONLY DEVICE RESET AND CLE
1042                            TEST THAT BIT 15 CANNOT BE WRITTEN AND READ
1043                            BACK THE SAME.  READ BIT 15,COMPLEMENT IT AND WRITE IT
1044                            VERIFYING THAT IT  DID NOT CHANGE. WRITE BIT 15,
1045                            DO A DEVICE RESET AND VERIFY THE BIT WAS  CLEARED.
1046                            REPEAT FOR A CLR INSTRUCTION.
1047
1048                    4077    ********************** TEST 123 **************************
1049                            TRANSMITTER CONTROL REGISTER BIT 8 WRITE ONLY DEVICE RESET AND C
1050                            TEST THAT BIT 8 CANNOT BE WRITTEN AND READ
1051                            BACK THE SAME.  READ BIT 8,COMPLEMENT IT AND WRITE IT
1052                            VERIFYING THAT IT  DID NOT CHANGE. WRITE BIT 8,
1053                            DO A DEVICE RESET AND VERIFY THE BIT WAS  CLEARED.
1054                            REPEAT FOR A CLR INSTRUCTION.
1055
1056                    4125    ********************** TEST 124 **************************
1057                            RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
1058                            SET BIT8, VERIFY BIT8 WAS SET.
1059                            CLEAR BIT8, VERIFY BIT8 WAS CLEARED.
1060
1061                    4155    ********************** TEST 125 **************************
1062                            RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
1063                            SET BIT6, VERIFY BIT6 WAS SET.
1064                            CLEAR BIT6, VERIFY BIT6 WAS CLEARED.
1065
1066                    4185    ********************** TEST 126 **************************
1067                            RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
1068                            SET BIT5, VERIFY BIT5 WAS SET.
1069                            CLEAR BIT5, VERIFY BIT5 WAS CLEARED.
1070
```

```
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
```

```
4215    ************************** TEST 127 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
        SET BIT4, VERIFY BIT4 WAS SET.
        CLEAR BIT4, VERIFY BIT4 WAS CLEARED.

4245    ************************** TEST 130 ****************************
        TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
        SET BIT6, VERIFY BIT6 WAS SET.
        CLEAR BIT6, VERIFY BIT6 WAS CLEARED.

4277    ************************** TEST 131 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
        SET BIT1, VERIFY BIT1 WAS SET.
        CLEAR BIT1, VERIFY BIT1 WAS CLEARED.

4309    ************************** TEST 132 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
        SET BIT2, VERIFY BIT2 WAS SET.
        CLEAR BIT2, VERIFY BIT2 WAS CLEARED.

4341    ************************** TEST 133 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
        SET BIT3, VERIFY BIT3 WAS SET.
        CLEAR BIT3, VERIFY BIT3 WAS CLEARED.

4373    ************************** TEST 134 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT 6 RESET AND CLEAR TEST
        WRITE BIT 6,AND TEST THAT IT WILL BE CLEARED AFTER A
        DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION

4405    ************************** TEST 135 ****************************
        RECEIVER CONTROL REGISTER READ/WRITE BIT 5 RESET AND CLEAR TEST
        WRITE BIT 5,AND TEST THAT IT WILL BE CLEARED AFTER A
        DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION

4437    ************************** TEST 136 ****************************
        TRANSMITTER CONTROL REGISTER READ/WRITE BIT 6 RESET AND CLEAR TE
        WRITE BIT 6,AND TEST THAT IT WILL BE CLEARED AFTER A
        DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION

4469    ************************** TEST 137 ****************************
        THIS TEST CHECKS THE MAINTENANCE CLOCK
        USED THROUGHOUT THE REMAINING DIAGNOSTICS

4506    ************************** TEST 140 ****************************
        THIS TEST WILL PERFORM STATIC TRANSMITTER FUNCTIONS
        IN MAINTENANCE MODE.  IT WILL PROVE THE INTERACTION
        OF SEND, DONE AND TSOM.
```

```
1121
1122                    4553    ************************** TEST 141 ****************************
1123                            THIS TEST CHECKS THE STATIC FUNCTIONS OF THE TRANSMITTER
1124                            IN MAINTENANCE MODE.  THIS TEST PROVES THE INTERACTION OF
1125                            TXDBUF, TXACT, TSOM,TRANSMITTED DATA AND DONE.
1126
1127                    4628    ************************** TEST 142 ****************************
1128                            THIS TEST VERIFIES THAT THE DEVICE IDLES FLAGS
1129                            IDLE A MINIMUM OF 64. FLAGS.
1130
1131                    4654    ************************** TEST 143 ****************************
1132                            THIS TEST PUSHES DATA THRU THE TRANSMITTER
1133                            IN MAINTENANCE MODE. THE TEST SENDS A FLAG,
1134                            AND TWO ALTERNATING ONES AND ZEROES CHARACTERS,
1135                            AN ALL ZEROES CHARACTER AND AN ALL ONES
1136                            CHARACTER TO VERIFY THE BIT STUFF CAPABILITY OF
1137                            THE DUP WITHOUT A CRC CHECK.THE TEST ROTATES
1138                            THE BITS THRU,SAMPLING THE DATA ON A BIT-BY-BIT
1139                            BASIS,LSB FIRST. IT STORES THE BIT IN THE MSB OF
1140                            THE SAVE LOCATION,COMPARES AND ROTATES RIGHT UNTIL
1141                            THE CHARACTER IS ASSEMBLED.
1142
1143                    4763    ************************** TEST 144 ****************************
1144                            THIS TEST VERIFIES THE ABORT SEQUENCE AND
1145                            NORMAL DATA SEQUENCE OF FLAG, DATA, FLAG
1146                            FOLLOWED BY IDLE LINE. THIS TEST ALSO PROVES
1147                            THE FUNCTIONING OF ACTIVE, TSOM, TEOM, SENC AND DONE.
1148                            WITHOUT USING A CRC CHECK.
1149
1150                    4831    ************************** TEST 145 ****************************
1151                            THIS TEST VERIFIES THAT A DATA
1152                            UNDER RUN CONDITION WILL CAUSE
1153                            THE TRANSMITTER DATA LATE BIT TO SET
1154                            AND THAT THE DEVICE WILL ABORT
1155                            UNTIL SEND IS CLEARED WHEN THE OUTPUT GOES TO A SPACE
1156
1157                    4879    ************************** TEST 146 ****************************
1158                            THIS TEST VERIFIES THAT DROPPING OF
1159                            SEND BEFORE SETTING TEOM CAUSES
1160                            A SPACE TO BE OUTPUT AFTER COMPLETION OF
1161                            A CHARACTER WITH AND WITHOUT BIT STUFF.
1162
1163                    4952    ************************** TEST 147 ****************************
1164                            THIS TEST VERIFIES THAT A DATA UNDERRUN
1165                            CONDITION WILL CAUSE THE TRANSMITTER DATA
1166                            LATE BIT TO SET AND THAT THE DUP WILL GO TO
1167                            A MARK OUTPUT
1168
1169                    4998    ************************** TEST 150 ****************************
1170                            THIS TEST VERIFIES THAT SETTING TEOM
1171                            AND TSOM AT THE SAME TIME WILL HOLD
1172                            ACTIVE UP AND SEND A FLAG
1173
```

```
1174
1175                                    5033    ************************** TEST 151 ****************************
1176                                            TEST OF THE BCC OPERATION USING
1177                                            CRC.CCITT FOR THE POLYNOMINAL. SPECIFIC
1178                                            DATA PATTERNS ARE USED TO ISOLATE FAULTS.
1179
1180                                    5213    ************************** TEST 152 ****************************
1181                                            THIS TEST IS AN AID FOR DEBUGGING CRC
1182                                            ERRORS.  A CHARACTER IS LOADED INTO THE
1183                                            DUP AND PUSHED OUT BIT BY BIT WHILE
1184                                            ALLOWING THE OPERATOR TO MONITOR THE CRC
1185                                            CHARACTER AS IT IS GENERATED.  THE DATA CHARACTER
1186                                            CAN ALSO BE CHANGED BY THE OPERATOR.
1187                                            PUT SW09=1 TO LOCK ON BITS.  TO CONTINUE HIT
1188                                            ANY KEY ON THE TTY. AFTER 16 TIMES PUT DOWN SW09 TO LEAVE
1189                                  ;
```

```
                                ;*MAINDEC-11-DZDPB-B /(377)/BASIC DUP11 AND OFFLINE SDLC TRANSMITTER TESTS
                                ;*COPYRIGHT(C) 1975,1977, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
                                ;*---------------------------------------------------------------------

                                        ;STARTING PROCEDURE
                                        ;LOAD PROGRAM
                                        ;LOAD ADDRESS 000200
                                        ;PRESS START
                                        ;PROGRAM WILL TYPE "MAINDEC-11-DZDPB-B /(377)/BASIC DUP11 AND OFFLINE SDLC TRANS
                                        ;PROGRAM WILL TYPE "R" TO INDICATE THAT TESTING HAS STARTED
                                        ;AT THE END OF A PASS, PROGRAM WILL TYPE PASS COMPLETE MESSAGE
                                        ;AND THEN RESUME TESTING


                                ;SWITCH REGISTER OPTIONS
                                ;----------------------

  100000                        SW15=100000             ;=1,HALT ON ERROR
  040000                        SW14=40000              ;=1,LOOP ON CURRENT TEST
  020000                        SW13=20000              ;=1,INHIBIT ERROR TYPEOUT
  010000                        SW12=10000              ;=1,DELETE TYPEOUT/BELL ON ERROR.
  004000                        SW11=4000               ;=1,INHIBIT ITERATIONS
  002000                        SW10=2000               ;=1,ESCAPE TO NEXT TEST ON ERROR
  001000                        SW09=1000               ;=1,LOOP WITH CURRENT DATA
  000400                        SW08=400                ;=1,LOOP ON ERROR
  000200                        SW07=200
  000100                        SW06=100
  000040                        SW05=40
  000020                        SW04=20
  000010                        SW03=10                 ;SELECT DUP'S DESIRED ACTIVE
                                ;NOTE:THIS MUST NOT EXCEED ORIGINAL COUNT
  000004                        SW02=4                  ;LOCK ON TEST SELECT
  000002                        SW01=2                  ;RESTART PROGRAM AT SELECTED TEST
  000001                        SW00=1                  ;ENTER PARAMETERS
```

```
                                ;REGISTER DEFINITIONS
                                ;--------------------

000000                          R0=%0           ;GENERAL REGISTER
000001                          R1=%1           ;GENERAL REGISTER
000002                          R2=%2           ;GENERAL REGISTER
000003                          R3=%3           ;GENERAL REGISTER
000004                          R4=%4           ;GENERAL REGISTER
000005                          R5=%5           ;GENERAL REGISTER
000006                          SP=%6           ;PROCESSOR STACK POINTER
000007                          PC=%7           ;PROGRAM COUNTER

                                ;LOCATION EQUIVALENCIES
                                ;----------------------

177776                          PS=177776       ;PROCESSOR STATUS WORD
001150                          STACK=1150      ;START OF PROCESSOR STACK

                                ;INSTRUCTION DEFINITIONS
                                ;-----------------------

005746                          PUSH1SP=5746    ;DECREMENT PROCESSOR STACK 1 WORD
005726                          POP1SP=5726     ;INCREMENT PROCESSOR STACK 1 WORD
010046                          PUSHR0=10046    ;SAVE R0 ON STACK
012600                          POPR0=12600     ;RESTORE R0 FROM STACK
024646                          PUSH2SP=24646   ;DECREMENT STACK TWICE
022626                          POP2SP=22626    ;INCREMENT STACK TWICE
                                .EQUIV  EMT,HLT ;BASIC DEFINITION OF ERROR CALL


100000                          BIT15=100000
040000                          BIT14=40000
020000                          BIT13=20000
010000                          BIT12=10000
004000                          BIT11=4000
002000                          BIT10=2000
001000                          BIT9=1000
000400                          BIT8=400
000200                          BIT7=200
000100                          BIT6=100
000040                          BIT5=40
000020                          BIT4=20
000010                          BIT3=10
000004                          BIT2=4
000002                          BIT1=2
000001                          BIT0=1
```

```
                                        ;;*****************************************************************************
                                        ;-----------------------------------------------------------------------------
                                        ;TRAPCATCHER FOR ILLEGAL INTERRUPTS
                                        ;THE STANDARD "TRAP CATCHER" IS PLACED
                                        ;BETWEEN ADDRESS 0 TO ADDRESS 776.
                                        ;IT LOOKS LIKE "PC+2 HALT".
                                        ;-----------------------------------------------------------------------------
                                        ;;*****************************************************************************

                                        .=0
                                        ;STANDARD INTERRUPT VECTORS
                                        ;-------------------------

                000024                  .=24
000024  005050                          .PFAIL                  ;POWER FAIL HANDLER
000026  000340                          340                     ;SERVICE AT LEVEL 7
000030  004350                          .HLT                    ;ERROR HANDLER
000032  000340                          340                     ;SERVICE AT LEVEL 7
000034  004316                          .TRPSRV                 ;GENERAL HANDLER DISPATCH SERVICE
000036  000340                          340                     ;SERVICE AT LEVEL 7
                000040                  .=40
000040  000000                          0                       ;SAVE FOR ACT-11 OR DDP2
000042  000000                          0                       ;RETURN ADDRESS IF UNDER ACT-11 OR DDP2
000044  000000                          0                       ;SAVE FOR ACT-11 OR DDP2
000046  003104                          $ENDAD                  ;FOR USE WITH ACT-11 OR DDP2
                000052                  .=52
000052  000000                          0                       ;ACT-11 PROGRAM CHARACTERISTICS

                000174                  .=174
000174  000000                  DISPREG:0                       ;SOFTWARE DISPLAY REGISTER
000176  000000                  SWREG:  0                       ;SOFTWARE SWITCH REGISTER
                000200                  .=200
000200  000137  001562                          JMP     .START  ;GO TO START OF PROGRAM

                001000                  .=1000
001000  005377  040515  047111  MTITLE: .ASCIZ  <377><12>MAINDEC-11-DZDPB-B <377>BASIC DJP11 AND OFFLINE SDLC TRANSMI
(2)
                001200                  .=1200
                                        ;SWR AND LIGHTS
                                        ;--------------

001200  177570                  DISPLAY:       177570           ;11/45 CONSOLE LIGHTS
001202  177570                  SWR:    177570                  ;INDIRECT POINTER TO SWITCH REGISTER

                                        ;INDIRECT POINTERS TO TELETYPE VECTORS AND REGISTERS
                                        ;--------------------------------------------------

001204  177560                  TKCSR:  177560                  ;TELETYPE KEYBOARD CONTROL REGISTER
001206  177562                  TKDBR:  177562                  ;TELETYPE KEYBOARD DATA BUFFER
001210  177564                  TPCSR:  177564                  ;TELEPRINTER CONTROL REGISTER
001212  177566                  TPDBR:  177566                  ;TELEPRINTER DATA BUFFER

                                        ;PROGRAM CONTROL PARAMETERS
                                        ;--------------------------
```

```
1329  001214  000000                  RETURN: 0                       ;SCOPE ADDRESS FOR LOOP ON TEST
1330  001216  000000                  NEXT:   0                       ;ADDRESS OF NEXT TEST TO BE EXECUTED
1331  001220  000000                  LOCK:   0                       ;ADDRESS FOR LOCK ON CURRENT DATA
1332  001222  000001                  ICOUNT: 1                       ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
1333  001224  000000                  LPCNT:  0                       ;NUMBER OF ITEREATIONS COMPLETED
1334  001226  000000                  TSTNO:  0                       ;NUMBER OF TEST IN PROGRESS
1335  001230  000000                  PASCNT: 0                       ;NUMBER OF PASSES COMPLETED
1336  001232  000000                  ERRCNT: 0                       ;TOTAL NUMBER OF ERRORS
1337  001234  000000                  LSTERR: 0                       ;PC OF LAST ERROR CALL
1338
1339                                  ;PROGRAM VARIABLES
1340                                  ;-----------------
1341
1342  001236  000000                  TEMP1:  0                       ;TEMPORARY STORAGE
1343  001240  000000                  TEMP2:  0                       ;TEMPORARY STORAGE
1344  001242  000000                  TEMP3:  0                       ;TEMPORARY STORAGE
1345  001244  000000                  TEMP4:  0                       ;TEMPORARY STORAGE
1346  001246  000000                  TEMP5:  0                       ;TEMPORARY STORAGE
1347  001250  000000                  SAVR0:  0                       ;R0 STORAGE
1348  001252  000000                  SAVR1:  0                       ;R1 STORAGE
1349  001254  000000                  SAVR2:  0                       ;R2 STORAGE
1350  001256  000000                  SAVR3:  0                       ;R3 STORAGE
1351  001260  000000                  SAVR4:  0                       ;R4 STORAGE
1352  001262  000000                  SAVR5:  0                       ;R5 STORAGE
1353  001264  000000                  SAVSP:  0                       ;STACK POINTER STORAGE
1354  001266  000000                  SAVPC:  0                       ;PROGRAM COUNTER STORAGE
1355
1356  001270  000000                  SAVR0A: 0                       ;R0 STORAGE
1357  001272  000000                  SAVR1A: 0                       ;R1 STORAGE
1358  001274  000000                  SAVR2A: 0                       ;R2 STORAGE
1359  001276  000000                  SAVR3A: 0                       ;R3 STORAGE
1360  001300  000000                  SAVR4A: 0                       ;R4 STORAGE
1361  001302  000000                  SAVR5A: 0                       ;R5 STORAGE
1362  001304  000000                  SAVSPA: 0                       ;STACK POINTER STORAGE
1363  001306  000000                  SAVPCA: 0                       ;PROGRAM COUNTER STORAGE
1364
1365  001310  000001                  DUPACTV: .BLKB  1               ;DUP11'S SELECTED ACTIVE.
1366  001311  000001                  DUPNUM: .BLKB   1               ;OCTAL NUMBER OF DUP11'S.
1367  001312  000001                  SAVACT: .BLKB   1               ;ORIGINAL ACTV.  DEVICES.
1368  001313  000001                  SAVNUM: .BLKB   1               ;WORKABLE NUMBER.
1369  001314  000001                  RUN:    .BLKB   1               ;POINTER ONE PAST RUNNING DEVICE.
1370          001316                  .EVEN
1371  001316  001500                  CREAM:  DUP.MAP                 ;TABLE POINTER.
```

```
1372
1373                                    ;CONTROL REGISTER DEFINITIONS
1374                                    ;---------------------------
1375                            ;RXCSR BIT DEFINITIONS
1376    100000          DSCA=BIT15              ;DATA SET CHANGE A
1377    040000          RING=BIT14              ;RING
1378    020000          CTS=BIT13               ;CLR TO SEND
1379    010000          CARDET=BIT12            ;CARRIER DETECT
1380    004000          RECACT=BIT11            ;REC ACTIVE
1381    002000          SRD=BIT10               ;SEC REC DATA
1382    001000          DSR=BIT9                ;DATA SET RDY
1383    000400          STPSYN=BIT8             ;STRIP SYNC
1384    000200          RXDONE=BIT7             ;REC DONE
1385    000100          RINTEN=BIT6             ;REC INTR ENABLE
1386    000040          DSINTE=BIT5             ;DSC INTR ENABLE
1387    000020          RCVEN=BIT4              ;REC ENABLE
1388    000010          STD=BIT3                ;SEC XMIT DATA
1389    000004          RTS=BIT2                ;REQ TO SEND
1390    000002          DTR=BIT1                ;DATA TERM RDY
1391    000001          DSCB=BIT0               ;DATA SET CHANGE B
1392                            ;RXDBUF BIT DEFINITIONS
1393    100000          RXDERR=BIT15            ;REC DATA ERROR
1394    040000          OVRRUN=BIT14            ;OVERRUN ERROR
1395    010000          CRCERR=BIT12            ;CRC ERROR
1396    002000          RABORT=BIT10            ;REC ABORT
1397    001000          REOM=BIT9               ;REC END OF MESSAGE
1398    000400          RSOM=BIT8               ;REC START OF MESSAGE
1399                            ;PARCSR BIT DEFINITIONS
1400    100000          DECMOD=BIT15            ;DEC MODE (DDCMP)
1401    001000          CRCEN=BIT9              ;CRC ENABLE
1402    010000          PRISEC=BIT12            ;PRI/SEC SELECT
1403                            ;TXCSR BIT DEFINITIONS
1404    100000          TXDLAT=BIT15            ;TX DATA LATE
1405    040000          MTDATA=BIT14            ;MAINT DATA OUT
1406    020000          CLK=BIT13               ;CLK
1407    010000          MMODEB=BIT12            ;MAINT MODE B
1408    004000          MMODEA=BIT11            ;MAINT MODE A
1409    002000          BITW=BIT10              ;BIT WINDOW INPUT
1410    001000          TXACT=BIT9              ;TX ACTIVE
1411    000400          MRESET=BIT8             ;MASTER RESET
1412    000200          TXDONE=BIT7             ;XMIT DONE
1413    000100          TXINTE=BIT6             ;XMIT DONE INTR ENABLE
1414    000020          SEND=BIT4               ;SEND
1415    000010          HDXEN=BIT3              ;HDX/FDX
1416                            ;TXCSR WRD DEFINITIONS
1417    000000          USER=0                  ;USER MODE
1418    014000          MMODE=14000             ;MAINT INT MODE
1419    010000          MEXT=10000              ;MAINT EXT MODE
1420    004000          SYSTST=4000             ;SYSTEM TEST MODE
1421
1422                            ;TXDBUF BIT DEFINITIONS
1423                            ;----------------------
1424    100000          RCRC7T=BIT15
1425    040000          RCRCIN=BIT14
1426    020000          TCRC7T=BIT13
1427    010000          TCRCIN=BIT12
```

```
1428            004000                  TIMER=BIT11             ;MAINTENANCE TIMER
1429            002000                  TABORT=BIT10            ;TRANSMIT ABORT
1430            001000                  TEOM=BIT9               ;TRANSMIT END OF MESSAGE
1431            000400                  TSOM=BIT8               ;TRANSMIT START OF MESSAGE
1432
1433                                    ;MISC. PROGRAM DEFINITIONS
1434                                    ;-------------------------
1435  001320    000000                  PRIRTY:.WORD 0
1436  001322    000001                  TCNFLG: .BLKB 1
1437  001323    000001                  OPCLRJ: .BLKB 1
1438  001324    000000                  DATA:   .WORD   0
1439  001326    000000                  SHIFTS: .WORD   0
1440  001330    000000                  MIND:   .WORD   0
1441  001332    000000                  FLAG:   .WORD   0
1442  001334    000001                  STJMFL: .BLKW 1
1443  001336    000001                  SRJMFL: .BLKW 1
1444
1445
```

```
1446
1447                                        ;PROGRAM CONTROL FLAGS
1448                                        ;--------------------
1449
1450  001340     000            INIFLG: .BYTE   0               ;PROGRAM INITIALIZATION FLAG
1451  001341     000            ERRFLG: .BYTE   0               ;ERROR OCCURED FLAG
1452  001342     000            LOKFLG: .BYTE   0               ;LOCK ON CURRENT TEST FLAG
1453  001343     000            QV.FLG: .BYTE   0               ;QUICK VERIFY FLAG.
1454                                                           ;ON FIRST PASS OF EACH DUP11 ITERATIONS
1455                                                           ;WILL BE SUPPRESSED
1456                                    .EVEN
1457           000000           $Y=0
1458
1459                                        ;DEFINITIONS FOR TRAP SUBROUTINE CALLS
1460                                        ;POINTERS TO SUBROUTINES CAN BE FOUND
1461                                        ;IN THE TABLE IMMEDIATLY FOLLOWING THE DEFINITIONS
1462
1463                                    ;;**********************************************************************
1464                                    ;----------------------------------------------------------------------
1465  001344                            .TRPTAB:
1466           104400           SCOPE=TRAP+0            ;CALL TO SCOPE LOOP AND ITERATION HANDLER
1467  001344   003160                    .SCOPE
1468           104401           SCOP1=TRAP+1            ;CALL TO LOOP ON CURRENT DATA HANDLER
1469  001346   003312                    .SCOP1
1470           104402           TYPE=TRAP+2             ;CALL TO TELETYPE OUTPUT ROUTINE
1471  001350   003336                    .TYPE
1472           104403           INSTR=TRAP+3            ;CALL TO ASCII STRING INPUT ROUTINE
1473  001352   003412                    .INSTR
1474           104404           INSTER=TRAP+4           ;CALL TO INPUT ERROR HANDLER
1475  001354   003516                    .INSTER
1476           104405           PARAM=TRAP+5            ;CALL TO NUMERICAL DATA INPUT ROUTINE
1477  001356   003536                    .PARAM
1478           104406           SAVO5=TRAP+6            ;CALL TO REGISTER SAVE ROUTINE
1479  001360   003736                    .SAVO5
1480           104407           RESO5=TRAP+7            ;CALL TO REGISTER RESTORE ROUTINE
1481  001362   003776                    .RESO5
1482           104410           CONVRT=TRAP+10          ;CALL TO DATA OUTPUT ROUTINE
1483  001364   004030                    .CONVRT
1484           104411           CNVRT=TRAP+11           ;CALL TO DATA OUTPUT ROUNTINE WITHOUT CR/LF.
1485  001366   004034                    .CNVRT
1486           104412           PKCLK=TRAP+12           ;CALL TO CLOCK ROUTINE
1487  001370   005006                    .PKCLK
1488           104413           SETFLG=TRAP+13          ;CALL TO TELETYPE INPUT ROUTINE
1489  001372   004242                    .SETFLG
1490
1491                                    ;----------------------------------------------------------------------
1492                                    ;;**********************************************************************
```

```
1493                                    ;DUP11 VECTOR AND REGISTER INDIRECT POINTERS
1494
1495  001374  000000            DUPRVC: 0               ;POINTER TO DUP11 RECEIVER INTERRUPT VECTOR
1496  001376  000000            DUPRPS: 0               ;POINTER TO DUP11 RECEIVER INTERRUPT SERVICE PS
1497  001400  000000            DUPTVC: 0               ;POINTER TO DUP11 TRANSMITTER INTERRUPT VECTOR
1498  001402  000000            DUPTPS: 0               ;POINTER TO DUP11 TRANSMITTER INTERRUPT SERVICE PS
1499  001404  000000            RXCSR: 0                ;POINTER TO DUP11 RECEIVER STATUS REGISTER
1500  001406  000000            RXDBUF: 0               ;POINTER TO DUP11 RECEIVER DATA BUFFER
1501  001410  000000            PARCSR: 0               ;POINTER TO DUP11 PARAMETER STATUS REGISTER
1502  001412  000000            TXCSR: 0                ;POINTER TO DUP11 TRANSMITTER STATUS REGISTER
1503  001414  000000            TXDBUF: 0               ;POINTER TO DUP11 TRANSMITTER DATA BUFFER
1504  001416  000000            DUPSEC: 0               ;POINTER TO DUP11 SECONDARY REGISTER SELECT REGISTER
1505  001420  000000            HUPPSR: 0               ;POINTER TO PARAMETER STATUS HIGH BYTE
1506  001422  000000            HUPRBF: 0               ;POINTER TO RECEIVER BUFFER HIGH BYTE
1507  001424  000000            HUPRCR: 0               ;POINTER TO RECEIVER CONTROL REG HIGH BYTE
1508  001426  000000            HUPTBF: 0               ;POINTER TO TRANSMITTER BUFFER HIGH BYTE .
1509  001430  000000            HUPTCR: 0               ;POINTER TO TRANSMITTER CONTROL REG HIGH BYTE
1510
1511
1512                                    ;DUP11 CONTROL INDICATORS FOR CURRENT DUP11 UNDER TEST
1513                                    ;---------------------------------------------------
1514
1515  001432     000             MASK.A: .BYTE 000       ;LAST CHAR TO TEST AND PARITY MASK
1516
1517  001433     010             CLK.A:  .BYTE 8.        ;NUMBER OF CLOCKS NEEDED FOR ONE CHAR
1518
1519  001434  000000             LOO.00: 000000          ;PARAMETERS
1520
```

```
1521                                            ;DUP11 STATUS TABLE AND ADDRESS ASSIGNMENTS
1522                                            ;-----------------------------------------
1523
1524            001500                  .=1500
1525    001500                  DUP.MAP:
1526    001500  000001          DUPCR0: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 0
1527    001502  000001          DUPTR0: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 0
1528    001504  000001          DUP0.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 0
1529
1530    001506  000001          DUPCR1: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 1
1531    001510  000001          DUPTR1: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 1
1532    001512  000001          DUP1.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 1
1533
1534    001514  000001          DUPCR2: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 2
1535    001516  000001          DUPTR2: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 2
1536    001520  000001          DUP2.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 2
1537
1538    001522  000001          DUPCR3: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 3
1539    001524  000001          DUPTR3: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 3
1540    001526  000001          DUP3.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 3
1541
1542    001530  000001          DUPCR4: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 4
1543    001532  000001          DUPTR4: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 4
1544    001534  000001          DUP4.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 4
1545
1546    001536  000001          DUPCR5: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 5
1547    001540  000001          DUPTR5: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 5
1548    001542  000001          DUP5.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 5
1549
1550    001544  000001          DUPCR6: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 6
1551    001546  000001          DUPTR6: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 6
1552    001550  000001          DUP6.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 6
1553
1554    001552  000001          DUPCR7: .BLKW 1         ;CONTROL STATUS REGISTER FOR DUP11 NUMBER 7
1555    001554  000001          DUPTR7: .BLKW 1         ;VECTOR "A" FOR DUP11 NUMBER 7
1556    001556  000001          DUP7.A: .BLKW 1         ;PARAMETER  FOR DUP11 NUMBER 7
1557
1558    001560  000000          DUP.END:        000000
1559
1560
1561
1562
1563
```

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
CONTROL								REGISTER							
VECTOR															
A	B	C	D	E	F	G	H	*	*	SYNC		*		*	

DEFINITIONS

A- OPTIONAL CLEAR JUMPER IN=1
B- TURNAROUND CONNECTOR ON=1
C-
D-

```
1583
1584                                        ;PROGRAM INITIALIZATION
1585                                        ;LOCK OUT INTERRUPTS
1586                                        ;SET UP PROCESSOR STACK
1587                                        ;SET UP POWER FAIL VECTOR
1588                                        ;CLEAR PROGRAM CONTROL FLAGS AND COUNTS
1589                                        ;TYPE TITLE MESSAGE
1590
1591  001562  012737  000340  177776  .START: MOV     #340,PS                 ;LOCK OUT INTERRUPTS
1592  001570  012706  001150                  MOV     #STACK,SP               ;SET UP STACK
1593  001574  012737  005050  000024          MOV     #.PFAIL,@#24            ;SET UP POWER FAIL VECTOR
1594  001602  113737  001311  001313          MOVB    DUPNUM,SAVNUM           ;SAVE NUMBER OF DEVICES IN SYSTEM
1595  001610  005037  001230                  CLR     PASCNT                  ;CLEAR PASS COUNT
1596  001614  105037  001341                  CLRB    ERRFLG                  ;CLEAR ERROR FLAG
1597  001620  105037  001343                  CLRB    QV.FLG                  ;ZERO QUICK VERIFY FLAG
1598  001624  012737  001500  001316          MOV     #DUP.MAP,CREAM          ;GET MAP POINTER.
1599  001632  112737  000001  001314          MOVB    #1,RUN                  ;POINT POINTER TO FIRST DEVICE.
1600  001640  005037  001232                  CLR     ERRCNT                  ;CLEAR ERROR COUNT
1601  001644  005037  001234                  CLR     LSTERR                  ;CLEAR LAST ERROR POINTER
1602  001650  012737  000001  001226          MOV     #1,TSTNO                ;SET UP FOR TEST 1
1603  001656  012737  001562  001214          MOV     #.START,RETURN          ;SET UP FOR POWER FAIL BEFORE
1604                                                                          ;TESTING STARTS
1605  001664  013746  000006                  MOV     @#6,-(SP)               ;SAVE CURRENT VECTORS
1606  001670  013746  000004                  MOV     @#4,-(SP)               ;
1607  001674  012737  001710  000004          MOV     #12$,@#4                ;SETUP FOR TIMEOUT
1608  001702  005777  177274                  TST     @SWR                    ;REFERENCE HARDWARE SWITCH REG
1609  001706  000407                          BR      13$                     ;BR IF IT EXISTS
1610  001710  012737  000176  001202  12$:     MOV     #SWREG,SWR      ;POINT TO SOFT SWR
1611  001716  012737  000174  001200          MOV     #DISPREG,DISPLAY        ;POINT TO SOFT DISPLAY REG
1612  001724  022626                          CMP     (SP)+,(SP)+             ;ADJUST STACK
1613  001726  012637  000004          13$:    MOV     (SP)+,@#4               ;RESTORE VECTORS
1614  001732  012637  000006                  MOV     (SP)+,@#6               ;
1615  001736  105737  001340                  TSTB    INIFLG                  ;HAS INITIALIZATION BEEN PERFORMED
1616  001742  001401                          BEQ     11$
1617  001744  000410                          BR      6$
1618  001746  022737  003104  000042  11$:     CMP     #$ENDAD,@#42    ;IF ACT-11 AUTO MODE,
1619  001754  001404                          BEQ     6$              ;DON'T TYPE ID
1620  001756  104402  001000                  TYPE    ,MTITLE                 ;TYPE TITLE MESSAGE
1621  001762  105137  001340                  COMB    INIFLG                  ;IF NOT SET FLAG AND DO
1622  001766  105777  177210          6$:     TSTB    @SWR                    ;BIT7=1?
1623  001772  100002                          BPL     10$
1624  001774  000137  002520                  JMP     1$
1625  002000                          10$:
1626  002000  032777  000001  177174          BIT     #SW00,@SWR      ;ENTER PARAMETERS
1627  002006  001002                          BNE     .+6             ;YES
1628  002010  000137  002360                  JMP     21$             ;NO
1629  002014  105137  001332                  COMB    FLAG
1630  002020  112737  000001  001340          MOVB    #1,INIFLG       ;SET TO MANUAL ENTRY
1631  002026  012700  001500                  MOV     #DUP.MAP,R0     ;CLR MAP
1632  002032  005020                  68$:    CLR     (R0)+
1633  002034  020027  001560                  CMP     R0,#DUP.END     ;DONE WITH MAP?
1634  002040  001374                          BNE     68$             ;BR IF NO
1635  002042  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
1636  002044  005474                          MCSR    ;MESSAGE
1637  002046  104405                          PARAM   ;CONVERT STRING
1638  002050  160000                          160000  ;LOW LIMIT
```

```
1639  002052  175500                          175500  ;HIGH LIMIT
1640  002054  001500                          DUPCR0  ;STORE AT THIS LOCATION
1641  002056     001                  .BYTE   1       ;MASK
1642  002057     001                  .BYTE   1       ;HOW MANY TIMES + 2
1643  002060  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
1644  002062  005513                          MVEC    ;MESSAGE
1645  002064  104405                          PARAM   ;CONVERT STRING
1646  002066  000300                          300     ;LOW LIMIT
1647  002070  000770                          770     ;HIGH LIMIT
1648  002072  001502                          DUPTR0  ;STORE AT THIS LOCATION
1649  002074     001                  .BYTE   1       ;MASK
1650  002075     001                  .BYTE   1       ;HOW MANY TIMES + 2
1651  002076  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
1652  002100  005703                          MPAR    ;MESSAGE
1653  002102  104405                          PARAM   ;CONVERT STRING
1654  002104  000004                          4       ;LOW LIMIT
1655  002106  000007                          7       ;HIGH LIMIT
1656  002110  001240                          TEMP2   ;STORE AT THIS LOCATION
1657  002112     000                  .BYTE   0       ;MASK
1658  002113     001                  .BYTE   1       ;HOW MANY TIMES + 2
1659  002114  013737  001240  001320          MOV     TEMP2,PRIRTY    ;SAVE PRIORITY
1660  002122  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
1661  002124  005650                          MTOTAL  ;MESSAGE
1662  002126  104405                          PARAM   ;CONVERT STRING
1663  002130  000001                          1       ;LOW LIMIT
1664  002132  000010                          8.      ;HIGH LIMIT
1665  002134  001236                          TEMP1   ;STORE AT THIS LOCATION
1666  002136     000                  .BYTE   0       ;MASK
1667  002137     001                  .BYTE   1       ;HOW MANY TIMES + 2
1668  002140  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
1669  002142  005526                          MJMPR   ;MESSAGE
1670  002144  104413                          SETFLG  ;SET FLAG BASED UPON INPUT STRING
1671  002146  001323                          OPCLRJ  ;THIS FLAG
1672  002150  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
1673  002152  005601                          MTCN    ;MESSAGE
1674  002154  104413                          SETFLG  ;SET FLAG BASED UPON INPUT STRING
1675  002156  001322                          TCNFLG  ;THIS FLAG
1676  002160  105737  001322                  TSTB    TCNFLG
1677  002164  001410                          BEQ     71$
1678  002166  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
1679  002170  005727                          MSTJM   ;MESSAGE
1680  002172  104413                          SETFLG  ;SET FLAG BASED UPON INPUT STRING
1681  002174  001334                          STJMFL  ;THIS FLAG
1682  002176  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
1683  002200  005762                          MSRJM   ;MESSAGE
1684  002202  104413                          SETFLG  ;SET FLAG BASED UPON INPUT STRING
1685  002204  001336                          SRJMFL  ;THIS FLAG
1686  002206  105737  001323          71$:    TSTB    OPCLRJ
1687  002212  001403                          BEQ     69$
1688  002214  052737  100000  001504          BIS     #BIT15,DUPO.A
1689  002222  105737  001322          69$:    TSTB    TCNFLG
1690  002226  001403                          BEQ     70$
1691  002230  052737  040000  001504          BIS     #BIT14,DUPO.A
1692  002236  112737  000001  001312  70$:    MOVB    #1,SAVACT
1693  002244  113737  001236  001311          MOVB    TEMP1,DUPNUM
1694  002252  113737  001236  001313          MOVB    TEMP1,SAVNUM
```

```
1695  002260  005337  001236          65$:    DEC     TEMP1
1696  002264  001404                          BEQ     64$
1697  002266  000261                          SEC
1698  002270  106137  001312                  ROLB    SAVACT
1699  002274  000771                          BR      65$
1700  002276  113737  001312  001240  64$:    MOVB    SAVACT,TEMP2    ;# OF TIMES
1701  002304  113737  001312  001310          MOVB    SAVACT,DUPACTV
1702  002312  000241                          CLC
1703  002314  106037  001240                  RORB    TEMP2
1704  002320  012700  001500                  MOV     #DUPCR0,R0
1705  002324  012701  001506                  MOV     #DUPCR1,R1
1706  002330  000241                  67$:    CLC
1707  002332  106037  001240                  RORB    TEMP2
1708  002336  103051                          BCC     66$
1709  002340  012011                          MOV     (R0)+,(R1)
1710  002342  062721  000010                  ADD     #10,(R1)+       ;CSR
1711  002346  012011                          MOV     (R0)+,(R1)
1712  002350  062721  000010                  ADD     #10,(R1)+       ;VECTOR
1713  002354  012021                          MOV     (R0)+,(R1)+     ;PARAMETERS
1714  002356  000764                          BR      67$
1715  002360  012700  001500          21$:    MOV     #DUP.MAP,R0     ;SETUP TO CLEAR MAP
1716  002364  005020                  20$:    CLR     (R0)+           ;CLEAR
1717  002366  020027  001560                  CMP     R0,#DUP.END     ;CHECK FOR FINISH
1718  002372  001374                          BNE     20$             ;BR IF MORE TO GO
1719  002374  012700  001500                  MOV     #DUP.MAP,R0     ;SETUP TO DEFAULT
1720  002400  012710  160050                  MOV     #160050,(R0)    ;LOAD CSR
1721  002404  012760  000770  000002          MOV     #770,2(R0)      ;LOAD VECTOR
1722  002412  012760  140026  000004          MOV     #140026,4(R0)   ;LOAD PARAMETERS AND SYNC
1723  002420  112737  000005  001320          MOVB    #5,PRIRTY       ;LOAD PRIORITY
1724  002426  012700  000001                  MOV     #1,R0           ;SAVE CORE THIS WAY
1725  002432  110037  001310                  MOVB    R0,DUPACTV      ;PRESET PROGRAM CONTROLS
1726  002436  110037  001311                  MOVB    R0,DUPNUM       ;DITTO
1727  002442  110037  001312                  MOVB    R0,SAVACT       ;DITTO
1728  002446  110037  001313                  MOVB    R0,SAVNUM       ;DITTO
1729  002452  110037  001322                  MOVB    R0,TCNFLG       ;DITTO
1730  002456  110037  001323                  MOVB    R0,OPCLRJ       ;DITTO
1731  002462                          66$:
1732  002462  104402  006015          16$:    TYPE    ,XHEAD                  ;TYPE HEADER
1733  002466  012737  001500  001236          MOV     #DUP.MAP,TEMP1          ;SET POINTER
1734  002474  017737  176536  001240  5$:     MOV     @TEMP1,TEMP2            ;SET DATA
1735  002502  001406                          BEQ     1$                      ;ALL DONE WITH DATA
1736  002504  104410                          CONVRT                          ;
1737  002506  006044                          XSTATQ                          ;
1738  002510  062737  000002  001236          ADD     #2,TEMP1                ;UPDATE POINTER
1739  002516  000766                          BR      5$
1740  002520  032777  000001  176454  1$:     BIT     #SW00,@SWR
1741  002526  001405                          BEQ     7$
1742  002530  005737  001332                  TST     FLAG
1743  002534  001002                          BNE     7$
1744  002536  000137  002000                  JMP     10$
1745  002542  005037  001332          7$:     CLR     FLAG
1746  002546  005737  000042                  TST     @#42                    ;IS PROGRAM RUNNING UNDER MONITOR
1747  002552  001030                          BNE     3$                      ;BR IF YES
1748  002554  032777  000010  176420          BIT     #SW03,@SWR              ;SELECT SPECIFIC DEVICES??
1749  002562  001424                          BEQ     3$                      ;BR IF NO.
1750  002564  104402  005414                  TYPE    ,MNEW                   ;TYPE THE MESSAGE.
```

```
1751  002570  005000                          CLR     R0              ;ZERO DATA LIGHTS
1752  002572  000000                          HALT                    ;WAIT FOR USER TO TELL WHAT DEVICES TO RUN
1753  002574  127737  176402  001312          CMPB    @SWR,SAVACT     ;IS THE NUMBER VALID?
1754  002602  101404                          BLOS    2$              ;BR IF NUMBER IS OK.
1755  002604  104402  005255                  TYPE    ,MERR3          ;TELL USER OF INVALID NUMBER.
1756  002610  000000                          HALT                    ;STOP EVERY THING.
1757  002612  000776                          BR      .-2             ;RESTART THE PROGRAM AGAIN.
1758  002614  117737  176362  001310  2$:     MOVB    @SWR,DUPACTV    ;GET NEW DEVICE PATTERN
1759  002622  113700  001310                  MOVB    DUPACTV,R0      ;SHOW THE USER WHAT HE SELECTED.
1760  002626  042700  177400                  BIC     #^C<377>,R0     ;USE ONLY LOW BYTE.
1761  002632  000000                          HALT                    ;CONTINUE DYNAMIC SWITCHES.
1762  002634  012700  000300          3$:     MOV     #300,R0         ;PREPARE TO CLEAR THE FLOATING
1763  002640  012701  000302                  MOV     #302,R1         ;VECTOR AREA. 300-776
1764  002644  010120                  4$:     MOV     R1,(R0)+        ;START PUTTING "PC+2 - HALT"
1765  002646  005021                          CLR     (R1)+           ;IN VECTOR AREA.
1766  002650  022021                          CMP     (R0)+,(R1)+     ;POP POINTERS
1767  002652  022700  001000                  CMP     #1000,R0        ;ALL DONE??
1768  002656  001372                          BNE     4$              ;BR IF NO.
1769
1770                                          ;TEST START AND RESTART
1771                                          ;----------------------
1772
1773  002660  012737  000340  177776  .BEGIN: MOV     #340,PS         ;LOCK OUT INTERRUPTS
1774  002666  012706  001150                  MOV     #STACK,SP       ;SET UP STACK
1775  002672  005737  000042                  TST     @#42            ;IS PROGRAM UNDER MONITOR CONTROL
1776  002676  001023                          BNE     2$              ;BR IF YES
1777  002700  032777  000004  176274          BIT     #BIT2,@SWR      ;CHECK FOR LOCK ON TEST
1778  002706  001411                          BEQ     1$              ;BR IF NO LOCK DESIRED.
1779  002710  104402  005313                  TYPE    ,MLOCK          ;TYPE LOCK SELECTED.
1780  002714  012737  000240  003174          MOV     #NOP,TTST       ;ADJUST SCOPE ROUTINE.
1781  002722  012737  000240  003176          MOV     #NOP,TTST+2     ;SET UP TO LOCK
1782  002730  000406                          BR      2$              ;CONTINUE ALONG.
1783  002732  013737  003306  003174  1$:     MOV     BRW,TTST        ;PREPARE NORMAL SCOPE ROUTINE
1784  002740  013737  003310  003176          MOV     BRX,TTST+2      ;LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
1785  002746  012737  006226  001214  2$:     MOV     #CYCLE,RETURN   ;START AT "CYCLE" FIND WHICH DEVICE TO TEST
1786  002754  104402  005203                  TYPE    ,MR             ;TYPE R
1787  002760  000177  176230                  JMP     @RETURN         ;START TESTING
```

```
                                                        ;END OF PASS
                                                        ;TYPE NAME OF TEST
                                                        ;UPDATE PASS COUNT
                                                        ;CHECK FOR EXIT TO ACT-11
                                                        ;RESTART TEST

002764  005037  001234          .EOP:   CLR     LSTERR          ;CLEAR LAST ERROR PC
002770  105037  001341                  CLRB    ERRFLG          ;CLEAR ERROR FLAG
002774  005237  001230                  INC     PASCNT          ;UPDATE PASS COUNT
003000  013777  001230  176172          MOV     PASCNT,@DISPLAY ;DISPLAY PASS COUNT
003006  104402  005157                  TYPE    ,MEPASS         ;TYPE END PASS
003012  104402  005342                  TYPE    ,MCSRX          ;TYPE CSR
003016  104411  003130                  CNVRT   ,XCSR           ;SHOW IT
003022  104402  005350                  TYPE    ,MVECX          ;TYPE VECTOR
003026  104411  003136                  CNVRT   ,XVEC           ;SHOW IT
003032  104402  005356                  TYPE    ,MPASSX         ;TYPE PASSES
003036  104411  003144                  CNVRT   ,XPASS          ;SHOW IT
003042  104402  005367                  TYPE    ,MERRX          ;TYPE ERRORS
003046  104411  003152                  CNVRT   ,XERR           ;SHOW IT
003052  105337  001313                  DECB    SAVNUM          ;ARE ALL DEVICES TESTED?
003056  001017                          BNE     RESTRT          ;BR IF NO.
003060  112737  000377  001343          MOVB    #377,QV.FLG     ;SET THE QUICK VERIFY FLAG.
003066  113737  001311  001313          MOVB    DUPNUM,SAVNUM   ;RESTORE THE COUNT
003074  013701  000042                  MOV     @#42,R1         ;CHECK FOR ACT-11 OR DDP
003100  001406                          BEQ     RESTRT          ;IF NOT, CONTINUE TESTING
003102  000005                          RESET                   ;STOP THE SHOW--CLEAR THE WORLD
003104                          $ENDAD:
003104  004711                          JSR     PC,(R1)
003106  000240                          NOP
003110  000240                          NOP
003112  000240                          NOP
003114  000240                          NOP
003116  012737  006226  001214  RESTRT: MOV     #CYCLE,RETURN
003124  000137  006226                  JMP     CYCLE
003130  000001                  XCSR:   1
003132     006     002                  .BYTE   6,2
003134  001404                          RXCSR
003136  000001                  XVEC:   1
003140     003     002                  .BYTE   3,2
003142  001374                          DUPRVC
003144  000001                  XPASS:  1
003146     006     002                  .BYTE   6,2
003150  001230                          PASCNT
003152  000001                  XERR:   1
003154     006     002                  .BYTE   6,2
003156  001232                          ERRCNT

                                        ;SCOPE LOOP AND INTERATION HANDLER

003160  005037  001234          .SCOPE: CLR     LSTERR          ;CLEAR LAST ERROR PC
003164  010016                          MOV     R0,(SP)         ;SAVE R0 ON STACK
003166  032777  040000  176006          BIT     #BIT14,@SWR     ;LOOP ON TEST?
003174  001407                  TTST:   BEQ     1$              ;BR IF NO (IF LOCK SW01 = 1;THIS LOCATION  = 240)
003176  000437                          BR      3$              ;GO TO 3$       (DITTO)
003200  105777  176000                  TSTB    @TKCSR          ;KYBD DONE?
003204  100034                          BPL     3$              ;BR IF NO (LOCK:  HIT A KEY ON TTY TO GO TO NEXT TEST)
```

```
1844  003206  017700  175774                MOV     @TKDBR,R0       ;CLR DONE BIT
1845  003212  000415                       BR      2$              ;CONTINUE
1846  003214  032777  004000  175760  1$:   BIT     #SW11,@SWR      ;DELETE ITERATION (QUICK PASS)?
1847  003222  001011                       BNE     2$              ;BR IF YES
1848  003224  105737  001343                TSTB    QV.FLG          ;HAS FIRST PASS BEEN COMPLETED?
1849  003230  001406                       BEQ     2$              ;BR IF QUICK VERIFY
1850  003232  005237  001224                INC     LPCNT           ;UPDATE ITERATION COUNTER
1851  003236  023737  001224  001222        CMP     LPCNT,ICOUNT    ;ALL ITERATIONS DONE?
1852  003244  001014                       BNE     3$              ;BR IF NOT YET
1953  003246  105037  001341          2$:   CLRB    ERRFLG          ;PREPARE FOR NEW TEST
1854  003252  005037  001224                CLR     LPCNT           ;START ICOUNT AT ZERO
1855  003256  005037  001220                CLR     LOCK
1856  003262  012737  000050  001222        MOV     #50,ICOUNT      ;RESET ITERATIONS
1857  003270  013737  001216  001214        MOV     NEXT,RETURN     ;GET NEXT TEST
1858  003276  011600                 3$:   MOV     (SP),R0         ;POP R0 OFF STACK
1859  003300  022626                       POP2SP                  ;FAKE AN RTI
1860  003302  000177  175706                JMP     @RETURN         ;GO DO THE TEST
1861  003306  001407                 BRW:  1407
1862  003310  000437                 BRX:  437
1863
1864                                        ;CHECK FOR FREEZE ON CURRENT DATA
1865                                        ;-------------------------------
1866
1867  003312  032777  001000  175662  .SCOP1: BIT     #SW09,@SWR      ;IS SW09=1(SET)?
1868  003320  001405                       BEQ     1$              ;BR IF NOT SET.
1869  003322  005737  001220                TST     LOCK
1870  003326  001402                       BEQ     1$
1871  003330  013716  001220                MOV     LOCK,(SP)       ;GOTO THE ADDRESS IN LOCK.
1872  003334  000002                 1$:   RTI                     ;GO BACK.
1873
1874                                        ;TELETYPE OUTPUT ROUTINE
1875                                        ;----------------------
1876
1877  003336  010546                 .TYPE: MOV     R5,-(SP)        ;SAVE R5 ON THE STACK.
1878  003340  017605  000002                MOV     @2(SP),R5       ;GET ADDRESS OF MESSAGE.
1879  003344  062766  000002  000002        ADD     #2,2(SP)        ;POP OVER ADDRESS.
1880  003352  032777  010000  175622  1$:   BIT     #SW12,@SWR      ;INHIBIT ALL PRINT OUT?
1881  003360  001012                       BNE     3$              ;BR IF NO PRINT OUT WANTED (SW12=1
1882  003362  105715                       TSTB    (R5)            ;IS NUMBER MINUS? (MSB=1(BIT7))
1883  003364  100002                       BPL     2$              ;BR IF NUMBER IS PLUS
1884  003366  104402  005136                TYPE    ,MCRLF          ;TYPE A CR/LF!
1885  003372  105777  175612          2$:   TSTB    @TPCSR          ;TTY READY?
1886  003376  100375                       BPL     2$              ;BR IF NO.
1887  003400  112577  175606                MOVB    (R5)+,@TPDBR    ;PRINT CURRENT CHAR.
1888  003404  001362                       BNE     1$              ;IF NOT ZERO KEEP PRINTING!
1889  003406  012605                 3$:   MOV     (SP)+,R5        ;END OF OUTPUT. RESTORE R5
1890  003410  000002                       RTI                     ;GO HOME
1891                                        ;-------------------------
1892
1893  003412  010346                 .INSTR: MOV    R3,-(SP)        ;SAVE R3 ON STACK
1894  003414  010446                       MOV     R4,-(SP)        ;SAVE R4 ON STACK
1895  003416  017637  000004  003434        MOV     @4(SP),.MSG
1896  003424  062766  000002  000004        ADD     #2,4(SP)
1897  003432  104402                 .INST1: TYPE
1898  003434  000000                 .MSG:   0
1899  003436  012704  006162                MOV     #INBUF,R4
```

```
1900  003442  012703  000007              MOV     #7,R3
1901  003446  105777  175532      1$:     TSTB    @TKCSR
1902  003452  100375                      BPL     1$
1903  003454  117714  175526              MOVB    @TKDBR,(R4)
1904  003460  142714  000200              BICB    #200,(R4)
1905  003464  122427  000015              CMPB    (R4)+,#15
1906  003470  001417                      BEQ     INSTR2
1907  003472  105777  175512      2$:     TSTB    @TPCSR
1908  003476  100375                      BPL     2$
1909  003500  017777  175502  .75504      MOV     @TKDBR,@TPDBR
1910  003506  005303                      DEC     R3
1911  003510  001356                      BNE     1$
1912  003512  012604                      MOV     (SP)+,R4
1913  003514  012603                      MOV     (SP)+,R3
1914  003516  010346              .INSTE: MOV     R3,-(SP)
1915  003520  010446                      MOV     R4,-(SP)
1916  003522  104402  005132              TYPE    ,MQM
1917  003526  000741                      BR      .INST1
1918  003530  012604              INSTR2: MOV     (SP)+,R4        ;RESTORE R4
1919  003532  012603                      MOV     (SP)+,R3        ;RESTORE R3
1920  003534  000002                      RTI
1921
1922                                      ;CONVERT ASCII STRING TO OCTAL
1923                                      ;----------------------------
1924
1925  003536  010546              .PARAM: MOV     R5,-(SP)
1926  003540  010446                      MOV     R4,-(SP)
1927  003542  016605  000004              MOV     4(SP),R5
1928  003546  012537  003726              MOV     (R5)+,LOLIM
1929  003552  012537  003730              MOV     (R5)+,HILIM
1930  003556  012537  003732              MOV     (R5)+,DEVADR
1931  003562  112537  003734              MOVB    (R5)+,LOBITS
1932  003566  112537  003735              MOVB    (R5)+,ADRCNT
1933  003572  010566  000004              MOV     R5,4(SP)
1934  003576  005005              PARAM1: CLR     R5
1935  003600  012704  006162              MOV     #INBUF,R4
1936  003604  122714  000015              CMPB    #15,(R4)
1937  003610  001420                      BEQ     PARERR
1938  003612  121427  000060      1$:     CMPB    (R4),#60
1939  003616  002415                      BLT     PARERR
1940  003620  121427  000067              CMPB    (R4),#67
1941  003624  003012                      BGT     PARERR
1942  003626  142714  000060              BICB    #60,(R4)
1943  003632  152405                      BISB    (R4)+,R5
1944  003634  122714  000015              CMPB    #15,(R4)
1945  003640  001406                      BEQ     LIMITS
1946  003642  006305                      ASL     R5
1947  003644  006305                      ASL     R5
1948  003646  006305                      ASL     R5
1949  003650  000760                      BR      1$
1950  003652  104404              PARERR: INSTER
1951  003654  000750                      BR      PARAM1
1952
1953                                      ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
1954                                      ;--------------------------------------
1955
```

```
1956  003656  020537  003730          LIMITS: CMP     R5,HILIM
1957  003662  101373                          BHI     PARERR
1958  003664  020537  003726                  CMP     R5,LOLIM
1959  003670  103770                          BLO     PARERR
1960  003672  133705  003734                  BITB    LOBITS,R5
1961  003676  001365                          BNE     PARERR
1962
1963                                          ;STORE NUMBER AT SPECIFIED ADDRESS
1964
1965  003700  013704  003732                  MOV     DEVADR,R4
1966  003704  010524                  1$:     MOV     R5,(R4)+
1967  003706  062705  000002                  ADD     #2,R5
1968  003712  105337  003735                  DECB    ADRCNT
1969  003716  001372                          BNE     1$
1970  003720  012604                          MOV     (SP)+,R4
1971  003722  012605                          MOV     (SP)+,R5
1972  003724  000002                          RTI
1973  003726  000000                  LOLIM:  0
1974  003730  000000                  HILIM:  0
1975  003732  000000                  DEVADR: 0
1976  003734  000000                  LOBITS: 0
1977          003735                  ADRCNT=LOBITS+1
1978
1979                                          ;SAVE PC OF TEST THAT FAILED AND R0-R5
1980                                          ;-------------------------------------
1981
1982  003736  016637  000004  001266  .SAV05: MOV     4(SP),SAVPC     ;SAVE R7 (PC)
1983
1984                                          ;SAVE R0-R5
1985
1986  003744  010537  001262          SV05:   MOV     R5,SAVR5        ;SAVE R5
1987  003750  010437  001260                  MOV     R4,SAVR4        ;SAVE R4
1988  003754  010337  001256                  MOV     R3,SAVR3        ;SAVE R3
1989  003760  010237  001254                  MOV     R2,SAVR2        ;SAVE R2
1990  003764  010137  001252                  MOV     R1,SAVR1        ;SAVE R1
1991  003770  010037  001250                  MOV     R0,SAVR0        ;SAVE R0
1992  003774  000002                          RTI                     ;LEAVE.
1993
1994                                          ;RESTORE R0-R5
1995
1996  003776  013700  001250          .RES05: MOV     SAVR0,R0        ;RESTORE R0
1997  004002  013701  001252                  MOV     SAVR1,R1        ;RESTORE R1
1998  004006  013702  001254                  MOV     SAVR2,R2        ;RESTORE R2
1999  004012  013703  001256                  MOV     SAVR3,R3        ;RESTORE R3
2000  004016  013704  001260                  MOV     SAVR4,R4        ;RESTORE R4
2001  004022  013705  001262                  MOV     SAVR5,R5        ;RESTORE R5
2002  004026  000002                          RTI                     ;LEAVE
2003
2004
2005                                          ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
2006                                          ;-----------------------------------------------------
2007
2008  004030  104402  005136          .CONVR: TYPE    ,MCRLF
2009  004034  010046                  .CNVRT: MOV     R0,-(SP)
2010  004036  010146                          MOV     R1,-(SP)
2011  004040  010346                          MOV     R3,-(SP)
```

K04


```
 2012  004042  010446                          MOV     R4,-(SP)
 2013  004044  010546                          MOV     R5,-(SP)
 2014  004046  017601  000012                  MOV     @12(SP),R1
 2015  004052  062766  000002  000012          ADD     #2,12(SP)
 2016  004060  012137  004234                  MOV     (R1)+,WRDCNT
 2017  004064  112137  004236          1$:     MOVB    (R1)+,CHRCNT
 2018  004070  112137  004237                  MOVB    (R1)+,SPACNT
 2019  004074  013137  004240                  MOV     @(R1)+,BINWRD
 2020  004100  013704  004240          2$:     MOV     BINWRD,R4
 2021  004104  113705  004236                  MOVB    CHRCNT,R5
 2022  004110  012700  006056                  MOV     #TEMP,R0
 2023  004114  010403                  3$:     MOV     R4,R3
 2024  004116  042703  177770                  BIC     #177770,R3
 2025  004122  062703  000060                  ADD     #060,R3
 2026  004126  110320                          MOVB    R3,(R0)+
 2027  004130  000241                          CLC
 2028  004132  006004                          ROR     R4
 2029  004134  000241                          CLC
 2030  004136  006004                          ROR     R4
 2031  004140  000241                          CLC
 2032  004142  006004                          ROR     R4
 2033  004144  005305                          DEC     R5
 2034  004146  001362                          BNE     3$
 2035  004150  012703  006120                  MOV     #MDATA,R3
 2036  004154  114023                  4$:     MOVB    -(R0),(R3)+
 2037  004156  105337  004236                  DECB    CHRCNT
 2038  004162  001374                          BNE     4$
 2039  004164  105737  004237                  TSTB    SPACNT
 2040  004170  001405                          BEQ     6$
 2041  004172  112723  000040          5$:     MOVB    #040,(R3)+
 2042  004176  105337  004237                  DECB    SPACNT
 2043  004202  001373                          BNE     5$
 2044  004204  105013                  6$:     CLRB    (R3)
 2045  004206  104402  006120                  TYPE    ,MDATA
 2046  004212  005337  004234                  DEC     WRDCNT
 2047  004216  001322                          BNE     1$
 2048  004220  012605                          MOV     (SP)+,R5
 2049  004222  012604                          MOV     (SP)+,R4
 2050  004224  012603                          MOV     (SP)+,R3
 2051  004226  012601                          MOV     (SP)+,R1
 2052  004230  012600                          MOV     (SP)+,R0
 2053  004232  000002                          RTI
 2054  004234  000000                  WRDCNT: 0
 2055  004236  000000                  CHRCNT: 0
 2056          004237                  SPACNT=CHRCNT+1
 2057  004240  000000                  BINWRD: 0
 2058
 2059
 2060                                          ;COMPARE THE FIRST CHARACTER IN THE TELETYPE INPUT
 2061                                          ;BUFFER TO THE CHARACTERS "N" AND "Y".
 2062                                          ;IF THE CHARACTER IS "N" CLEAR THE FLAG
 2063                                          ;IF THE CHARACTER IS "Y" SET THE FLAG
 2064
 2065  004242  017605  000000          .SETFLG:MOV     @(SP),R5
 2066  004246  042737  000040  006162          BIC     #40,INBUF
 2067  004254  122737  000116  006162          CMPB    #'N,INBUF       ;IS IT "N" ?
```

```
2068  004262  001002                          BNE     1$
2069  004264  105015                          CLRB    (R5)    ;000
2070  004266  000406                          BR      2$
2071  004270  122737  000131  006162  1$:     CMPB    #'Y,INBUF       ;IS IT "Y" ?
2072  004276  001005                          BNE     3$
2073  004300  112715  177777                  MOVB    #-1,(R5)        ;377
2074  004304  062716  000002          2$:     ADD     #2,(SP)
2075  004310  000002                          RTI
2076  004312  104404                  3$:     INSTER  ;RETRY
2077  004314  000752                          BR      .SETFLG
2078
2079
2080                                          ;TRAP DISPATCH SERVICE
2081                                          ;ARGUMENT OF TRAP IS EXTRACTED
2082                                          ;AND USED AS OFFSET TO OBTAIN POINTER
2083                                          ;TO SELECTED SUBROUTINE
2084
2085  004316  011646                  .TRPSR: MOV     (SP),-(SP)              ;GET PC OF RETURN
2086  004320  162716  000002                  SUB     #2,(SP)                 ;=PC OF TRAP
2087  004324  017616  000000                  MOV     @(SP),(SP)              ;GET TRP
2088  004330  006316                  TRPOK:  ASL     (SP)                    ;MULTIPLY TRAP ARG BY 2
2089  004332  042716  177001                  BIC     #177001,(SP)            ;CLEAR UNWANTED BITS
2090  004336  062716  001344                  ADD     #.TRPTAB,(SP)           ;POINTER TO SUBROUTINE ADDRESS
2091  004342  017616  000000                  MOV     @(SP),(SP)              ;SUBROUTINE ADDRESS
2092  004346  000136                          JMP     @(SP)+                  ;GO TO SUBROUTINE
2093
2094                                          ;ERROR HANDLER
2095                                          ;-------------
2096
2097  004350  032777  010000  174624  .HLT:   BIT     #SW12,@SWR      ;BELL ON ERROR?
2098  004356  001406                          BEQ     XBX             ;BR IF NO BELL
2099  004360  105777  174624                  TSTB    @TPCSR          ;TTY READY.
2100  004364  100003                          BPL     XBX             ;DON'T WAIT IF TTY NOT READY.
2101  004366  112777  000207  174616          MOVB    #207,@TPDBR     ;PUSH A BELL AT THE TTY.
2102  004374  032777  020000  174600  XBX:    BIT     #SW13,@SWR      ;DELETE ERROR PRINT OUT?
2103  004402  001105                          BNE     HALTS           ;BR IF NO PRINT OUT WANTED.
2104  004404  021637  001234                  CMP     (SP),LSTERR     ;WAS THIS ERROR FOUND LAST TIME?
2105  004410  001404                          BEQ     1$              ;BR IF YES
2106  004412  011637  001234                  MOV     (SP),LSTERR     ;RECORD BEING HERE
2107  004416  105037  001341                  CLRB    ERRFLG          ;PREPARE HEADER
2108  004422  104406                  1$:     SAV05                   ;SAVE ALL PROC REGISTERS
2109  004424  011605                          MOV     (SP),R5         ;GET THE PC OF ERROR
2110  004426  162705  000002                  SUB     #2,R5           ;GET ADDRESS OF TRAP CALL
2111  004432  011504                          MOV     (R5),R4         ;GET HLT INSTRUCTION
2112  004434  006304                          ASL     R4              ;MULT BY TWO
2113  004436  061504                          ADD     (R5),R4         ;DOUBLE IT
2114  004440  006304                          ASL     R4              ;MULT AGAIN
2115  004442  042704  177001                  BIC     #177001,R4      ;CLEAR JUNK
2116  004446  062704  027732                  ADD     #.ERRTAB,R4     ;GET POINTER
2117  004452  012437  004566                  MOV     (R4)+,ERRMSG    ;GET ERROR MESSAGE
2118  004456  012437  004600                  MOV     (R4)+,DATAHD    ;GET DATA HEADRER
2119  004462  011437  004612                  MOV     (R4),DATABP     ;GET DATA TABLE
2120  004466  105737  001341                  TSTB    ERRFLG          ;TYPE HEADREER
2121  004472  001403                          BEQ     TYPMSG          ;BR IF YES
2122  004474  005737  004612                  TST     DATABP          ;DOES DATA TABLE EXIST?
2123  004500  001040                          BNE     TYPDAT          ;BR IF YES.
```

```
2124  004502  104402  005136          TYPMSG: TYPE    ,MCRLF
2125  004506  104402  005136                  TYPE    ,MCRLF
2126  004512  005737  001220                  TST     LOCK
2127  004516  001402                          BEQ     1$
2128  004520  104402  005412                  TYPE    ,MASTEK
2129  004524  104402  005400          1$:     TYPE    ,MTSTN
2130  004530  104411  005000                  CNVRT   ,XTSTN          ;SHOW IT
2131  004534  104402  005467                  TYPE    ,MERRPC         ;TYPE PC.
2132  004540  104411  004772                  CNVRT   ,ERTABO         ;SHOW IT
2133  004544  104402  005136                  TYPE    ,MCRLF          ;GIVE A CR/LF
2134  004550  112737  177777  001341          MOVB    #-1,ERRFLG      ;NO MORE HEADER UNLESS NO DATA TABLE.
2135  004556  005737  004566                  TST     ERRMSG          ;IS THERE AN ERROR MESSAGE?
2136  004562  001402                          BEQ     WRKO.FM         ;BR IF NO.
2137  004564  104402                          TYPE                    ;TYPE
2138  004566  000000                  ERRMSG: 0                       ;    ERROR MESSAGE
2139  004570                          WRKO.FM:                        ;
2140  004570  005737  004600                  TST     DATAHD          ;DATA HEADER?
2141  004574  001402                          BEQ     TYPDAT          ;BR IF NO
2142  004576  104402                          TYPE                    ;TYPE
2143  004600  000000                  DATAHD: 0                       ;    DATA HEADER
2144  004602  005737  004612          TYPDAT: TST     DATABP          ;DATA TABLE?
2145  004606  001402                          BEQ     RESREG          ;BR IF NO.
2146  004610  104410                          CONVRT                  ;SHOW
2147  004612  000000                  DATABP: 0                       ;    DATA TABLE
2148  004614  104407                  RESREG: RESO5                   ;RESTORE PROC REGISTERS
2149  004616  022737  003104  000042  HALTS:  CMP     #$ENDAD,@#42    ;IF ACT-11 AUTO MODE--HALT!!
2150  004624  001403                          BEQ     1$
2151  004626  005777  174350                  TST     @SWR            ;HALT ON ERROR?
2152  004632  100035                          BPL     EXITER          ;BR IF NO HALT ON ERROR
2153  004634  010046                  1$:     PUSHRO                  ;SAVE RO
2154  004636  016600  000002                  MOV     2(SP),RO        ;SHOW ERROR PC IN DATA LIGHTS
2155  004642  013746  000004                  MOV     4,-(SP)         ;SAVE OLD TRAP
2156  004646  013746  000006                  MOV     6,-(SP)
2157  004652  012737  004710  000004          MOV     #22$,4          ;FORCE HALT IF TIME-OUT
2158  004660  012737  000340  000006          MOV     #340,6          ;WHEN REFERENCING TXCSR
2159  004666  042777  014000  174516          BIC     #SYSTST!MEXT,@TXCSR
2160  004674  000000                          HALT                    ;HALT
2161  004676  012637  000006                  MOV     (SP)+,6         ;RESTORE TRAP
2162  004702  012637  000004                  MOV     (SP)+,4
2163  004706  000406                          BR      33$
2164  004710  000000                  22$:    HALT                    ;HALT
2165  004712  022626                          CMP     (SP)+,(SP)+     ;POP STACK
2166  004714  012637  000006                  MOV     (SP)+,6         ;RESTORE TRAP
2167  004720  012637  000004                  MOV     (SP)+,4
2168  004724  012600                  33$:    POPRO                   ;GET RO
2169  004726  005237  001232          EXITER: INC     ERRCNT          ;UPDATE ERROR COUNT
2170  004732  032777  000400  174242          BIT     #SW08,@SWR      ;GOTO TOP OF TEST?
2171  004740  001007                          BNE     1$              ;BR IF YES
2172  004742  032777  002000  174232          BIT     #SW10,@SWR      ;GOTO NEXT TEST?
2173  004750  001407                          BEQ     2$              ;BR IF NO
2174  004752  013737  001216  001214          MOV     NEXT,RETURN     ;SET FOR NEXT TEST
2175  004760  012706  001150          1$:     MOV     #STACK,SP       ;RESET SP
2176  004764  000177  174224                  JMP     @RETURN         ;GOTO SPECIFIED TEST
2177  004770  000002                  2$:     RTI                     ;RETURN
2178  004772  000001                  ERTABO: 1
2179  004774     006     002                  .BYTE   6,2
```

```
 2180  004776  001266                          SAVPC
 2181  005000  000001                  XTSTN:  1
 2182  005002     003     002                  .BYTE   3,2
 2183  005004  001226                          TSTNO
 2184  005006  017600  000000          .PKCLK: MOV     @(SP),R0        ;GET THE # OF TICKS TO POKE
 2185  005012  062716  000002                  ADD     #2,(SP)         ;POP OVER THE #
 2186  005016                          1$:
 2187  005016  052777  020000  174366          BIS     #CLK,@TXCSR     ;POKE CLOCK UP
 2188  005024  005300                          DEC     R0              ;ARE WE DONE?
 2189  005026  001405                          BEQ     2$              ;YES-GO TO 2$
 2190  005030  042777  020000  174354          BIC     #CLK,@TXCSR             ;POKE CLOCK DOWN
 2191  005036  005300                          DEC     R0              ;ARE WE DONE?
 2192  005040  001366                          BNE     1$              ;NO-REPEAT
 2193  005042  000002                  2$:     RTI                     ;RETURN
 2194
 2195
 2196                                  ;WAIT ROUTINE
 2197  005044  000240                  $MALL:  NOP                     ;STALL
 2198  005046  000207                          RTS     PC              ;RETURN
 2199
 2200                                  ;POWER FAIL ROUTINE
 2201
 2202  005050  012737  005060  000024  .PFAIL: MOV     #PWRUP,24       ;LOAD PFAIL VECTOR FOR POWER UP
 2203  005056  000000                          HALT                    ;
 2204  005060  000005                  PWRUP:  RESET                   ;WAIT TTY TO COME UP
 2205  005062  012706  001150                  MOV     #STACK,SP       ;REINIT STACK POINTER
 2206  005066  012737  005050  000024          MOV     #.PFAIL,24      ;LOAD PFAIL VECTOR FOR POWER DOWN
 2207  005074  104402                          TYPE
 2208  005076  005141                          MPOWER
 2209  005100  000177  174110                  JMP     @RETURN
 2210                                  ;CLRVEC,ROUTINE TO FILL COMMUNICATION VECTOR AREA WITH .+2,HALT
 2211
 2212  005104  012702  000300          CLRVEC: MOV     #300,R2         ;R2 COMM VECTOR AREA ADRS
 2213  005110  012701  000302                  MOV     #302,R1         ;INIT R1 WITH ADRS OF HALT
 2214  005114  010122                  1$:     MOV     R1,(R2)+        ;MOV .+2 TO PC
 2215  005116  005022                          CLR     (R2)+           ;MOV HALT TO PC
 2216  005120  022121                          CMP     (R1)+,(R1)+     ;INC TO NEXT VECTOR AREA
 2217  005122  022701  000776                  CMP     #776,R1         ;END OF VECTOR AREA
 2218  005126  001372                          BNE     1$              ;NO
 2219  005130  000207                          RTS     PC              ;RETURN
 2220
 2221
 2222
 2223  005132  020040  000077          MQM:    .ASCIZ  /  ?/
  (2)  005136  005015     000          MCRLF:  .ASCIZ  <15><12>
  (2)  005141     377  053520  020122  MPOWER: .ASCIZ  <377>/PWR FAILED. /
  (2)  005157     015  042777  042116  MEPASS: .ASCIZ  <15><377>/END PASS DZDP88  /
  (2)  005203     377  000122          MR:     .ASCIZ  <377>/R/
  (2)  005206  050377  047522  051107  MERR2:  .ASCIZ  <377>/PROGRAM INDICATES NO DEVICES PRESENT./
  (2)  005255     377  047111  052523  MERR3:  .ASCIZ  <377>/INSUFFICIENT DATA!/
  (2)  005301     377  042524  052123  MTSTPC: .ASCIZ  <377>/TEST PC-/
  (2)  005313     377  047514  045503  MLOCK:  .ASCIZ  <377>/LOCK ON SELECTED TEST
  (2)  005342  051503  035122  000040  MCSRX:  .ASCIZ   /CSR: /
  (2)  005350  042526  035103  000040  MVECX:  .ASCIZ  /VEC: /
  (2)  005356  040520  051523  051505  MPASSX: .ASCIZ  /PASSES: /
  (2)  005367     105  051122  051117  MERRX:  .ASCIZ  /ERRORS: /
```

```
 (2)  005400 042524 052123 047040 MTSTN:  .ASCIZ  /TEST NO: /
 (2)  005412 000052               MASTEK: .ASCIZ  /*/
 (2)  005414 051777 052105 051440 MNEW:   .ASCIZ  <377>/SET SWITCH REG TO DUP11'S DESIRED ACTIVE./
 (2)  005467    120 035103 000040 MERRPC: .ASCIZ  /PC: /
 (2)  005474 051377 041505 041440 MCSR:   .ASCIZ  <377>/REC CSR ADRS /
 (2)  005513    377 042526 020103 MVEC:   .ASCIZ  <377>/VEC ADRS /
 (2)  005526 044777 020123 044124 MJMPR:  .ASCIZ  <377>/IS THE OPTIONAL CLR JMPR IN?  (Y OR N)   /
 (2)  005601    377 051511 052040 MTCN:   .ASCIZ  <377>/IS THE H325 CONNECTOR ON? (Y OR N)   /
 (2)  005650 021777 047440 020106 MTOTAL: .ASCIZ  <377>/# OF DUP'S (IN OCTAL)    /
 (2)  005703    377 051120 047511 MPAR:   .ASCIZ  <377>/PRIORITY (4 TO 7) /
 (2)  005727    377 042523 020103 MSTJM:  .ASCIZ  <377>/SEC TX JMPR IN? (Y OR N) /
 (2)  005762 051777 041505 051040 MSRJM:  .ASCIZ  <377>/SEC RX JMPR IN? (Y OR N) /
 (2)  006015    377 040515 020120 XHEAD:  .ASCIZ  <377>/MAP OF DUP11 STATUS/<377>
 (2)         006044               .EVEN
 (2)  006044 000002               XSTATQ: 2
2224  006046    006      003              .BYTE   6,3
2225  006050 001236                      TEMP1
2226  006052    006      002              .BYTE   6,2
2227  006054 001240                      TEMP2
2228                               .EVEN
2229
2230  006056 000000               TEMP:   0
2231         006120               .=.+40
2232  006120 000000               MDATA:  0
2233         006162               .=.+40
2234  006162 000000               INBUF:  0
2235         006224               .=.+10
2236  006224 000001               TRP.PC: .BLKW 1
2237
```

```
2238                                                    ;
2239                                                    ;ROUTINE USED TO "CYCLE" THROUGH UP TO EIGHT DUP11'S
2240                                                    ;THIS ROUTINE SETS UP THE CONTROL ADDRESS FOR THE DIAGNOSTIC
2241                                                    ;AND RUNS THE SPECIFIED DUP11'S.   THIS ROUTINE *MUST*
2242                                                    ;BE RUN FIRST BEFORE ENTERING THE DIAGNOSTIC FOR THE
2243                                                    ;SETUP NECESSARY.
2244                                                    ;
2245
2246
2247  006226  105737  001310          CYCLE:  TSTB    DUPACTV         ;ARE ANY DUP11'S TO BE TESTED?
2248  006232  001004                          BNE     1$              ;BR IF OK.
2249  006234  104402  005206                  TYPE    ,MERR2          ;NO DUP11'S SELECTED!!
2250  006240  000000                          HALT                    ;STOP THE SHOW.
2251  006242  000776                          BR      .-2             ;DISQUALIFY CONT. SW.
2252  006244  133737  001314  001310  1$:     BITB    RUN,DUPACTV     ;IS THIS ONE "ACTIVE"
2253  006252  001020                          BNE     2$              ;BR IF GOOD ONE FOUND.
2254  006254  000241                          CLC                     ;CLEAR PROC. CARRY BIT.
2255  006256  106137  001314                  ROLB    RUN             ;UPDATE POINTER
2256  006262  105537  001314                  ADCB    RUN             ;CATCH CARRY FROM RUN
2257  006266  062737  000006  001316          ADD     #6,CREAM        ;UPDATE ADDRESS POINTER.
2258  006274  022737  001560  001316          CMP     #DUP.END,CREAM
2259  006302  001360                          BNE     1$
                                                        ;KEEP GOING; NOT ALL TESTED FOR.
2260  006304  012737  001500  001316          MOV     #DUP.MAP,CREAM  ;RESET ADDRESS POINTER.
2261  006312  000754                          BR      1$              ;KEEP LOOKING FOR ACTIVE DUP11
2262  006314  000241                  2$:     CLC                     ;CLEAR PROC. CARRY.
2263  006316  106137  001314                  ROLB    RUN             ;UPDATE POINTER.
2264  006322  105537  001314                  ADCB    RUN             ;CATCH CARRY.
2265  006326  013700  001316                  MOV     CREAM,R0        ;GET ADDRESS POINTER.
2266  006332  062737  000006  001316          ADD     #6,CREAM        ;UPDATE.
2267  006340  022737  001560  001316          CMP     #DUP.END,CREAM
2268                                                  ;ALL DONE?
2269  006346  001003                          BNE     3$              ;BR IF NO.
2270  006350  012737  001500  001316          MOV     #DUP.MAP,CREAM  ;RESTORE POINTER.
2271  006356  012037  001404          3$:     MOV     (R0)+,RXCSR     ;LOAD SYSTEM CTRL. REG
2272  006362  012037  001374                  MOV     (R0)+,DUPRVC    ;LOAD VECTOR
2273  006366  012037  001434                  MOV     (R0)+,LOO.OO    ;GET PARAMETERS
2274  006372  012700  000002                  MOV     #2,R0           ;SAVE CORE THIS WAY!
2275  006376  013737  001404  001424          MOV     RXCSR,HUPRCR    ;GET CONTROL REG HIGH BYTE
2276  006404  005237  001424                  INC     HUPRCR          ;GOT IT
2277  006410  013737  001424  001406          MOV     HUPRCR,RXDBUF   ;GET RX CONTROL REG BUFFER
2278  006416  005237  001406                  INC     RXDBUF          ;GOT IT
2279  006422  013737  001406  001416          MOV     RXDBUF,DUPSEC   ;GOT SECONDARY REG SELECT REG
2280  006430  013737  001406  001410          MOV     RXDBUF,PARCSR   ;GOT PARAMETER STATUS REGISTER
2281  006436  013737  001406  001422          MOV     RXDBUF,HUPRBF   ;GET RX BUFFER HIGH BYTE
2282  006444  005237  001422                  INC     HUPRBF          ;GOT IT
2283  006450  013737  001422  001420          MOV     HUPRBF,HUPPSR   ;GOT PAR STATUS REG HIGH BYTE
2284  006456  013737  001420  001412          MOV     HUPPSR,TXCSR    ;GET TX CONTROL REGISTER
2285  006464  005237  001412                  INC     TXCSR           ;GOT IT
2286  006470  013737  001412  001430          MOV     TXCSR,HUPTCR    ;GET TX CONTROL REG HIGH BYTE
2287  006476  005237  001430                  INC     HUPTCR          ;GOT IT
2288  006502  013737  001430  001414          MOV     HUPTCR,TXDBUF   ;BET TX BUFFER
2289  006510  005237  001414                  INC     TXDBUF          ;GOT IT
2290  006514  013737  001414  001426          MOV     TXDBUF,HUPTBF   ;GET TX BUFFER HIGH BYTE
2291  006522  005237  001426                  INC     HUPTBF          ;GOT IT
2292
2293  006526  013737  001374  001376          MOV     DLPRVC,DUPRPS   ;RX VECTOR
```

```
2294  006534  060037  001376                  ADD     R0,DUPRPS       ;RX PRIORITY LEVEL
2295  006540  013737  001376  001400          MOV     DUPRPS,DUPTVC
2296  006546  060037  001400                  ADD     R0,DUPTVC       ;TX VECTOR
2297  006552  013737  001400  001402          MOV     DUPTVC,DUPTPS
2298  006560  060037  001402                  ADD     R0,DUPTPS       ;TX PRIORITY LEVEL
2299
2300
2301  006564  012700  001434                  MOV     #LOO.00,R0      ;LOAD STAUS 00-00
2302  006570  012701  001432                  MOV     #MASK.A,R1      ;PREPARE MASK.
2303  006574  012702  001433                  MOV     #CLK.A,R2       ;PREPARE CLOCKS
2304  006600  004737  006744                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
2305  006604  005737  000042                  TST     @#42
2306  006610  001050                          BNE     4$
2307  006612  032777  000002  172362          BIT     #SW01,@SWR      ;IF SW01=1,GET STARTING TEST #
2308  006620  001444                          BEQ     4$
2309  006622  104402  005136          7$:     TYPE    ,MCRLF
2310  006626  104403                          INSTR   ;OUTPUT MESSAGE & GET INPUT STRING
2311  006630  005400                          MTSTN   ;MESSAGE
2312  006632  104405                          PARAM   ;CONVERT STRING
2313  006634  000001                          1       ;LOW LIMIT
2314  006636  001000                          1000    ;HIGH LIMIT
2315  006640  001226                          TSTNO   ;STORE AT THIS LOCATION
2316  006642     000                  .BYTE   0       ;MASK
2317  006643     001                  .BYTE   1       ;HOW MANY TIMES + 2
2318  006644  012700  007160                  MOV     #TST1,R0
2319  006650  022710  012737          5$:     CMP     #12737,(R0)
2320  006654  001017                          BNE     6$
2321  006656  023760  001226  000002          CMP     TSTNO,2(R0)
2322  006664  001013                          BNE     6$
2323  006666  022760  001226  000004          CMP     #TSTNO,4(R0)
2324  006674  001007                          BNE     6$
2325  006676  010037  001214                  MOV     R0,RETURN       ;SAVE PC
2326  006702  104402  005136                  TYPE    ,MCRLF
2327  006706  104402  005203                  TYPE    ,MR
2328  006712  000412                          BR      8$
2329  006714  005720                  6$:     TST     (R0)+
2330  006716  020027  026316                  CMP     R0,#TLAST+10
2331  006722  001352                          BNE     5$
2332  006724  104402  005132                  TYPE    ,MQM
2333  006730  000734                          BR      7$
2334
2335  006732  012737  007160  001214  4$:     MOV     #TST1,RETURN    ;PREPARE RETURN ADDRESS
2336  006740  000177  172250          8$:     JMP     @RETURN         ;GO START TESTING.
2337
2338  006744  011003                  FIX.00: MOV     (R0),R3         ;GET PARAMETERS.
2339  006746  000207                  5$:     RTS     PC              ;
```

```
2340
2341
2342
2343  006750  012104                  ACC:    MOV     (R1)+,R4        ;GET THE FLAG FOR # OF CLOCK TICKS
2344  006752  104412  000002          1$:     PKCLK   ,2
2345  006756  000241                          CLC                     ;PUT CARRY IN A KNOWN STATE
2346  006760  032777  040000  172424          BIT     #MTDATA,@TXCSR  ;FIND OUT IF BIT IS A 1 OR 0
2347  006766  001401                          BEQ     .+4             ;BR IF 0
2348  006770  000261                          SEC                     ;SET THE BIT
2349  006772  006004                          ROR     R4              ;PICK UP CARRY AND PUSH INTO R4
2350  006774  103366                          BCC     1$              ;BRANCH IF MORE TO GO
2351  006776  000201                          RTS     R1
2352  007000  005037  001246          ABRT:   CLR     TEMP5
2353  007004  012137  001244                  MOV     (R1)+,TEMP4     ;GET THE # OF ABORTS TO DO
2354  007010  104412  000002          1$:     PKCLK   ,2      ;POKE OUT A BIT
2355  007014  032777  040000  172370          BIT     #MTDATA,@TXCSR  ;CHECK MAINT DATA OUT
2356  007022  001001                          BNE     2$              ;BRANCH IF IT IS A ONE
2357  007024  104013                          HLT     13              ;OUTPUT WAS A ZERO--NG
2358  007026  005237  001246          2$:     INC     TEMP5           ;INC THE # OF BITS OUTPUT
2359  007032  022737  000010  001246          CMP     #8.,TEMP5       ;IS THE CHARACTER DONE?
2360  007040  001363                          BNE     1$              ;BRANCH IF NOT DONE
2361  007042  005337  001244                  DEC     TEMP4           ;LOWER THE #TO DO
2362  007046  001360                          BNE     1$              ;BRANCH IF NOT DONE
2363  007050  000201                          RTS     R1
2364
2365  007052  012137  001244          FLG:    MOV     (R1)+,TEMP4     ;GET THE # OF FLAGS
2366  007056  104412  000002          64$:    PKCLK   ,2              ;POKE OUT THE FIRST BIT OF THE FLAG
2367  007062  032777  040000  172322          BIT     #MTDATA,@TXCSR  ;CHECK MAINT DATA OUT
2368  007070  001401                          BEQ     65$             ;BRANCH IF 0
2369  007072  104012                          HLT     12              ;DUP FAILED TO CLOCK OUT FIRST BIT
2370  007074  005037  001246          65$:    CLR     TEMP5           ;SETUP FOR 1'S OUTPUT
2371  007100  104412  000002          1$:     PKCLK   ,2              ;CONTINUE TO POKE OUT BITS
2372  007104  032777  040000  172300          BIT     #MTDATA,@TXCSR  ;TEST MAINT DATA OUT
2373  007112  001001                          BNE     2$              ;BRANCH IF A 1
2374  007114  104013                          HLT     13              ;DUP FAILED TO CLOCK A ONE
2375  007116  005237  001246          2$:     INC     TEMP5           ;KEEP UP WITH THE # OF 1'S OUTPUT
2376  007122  022737  000006  001246          CMP     #6,TEMP5        ;ARE WE DONE WITH SIX ONES?
2377  007130  001363                          BNE     1$              ;NO-BRANCH BACK
2378  007132  104412  000002                  PKCLK   ,2              ;YES,OUTPUT THE LAST 0
2379  007136  032777  040000  172246          BIT     #MTDATA,@TXCSR  ;CHECK IT
2380  007144  001401                          BEQ     3$              ;BRANCH IF A 0
2381  007146  104012                          HLT     12              ;LAST BIT OF FLAG WAS NOT CORECT
2382  007150  005337  001244          3$:     DEC     TEMP4           ;ARE WE DONE WITH FLAGS?
2383  007154  001340                          BNE     64$             ;BR IF NO
2384  007156  000201                          RTS     R1
2385
```

```
                                        ;*************************** TEST 1 ****************************
                                        ;*VERIFY THAT ADDRESSING DEVICE DOES *NOT* CAUSE
                                        ;*A TIME-OUT TRAP.
                                        ;;***************************************************************

                                ;;************
                                ;            *
                                ; TEST 1
                                ;            *
                                ;;************
                                ;;***************************************************************************
007160  012737  000001  001226  TST1:   MOV     #1,@#TSTNO
007166  012737  007264  001216          MOV     #TST2,NEXT
007174  012737  007226  001220          MOV     #1$,LOCK
007202  012700  000004                  MOV     #4,R0           ;SET FOR MAX. 4 PRI. REGISTERS
007206  013701  001404                  MOV     RXCSR,R1        ;GET FIRST PRI. ADDRESS
007212  012737  007256  000004          MOV     #2$,4           ;SET FOR TIME-OUT TRAP.
007220  012737  000340  000006          MOV     #340,6          ;SAFE GUARD.
007226  005711                  1$:     TST     (R1)            ;REFERENCE THE ADDRESS.
007230  104401                          SCOP1                   ;IF SW09=1; GOTO 1$
007232  062701  000002                  ADD     #2,R1           ;UPDATE TO NEXT ADDRESS.
007236  005300                          DEC     R0              ;ARE ALL ADDRESS CHECKED?
007240  001372                          BNE     1$              ;BR IF NO.
007242  012737  000006  000004          MOV     #6,4            ;RESET TRAP ZONE.
007250  005037  000006                  CLR     @#6             ;
007254  104400                          SCOPE                   ;SCOPE THIS TEST
007256  011602                  2$:     MOV     (SP),R2         ;SAVE THE TRAP PC
007260  104001                          HLT     1               ;REPORT TIME-OUT TRAP
007262  000002                          RTI                     ;RETURN TO MAIN PROGRAM

                                        ;*************************** TEST 2 ****************************
                                        ;*PRIMARY REGISTER ADDRESSING TEST
                                        ;*LOAD EACH PRIMARY REGISTER WITH A
                                        ;*DIFFERENT NUMBER AND VERIFY EACH
                                        ;*WAS INDIVIDUALLY ADDRESSED
                                        ;;***************************************************************

                                ;;************
                                ;            *
                                ; TEST 2
                                ;            *
                                ;;************
                                ;;***************************************************************************
007264  012737  000002  001226  TST2:   MOV     #2,@#TSTNO
007272  012737  007442  001216          MOV     #TST3,NEXT
007300  012737  007324  001220          MOV     #1$,LOCK
007306  012700  007432                  MOV     #3$,R0          ;SET THE TABLE POINTER
007312  013703  001404                  MOV     RXCSR,R3        ;SET THE DUP HARDWARE POINTER
007316  005013                          CLR     (R3)            ;CLR THE REGISTER BEFORE STARTING
007320  012702  000004                  MOV     #4,R2           ;SET FOR 4 PRIMARY REGISTERS
007324  011005                  1$:     MOV     (R0),R5         ;SET "EXPECTED"
007326  010513                          MOV     R5,(R3)         ;WRITE "EXPECTED" TO THE REGISTER
007330  011304                          MOV     (R3),R4         ;READ THE REGISTER BACK
007332  042704  000200                  BIC     #BIT7,R4        ;CLR UNWANTED BIT
007336  042704  000200                  BIC     #BIT7,R4        ;CLR UNWANTED BITS
007342  020504                          CMP     R5,R4           ;DOES EXPECTED=RECEIVED?
```

```
2442  007344  001401                        BEQ     2$              ;BR IF YES
2443  007346  104003                        HLT     3               ;THIS IS A DATA ERROR.  IT IS **NOT**
2444                                                                  ;A DUAL ADDRESSING ERROR!!!!!!
2445  007350  104401                2$:     SCOP1                   ;SW09=1?
2446  007352  022023                        CMP     (R0)+,(R3)+     ;POP DATA AND HARDWARE POINTERS
2447  007354  005302                        DEC     R2              ;UPDATE THE REGISTER COUNTER
2448  007356  001362                        BNE     1$              ;BRANCH IF MORE TO GO
2449                                ;:NOW CHECK FOR DUAL ADDRESSING
2450  007360  012700  007432               MOV     #3$,R0          ;SET THE TABLE POINTER
2451  007364  013703  001404               MOV     RXCSR,R3        ;SET THE DUP HARDWARE POINTER
2452  007370  012737  007402  001220       MOV     #4$,LOCK        ;SET FOR SW09=1
2453  007376  012702  000004               MOV     #4,R2           ;SET FOR 4 PRIMARY REGISTERS
2454  007402  011005                4$:     MOV     (R0),R5         ;SET "EXPECTED"
2455  007404  011304                        MOV     (R3),R4         ;READ THE REGISTER BACK
2456  007406  042704  000200               BIC     #BIT7,R4        ;CLR UNWANTED BITS
2457  007412  020504                        CMP     R5,R4
2458  007414  001401                        BEQ     5$              ;BRANCH IF OK
2459  007416  104003                        HLT     3               ;THIS IS A DUAL ADDRESSING ERROR
2460  007420  104401                5$:     SCOP1                   ;SW09=1?
2461  007422  022023                        CMP     (R0)+,(R3)+     ;POP POINTERS
2462  007424  005302                        DEC     R2              ;UPDATE THE REGISTER COUNTER
2463  007426  001365                        BNE     4$              ;BRANCH IF MORE TO GO
2464  007430  104400                        SCOPE                   ;SCOPE THIS TEST
2465  007432  000020                3$:     .WORD   00020           ;RXCSR
2466  007434  000000                        .WORD   00000           ;RXDBUF AND PARCSR
2467  007436  000010                        .WORD   0C010           ;TXCSR
2468  007440  000100                        .WORD   00100           ;TXDBUF
2469
2470
2471                                        ;*************************** TEST 3 ****************************
2472                                        ;*RECEIVER CONTROL REGISTER RESET TEST.  TEST THAT AFTER
2473                                        ;*RECEIVER CONTROL REGISTER IS WRITTEN AND A BUS RESET IS
2474                                        ;*DONE THAT RECEIVER CONTROL REGISTER IS CLEARED.
2475                                        ;;************************************************************************
2476
2477                                ;::************
2478                                ;             *
2479                                ; TEST 3
2480                                ;             *
2481                                ;::************
2482                                ;;************************************************************************
2483  007442  012737  000003  001226 TST3:   MOV     #3,@#TSTNO
2484  007450  012737  007504  001216        MOV     #TST4,NEXT
2485  007456  005005                        CLR     R5              ;SET "EXPECTED"
2486  007460  013703  001404               MOV     RXCSR,R3        ;GET  THE RECEIVER CONTROL REGISTER
2487  007464  012713  177777               MOV     #-1,(R3)        ;LOAD RECEIVER CONTROL REGISTER WITH ALL ONES
2488  007470  000005                        RESET
2489  007472  011304                        MOV     (R3),R4         ;READ THE RECEIVER CONTROL REGISTER
2490  007474  020504                        CMP     R5,R4           ;R5=GOOD,R4= ?
2491  007476  001401                        BEQ     1$              ;BR IF OK
2492  007500  104002                        HLT     2               ;COMPARISON ERROR
2493  007502  104400                1$:     SCOPE                   ;SCOPE THIS TEST
2494
2495
2496                                        ;*************************** TEST 4 ****************************
2497                                        ;*RECEIVER BUFFER REGISTER RESET TEST.  TEST THAT AFTER A BUS
```

```
2498                                    ;*RESET IS DONE THAT RECEIVER BUFFER REGISTER IS CLEARED.
2499                                    ;;*****************************************************************
2500
2501                                    ;;*************
2502                                    ;               *
2503                                    ; TEST 4
2504                                    ;               *
2505                                    ;;*************
2506                                    ;;*****************************************************************
2507  007504  012737  000004  001226  TST4:   MOV     #4,@#TSTNO
2508  007512  012737  007542  001216          MOV     #TST5,NEXT
2509  007520  005005                         CLR     R5                      ;SET "EXPECTED"
2510  007522  013703  001406                 MOV     RXDBUF,R3               ;GET  THE RECEIVER BUFFER REGISTER
2511  007526  000005                         RESET
2512  007530  011304                         MOV     (R3),R4                 ;READ THE RECEIVER BUFFER REGISTER
2513  007532  020504                         CMP     R5,R4                   ;R5=GOOD,R4= ?
2514  007534  001401                         BEQ     1$                      ;BR IF OK
2515  007536  104002                         HLT     2                       ;COMPARISON ERROR
2516  007540  104400                 1$:     SCOPE                           ;SCOPE THIS TEST
2517
2518
2519                                    ;*************************** TEST 5 ****************************
2520                                    ;*PARAMETER STATUS REGISTER RESET TEST.  TEST THAT AFTER
2521                                    ;*PARAMETER STATUS REGISTER IS WRITTEN AND A BUS RESET IS
2522                                    ;*DONE THAT PARAMETER STATUS REGISTER IS CLEARED.
2523                                    ;;*****************************************************************
2524
2525                                    ;;*************
2526                                    ;               *
2527                                    ; TEST 5
2528                                    ;               *
2529                                    ;;*************
2530                                    ;;*****************************************************************
2531  007542  012737  000005  001226  TST5:   MOV     #5,@#TSTNO
2532  007550  012737  007604  001216          MOV     #TST6,NEXT
2533  007556  005005                         CLR     R5                      ;SET "EXPECTED"
2534  007560  013703  001410                 MOV     PARCSR,R3               ;GET  THE PARAMETER STATUS REGISTER
2535  007564  012713  177777                 MOV     #-1,(R3)                ;LOAD PARAMETER STATUS REGISTER WITH ALL ONES
2536  007570  000005                         RESET
2537  007572  011304                         MOV     (R3),R4                 ;READ THE PARAMETER STATUS REGISTER
2538  007574  020504                         CMP     R5,R4                   ;R5=GOOD,R4= ?
2539  007576  001401                         BEQ     1$                      ;BR IF OK
2540  007600  104002                         HLT     2                       ;COMPARISON ERROR
2541  007602  104400                 1$:     SCOPE                           ;SCOPE THIS TEST
2542
2543
2544                                    ;*************************** TEST 6 ****************************
2545                                    ;*TRANSMITTER CONTROL REGISTER RESET TEST.  TEST THAT AFTER
2546                                    ;*TRANSMITTER CONTROL REGISTER IS WRITTEN AND A BUS RESET IS
2547                                    ;*DONE THAT TRANSMITTER CONTROL REGISTER IS CLEARED.
2548                                    ;;*****************************************************************
2549
2550                                    ;;*************
2551                                    ;               *
2552                                    ; TEST 6
2553                                    ;               *
```

```
                                        ;:************
                                        ;:*************************************************************************
007604  012737  000006  001226  TST6:   MOV     #6,@#TSTNO
007612  012737  007652  001216          MOV     #TST7,NEXT
007620  005005                          CLR     R5                      ;SET "EXPECTED"
007622  013703  001412                  MOV     TXCSR,R3                ;GET  THE TRANSMITTER CONTROL REGISTER
007626  012713  177777                  MOV     #-1,(R3)                ;LOAD TRANSMITTER CONTROL REGISTER WITH ALL ONES
007632  000005                          RESET
007634  011304                          MOV     (R3),R4                 ;READ THE TRANSMITTER CONTROL REGISTER
007636  042704  000200                  BIC     #BIT7,R4                ;CLR UNWANTED BITS
007642  020504                          CMP     R5,R4                   ;R5=GOOD,R4= ?
007644  001401                          BEQ     1$                      ;BR IF OK
007646  104002                          HLT     2                       ;COMPARISON ERROR
007650  104400                  1$:     SCOPE                           ;SCOPE THIS TEST

                                        ;*************************** TEST 7 ****************************
                                        ;*TRANSMITTER BUFFER REGISTER RESET TEST.  TEST THAT AFTER
                                        ;*TRANSMITTER BUFFER REGISTER IS WRITTEN AND A BUS RESET IS
                                        ;*DONE THAT TRANSMITTER BUFFER REGISTER IS CLEARED.
                                        ;:*************************************************************

                                        ;:************
                                        ;               *
                                        ; TEST 7
                                        ;               *
                                        ;:************
                                        ;:*************************************************************************
007652  012737  000007  001226  TST7:   MOV     #7,@#TSTNO
007660  012737  007714  001216          MOV     #TST10,NEXT
007666  005005                          CLR     R5                      ;SET "EXPECTED"
007670  013703  001414                  MOV     TXDBUF,R3               ;GET  THE TRANSMITTER BUFFER REGISTER
007674  012713  177777                  MOV     #-1,(R3)                ;LOAD TRANSMITTER BUFFER REGISTER WITH ALL ONES
007700  000005                          RESET
007702  011304                          MOV     (R3),R4                 ;READ THE TRANSMITTER BUFFER REGISTER
007704  020504                          CMP     R5,R4                   ;R5=GOOD,R4= ?
007706  001401                          BEQ     1$                      ;BR IF OK
007710  104002                          HLT     2                       ;COMPARISON ERROR
007712  104400                  1$:     SCOPE                           ;SCOPE THIS TEST

                                        ;*************************** TEST 10 ***************************
                                        ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
                                        ;*SET BIT3, VERIFY BIT3 WAS SET.
                                        ;*CLEAR BIT3, VERIFY BIT3 WAS CLEARED.
                                        ;:*************************************************************

                                        ;:************
                                        ;               *
                                        ; TEST 10
                                        ;               *
                                        ;:************
                                        ;:*************************************************************************
007714  012737  000010  001226  TST10:  MOV     #10,@#TSTNO
007722  012737  010000  001216          MOV     #TST11,NEXT
007730  013703  001412                  MOV     TXCSR,R3                ;SET REGISTER TO BE TESTED.
```

```
2610  007734  012705  000010                  MOV     #BIT3,R5        ;SET "EXPECTED "
2611  007740  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
2612  007742  011304                          MOV     (R3),R4         ;READ THE REGISTER.
2613  007744  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
2614  007750  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
2615  007752  001401                          BEQ     1$              ;ARE THEY THE SAME?
2616  007754  104003                          HLT     3               ;COMPARISON ERROR.
2617  007756  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT3
2618  007760  011304                          MOV     (R3),R4         ;READ THE REGISTER.
2619  007762  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
2620  007766  005005                          CLR     R5              ;SET "EXPECTED"
2621  007770  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
2622  007772  001401                          BEQ     2$              ;BR IF OK
2623  007774  104003                          HLT     3               ;COMPARISON ERROR
2624  007776  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
2625
2626
2627                                          ;************************** TEST 11 **************************
2628                                          ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
2629                                          ;*SET BIT4, VERIFY BIT4 WAS SET.
2630                                          ;*CLEAR BIT4, VERIFY BIT4 WAS CLEARED.
2631                                          ;;**********************************************************
2632
2633                                  ;;*************
2634                                  ;               *
2635                                  ; TEST 11       *
2636                                  ;               *
2637                                  ;;*************
2638                                  ;;**********************************************************************
2639  010000  012737  000011  001226  TST11:  MOV     #11,@#TSTNO
2640  010006  012737  010064  001216          MOV     #TST12,NEXT
2641  010014  013703  001412                  MOV     TXCSR,R3        ;SET REGISTER TO BE TESTED.
2642  010020  012705  000020                  MOV     #BIT4,R5        ;SET "EXPECTED ".
2643  010024  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
2644  010026  011304                          MOV     (R3),R4         ;READ THE REGISTER.
2645  010030  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
2646  010034  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
2647  010036  001401                          BEQ     1$              ;ARE THEY THE SAME?
2648  010040  104003                          HLT     3               ;COMPARISON ERROR.
2649  010042  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT4
2650  010044  011304                          MOV     (R3),R4         ;READ THE REGISTER.
2651  010046  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
2652  010052  005005                          CLR     R5              ;SET "EXPECTED"
2653  010054  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
2654  010056  001401                          BEQ     2$              ;BR IF OK
2655  010060  104003                          HLT     3               ;COMPARISON ERROR
2656  010062  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
2657
2658
2659                                          ;************************** TEST 12 **************************
2660                                          ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
2661                                          ;*SET BIT10, VERIFY BIT10 WAS SET.
2662                                          ;*CLEAR BIT10, VERIFY BIT10 WAS CLEARED.
2663                                          ;;**********************************************************
2664
2665                                  ;;*************
```

```
                                        ;              *
                                        ; TEST 12
                                        ;              *
                                        ;:*************
                                        ;:*****************************************************************************
010064  012737  000012  001226 TST12:   MOV     #12,@#TSTNO
010072  012737  010150  001216          MOV     #TST13,NEXT
010100  013703  001412                  MOV     TXCSR,R3        ;SET REGISTER TO BE TESTED.
010104  012705  002000                  MOV     #BIT10,R5       ;SET "EXPECTED "
010110  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
010112  011304                          MOV     (R3),R4         ;READ THE REGISTER.
010114  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
010120  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
010122  001401                          BEQ     1$              ;ARE THEY THE SAME?
010124  104003                          HLT     3               ;COMPARISON ERROR.
010126  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT10
010130  011304                          MOV     (R3),R4         ;READ THE REGISTER.
010132  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
010136  005005                          CLR     R5              ;SET "EXPECTED"
010140  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
010142  001401                          BEQ     2$              ;BR IF OK
010144  104003                          HLT     3               ;COMPARISON ERROR
010146  104400                  2$:     SCOPE                   ;SCOPE THIS TEST

                                        ;************************** TEST 13 ****************************
                                        ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
                                        ;*SET BIT11, VERIFY BIT11 WAS SET.
                                        ;*CLEAR BIT11, VERIFY BIT11 WAS CLEARED.
                                        ;:*************************************************************
                                        ;:*************
                                        ;              *
                                        ; TEST 13
                                        ;              *
                                        ;:*************
                                        ;:*****************************************************************************
010150  012737  000013  001226 TST13:   MOV     #13,@#TSTNO
010156  012737  010234  001216          MOV     #TST14,NEXT
010164  013703  001412                  MOV     TXCSR,R3        ;SET REGISTER TO BE TESTED.
010170  012705  004000                  MOV     #BIT11,R5       ;SET "EXPECTED ".
010174  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
010176  011304                          MOV     (R3),R4         ;READ THE REGISTER.
010200  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
010204  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
010206  001401                          BEQ     1$              ;ARE THEY THE SAME?
010210  104003                          HLT     3               ;COMPARISON ERROR.
010212  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT11
010214  011304                          MOV     (R3),R4         ;READ THE REGISTER.
010216  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
010222  005005                          CLR     R5              ;SET "EXPECTED"
010224  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
010226  001401                          BEQ     2$              ;BR IF OK
010230  104003                          HLT     3               ;COMPARISON ERROR
010232  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
```

```
                                        ;*************************** TEST 14 ****************************
                                        ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
                                        ;*SET BIT12, VERIFY BIT12 WAS SET.
                                        ;*CLEAR BIT12, VERIFY BIT12 WAS CLEARED.
                                        ;;************************************************************************

                                ;;*************
                                ;               *
                                ; TEST 14
                                ;               *
                                ;;*************
                                ;;************************************************************************
010234  012737  000014  001226  TST14:  MOV     #14,@#TSTNO
010242  012737  010320  001216          MOV     #TST15,NEXT
010250  013703  001412                  MOV     TXCSR,R3        ;SET REGISTER TO BE TESTED.
010254  012705  010000                  MOV     #BIT12,R5       ;SET "EXPECTED ".
010260  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
010262  011304                          MOV     (R3),R4         ;READ THE REGISTER.
010264  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
010270  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
010272  001401                          BEQ     1$              ;ARE THEY THE SAME?
010274  104003                          HLT     3               ;COMPARISON ERROR.
010276  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT12
010300  011304                          MOV     (R3),R4         ;READ THE REGISTER.
010302  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
010306  005005                          CLR     R5              ;SET "EXPECTED"
010310  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
010312  001401                          BEQ     2$              ;BR IF OK
010314  104003                          HLT     3               ;COMPARISON ERROR
010316  104400                  2$:     SCOPE                   ;SCOPE THIS TEST


                                        ;*************************** TEST 15 ****************************
                                        ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
                                        ;*SET BIT13, VERIFY BIT13 WAS SET.
                                        ;*CLEAR BIT13, VERIFY BIT13 WAS CLEARED.
                                        ;;************************************************************************

                                ;;*************
                                ;               *
                                ; TEST 15
                                ;               *
                                ;;*************
                                ;;************************************************************************
010320  012737  000015  001226  TST15:  MOV     #15,@#TSTNO
010326  012737  010404  001216          MOV     #TST16,NEXT
010334  013703  001412                  MOV     TXCSR,R3        ;SET REGISTER TO BE TESTED.
010340  012705  020000                  MOV     #BIT13,R5       ;SET "EXPECTED ".
010344  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
010346  011304                          MOV     (R3),R4         ;READ THE REGISTER.
010350  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
010354  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
010356  001401                          BEQ     1$              ;ARE THEY THE SAME?
010360  104003                          HLT     3               ;COMPARISON ERROR.
010362  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT13
```

```
2778  010364  011304                        MOV     (R3),R4         ;READ THE REGISTER.
2779  010366  042704  000200                BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
2780  010372  005005                        CLR     R5              ;SET "EXPECTED"
2781  010374  020504                        CMP     R5,R4           ;R5=GOOD; R4=?
2782  010376  001401                        BEQ     2$              ;BR IF OK
2783  010400  104003                        HLT     3               ;COMPARISON ERROR
2784  010402  104400                2$:     SCOPE                   ;SCOPE THIS TEST
2785
2786
2787                                        ;*************************** TEST 16 ****************************
2788                                        ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
2789                                        ;*SET BIT0, VERIFY BIT0 WAS SET.
2790                                        ;*CLEAR BIT0, VERIFY BIT0 WAS CLEARED.
2791                                        ;:************************************************************
2792
2793                                ;:*************
2794                                ;             *
2795                                ; TEST 16
2796                                ;             *
2797                                ;:*************
2798                                ;:************************************************************************
2799  010404  012737  000016  001226 TST16:  MOV     #16,@#TSTNO
2800  010412  012737  010460  001216         MOV     #TST17,NEXT
2801  010420  013703  001414                MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
2802  010424  012705  000001                MOV     #BIT0,R5        ;SET "EXPECTED ".
2803  010430  010513                        MOV     R5,(R3)         ;WRITE THE REGISTER.
2804  010432  011304                        MOV     (R3),R4         ;READ THE REGISTER.
2805  010434  020504                        CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
2806  010436  001401                        BEQ     1$              ;ARE THEY THE SAME?
2807  010440  104003                        HLT     3               ;COMPARISON ERROR.
2808  010442  040513                1$:     BIC     R5,(R3)         ;CLEAR BIT0
2809  010444  011304                        MOV     (R3),R4         ;READ THE REGISTER.
2810  010446  005005                        CLR     R5              ;SET "EXPECTED"
2811  010450  020504                        CMP     R5,R4           ;R5=GOOD; R4=?
2812  010452  001401                        BEQ     2$              ;BR IF OK
2813  010454  104003                        HLT     3               ;COMPARISON ERROR
2814  010456  104400                2$:     SCOPE                   ;SCOPE THIS TEST
2815
2816
2817                                        ;*************************** TEST 17 ****************************
2818                                        ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
2819                                        ;*SET BIT1, VERIFY BIT1 WAS SET.
2820                                        ;*CLEAR BIT1, VERIFY BIT1 WAS CLEARED.
2821                                        ;:************************************************************
2822
2823                                ;:*************
2824                                ;             *
2825                                ; TEST 17
2826                                ;             *
2827                                ;:*************
2828                                ;:************************************************************************
2829  010460  012737  000017  001226 TST17:  MOV     #17,@#TSTNO
2830  010466  012737  010534  001216         MOV     #TST20,NEXT
2831  010474  013703  001414                MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
2832  010500  012705  000002                MOV     #BIT1,R5        ;SET "EXPECTED ".
2833  010504  010513                        MOV     R5,(R3)         ;WRITE THE REGISTER.
```

```
2834  010506  011304                        MOV     (R3),R4         ;READ THE REGISTER.
2835  010510  020504                        CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
2836  010512  001401                        BEQ     1$              ;ARE THEY THE SAME?
2837  010514  104003                        HLT     3               ;COMPARISON ERROR.
2838  010516  040513                1$:     BIC     R5,(R3)         ;CLEAR BIT1
2839  010520  011304                        MOV     (R3),R4         ;READ THE REGISTER.
2840  010522  005005                        CLR     R5              ;SET "EXPECTED"
2841  010524  020504                        CMP     R5,R4           ;R5=GOOD; R4=?
2842  010526  001401                        BEQ     2$              ;BR IF OK
2843  010530  104003                        HLT     3               ;COMPARISON ERROR
2844  010532  104400                2$:     SCOPE                   ;SCOPE THIS TEST
2845
2846
2847                                        ;*************************** TEST 20 ****************************
2848                                        ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
2849                                        ;*SET BIT2, VERIFY BIT2 WAS SET.
2850                                        ;*CLEAR BIT2, VERIFY BIT2 WAS CLEARED.
2851                                        ;;*************************************************************
2852
2853                                    ;;*************
2854                                    ;             *
2855                                    ; TEST 20
2856                                    ;             *
2857                                    ;;*************
2858                                    ;;*************************************************************
2859  010534  012737  000020  001226  TST20:  MOV     #20,@#TSTNO
2860  010542  012737  010610  001216          MOV     #TST21,NEXT
2861  010550  013703  001414                  MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
2862  010554  012705  000004                  MOV     #BIT2,R5        ;SET "EXPECTED".
2863  010560  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
2864  010562  011304                          MOV     (R3),R4         ;READ THE REGISTER.
2865  010564  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
2866  010566  001401                          BEQ     1$              ;ARE THEY THE SAME?
2867  010570  104003                          HLT     3               ;COMPARISON ERROR.
2868  010572  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT2
2869  010574  011304                          MOV     (R3),R4         ;READ THE REGISTER.
2870  010576  005005                          CLR     R5              ;SET "EXPECTED"
2871  010600  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
2872  010602  001401                          BEQ     2$              ;BR IF OK
2873  010604  104003                          HLT     3               ;COMPARISON ERROR
2874  010606  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
2875
2876
2877                                        ;*************************** TEST 21 ****************************
2878                                        ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
2879                                        ;*SET BIT3, VERIFY BIT3 WAS SET.
2880                                        ;*CLEAR BIT3, VERIFY BIT3 WAS CLEARED.
2881                                        ;;*************************************************************
2882
2883                                    ;;*************
2884                                    ;             *
2885                                    ; TEST 21
2886                                    ;             *
2887                                    ;;*************
2888                                    ;;*************************************************************
2889  010610  012737  000021  001226  TST21:  MOV     #21,@#TSTNO
```

```
2890  010616  012737  010664  001216          MOV     #TST22,NEXT
2891  010624  013703  001414                  MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
2892  010630  012705  000010                  MOV     #BIT3,R5        ;SET "EXPECTED ".
2893  010634  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
2894  010636  011304                          MOV     (R3),R4         ;READ THE REGISTER.
2895  010640  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
2896  010642  001401                          BEQ     1$              ;ARE THEY THE SAME?
2897  010644  104003                          HLT     3               ;COMPARISON ERROR.
2898  010646  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT3
2899  010650  011304                          MOV     (R3),R4         ;READ THE REGISTER.
2900  010652  005005                          CLR     R5              ;SET "EXPECTED"
2901  010654  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
2902  010656  001401                          BEQ     2$              ;BR IF OK
2903  010660  104003                          HLT     3               ;COMPARISON ERROR
2904  010662  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
2905
2906
2907                                          ;************************** TEST 22 **************************
2908                                          ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
2909                                          ;*SET BIT4, VERIFY BIT4 WAS SET.
2910                                          ;*CLEAR BIT4, VERIFY BIT4 WAS CLEARED.
2911                                          ;;**********************************************************
2912
2913                                  ;;************
2914                                  ;               *
2915                                  ; TEST 22
2916                                  ;               *
2917                                  ;;************
2918                                  ;;**********************************************************************
2919  010664  012737  000022  001226  TST22:  MOV     #22,@#TSTNO
2920  010672  012737  010740  001216          MOV     #TST23,NEXT
2921  010700  013703  001414                  MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
2922  010704  012705  000020                  MOV     #BIT4,R5        ;SET "EXPECTED ".
2923  010710  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
2924  010712  011304                          MOV     (R3),R4         ;READ THE REGISTER.
2925  010714  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
2926  010716  001401                          BEQ     1$              ;ARE THEY THE SAME?
2927  010720  104003                          HLT     3               ;COMPARISON ERROR.
2928  010722  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT4
2929  010724  011304                          MOV     (R3),R4         ;READ THE REGISTER.
2930  010726  005005                          CLR     R5              ;SET "EXPECTED"
2931  010730  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
2932  010732  001401                          BEQ     2$              ;BR IF OK
2933  010734  104003                          HLT     3               ;COMPARISON ERROR
2934  010736  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
2935
2936
2937                                          ;************************** TEST 23 **************************
2938                                          ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
2939                                          ;*SET BIT5, VERIFY BIT5 WAS SET.
2940                                          ;*CLEAR BIT5, VERIFY BIT5 WAS CLEARED.
2941                                          ;;**********************************************************
2942
2943                                  ;;************
2944                                  ;               *
2945                                  ; TEST 23
```

```
2946                                    ;            *
2947                                    ;:************
2948                                    ;:*************************************************************************
2949  010740  012737  000023  001226    TST23:  MOV     #23,@#TSTNO
2950  010746  012737  011014  001216            MOV     #TST24,NEXT
2951  010754  013703  001414                    MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
2952  010760  012705  000040                    MOV     #BIT5,R5        ;SET "EXPECTED ".
2953  010764  010513                            MOV     R5,(R3)         ;WRITE THE REGISTER.
2954  010766  011304                            MOV     (R3),R4         ;READ THE REGISTER.
2955  010770  020504                            CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
2956  010772  001401                            BEQ     1$              ;ARE THEY THE SAME?
2957  010774  104003                            HLT     3               ;COMPARISON ERROR.
2958  010776  040513                    1$:     BIC     R5,(R3)         ;CLEAR BIT5
2959  011000  011304                            MOV     (R3),R4         ;READ THE REGISTER.
2960  011002  005005                            CLR     R5              ;SET "EXPECTED"
2961  011004  020504                            CMP     R5,R4           ;R5=GOOD; R4=?
2962  011006  001401                            BEQ     2$              ;BR IF OK
2963  011010  104003                            HLT     3               ;COMPARISON ERROR
2964  011012  104400                    2$:     SCOPE                   ;SCOPE THIS TEST
2965
2966
2967                                            ;************************** TEST 24 ****************************
2968                                            ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
2969                                            ;*SET BIT6, VERIFY BIT6 WAS SET.
2970                                            ;*CLEAR BIT6, VERIFY BIT6 WAS CLEARED.
2971                                            ;:*************************************************************
2972
2973                                    ;:************
2974                                    ;            *
2975                                    ; TEST 24
2976                                    ;            *
2977                                    ;:************
2978                                    ;:*************************************************************,************
2979  011014  012737  000024  001226    TST24:  MOV     #24,@#TSTNO
2980  011022  012737  011070  001216            MOV     #TST25,NEXT
2981  011030  013703  001414                    MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
2982  011034  012705  000100                    MOV     #BIT6,R5        ;SET "EXPECTED ".
2983  011040  010513                            MOV     R5,(R3)         ;WRITE THE REGISTER.
2984  011042  011304                            MOV     (R3),R4         ;READ THE REGISTER.
2985  011044  020504                            CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
2986  011046  001401                            BEQ     1$              ;ARE THEY THE SAME?
2987  011050  104003                            HLT     3               ;COMPARISON ERROR.
2988  011052  040513                    1$:     BIC     R5,(R3)         ;CLEAR BIT6
2989  011054  011304                            MOV     (R3),R4         ;READ THE REGISTER.
2990  011056  005005                            CLR     R5              ;SET "EXPECTED"
2991  011060  020504                            CMP     R5,R4           ;R5=GOOD; R4=?
2992  011062  001401                            BEQ     2$              ;BR IF OK
2993  011064  104003                            HLT     3               ;COMPARISON ERROR
2994  011066  104400                    2$:     SCOPE                   ;SCOPE THIS TEST
2995
2996
2997                                            ;************************** TEST 25 ****************************
2998                                            ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
2999                                            ;*SET BIT7, VERIFY BIT7 WAS SET.
3000                                            ;*CLEAR BIT7, VERIFY BIT7 WAS CLEARED.
3001                                            ;:*************************************************************
```

```
3002
3003                                    ;:************
3004                                    ;            *
3005                                    ; TEST 25
3006                                    ;            *
3007                                    ;:************
3008                                    ;:*************************************************************************
3009  011070  012737  000025  001226  TST25:  MOV     #25,@#TSTNO
3010  011076  012737  011144  001216         MOV     #TST26,NEXT
3011  011104  013703  001414                 MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
3012  011110  012705  000200                 MOV     #BIT7,R5        ;SET "EXPECTED ".
3013  011114  010513                         MOV     R5,(R3)         ;WRITE THE REGISTER.
3014  011116  011304                         MOV     (R3),R4         ;READ THE REGISTER.
3015  011120  020504                         CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
3016  011122  001401                         BEQ     1$              ;ARE THEY THE SAME?
3017  011124  104003                         HLT     3               ;COMPARISON ERROR.
3018  011126  040513                  1$:    BIC     R5,(R3)         ;CLEAR BIT7
3019  011130  011304                         MOV     (R3),R4         ;READ THE REGISTER.
3020  011132  005005                         CLR     R5              ;SET "EXPECTED"
3021  011134  020504                         CMP     R5,R4           ;R5=GOOD; R4=?
3022  011136  001401                         BEQ     2$              ;BR IF OK
3023  011140  104003                         HLT     3               ;COMPARISON ERROR
3024  011142  104400                  2$:    SCOPE                   ;SCOPE THIS TEST
3025
3026
3027                                            ;************************** TEST 26 **************************
3028                                            ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
3029                                            ;*SET BIT8, VERIFY BIT8 WAS SET.
3030                                            ;*CLEAR BIT8, VERIFY BIT8 WAS CLEARED.
3031                                            ;:*********************************************************************
3032
3033                                    ;:************
3034                                    ;            *
3035                                    ; TEST 26
3036                                    ;            *
3037                                    ;:************
3038                                    ;:*************************************************************************
3039  011144  012737  000026  001226  TST26:  MOV     #26,@#TSTNO
3040  011152  012737  011220  001216         MOV     #TST27,NEXT
3041  011160  013703  001414                 MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
3042  011164  012705  000400                 MOV     #BIT8,R5        ;SET "EXPECTED ".
3043  011170  010513                         MOV     R5,(R3)         ;WRITE THE REGISTER.
3044  011172  011304                         MOV     (R3),R4         ;READ THE REGISTER.
3045  011174  020504                         CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
3046  011176  001401                         BEQ     1$              ;ARE THEY THE SAME?
3047  011200  104003                         HLT     3               ;COMPARISON ERROR.
3048  011202  040513                  1$:    BIC     R5,(R3)         ;CLEAR BIT8
3049  011204  011304                         MOV     (R3),R4         ;READ THE REGISTER.
3050  011206  005005                         CLR     R5              ;SET "EXPECTED"
3051  011210  020504                         CMP     R5,R4           ;R5=GOOD; R4=?
3052  011212  001401                         BEQ     2$              ;BR IF OK
3053  011214  104003                         HLT     3               ;COMPARISON ERROR
3054  011216  104400                  2$:    SCOPE                   ;SCOPE THIS TEST
3055
3056
3057                                            ;************************** TEST 27 **************************
```

```
                                        ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
                                        ;*SET BIT9, VERIFY BIT9 WAS SET.
                                        ;*CLEAR BIT9, VERIFY BIT9 WAS CLEARED.
                                        ;;*****************************************************************

                                    ;;************
                                    ;            *
                                    ; TEST 27
                                    ;            *
                                    ;;************
                                    ;;*****************************************************************************
011220  012737  000027  001226  TST27:  MOV     #27,@#TSTNO
011226  012737  011274  001216          MOV     #TST30,NEXT
011234  013703  001414                  MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
011240  012705  001000                  MOV     #BIT9,R5        ;SET "EXPECTED ".
011244  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
011246  011304                          MOV     (R3),R4         ;READ THE REGISTER.
011250  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
011252  001401                          BEQ     1$              ;ARE THEY THE SAME?
011254  104003                          HLT     3               ;COMPARISON ERROR.
011256  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT9
011260  011304                          MOV     (R3),R4         ;READ THE REGISTER.
011262  005005                          CLR     R5              ;SET "EXPECTED"
011264  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
011266  001401                          BEQ     2$              ;BR IF OK
011270  104003                          HLT     3               ;COMPARISON ERROR
011272  104400                  2$:     SCOPE                   ;SCOPE THIS TEST

                                        ;************************** TEST 30 ***************************************
                                        ;*TRANSMITTER DATA BUFFER REGISTER READ/WRITE BIT TEST.
                                        ;*SET BIT10, VERIFY BIT10 WAS SET.
                                        ;*CLEAR BIT10, VERIFY BIT10 WAS CLEARED.
                                        ;;*****************************************************************

                                    ;;************
                                    ;            *
                                    ; TEST 30
                                    ;            *
                                    ;;************
                                    ;;*****************************************************************************
011274  012737  000030  001226  TST30:  MOV     #30,@#TSTNO
011302  012737  011350  001216          MOV     #TST31,NEXT
011310  013703  001414                  MOV     TXDBUF,R3       ;SET REGISTER TO BE TESTED.
011314  012705  002000                  MOV     #BIT10,R5       ;SET "EXPECTED ".
011320  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
011322  011304                          MOV     (R3),R4         ;READ THE REGISTER.
011324  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
011326  001401                          BEQ     1$              ;ARE THEY THE SAME?
011330  104003                          HLT     3               ;COMPARISON ERROR.
011332  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT10
011334  011304                          MOV     (R3),R4         ;READ THE REGISTER.
011336  005005                          CLR     R5              ;SET "EXPECTED"
011340  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
011342  001401                          BEQ     2$              ;BR IF OK
011344  104003                          HLT     3               ;COMPARISON ERROR
```

```
3114  011346  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3115
3116
3117                                          ;*************************** TEST 31 ****************************
3118                                          ;*RECEIVER CONTROL REGISTER READ/WRITE BIT 1 RESET AND CLEAR TEST
3119                                          ;*WRITE BIT 1,AND TEST THAT IT WILL BE CLEARED AFTER A
3120                                          ;*DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION
3121                                          ;;************************************************************
3122
3123                                  ;;*************
3124                                  ;              *
3125                                  ; TEST 31
3126                                  ;              *
3127                                  ;;*************
3128                                  ;;************************************************************************
3129  011350  012737  000031  001226  TST31:  MOV     #31,@#TSTNO
3130  011356  012737  011444  001216          MOV     #TST32,NEXT
3131  011364  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
3132  011370  052713  000002                  BIS     #BIT1,(R3)      ;SET BIT 1 AT RECEIVER CONTROL REGISTER
3133  011374  005005                          CLR     R5              ;SET "EXPECTED"
3134  011376  052777  000400  170006          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3135  011404  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3136  011410  032713  000002                  BIT     #BIT1,(R3)      ;TEST BIT 1
3137  011414  001402                          BEQ     1$              ;BIT 1 IS CLEARED
3138  011416  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3139  011420  104003                          HLT     3               ;BIT 1 IS SET AND SHOULDN'T BE
3140  011422  052713  000002          1$:     BIS     #BIT1,(R3)      ;SET BIT 1 AGAIN
3141  011426  005013                          CLR     (R3)            ;CLEAR THE RECEIVER CONTROL REGISTER
3142  011430  032713  000002                  BIT     #BIT1,(R3)      ;TEST TO SEE IF BIT 1 CLEARED
3143  011434  001402                          BEQ     2$              ;BIT 1 IS OK
3144  011436  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3145  011440  104003                          HLT     3               ;BIT 1 FAILED TO CLEAR
3146  011442  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3147
3148
3149                                          ;*************************** TEST 32 ****************************
3150                                          ;*RECEIVER CONTROL REGISTER READ/WRITE BIT 2 RESET AND CLEAR TES
3151                                          ;*WRITE BIT 2,AND TEST THAT IT WILL BE CLEARED AFTER A
3152                                          ;*DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION
3153                                          ;;************************************************************
3154
3155                                  ;;*************
3156                                  ;              *
3157                                  ; TEST 32
3158                                  ;              *
3159                                  ;;*************
3160                                  ;;************************************************************************
3161  011444  012737  000032  001226  TST32:  MOV     #32,@#TSTNO
3162  011452  012737  011540  001216          MOV     #TST33,NEXT
3163  011460  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
3164  011464  052713  000004                  BIS     #BIT2,(R3)      ;SET BIT 2 AT RECEIVER CONTROL REGISTER
3165  011470  005005                          CLR     R5              ;SET "EXPECTED"
3166  011472  052777  000400  167712          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3167  011500  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3168  011504  032713  000004                  BIT     #BIT2,(R3)      ;TEST BIT 2
3169  011510  001402                          BEQ     1$              ;BIT 2 IS CLEARED
```

```
3170  011512  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3171  011514  104003                          HLT     3               ;BIT 2 IS SET AND SHOULDN'T BE
3172  011516  052713  000004          1$:     BIS     #BIT2,(R3)      ;SET BIT 2 AGAIN
3173  011522  005013                          CLR     (R3)            ;CLEAR THE RECEIVER CONTROL REGISTER
3174  011524  032713  000004                  BIT     #BIT2,(R3)      ;TEST TO SEE IF BIT 2 CLEARED
3175  011530  001402                          BEQ     2$              ;BIT 2 IS OK
3176  011532  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3177  011534  104003                          HLT     3               ;BIT 2 FAILED TO CLEAR
3178  011536  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3179
3180
3181                                          ;*************************** TEST 33 ****************************
3182                                          ;*RECEIVER CONTROL REGISTER READ/WRITE BIT 3 RESET AND CLEAR TEST
3183                                          ;*WRITE BIT 3,AND TEST THAT IT WILL BE CLEARED AFTER A
3184                                          ;*DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION
3185                                          ;;*************************************************************
3186
3187                                  ;;*************
3188                                  ;             *
3189                                  ; TEST 33
3190                                  ;             *
3191                                  ;;*************
3192                                  ;;*************************************************************************
3193  011540  012737  000033  001226  TST33:  MOV     #33,@#TSTNO
3194  011546  012737  011634  001216          MOV     #TST34,NEXT
3195  011554  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
3196  011560  052713  000010                  BIS     #BIT3,(R3)      ;SET BIT 3 AT RECEIVER CONTROL REGISTER
3197  011564  005005                          CLR     R5              ;SET "EXPECTED"
3198  011566  052777  000400  167616        BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3199  011574  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3200  011600  032713  000010                  BIT     #BIT3,(R3)      ;TEST BIT 3
3201  011604  001402                          BEQ     1$              ;BIT 3 IS CLEARED
3202  011606  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3203  011610  104003                          HLT     3               ;BIT 3 IS SET AND SHOULDN'T BE
3204  011612  052713  000010          1$:     BIS     #BIT3,(R3)      ;SET BIT 3 AGAIN
3205  011616  005013                          CLR     (R3)            ;CLEAR THE RECEIVER CONTROL REGISTER
3206  011620  032713  000010                  BIT     #BIT3,(R3)      ;TEST TO SEE IF BIT 3 CLEARED
3207  011624  001402                          BEQ     2$              ;BIT 3 IS OK
3208  011626  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3209  011630  104003                          HLT     3               ;BIT 3 FAILED TO CLEAR
3210  011632  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3211
3212
3213                                          ;*************************** TEST 34 ****************************
3214                                          ;*RECEIVER CONTROL REGISTER READ/WRITE BIT 4 RESET AND CLEAR TEST
3215                                          ;*WRITE BIT 4,AND TEST THAT IT WILL BE CLEARED AFTER A
3216                                          ;*DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION
3217                                          ;;*************************************************************
3218
3219                                  ;;*************
3220                                  ;             *
3221                                  ; TEST 34
3222                                  ;             *
3223                                  ;;*************
3224                                  ;;*************************************************************************
3225  011634  012737  000034  001226  TST34:  MOV     #34,@#TSTNO
```

```
3226  011642  012737  011730  001216          MOV     #TST35,NEXT
3227  011650  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
3228  011654  052713  000020                  BIS     #BIT4,(R3)      ;SET BIT 4 AT RECEIVER CONTROL REGISTER
3229  011660  005005                          CLR     R5              ;SET "EXPECTED"
3230  011662  052777  000400  167522          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3231  011670  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3232  011674  032713  000020                  BIT     #BIT4,(R3)      ;TEST BIT 4
3233  011700  001402                          BEQ     1$              ;BIT 4 IS CLEARED
3234  011702  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3235  011704  104003                          HLT     3               ;BIT 4 IS SET AND SHOULDN'T BE
3236  011706  052713  000020          1$:     BIS     #BIT4,(R3)      ;SET BIT 4 AGAIN
3237  011712  005013                          CLR     (R3)            ;CLEAR THE RECEIVER CONTROL REGISTER
3238  011714  032713  000020                  BIT     #BIT4,(R3)      ;TEST TO SEE IF BIT 4 CLEARED
3239  011720  001402                          BEQ     2$              ;BIT 4 IS OK
3240  011722  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3241  011724  104003                          HLT     3               ;BIT 4 FAILED TO CLEAR
3242  011726  104400          2$:     SCOPE                           ;SCOPE THIS TEST
3243
3244
3245                                  ;*************************** TEST 35 ****************************
3246                                  ;*RECEIVER CONTROL REGISTER READ/WRITE BIT 8 RESET AND CLEAR TEST
3247                                  ;*WRITE BIT 8,AND TEST THAT IT WILL BE CLEARED AFTER A
3248                                  ;*DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION
3249                                  ;;*************************************************************
3250
3251                                  ;;************
3252                                  ;               *
3253                                  ; TEST 35
3254                                  ;               *
3255                                  ;;************
3256                                  ;;*************************************************************
3257  011730  012737  000035  001226  TST35:  MOV     #35,@#TSTNO
3258  011736  012737  012024  001216          MOV     #TST36,NEXT
3259  011744  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
3260  011750  052713  000400                  BIS     #BIT8,(R3)      ;SET BIT 8 AT RECEIVER CONTROL REGISTER
3261  011754  005005                          CLR     R5              ;SET "EXPECTED"
3262  011756  052777  000400  167426          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3263  011764  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3264  011770  032713  000400                  BIT     #BIT8,(R3)      ;TEST BIT 8
3265  011774  001402                          BEQ     1$              ;BIT 8 IS CLEARED
3266  011776  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3267  012000  104003                          HLT     3               ;BIT 8 IS SET AND SHOULDN'T BE
3268  012002  052713  000400          1$:     BIS     #BIT8,(R3)      ;SET BIT 8 AGAIN
3269  012006  005013                          CLR     (R3)            ;CLEAR THE RECEIVER CONTROL REGISTER
3270  012010  032713  000400                  BIT     #BIT8,(R3)      ;TEST TO SEE IF BIT 8 CLEARED
3271  012014  001402                          BEQ     2$              ;BIT 8 IS OK
3272  012016  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3273  012020  104003                          HLT     3               ;BIT 8 FAILED TO CLEAR
3274  012022  104400          2$:     SCOPE                           ;SCOPE THIS TEST
3275
3276
3277                                  ;*************************** TEST 36 ****************************
3278                                  ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT 3 RESET AND CLEAR TEST
3279                                  ;*WRITE BIT 3,AND TEST THAT IT WILL BE CLEARED AFTER A
3280                                  ;*DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION
3281                                  ;;*************************************************************
```

```
3282
3283                                    ;:************
3284                                    ;            *
3285                                    ; TEST 36
3286                                    ;            *
3287                                    ;:************
3288                                    ;:*************************************************************************
3289  012024  012737  000036  001226  TST36:  MOV     #36,@#TSTNO
3290  012032  012737  012120  001216          MOV     #TST37,NEXT
3291  012040  013703  001412                 MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
3292  012044  052713  000010                 BIS     #BIT3,(R3)      ;SET BIT 3 AT TRANSMITTER CONTROL REGISTER
3293  012050  005005                         CLR     R5              ;SET "EXPECTED"
3294  012052  052777  000400  167332         BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3295  012060  004737  005044                 JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3296  012064  032713  000010                 BIT     #BIT3,(R3)      ;TEST BIT 3
3297  012070  001402                         BEQ     1$              ;BIT 3 IS CLEARED
3298  012072  011304                         MOV     (R3),R4         ;LOAD "FOUND"
3299  012074  104003                         HLT     3               ;BIT 3 IS SET AND SHOULDN'T BE
3300  012076  052713  000010          1$:    BIS     #BIT3,(R3)      ;SET BIT 3 AGAIN
3301  012102  005013                         CLR     (R3)            ;CLEAR THE TRANSMITTER CONTROL REGISTER
3302  012104  032713  000010                 BIT     #BIT3,(R3)      ;TEST TO SEE IF BIT 3 CLEARED
3303  012110  001402                         BEQ     2$              ;BIT 3 IS OK
3304  012112  011304                         MOV     (R3),R4         ;LOAD "FOUND"
3305  012114  104003                         HLT     3               ;BIT 3 FAILED TO CLEAR
3306  012116  104400                  2$:    SCOPE                   ;SCOPE THIS TEST
3307
3308
3309                                    ;************************* TEST 37 *************************
3310                                    ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT 4 RESET AND CLEAR TEST
3311                                    ;*WRITE BIT 4 AND TEST THAT IT WILL BE CLEARED AFTER A
3312                                    ;*DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION
3313                                    ;:*************************************************************
3314
3315                                    ;:************
3316                                    ;            *
3317                                    ; TEST 37
3318                                    ;            *
3319                                    ;:************
3320                                    ;:*************************************************************************
3321  012120  012737  000037  001226  TST37:  MOV     #37,@#TSTNO
3322  012126  012737  012214  001216          MOV     #TST40,NEXT
3323  012134  013703  001412                 MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
3324  012140  052713  000020                 BIS     #BIT4,(R3)      ;SET BIT 4 AT TRANSMITTER CONTROL REGISTER
3325  012144  005005                         CLR     R5              ;SET "EXPECTED"
3326  012146  052777  000400  167236         BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3327  012154  004737  005044                 JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3328  012160  032713  000020                 BIT     #BIT4,(R3)      ;TEST BIT 4
3329  012164  001402                         BEQ     1$              ;BIT 4 IS CLEARED
3330  012166  011304                         MOV     (R3),R4         ;LOAD "FOUND"
3331  012170  104003                         HLT     3               ;BIT 4 IS SET AND SHOULDN'T BE
3332  012172  052713  000020          1$:    BIS     #BIT4,(R3)      ;SET BIT 4 AGAIN
3333  012176  005013                         CLR     (R3)            ;CLEAR THE TRANSMITTER CONTROL REGISTER
3334  012200  032713  000020                 BIT     #BIT4,(R3)      ;TEST TO SEE IF BIT 4 CLEARED
3335  012204  001402                         BEQ     2$              ;BIT 4 IS OK
3336  012206  011304                         MOV     (R3),R4         ;LOAD "FOUND"
3337  012210  104003                         HLT     3               ;BIT 4 FAILED TO CLEAR
```

```
3338  012212  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3339
3340
3341                                          ;*************************** TEST 40 ****************************
3342                                          ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT 10 RESET AND CLEAR TEST
3343                                          ;*WRITE BIT 10,AND TEST THAT IT WILL BE CLEARED AFTER A
3344                                          ;*DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION
3345                                          ;;************************************************************************
3346
3347                                  ;;*************
3348                                  ;             *
3349                                  ; TEST 40
3350                                  ;             *
3351                                  ;;*************
3352                                  ;;************************************************************************
3353  012214  012737  000040  001226  TST40:  MOV     #40,@#TSTNO
3354  012222  012737  012310  001216          MOV     #TST41,NEXT
3355  012230  013703  001412                  MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
3356  012234  052713  002000                  BIS     #BIT10,(R3)     ;SET BIT 10 AT TRANSMITTER CONTROL REGISTER
3357  012240  005005                          CLR     R5              ;SET "EXPECTED"
3358  012242  052777  000400  167142          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3359  012250  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3360  012254  032713  002000                  BIT     #BIT10,(R3)     ;TEST BIT 10
3361  012260  001402                          BEQ     1$              ;BIT 10 IS CLEARED
3362  012262  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3363  012264  104003                          HLT     3               ;BIT 10 IS SET AND SHOULDN'T BE
3364  012266  052713  002000          1$:     BIS     #BIT10,(R3)     ;SET BIT 10 AGAIN
3365  012272  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER CONTROL REGISTER
3366  012274  032713  002000                  BIT     #BIT10,(R3)     ;TEST TO SEE IF BIT 10 CLEARED
3367  012300  001402                          BEQ     2$              ;BIT 10 IS OK
3368  012302  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3369  012304  104003                          HLT     3               ;BIT 10 FAILED TO CLEAR
3370  012306  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3371
3372
3373                                          ;*************************** TEST 41 ****************************
3374                                          ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT 11 RESET AND CLEAR TEST
3375                                          ;*WRITE BIT 11,AND TEST THAT IT WILL BE CLEARED AFTER A
3376                                          ;*DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION
3377                                          ;;************************************************************************
3378
3379                                  ;;*************
3380                                  ;             *
3381                                  ; TEST 41
3382                                  ;             *
3383                                  ;;*************
3384                                  ;;************************************************************************
3385  012310  012737  000041  001226  TST41:  MOV     #41,@#TSTNO
3386  012316  012737  012404  001216          MOV     #TST42,NEXT
3387  012324  013703  001412                  MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
3388  012330  052713  004000                  BIS     #BIT11,(R3)     ;SET BIT 11 AT TRANSMITTER CONTROL REGISTER
3389  012334  005005                          CLR     R5              ;SET "EXPECTED"
3390  012336  052777  000400  167046          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3391  012344  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3392  012350  032713  004000                  BIT     #BIT11,(R3)     ;TEST BIT 11
3393  012354  001402                          BEQ     1$              ;BIT 11 IS CLEARED
```

```
3394  012356  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3395  012360  104003                          HLT     3               ;BIT 11 IS SET AND SHOULDN'T BE
3396  012362  052713  004000          1$:     BIS     #BIT11,(R3)     ;SET BIT 11 AGAIN
3397  012366  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER CONTROL REGISTER
3398  012370  032713  004000                  BIT     #BIT11,(R3)     ;TEST TO SEE IF BIT 11 CLEARED
3399  012374  001402                          BEQ     2$              ;BIT 11 IS OK
3400  012376  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3401  012400  104003                          HLT     3               ;BIT 11 FAILED TO CLEAR
3402  012402  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3403
3404
3405                                          ;*************************** TEST 42 ****************************
3406                                          ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT 12 RESET AND CLEAR TEST
3407                                          ;*WRITE BIT 12,AND TEST THAT IT WILL BE CLEARED AFTER A
3408                                          ;*DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION
3409                                          ;;*************************************************************
3410
3411                                  ;:*************
3412                                  ;             *
3413                                  ; TEST 42
3414                                  ;             *
3415                                  ;:*************
3416                                  ;;************************************************************************
3417  012404  012737  000042  001226  TST42:  MOV     #42,@#TSTNO
3418  012412  012737  012500  001216          MOV     #TST43,NEXT
3419  012420  013703  001412                  MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
3420  012424  052713  010000                  BIS     #BIT12,(R3)     ;SET BIT 12 AT TRANSMITTER CONTROL REGISTER
3421  012430  005005                          CLR     R5              ;SET "EXPECTED"
3422  012432  052777  000400  166752          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3423  012440  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3424  012444  032713  010000                  BIT     #BIT12,(R3)     ;TEST BIT 12
3425  012450  001402                          BEQ     1$              ;BIT 12 IS CLEARED
3426  012452  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3427  012454  104003                          HLT     3               ;BIT 12 IS SET AND SHOULDN'T BE
3428  012456  052713  010000          1$:     BIS     #BIT12,(R3)     ;SET BIT 12 AGAIN
3429  012462  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER CONTROL REGISTER
3430  012464  032713  010000                  BIT     #BIT12,(R3)     ;TEST TO SEE IF BIT 12 CLEARED
3431  012470  001402                          BEQ     2$              ;BIT 12 IS OK
3432  012472  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3433  012474  104003                          HLT     3               ;BIT 12 FAILED TO CLEAR
3434  012476  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3435
3436
3437                                          ;*************************** TEST 43 ****************************
3438                                          ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT 13 RESET AND CLEAR TEST
3439                                          ;*WRITE BIT 13,AND TEST THAT IT WILL BE CLEARED AFTER A
3440                                          ;*DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION
3441                                          ;;*************************************************************
3442
3443                                  ;:*************
3444                                  ;             *
3445                                  ; TEST 43
3446                                  ;             *
3447                                  ;:*************
3448                                  ;;************************************************************************
3449  012500  012737  000043  001226  TST43:  MOV     #43,@#TSTNO
```

```
3450  012506  012737  012574  001216          MOV     #TST44,NEXT
3451  012514  013703  001412                  MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
3452  012520  052713  020000                  BIS     #BIT13,(R3)     ;SET BIT 13 AT TRANSMITTER CONTROL REGISTER
3453  012524  005005                          CLR     R5              ;SET "EXPECTED"
3454  012526  052777  000400  166656          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3455  012534  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3456  012540  032713  020000                  BIT     #BIT13,(R3)     ;TEST BIT 13
3457  012544  001402                          BEQ     1$              ;BIT 13 IS CLEARED
3458  012546  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3459  012550  104003                          HLT     3               ;BIT 13 IS SET AND SHOULDN'T BE
3460  012552  052713  020000          1$:     BIS     #BIT13,(R3)     ;SET BIT 13 AGAIN
3461  012556  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER CONTROL REGISTER
3462  012560  032713  020000                  BIT     #BIT13,(R3)     ;TEST TO SEE IF BIT 13 CLEARED
3463  012564  001402                          BEQ     2$              ;BIT 13 IS OK
3464  012566  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3465  012570  104003                          HLT     3               ;BIT 13 FAILED TO CLEAR
3466  012572  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3467
3468
3469                                          ;*************************** TEST 44 ****************************
3470                                          ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 0 RESET AND CLEAR TEST
3471                                          ;*WRITE BIT 0,AND TEST THAT IT WILL BE CLEARED AFTER A
3472                                          ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
3473                                          ;;*************************************************************
3474
3475                                  ;;************
3476                                  ;           *
3477                                  ; TEST 44
3478                                  ;           *
3479                                  ;;************
3480                                  ;;*************************************************************************
3481  012574  012737  000044  001226  TST44:  MOV     #44,@#TSTNO
3482  012602  012737  012670  001216          MOV     #TST45,NEXT
3483  012610  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
3484  012614  052713  000001                  BIS     #BIT0,(R3)      ;SET BIT 0 AT TRANSMITTER BUFFER REGISTER
3485  012620  005005                          CLR     R5              ;SET "EXPECTED"
3486  012622  052777  000400  166562          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3487  012630  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3488  012634  032713  000001                  BIT     #BIT0,(R3)      ;TEST BIT 0
3489  012640  001402                          BEQ     1$              ;BIT 0 IS CLEARED
3490  012642  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3491  012644  104003                          HLT     3               ;BIT 0 IS SET AND SHOULDN'T BE
3492  012646  052713  000001          1$:     BIS     #BIT0,(R3)      ;SET BIT 0 AGAIN
3493  012652  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
3494  012654  032713  000001                  BIT     #BIT0,(R3)      ;TEST TO SEE IF BIT 0 CLEARED
3495  012660  001402                          BEQ     2$              ;BIT 0 IS OK
3496  012662  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3497  012664  104003                          HLT     3               ;BIT 0 FAILED TO CLEAR
3498  012666  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3499
3500
3501                                          ;*************************** TEST 45 ****************************
3502                                          ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 1 RESET AND CLEAR TEST
3503                                          ;*WRITE BIT 1,AND TEST THAT IT WILL BE CLEARED AFTER A
3504                                          ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
3505                                          ;;*************************************************************
```

```
3506
3507                                    ;:************
3508                                    ;            *
3509                                    ; TEST 45
3510                                    ;            *
3511                                    ;:************
3512                                    ;:*************************************************************************
3513  012670  012737  000045  001226  TST45:  MOV     #45,@#TSTNO
3514  012676  012737  012764  001216           MOV     #TST46,NEXT
3515  012704  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
3516  012710  052713  000002                  BIS     #BIT1,(R3)      ;SET BIT 1 AT TRANSMITTER BUFFER REGISTER
3517  012714  005005                          CLR     R5              ;SET "EXPECTED"
3518  012716  052777  000400  166466          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3519  012724  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3520  012730  032713  000002                  BIT     #BIT1,(R3)      ;TEST BIT 1
3521  012734  001402                          BEQ     1$              ;BIT 1 IS CLEARED
3522  012736  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3523  012740  104003                          HLT     3               ;BIT 1 IS SET AND SHOULDN'T BE
3524  012742  052713  000002          1$:     BIS     #BIT1,(R3)      ;SET BIT 1 AGAIN
3525  012746  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
3526  012750  032713  000002                  BIT     #BIT1,(R3)      ;TEST TO SEE IF BIT 1 CLEARED
3527  012754  001402                          BEQ     2$              ;BIT 1 IS OK
3528  012756  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3529  012760  104003                          HLT     3               ;BIT 1 FAILED TO CLEAR
3530  012762  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3531
3532
3533                                    ;************************* TEST 46 *************************
3534                                    ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 2 RESET AND CLEAR TEST
3535                                    ;*WRITE BIT 2,AND TEST THAT IT WILL BE CLEARED AFTER A
3536                                    ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
3537                                    ;:*******************************************************************
3538
3539                                    ;:************
3540                                    ;            *
3541                                    ; TEST 46
3542                                    ;            *
3543                                    ;:************
3544                                    ;:*************************************************************************
3545  012764  012737  000046  001226  TST46:  MOV     #46,@#TSTNO
3546  012772  012737  013060  001216           MOV     #TST47,NEXT
3547  013000  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
3548  013004  052713  000004                  BIS     #BIT2,(R3)      ;SET BIT 2 AT TRANSMITTER BUFFER REGISTER
3549  013010  005005                          CLR     R5              ;SET "EXPECTED"
3550  013012  052777  000400  166372          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3551  013020  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3552  013024  032713  000004                  BIT     #BIT2,(R3)      ;TEST BIT 2
3553  013030  001402                          BEQ     1$              ;BIT 2 IS CLEARED
3554  013032  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3555  013034  104003                          HLT     3               ;BIT 2 IS SET AND SHOULDN'T BE
3556  013036  052713  000004          1$:     BIS     #BIT2,(R3)      ;SET BIT 2 AGAIN
3557  013042  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
3558  013044  032713  000004                  BIT     #BIT2,(R3)      ;TEST TO SEE IF BIT 2 CLEARED
3559  013050  001402                          BEQ     2$              ;BIT 2 IS OK
3560  013052  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3561  013054  104003                          HLT     3               ;BIT 2 FAILED TO CLEAR
```

```
3562  013056  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3563
3564
3565                                          ;************************** TEST 47 ****************************
3566                                          ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 3 RESET AND CLEAR TEST
3567                                          ;*WRITE BIT 3 AND TEST THAT IT WILL BE CLEARED AFTER A
3568                                          ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
3569                                          ;:**************************************************************
3570
3571                                  ;:*************
3572                                  ;             *
3573                                  ; TEST 47
3574                                  ;             *
3575                                  ;:*************
3576                                  ;:*****************************************************************************
3577  013060  012737  000047  001226  TST47:  MOV     #47,@#TSTNO
3578  013066  012737  013154  001216          MOV     #TST50,NEXT
3579  013074  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
3580  013100  052713  000010                  BIS     #BIT3,(R3)      ;SET BIT 3 AT TRANSMITTER BUFFER REGISTER
3581  013104  005005                          CLR     R5              ;SET "EXPECTED"
3582  013106  052777  000400  166276          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3583  013114  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3584  013120  032713  000010                  BIT     #BIT3,(R3)      ;TEST BIT 3
3585  013124  001402                          BEQ     1$              ;BIT 3 IS CLEARED
3586  013126  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3587  013130  104003                          HLT     3               ;BIT 3 IS SET AND SHOULDN'T BE
3588  013132  052713  000010          1$:     BIS     #BIT3,(R3)      ;SET BIT 3 AGAIN
3589  013136  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
3590  013140  032713  000010                  BIT     #BIT3,(R3)      ;TEST TO SEE IF BIT 3 CLEARED
3591  013144  001402                          BEQ     2$              ;BIT 3 IS OK
3592  013146  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3593  013150  104003                          HLT     3               ;BIT 3 FAILED TO CLEAR
3594  013152  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3595
3596
3597                                          ;************************** TEST 50 ****************************
3598                                          ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 4 RESET AND CLEAR TEST
3599                                          ;*WRITE BIT 4 AND TEST THAT IT WILL BE CLEARED AFTER A
3600                                          ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
3601                                          ;:**************************************************************
3602
3603                                  ;:*************
3604                                  ;             *
3605                                  ; TEST 50
3606                                  ;             *
3607                                  ;:*************
3608                                  ;:*****************************************************************************
3609  013154  012737  000050  001226  TST50:  MOV     #50,@#TSTNO
3610  013162  012737  013250  001216          MOV     #TST51,NEXT
3611  013170  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
3612  013174  052713  000020                  BIS     #BIT4,(R3)      ;SET BIT 4 AT TRANSMITTER BUFFER REGISTER
3613  013200  005005                          CLR     R5              ;SET "EXPECTED"
3614  013202  052777  000400  166202          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3615  013210  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3616  013214  032713  000020                  BIT     #BIT4,(R3)      ;TEST BIT 4
3617  013220  001402                          BEQ     1$              ;BIT 4 IS CLEARED
```

```
 3618  013222  011304                          MOV     (R3),R4         ;LOAD "FOUND"
 3619  013224  104003                          HLT     3               ;BIT 4 IS SET AND SHOULDN'T BE
 3620  013226  052713  000020          1$:     BIS     #BIT4,(R3)      ;SET BIT 4 AGAIN
 3621  013232  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
 3622  013234  032713  000020                  BIT     #BIT4,(R3)      ;TEST TO SEE IF BIT 4 CLEARED
 3623  013240  001402                          BEQ     2$              ;BIT 4 IS OK
 3624  013242  011304                          MOV     (R3),R4         ;LOAD "FOUND"
 3625  013244  104003                          HLT     3               ;BIT 4 FAILED TO CLEAR
 3626  013246  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
 3627
 3628                                          ;*************************** TEST 51 ****************************
 3629                                          ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 5 RESET AND CLEAR TEST
 3630                                          ;*WRITE BIT 5,AND TEST THAT IT WILL BE CLEARED AFTER A
 3631                                          ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
 3632                                          ;;*************************************************************
 3633
 3634                                  ;;*************
 3635                                  ;             *
 3636                                  ; TEST 51     *
 3637                                  ;             *
 3638                                  ;;*************
 3639                                  ;;*************************************************************************
 3640  013250  012737  000051  001226  TST51:  MOV     #51,@#TSTNO
 3641  013256  012737  013344  001216          MOV     #TST52,NEXT
 3642  013264  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
 3643  013270  052713  000040                  BIS     #BIT5,(R3)      ;SET BIT 5 AT TRANSMITTER BUFFER REGISTER
 3644  013274  005005                          CLR     R5              ;SET "EXPECTED"
 3645  013276  052777  000400  166106          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
 3646  013304  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
 3647  013310  032713  000040                  BIT     #BIT5,(R3)      ;TEST BIT 5
 3648  013314  001402                          BEQ     1$              ;BIT 5 IS CLEARED
 3649  013316  011304                          MOV     (R3),R4         ;LOAD "FOUND"
 3650  013320  104003                          HLT     3               ;BIT 5 IS SET AND SHOULDN'T BE
 3651  013322  052713  000040          1$:     BIS     #BIT5,(R3)      ;SET BIT 5 AGAIN
 3652  013326  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
 3653  013330  032713  000040                  BIT     #BIT5,(R3)      ;TEST TO SEE IF BIT 5 CLEARED
 3654  013334  001402                          BEQ     2$              ;BIT 5 IS OK
 3655  013336  011304                          MOV     (R3),R4         ;LOAD "FOUND"
 3656  013340  104003                          HLT     3               ;BIT 5 FAILED TO CLEAR
 3657  013342  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
 3658
 3659
 3660                                          ;*************************** TEST 52 ****************************
 3661                                          ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 6 RESET AND CLEAR TEST
 3662                                          ;*WRITE BIT 6,AND TEST THAT IT WILL BE CLEARED AFTER A
 3663                                          ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
 3664                                          ;;*************************************************************
 3665
 3666                                  ;;*************
 3667                                  ;             *
 3668                                  ; TEST 52     *
 3669                                  ;             *
 3670                                  ;;*************
 3671                                  ;;*************************************************************************
 3672  013344  012737  000052  001226  TST52:  MOV     #52,@#TSTNO
```

```
3674  013352  012737  013440  001216          MOV     #TST53,NEXT
3675  013360  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
3676  013364  052713  000100                  BIS     #BIT6,(R3)      ;SET BIT 6 AT TRANSMITTER BUFFER REGISTER
3677  013370  005005                          CLR     R5              ;SET "EXPECTED"
3678  013372  052777  000400  166012          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3679  013400  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3680  013404  032713  000100                  BIT     #BIT6,(R3)      ;TEST BIT 6
3681  013410  001402                          BEQ     1$              ;BIT 6 IS CLEARED
3682  013412  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3683  013414  104003                          HLT     3               ;BIT 6 IS SET AND SHOULDN'T BE
3684  013416  052713  000100          1$:     BIS     #BIT6,(R3)      ;SET BIT 6 AGAIN
3685  013422  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
3686  013424  032713  000100                  BIT     #BIT6,(R3)      ;TEST TO SEE IF BIT 6 CLEARED
3687  013430  001402                          BEQ     2$              ;BIT 6 IS OK
3688  013432  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3689  013434  104003                          HLT     3               ;BIT 6 FAILED TO CLEAR
3690  013436  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3691
3692
3693                                          ;*************************** TEST 53 ****************************
3694                                          ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 7 RESET AND CLEAR TEST
3695                                          ;*WRITE BIT 7,AND TEST THAT IT WILL BE CLEARED AFTER A
3696                                          ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
3697                                          ;;*************************************************************
3698
3699                                  ;;*************
3700                                  ;*           *
3701                                  ; TEST 53
3702                                  ;*           *
3703                                  ;;*************
3704                                  ;;*************************************************************************
3705  013440  012737  000053  001226  TST53:  MOV     #53,@#TSTNO
3706  013446  012737  013534  001216          MOV     #TST54,NEXT
3707  013454  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
3708  013460  052713  000200                  BIS     #BIT7,(R3)      ;SET BIT 7 AT TRANSMITTER BUFFER REGISTER
3709  013464  005005                          CLR     R5              ;SET "EXPECTED"
3710  013466  052777  000400  165716          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3711  013474  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3712  013500  032713  000200                  BIT     #BIT7,(R3)      ;TEST BIT 7
3713  013504  001402                          BEQ     1$              ;BIT 7 IS CLEARED
3714  013506  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3715  013510  104003                          HLT     3               ;BIT 7 IS SET AND SHOULDN'T BE
3716  013512  052713  000200          1$:     BIS     #BIT7,(R3)      ;SET BIT 7 AGAIN
3717  013516  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
3718  013520  032713  000200                  BIT     #BIT7,(R3)      ;TEST TO SEE IF BIT 7 CLEARED
3719  013524  001402                          BEQ     2$              ;BIT 7 IS OK
3720  013526  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3721  013530  104003                          HLT     3               ;BIT 7 FAILED TO CLEAR
3722  013532  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3723
3724
3725                                          ;*************************** TEST 54 ****************************
3726                                          ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 8 RESET AND CLEAR TEST
3727                                          ;*WRITE BIT 8,AND TEST THAT IT WILL BE CLEARED AFTER A
3728                                          ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
3729                                          ;;*************************************************************
```

```
3730
3731                                    ;:************
3732                                    ;            *
3733                                    ; TEST 54
3734                                    ;            *
3735                                    ;:************
3736                                    ;:*************************************************************************
3737  013534  012737  000054  001226  TST54:  MOV     #54,@#TSTNO
3738  013542  012737  013630  001216           MOV     #TST55,NEXT
3739  013550  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
3740  013554  052713  000400                  BIS     #BIT8,(R3)      ;SET BIT 8 AT TRANSMITTER BUFFER REGISTER
3741  013560  005005                          CLR     R5              ;SET "EXPECTED"
3742  013562  052777  000400  165622          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3743  013570  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3744  013574  032713  000400                  BIT     #BIT8,(R3)      ;TEST BIT 8
3745  013600  001402                          BEQ     1$              ;BIT 8 IS CLEARED
3746  013602  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3747  013604  104003                          HLT     3               ;BIT 8 IS SET AND SHOULDN'T BE
3748  013606  052713  000400          1$:     BIS     #BIT8,(R3)      ;SET BIT 8 AGAIN
3749  013612  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
3750  013614  032713  000400                  BIT     #BIT8,(R3)      ;TEST TO SEE IF BIT 8 CLEARED
3751  013620  001402                          BEQ     2$              ;BIT 8 IS OK
3752  013622  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3753  013624  104003                          HLT     3               ;BIT 8 FAILED TO CLEAR
3754  013626  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3755
3756
3757                                    ;************************** TEST 55 **************************
3758                                    ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 9 RESET AND CLEAR TEST
3759                                    ;*WRITE BIT 9,AND TEST THAT IT WILL BE CLEARED AFTER A
3760                                    ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
3761                                    ;:*********************************************************************
3762
3763                                    ;:************
3764                                    ;            *
3765                                    ; TEST 55
3766                                    ;            *
3767                                    ;:************
3768                                    ;:*************************************************************************
3769  013630  012737  000055  001226  TST55:  MOV     #55,@#TSTNO
3770  013636  012737  013724  001216           MOV     #TST56,NEXT
3771  013644  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
3772  013650  052713  001000                  BIS     #BIT9,(R3)      ;SET BIT 9 AT TRANSMITTER BUFFER REGISTER
3773  013654  005005                          CLR     R5              ;SET "EXPECTED"
3774  013656  052777  000400  165526          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3775  013664  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3776  013670  032713  001000                  BIT     #BIT9,(R3)      ;TEST BIT 9
3777  013674  001402                          BEQ     1$              ;BIT 9 IS CLEARED
3778  013676  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3779  013700  104003                          HLT     3               ;BIT 9 IS SET AND SHOULDN'T BE
3780  013702  052713  001000          1$:     BIS     #BIT9,(R3)      ;SET BIT 9 AGAIN
3781  013706  005013                          CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
3782  013710  032713  001000                  BIT     #BIT9,(R3)      ;TEST TO SEE IF BIT 9 CLEARED
3783  013714  001402                          BEQ     2$              ;BIT 9 IS OK
3784  013716  011304                          MOV     (R3),R4         ;LOAD "FOUND"
3785  013720  104003                          HLT     3               ;BIT 9 FAILED TO CLEAR
```

```
3786  013722  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
3787
3788
3789                                          ;*************************** TEST 56 ****************************
3790                                          ;*TRANSMITTER BUFFER REGISTER READ/WRITE BIT 10 RESET AND CLEAR TEST
3791                                          ;*WRITE BIT 10,AND TEST THAT IT WILL BE CLEARED AFTER A
3792                                          ;*DEVICE RESET AND TRANSMITTER BUFFER REGISTER CLR INSTRUCTION
3793                                          ;;***********************************************************************
3794
3795                                     ;;*************
3796                                     ;             *
3797                                     ; TEST 56
3798                                     ;             *
3799                                     ;;*************
3800                                     ;;***********************************************************************
3801  013724  012737  000056  001226  TST56:  MOV     #56,@#TSTNO
3802  013732  012737  014020  001216          MOV     #TST57,NEXT
3803  013740  013703  001414                 MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
3804  013744  052713  002000                 BIS     #BIT10,(R3)     ;SET BIT 10 AT TRANSMITTER BUFFER REGISTER
3805  013750  005005                         CLR     R5              ;SET "EXPECTED"
3806  013752  052777  000400  165432         BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3807  013760  004737  005044                 JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3808  013764  032713  002000                 BIT     #BIT10,(R3)     ;TEST BIT 10
3809  013770  001402                         BEQ     1$              ;BIT 10 IS CLEARED
3810  013772  011304                         MOV     (R3),R4         ;LOAD "FOUND"
3811  013774  104003                         HLT     3               ;BIT 10 IS SET AND SHOULDN'T BE
3812  013776  052713  002000          1$:    BIS     #BIT10,(R3)     ;SET BIT 10 AGAIN
3813  014002  005013                         CLR     (R3)            ;CLEAR THE TRANSMITTER BUFFER REGISTER
3814  014004  032713  002000                 BIT     #BIT10,(R3)     ;TEST TO SEE IF BIT 10 CLEARED
3815  014010  001402                         BEQ     2$              ;BIT 10 IS OK
3816  014012  011304                         MOV     (R3),R4         ;LOAD "FOUND"
3817  014014  104003                         HLT     3               ;BIT 10 FAILED TO CLEAR
3818  014016  104400                  2$:    SCOPE                   ;SCOPE THIS TEST
3819
3820
3821                                          ;*************************** TEST 57 ****************************
3822                                          ;*RECEIVER BUFFER REGISTER TEST
3823                                          ;*TEST THAT RECEIVER BUFFER REGISTER CANNOT BE WRITTEN.
3824                                          ;*WRITE RECEIVER BUFFER REGISTER WITH ALL 1'S
3825                                          ;*AND VERIFY THAT ALL 0'S ARE READ BACK.
3826                                          ;;***********************************************************************
3827
3828                                     ;;*************
3829                                     ;             *
3830                                     ; TEST 57
3831                                     ;             *
3832                                     ;;*************
3833                                     ;;***********************************************************************
3834  014020  012737  000057  001226  TST57:  MOV     #57,@#TSTNO
3835  014026  012737  014060  001216          MOV     #TST60,NEXT
3836  014034  013703  001406                 MOV     RXDBUF,R3
3837                                                                  ;GET THE REGISTER.
3838  014040  005005                         CLR     R5              ;SET EXPECTED (ZERO)
3839  014042  012713  177777                 MOV     #-1,(R3)        ;WRITE REGISTER WITH ALL 1'S
3840  014046  011304                         MOV     (R3),R4         ;READ THE REGISTER.
3841  014050  020504                         CMP     R5,R4           ;IS THE REGISTER EQUAL TO ZERO.
```

```
3842  014052  001401                        BEQ     1$              ;BR IF OK.
3843  014054  104003                        HLT     3               ;REGISTER NOT ZERO.
3844  014056  104400                1$:     SCOPE                   ;SCOPE THIS TEST.
3845
3846
3847                                        ;*************************** TEST 60 ****************************
3848
3849                                        ;*PARAMETER STATUS REGISTER TEST
3850                                        ;*TEST THAT PARAMETER STATUS REGISTER CANNOT BE WRITTEN
3851                                        ;*READ THE PARAMETER STATUS REGISTER AND STORE THE DATA.
3852                                        ;*COMPLEMENT THE DATA AND  WRITE THE PARAMETER STATUS REGISTER
3853                                        ;*VERIFING THAT THE PARAMETER STATUS REGISTER DID NOT CHANGE.
3854                                        ;;************************************************************
3855
3856                                ;;*************
3857                                ;             *
3858                                ; TEST 60
3859                                ;             *
3860                                ;;*************
3861                                ;;************************************************************************
3862  014060  012737  000060  001226 TST60:  MOV     #60,@#TSTNO
3863  014066  012737  014122  001216         MOV     #TST61,NEXT
3864  014074  013703  001410                MOV     PARCSR,R3
3865                                                                ;GET THE REGISTER.
3866  014100  011305                        MOV     (R3),R5         ;READ THE REGISTER INTO R5
3867  014102  010504                        MOV     R5,R4           ;SAVE REG INTO R4
3868  014104  005104                        COM     R4              ;MAKE R4 OPPOSITE TO REGISTER
3869  014106  010413                        MOV     R4,(R3)         ;WRITE THE REGISTER WITH THE COMPLIMENT
3870  014110  011304                        MOV     (R3),R4         ;READ THE REGISTER.
3871  014112  020504                        CMP     R5,R4           ;IS THE REGISTER EQUAL TO ZERO.
3872  014114  001401                        BEQ     1$              ;BR IF OK.
3873  014116  104003                        HLT     3               ;REGISTER NOT ZERO.
3874  014120  104400                1$:     SCOPE                   ;SCOPE THIS TEST.
3875
3876
3877                                        ;*************************** TEST 61 ****************************
3878                                        ;*RECEIVER CONTROL REGISTER BIT 0 READ ONLY DEVICE RESET AND CLEAR TEST
3879                                        ;*WRITE BIT 0 A ONE AND VERIFY A ZERO IS READ BACK
3880                                        ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
3881                                        ;;************************************************************
3882
3883                                ;;*************
3884                                ;             *
3885                                ; TEST 61
3886                                ;             *
3887                                ;;*************
3888                                ;;************************************************************************
3889  014122  012737  000061  001226 TST61:  MOV     #61,@#TSTNO
3890  014130  012737  014234  001216         MOV     #TST62,NEXT
3891  014136  013703  001404                MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
3892  014142  012705  000001                MOV     #BIT0,R5        ;GET BIT 0
3893  014146  010513                        MOV     R5,(R3)         ;WRITE BIT 0 TO RECEIVER CONTROL REGISTER
3894  014150  011304                        MOV     (R3),R4         ;READ RECEIVER CONTROL REGISTER BACK
3895  014152  005005                        CLR     R5              ;SET "EXPECTED"
3896  014154  020504                        CMP     R5,R4           ;R5=GOOD,R4= ?
3897  014156  001401                        BEQ     5$              ;BIT 0 IS OK
```

```
3898  014160  104003                          HLT     3               ;BIT  FAILED TO CLR
3899  014162  012705  000001          5$:     MOV     #BIT0,R5        ;RELOAD THE BIT
3900  014166  010513                          MOV     R5,(R3)         ;WRITE BIT 0 TO THE REG
3901  014170  052777  000400  165214          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3902  014176  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3903  014202  011304                          MOV     (R3),R4         ;GET BIT 0
3904  014204  032704  000001                  BIT     #BIT0,R4        ;TEST BIT 0 FOR RESET CLR
3905  014210  001401                          BEQ     7$              ;BIT 0 IS OK
3906  014212  104003                          HLT     3               ;BIT  FAILED TO CLEAR
3907  014214  010513                  7$:     MOV     R5,(R3)         ;SET BIT 0
3908  014216  005013                          CLR     (R3)            ;CLR RECEIVER CONTROL REGISTER
3909  014220  011304                          MOV     (R3),R4         ;READ THE RECEIVER CONTROL REGISTER
3910  014222  005005                          CLR     R5              ;SET "EXPECTED"
3911  014224  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
3912  014226  001401                          BEQ     10$             ;BIT 0 IS OK
3913  014230  104003                          HLT     3               ;BIT  FAILED TO CLEAR
3914  014232  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
3915
3916
3917                                          ;*************************** TEST 62 ****************************
3918                                          ;*RECEIVER CONTROL REGISTER BIT 7 READ ONLY DEVICE RESET AND CLEAR TEST
3919                                          ;*WRITE BIT 7 A ONE AND VERIFY A ZERO IS READ BACK
3920                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
3921                                          ;;*****************************************************************
3922
3923                                  ;;*************
3924                                  ;             *
3925                                  ; TEST 62     *
3926                                  ;             *
3927                                  ;;*************
3928                                  ;;*****************************************************************************
3929  014234  012737  000062  001226  TST62:  MOV     #62,@#TSTNO
3930  014242  012737  014346  001216          MOV     #TST63,NEXT
3931  014250  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
3932  014254  012705  000200                  MOV     #BIT7,R5        ;GET BIT 7
3933  014260  010513                          MOV     R5,(R3)         ;WRITE BIT 7 TO RECEIVER CONTROL REGISTER
3934  014262  011304                          MOV     (R3),R4         ;READ RECEIVER CONTROL REGISTER BACK
3935  014264  005005                          CLR     R5              ;SET "EXPECTED"
3936  014266  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
3937  014270  001401                          BEQ     5$              ;BIT 7 IS OK
3938  014272  104003                          HLT     3               ;BIT  FAILED TO CLR
3939  014274  012705  000200          5$:     MOV     #BIT7,R5        ;RELOAD THE BIT
3940  014300  010513                          MOV     R5,(R3)         ;WRITE BIT 7 TO THE REG
3941  014302  052777  000400  165102          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
3942  014310  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
3943  014314  011304                          MOV     (R3),R4         ;GET BIT 7
3944  014316  032704  000200                  BIT     #BIT7,R4        ;TEST BIT 7 FOR RESET CLR
3945  014322  001401                          BEQ     7$              ;BIT 7 IS OK
3946  014324  104003                          HLT     3               ;BIT  FAILED TO CLEAR
3947  014326  010513                  7$:     MOV     R5,(R3)         ;SET BIT 7
3948  014330  005013                          CLR     (R3)            ;CLR RECEIVER CONTROL REGISTER
3949  014332  011304                          MOV     (R3),R4         ;READ THE RECEIVER CONTROL REGISTER
3950  014334  005005                          CLR     R5              ;SET "EXPECTED"
3951  014336  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
3952  014340  001401                          BEQ     10$             ;BIT 7 IS Ok
3953  014342  104003                          HLT     3               ;BIT  FAILED TO CLEAR
```

```
014344  104400                  10$:    SCOPE                   ;SCOPE THIS TEST


                                        ;*************************** TEST 63 ****************************
                                        ;*RECEIVER CONTROL REGISTER BIT 9 READ ONLY DEVICE RESET AND CLEAR TEST
                                        ;*WRITE BIT 9 A ONE AND VERIFY A ZERO IS READ BACK
                                        ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
                                        ;;*************************************************************


                                ;;*************
                                ;             *
                                ; TEST 63
                                ;             *
                                ;;*************
                                ;;*************************************************************************
014346  012737  000063  001226  TST63:  MOV     #63,@#TSTNO
014354  012737  014460  001216          MOV     #TST64,NEXT
014362  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
014366  012705  001000                  MOV     #BIT9,R5        ;GET BIT 9
014372  010513                          MOV     R5,(R3)         ;WRITE BIT 9 TO RECEIVER CONTROL REGISTER
014374  011304                          MOV     (R3),R4         ;READ RECEIVER CONTROL REGISTER BACK
014376  005005                          CLR     R5              ;SET "EXPECTED"
014400  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
014402  001401                          BEQ     5$              ;BIT 9 IS OK
014404  104003                          HLT     3               ;BIT  FAILED TO CLR
014406  012705  001000          5$:     MOV     #BIT9,R5        ;RELOAD THE BIT
014412  010513                          MOV     R5,(R3)         ;WRITE BIT 9 TO THE REG
014414  052777  000400  164770          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
014422  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
014426  011304                          MOV     (R3),R4         ;GET BIT 9
014430  032704  001000                  BIT     #BIT9,R4        ;TEST BIT 9 FOR RESET CLR
014434  001401                          BEQ     7$              ;BIT 9 IS OK
014436  104003                          HLT     3               ;BIT  FAILED TO CLEAR
014440  010513                  7$:     MOV     R5,(R3)         ;SET BIT 9
014442  005013                          CLR     (R3)            ;CLR RECEIVER CONTROL REGISTER
014444  011304                          MOV     (R3),R4         ;READ THE RECEIVER CONTROL REGISTER
014446  005005                          CLR     R5              ;SET "EXPECTED"
014450  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
014452  001401                          BEQ     10$             ;BIT 9 IS OK
014454  104003                          HLT     3               ;BIT  FAILED TO CLEAR
014456  104400                  10$:    SCOPE                   ;SCOPE THIS TEST


                                        ;*************************** TEST 64 ****************************
                                        ;*RECEIVER CONTROL REGISTER BIT 10 READ ONLY DEVICE RESET AND CLEAR TEST
                                        ;*WRITE BIT 10 A ONE AND VERIFY A ZERO IS READ BACK
                                        ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
                                        ;;*************************************************************


                                ;;*************
                                ;             *
                                ; TEST 64
                                ;             *
                                ;;*************
                                ;;*************************************************************************
014460  012737  000064  001226  TST64:  MOV     #64,@#TSTNO
```

```
4010  014466  012737  014572  001216          MOV     #TST65,NEXT
4011  014474  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
4012  014500  012705  002000                  MOV     #BIT10,R5       ;GET BIT 10
4013  014504  010513                          MOV     R5,(R3)         ;WRITE BIT 10 TO RECEIVER CONTROL REGISTER
4014  014506  011304                          MOV     (R3),R4         ;READ RECEIVER CONTROL REGISTER BACK
4015  014510  005005                          CLR     R5              ;SET "EXPECTED"
4016  014512  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4017  014514  001401                          BEQ     5$              ;BIT 10 IS OK
4018  014516  104003                          HLT     3               ;BIT  FAILED TO CLR
4019  014520  012705  002000          5$:     MOV     #BIT10,R5       ;RELOAD THE BIT
4020  014524  010513                          MOV     R5,(R3)         ;WRITE BIT 10 TO THE REG
4021  014526  052777  000400  164656          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4022  014534  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4023  014540  011304                          MOV     (R3),R4         ;GET BIT 10
4024  014542  032704  002000                  BIT     #BIT10,R4       ;TEST BIT 10 FOR RESET CLR
4025  014546  001401                          BEQ     7$              ;BIT 10 IS OK
4026  014550  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4027  014552  010513                  7$:     MOV     R5,(R3)         ;SET BIT 10
4028  014554  005013                          CLR     (R3)            ;CLR RECEIVER CONTROL REGISTER
4029  014556  011304                          MOV     (R3),R4         ;READ THE RECEIVER CONTROL REGISTER
4030  014560  005005                          CLR     R5              ;SET "EXPECTED"
4031  014562  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4032  014564  001401                          BEQ     10$             ;BIT 10 IS OK
4033  014566  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4034  014570  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
4035
4036
4037                                          ;************************** TEST 65 **************************
4038                                          ;*RECEIVER CONTROL REGISTER BIT 11 READ ONLY DEVICE RESET AND CLEAR TES
4039                                          ;*WRITE BIT 11 A ONE AND VERIFY A ZERO IS READ BACK
4040                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4041                                          ;;**********************************************************************
4042
4043                                  ;;*************
4044                                  ;             *
4045                                  ; TEST 65
4046                                  ;             *
4047                                  ;;*************
4048                                  ;;**********************************************************************
4049  014572  012737  000065  001226  TST65:  MOV     #65,@#TSTNO
4050  014600  012737  014704  001216          MOV     #TST66,NEXT
4051  014606  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
4052  014612  012705  004000                  MOV     #BIT11,R5       ;GET BIT 11
4053  014616  010513                          MOV     R5,(R3)         ;WRITE BIT 11 TO RECEIVER CONTROL REGISTER
4054  014620  011304                          MOV     (R3),R4         ;READ RECEIVER CONTROL REGISTER BACK
4055  014622  005005                          CLR     R5              ;SET "EXPECTED"
4056  014624  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4057  014626  001401                          BEQ     5$              ;BIT 11 IS OK
4058  014630  104003                          HLT     3               ;BIT  FAILED TO CLR
4059  014632  012705  004000          5$:     MOV     #BIT11,R5       ;RELOAD THE BIT
4060  014636  010513                          MOV     R5,(R3)         ;WRITE BIT 11 TO THE REG
4061  014640  052777  000400  164544          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4062  014646  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4063  014652  011304                          MOV     (R3),R4         ;GET BIT 11
4064  014654  032704  004000                  BIT     #BIT11,R4       ;TEST BIT 11 FOR RESET CLR
4065  014660  001401                          BEQ     7$              ;BIT 11 IS Ok
```

```
4066  014662  104003                  HLT     3               ;BIT  FAILED TO CLEAR
4067  014664  010513          7$:     MOV     R5,(R3)         ;SET BIT 11
4068  014666  005013                  CLR     (R3)            ;CLR RECEIVER CONTROL REGISTER
4069  014670  011304                  MOV     (R3),R4         ;READ THE RECEIVER CONTROL REGISTER
4070  014672  005005                  CLR     R5              ;SET "EXPECTED"
4071  014674  020504                  CMP     R5,R4           ;R5=GOOD,R4= ?
4072  014676  001401                  BEQ     10$             ;BIT 11 IS OK
4073  014700  104003                  HLT     3               ;BIT  FAILED TO CLEAR
4074  014702  104400          10$:    SCOPE                   ;SCOPE THIS TEST
4075
4076
4077                                  ;*************************** TEST 66 ****************************
4078                                  ;*RECEIVER CONTROL REGISTER BIT 12 READ ONLY DEVICE RESET AND CLEAR TEST
4079                                  ;*WRITE BIT 12 A ONE AND VERIFY A ZERO IS READ BACK
4080                                  ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4081                                  ;;*************************************************************
4082
4083                                  ;;************
4084                                  ;               *
4085                                  ; TEST 66
4086                                  ;               *
4087                                  ;;************
4088                                  ;;*************************************************************************
4089  014704  012737  000066  001226  TST66:  MOV     #66,@#TSTNO
4090  014712  012737  015016  001216          MOV     #TST67,NEXT
4091  014720  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
4092  014724  012705  010000                  MOV     #BIT12,R5       ;GET BIT 12
4093  014730  010513                          MOV     R5,(R3)         ;WRITE BIT 12 TO RECEIVER CONTROL REGISTER
4094  014732  011304                          MOV     (R3),R4         ;READ RECEIVER CONTROL REGISTER BACK
4095  014734  005005                          CLR     R5              ;SET "EXPECTED"
4096  014736  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4097  014740  001401                          BEQ     5$              ;BIT 12 IS OK
4098  014742  104003                          HLT     3               ;BIT  FAILED TO CLR
4099  014744  012705  010000          5$:     MOV     #BIT12,R5       ;RELOAD THE BIT
4100  014750  010513                          MOV     R5,(R3)         ;WRITE BIT 12 TO THE REG
4101  014752  052777  000400  164432          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4102  014760  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4103  014764  011304                          MOV     (R3),R4         ;GET BIT 12
4104  014766  032704  010000                  BIT     #BIT12,R4       ;TEST BIT 12 FOR RESET CLR
4105  014772  001401                          BEQ     7$              ;BIT 12 IS OK
4106  014774  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4107  014776  010513          7$:     MOV     R5,(R3)         ;SET BIT 12
4108  015000  005013                  CLR     (R3)            ;CLR RECEIVER CONTROL REGISTER
4109  015002  011304                  MOV     (R3),R4         ;READ THE RECEIVER CONTROL REGISTER
4110  015004  005005                  CLR     R5              ;SET "EXPECTED"
4111  015006  020504                  CMP     R5,R4           ;R5=GOOD,R4= ?
4112  015010  001401                  BEQ     10$             ;BIT 12 IS OK
4113  015012  104003                  HLT     3               ;BIT  FAILED TO CLEAR
4114  015014  104400          10$:    SCOPE                   ;SCOPE THIS TEST
4115
4116
4117                                  ;*************************** TEST 67 ****************************
4118                                  ;*RECEIVER CONTROL REGISTER BIT 13 READ ONLY DEVICE RESET AND CLEAR TEST
4119                                  ;*WRITE BIT 13 A ONE AND VERIFY A ZERO IS READ BACK
4120                                  ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4121                                  ;;*************************************************************
```

```
4122
4123                                    ;:************
4124                                    ;            *
4125                                    ; TEST 67
4126                                    ;            *
4127                                    ;:************
4128                                    ;:*************************************************************************
4129  015016  012737  000067  001226  TST67:  MOV     #67,@#TSTNO
4130  015024  012737  015130  001216           MOV     #TST70,NEXT
4131  015032  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
4132  015036  012705  020000                  MOV     #BIT13,R5       ;GET BIT 13
4133  015042  010513                          MOV     R5,(R3)         ;WRITE BIT 13 TO RECEIVER CONTROL REGISTER
4134  015044  011304                          MOV     (R3),R4         ;READ RECEIVER CONTROL REGISTER BACK
4135  015046  005005                          CLR     R5              ;SET "EXPECTED"
4136  015050  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4137  015052  001401                          BEQ     5$              ;BIT 13 IS OK
4138  015054  104003                          HLT     3               ;BIT  FAILED TO CLR
4139  015056  012705  020000           5$:     MOV     #BIT13,R5       ;RELOAD THE BIT
4140  015062  010513                          MOV     R5,(R3)         ;WRITE BIT 13 TO THE REG
4141  015064  052777  000400  164320          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4142  015072  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4143  015076  011304                          MOV     (R3),R4         ;GET BIT 13
4144  015100  032704  020000                  BIT     #BIT13,R4       ;TEST BIT 13 FOR RESET CLR
4145  015104  001401                          BEQ     7$              ;BIT 13 IS OK
4146  015106  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4147  015110  010513                   7$:    MOV     R5,(R3)         ;SET BIT 13
4148  015112  005013                          CLR     (R3)            ;CLR RECEIVER CONTROL REGISTER
4149  015114  011304                          MOV     (R3),R4         ;READ THE RECEIVER CONTROL REGISTER
4150  015116  005005                          CLR     R5              ;SET "EXPECTED"
4151  015120  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4152  015122  001401                          BEQ     10$             ;BIT 13 IS OK
4153  015124  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4154  015126  104400                   10$:   SCOPE                   ;SCOPE THIS TEST
4155
4156
4157                                          ;************************** TEST 70 **************************
4158                                          ;*RECEIVER CONTROL REGISTER BIT 14 READ ONLY DEVICE RESET AND CLEAR TEST
4159                                          ;*WRITE BIT 14 A ONE AND VERIFY A ZERO IS READ BACK
4160                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4161                                          ;:*************************************************************
4162
4163                                    ;:************
4164                                    ;            *
4165                                    ; TEST 70
4166                                    ;            *
4167                                    ;:************
4168                                    ;:*************************************************************************
4169  015130  012737  000070  001226  TST70:  MOV     #70,@#TSTNO
4170  015136  012737  015242  001216           MOV     #TST71,NEXT
4171  015144  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
4172  015150  012705  040000                  MOV     #BIT14,R5       ;GET BIT 14
4173  015154  010513                          MOV     R5,(R3)         ;WRITE BIT 14 TO RECEIVER CONTROL REGISTER
4174  015156  011304                          MOV     (R3),R4         ;READ RECEIVER CONTROL REGISTER BACK
4175  015160  005005                          CLR     R5              ;SET "EXPECTED"
4176  015162  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4177  015164  001401                          BEQ     5$              ;BIT 14 IS OK
```

```
4178  015166  104003                          HLT     3               ;BIT  FAILED TO CLR
4179  015170  012705  040000          5$:     MOV     #BIT14,R5       ;RELOAD THE BIT
4180  015174  010513                          MOV     R5,(R3)         ;WRITE BIT 14 TO THE REG
4181  015176  052777  000400  164206          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4182  015204  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4183  015210  011304                          MOV     (R3),R4         ;GET BIT 14
4184  015212  032704  040000                  BIT     #BIT14,R4       ;TEST BIT 14 FOR RESET CLR
4185  015216  001401                          BEQ     7$              ;BIT 14 IS OK
4186  015220  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4187  015222  010513                  7$:     MOV     R5,(R3)         ;SET BIT 14
4188  015224  005013                          CLR     (R3)            ;CLR RECEIVER CONTROL REGISTER
4189  015226  011304                          MOV     (R3),R4         ;READ THE RECEIVER CONTROL REGISTER
4190  015230  005005                          CLR     R5              ;SET "EXPECTED"
4191  015232  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4192  015234  001401                          BEQ     10$             ;BIT 14 IS OK
4193  015236  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4194  015240  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
4195
4196
4197                                          ;*************************** TEST 71 ****************************
4198                                          ;*RECEIVER CONTROL REGISTER BIT 15 READ ONLY DEVICE RESET AND CLEAR TEST
4199                                          ;*WRITE BIT 15 A ONE AND VERIFY A ZERO IS READ BACK
4200                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4201                                          ;:*************************************************************
4202
4203                                  ;:*************
4204                                  ;             *
4205                                  ; TEST 71
4206                                  ;             *
4207                                  ;:*************
4208                                  ;:*************************************************************************
4209  015242  012737  000071  001226  TST71:  MOV     #71,@#TSTNO
4210  015250  012737  015354  001216          MOV     #TST72,NEXT
4211  015256  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
4212  015262  012705  100000                  MOV     #BIT15,R5       ;GET BIT 15
4213  015266  010513                          MOV     R5,(R3)         ;WRITE BIT 15 TO RECEIVER CONTROL REGISTER
4214  015270  011304                          MOV     (R3),R4         ;READ RECEIVER CONTROL REGISTER BACK
4215  015272  005005                          CLR     R5              ;SET "EXPECTED"
4216  015274  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4217  015276  001401                          BEQ     5$              ;BIT 15 IS OK
4218  015300  104003                          HLT     3               ;BIT  FAILED TO CLR
4219  015302  012705  100000          5$:     MOV     #BIT15,R5       ;RELOAD THE BIT
4220  015306  010513                          MOV     R5,(R3)         ;WRITE BIT 15 TO THE REG
4221  015310  052777  000400  164074          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4222  015316  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4223  015322  011304                          MOV     (R3),R4         ;GET BIT 15
4224  015324  032704  100000                  BIT     #BIT15,R4       ;TEST BIT 15 FOR RESET CLR
4225  015330  001401                          BEQ     7$              ;BIT 15 IS OK
4226  015332  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4227  015334  010513                  7$:     MOV     R5,(R3)         ;SET BIT 15
4228  015336  005013                          CLR     (R3)            ;CLR RECEIVER CONTROL REGISTER
4229  015340  011304                          MOV     (R3),R4         ;READ THE RECEIVER CONTROL REGISTER
4230  015342  005005                          CLR     R5              ;SET "EXPECTED"
4231  015344  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4232  015346  001401                          BEQ     10$             ;BIT 15 IS OK
4233  015350  104003                          HLT     3               ;BIT  FAILED TO CLEAR
```

```
 4234  015352  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
 4235
 4236
 4237                                          ;*************************** TEST 72 ****************************
 4238                                          ;*RECEIVER BUFFER REGISTER BIT 0 READ ONLY DEVICE RESET AND CLEAR TEST
 4239                                          ;*WRITE BIT 0 A ONE AND VERIFY A ZERO IS READ BACK
 4240                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
 4241                                          ;;*************************************************************
 4242
 4243                                  ;:************
 4244                                  ;            *
 4245                                  ; TEST 72
 4246                                  ;            *
 4247                                  ;:************
 4248                                  ;;*************************************************************************
 4249  015354  012737  000072  001226  TST72:  MOV     #72,@#TSTNO
 4250  015362  012737  015466  001216          MOV     #TST73,NEXT
 4251  015370  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
 4252  015374  012705  000001                  MOV     #BIT0,R5        ;GET BIT 0
 4253  015400  010513                          MOV     R5,(R3)         ;WRITE BIT 0 TO RECEIVER BUFFER REGISTER
 4254  015402  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
 4255  015404  005005                          CLR     R5              ;SET "EXPECTED"
 4256  015406  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
 4257  015410  001401                          BEQ     5$              ;BIT 0 IS OK
 4258  015412  104003                          HLT     3               ;BIT  FAILED TO CLR
 4259  015414  012705  000001          5$:     MOV     #BIT0,R5        ;RELOAD THE BIT
 4260  015420  010513                          MOV     R5,(R3)         ;WRITE BIT 0 TO THE REG
 4261  015422  052777  000400  163762          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
 4262  015430  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
 4263  015434  011304                          MOV     (R3),R4         ;GET BIT 0
 4264  015436  032704  000001                  BIT     #BIT0,R4        ;TEST BIT 0 FOR RESET CLR
 4265  015442  001401                          BEQ     7$              ;BIT 0 IS OK
 4266  015444  104003                          HLT     3               ;BIT  FAILED TO CLEAR
 4267  015446  010513                  7$:     MOV     R5,(R3)         ;SET BIT 0
 4268  015450  005013                          CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
 4269  015452  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
 4270  015454  005005                          CLR     R5              ;SET "EXPECTED"
 4271  015456  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
 4272  015460  001401                          BEQ     10$             ;BIT 0 IS OK
 4273  015462  104003                          HLT     3               ;BIT  FAILED TO CLEAR
 4274  015464  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
 4275
 4276
 4277                                          ;*************************** TEST 73 ****************************
 4278                                          ;*RECEIVER BUFFER REGISTER BIT 1 READ ONLY DEVICE RESET AND CLEAR TEST
 4279                                          ;*WRITE BIT 1 A ONE AND VERIFY A ZERO IS READ BACk
 4280                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
 4281                                          ;;*************************************************************
 4282
 4283                                  ;:************
 4284                                  ;            *
 4285                                  ; TEST 73
 4286                                  ;            *
 4287                                  ;:************
 4288                                  ;;*************************************************************************
 4289  015466  012737  000073  001226  TST73:  MOV     #73,@#TSTNO
```

```
4290  015474  012737  015600  001216          MOV     #TST74,NEXT
4291  015502  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4292  015506  012705  000002                  MOV     #BIT1,R5        ;GET BIT 1
4293  015512  010513                          MOV     R5,(R3)         ;WRITE BIT 1 TO RECEIVER BUFFER REGISTER
4294  015514  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4295  015516  005005                          CLR     R5              ;SET "EXPECTED"
4296  015520  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4297  015522  001401                          BEQ     5$              ;BIT 1 IS OK
4298  015524  104003                          HLT     3               ;BIT  FAILED TO CLR
4299  015526  012705  000002          5$:     MOV     #BIT1,R5        ;RELOAD THE BIT
4300  015532  010513                          MOV     R5,(R3)         ;WRITE BIT 1 TO THE REG
4301  015534  052777  000400  163650          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4302  015542  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4303  015546  011304                          MOV     (R3),R4         ;GET BIT 1
4304  015550  032704  000002                  BIT     #BIT1,R4        ;TEST BIT 1 FOR RESET CLR
4305  015554  001401                          BEQ     7$              ;BIT 1 IS OK
4306  015556  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4307  015560  010513                  7$:     MOV     R5,(R3)         ;SET BIT 1
4308  015562  005013                          CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4309  015564  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4310  015566  005005                          CLR     R5              ;SET "EXPECTED"
4311  015570  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4312  015572  001401                          BEQ     10$             ;BIT 1 IS OK
4313  015574  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4314  015576  104400                 10$:     SCOPE                   ;SCOPE THIS TEST
4315
4316
4317                                          ;************************** TEST 74 **************************
4318                                          ;*RECEIVER BUFFER REGISTER BIT 2 READ ONLY DEVICE RESET AND CLEAR TEST
4319                                          ;*WRITE BIT 2 A ONE AND VERIFY A ZERO IS READ BACK
4320                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4321                                          ;*************************************************************
4322
4323                                  ;:************
4324                                  ;            *
4325                                  ; TEST 74
4326                                  ;            *
4327                                  ;:************
4328                                  ;:*************************************************************************
4329  015600  012737  000074  001226  TST74:  MOV     #74,@#TSTNO
4330  015606  012737  015712  001216          MOV     #TST75,NEXT
4331  015614  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4332  015620  012705  000004                  MOV     #BIT2,R5        ;GET BIT 2
4333  015624  010513                          MOV     R5,(R3)         ;WRITE BIT 2 TO RECEIVER BUFFER REGISTER
4334  015626  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4335  015630  005005                          CLR     R5              ;SET "EXPECTED"
4336  015632  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4337  015634  001401                          BEQ     5$              ;BIT 2 IS OK
4338  015636  104003                          HLT     3               ;BIT  FAILED TO CLR
4339  015640  012705  000004          5$:     MOV     #BIT2,R5        ;RELOAD THE BIT
4340  015644  010513                          MOV     R5,(R3)         ;WRITE BIT 2 TO THE REG
4341  015646  052777  000400  163536          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4342  015654  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4343  015660  011304                          MOV     (R3),R4         ;GET BIT 2
4344  015662  032704  000004                  BIT     #BIT2,R4        ;TEST BIT 2 FOR RESET CLR
4345  015666  001401                          BEQ     7$              ;BIT 2 IS OK
```

```
4346  015670  104003                  HLT     3               ;BIT  FAILED TO CLEAR
4347  015672  010513          7$:     MOV     R5,(R3)         ;SET BIT 2
4348  015674  005013                  CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4349  015676  011304                  MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4350  015700  005005                  CLR     R5              ;SET "EXPECTED"
4351  015702  020504                  CMP     R5,R4           ;R5=GOOD,R4= ?
4352  015704  001401                  BEQ     10$             ;BIT 2 IS OK
4353  015706  104003                  HLT     3               ;BIT  FAILED TO CLEAR
4354  015710  104400          10$:    SCOPE                   ;SCOPE THIS TEST
4355
4356
4357                                  ;************************** TEST 75 **************************
4358                                  ;*RECEIVER BUFFER REGISTER BIT 3 READ ONLY DEVICE RESET AND CLEAR TEST
4359                                  ;*WRITE BIT 3 A ONE AND VERIFY A ZERO IS READ BACK
4360                                  ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4361                                  ;:**********************************************************
4362
4363                                  ;:***********
4364                                  ;           *
4365                                  ; TEST 75
4366                                  ;           *
4367                                  ;:***********
4368                                  ;:**********************************************************
4369  015712  012737  000075  001226  TST75:  MOV     #75,@#TSTNO
4370  015720  012737  016024  001216          MOV     #TST76,NEXT
4371  015726  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4372  015732  012705  000010                  MOV     #BIT3,R5        ;GET BIT 3
4373  015736  010513                          MOV     R5,(R3)         ;WRITE BIT 3 TO RECEIVER BUFFER REGISTER
4374  015740  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4375  015742  005005                          CLR     R5              ;SET "EXPECTED"
4376  015744  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4377  015746  001401                          BEQ     5$              ;BIT 3 IS OK
4378  015750  104003                          HLT     3               ;BIT  FAILED TO CLR
4379  015752  012705  000010          5$:     MOV     #BIT3,R5        ;RELOAD THE BIT
4380  015756  010513                          MOV     R5,(R3)         ;WRITE BIT 3 TO THE REG
4381  015760  052777  000400  163424          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4382  015766  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4383  015772  011304                          MOV     (R3),R4         ;GET BIT 3
4384  015774  032704  000010                  BIT     #BIT3,R4        ;TEST BIT 3 FOR RESET CLR
4385  016000  001401                          BEQ     7$              ;BIT 3 IS OK
4386  016002  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4387  016004  010513          7$:     MOV     R5,(R3)         ;SET BIT 3
4388  016006  005013                  CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4389  016010  011304                  MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4390  016012  005005                  CLR     R5              ;SET "EXPECTED"
4391  016014  020504                  CMP     R5,R4           ;R5=GOOD,R4= ?
4392  016016  001401                  BEQ     10$             ;BIT 3 IS OK
4393  016020  104003                  HLT     3               ;BIT  FAILED TO CLEAR
4394  016022  104400          10$:    SCOPE                   ;SCOPE THIS TEST
4395
4396
4397                                  ;************************** TEST 76 **************************
4398                                  ;*RECEIVER BUFFER REGISTER BIT 4 READ ONLY DEVICE RESET AND CLEAR TEST
4399                                  ;*WRITE BIT 4 A ONE AND VERIFY A ZERO IS READ BACK
4400                                  ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4401                                  ;:**********************************************************
```

```
                                        ;:************
                                        ;            *
                                        ; TEST 76
                                        ;            *
                                        ;:************
                                        ;:*************************************************************************
016024  012737  000076  001226 TST76:   MOV     #76,@#TSTNO
016032  012737  016136  001216          MOV     #TST77,NEXT
016040  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
016044  012705  000020                  MOV     #BIT4,R5        ;GET BIT 4
016050  010513                          MOV     R5,(R3)         ;WRITE BIT 4 TO RECEIVER BUFFER REGISTER
016052  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
016054  005005                          CLR     R5              ;SET "EXPECTED"
016056  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
016060  001401                          BEQ     5$              ;BIT 4 IS OK
016062  104003                          HLT     3               ;BIT  FAILED TO CLR
016064  012705  000020          5$:     MOV     #BIT4,R5        ;RELOAD THE BIT
016070  010513                          MOV     R5,(R3)         ;WRITE BIT 4 TO THE REG
016072  052777  000400  163312          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
016100  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
016104  011304                          MOV     (R3),R4         ;GET BIT 4
016106  032704  000020                  BIT     #BIT4,R4        ;TEST BIT 4 FOR RESET CLR
016112  001401                          BEQ     7$              ;BIT 4 IS OK
016114  104003                          HLT     3               ;BIT  FAILED TO CLEAR
016116  010513                  7$:     MOV     R5,(R3)         ;SET BIT 4
016120  005013                          CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
016122  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
016124  005005                          CLR     R5              ;SET "EXPECTED"
016126  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
016130  001401                          BEQ     10$             ;BIT 4 IS OK
016132  104003                          HLT     3               ;BIT  FAILED TO CLEAR
016134  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
                                        ;
                                        ;
                                        ;*************************** TEST 77 ****************************
                                        ;*RECEIVER BUFFER REGISTER BIT 5 READ ONLY DEVICE RESET AND CLEAR TEST
                                        ;*WRITE BIT 5 A ONE AND VERIFY A ZERO IS READ BACK
                                        ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
                                        ;:*************************************************************************
                                        ;
                                        ;:************
                                        ;            *
                                        ; TEST 77
                                        ;            *
                                        ;:************
                                        ;:*************************************************************************
016136  012737  000077  001226 TST77:   MOV     #77,@#TSTNO
016144  012737  016250  001216          MOV     #TST100,NEXT
016152  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
016156  012705  000040                  MOV     #BIT5,R5        ;GET BIT 5
016162  010513                          MOV     R5,(R3)         ;WRITE BIT 5 TO RECEIVER BUFFER REGISTER
016164  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
016166  005005                          CLR     R5              ;SET "EXPECTED"
016170  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
016172  001401                          BEQ     5$              ;BIT 5 IS OK
```

```
4458  016174  104003                          HLT     3               ;BIT  FAILED TO CLR
4459  016176  012705  000040          5$:     MOV     #BIT5,R5        ;RELOAD THE BIT
4460  016202  010513                          MOV     R5,(R3)         ;WRITE BIT 5 TO THE REG
4461  016204  052777  000400  163200          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4462  016212  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4463  016216  011304                          MOV     (R3),R4         ;GET BIT 5
4464  016220  032704  000040                  BIT     #BIT5,R4        ;TEST BIT 5 FOR RESET CLR
4465  016224  001401                          BEQ     7$              ;BIT 5 IS OK
4466  016226  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4467  016230  010513                  7$.     MOV     R5,(R3)         ;SET BIT 5
4468  016232  005013                          CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4469  016234  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4470  016236  005005                          CLR     R5              ;SET "EXPECTED"
4471  016240  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4472  016242  001401                          BEQ     10$             ;BIT 5 IS OK
4473  016244  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4474  016246  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
4475
4476
4477                                          ;*************************** TEST 100 ****************************
4478                                          ;*RECEIVER BUFFER REGISTER BIT 6 READ ONLY DEVICE RESET AND CLEAR TEST
4479                                          ;*WRITE BIT 6 A ONE AND VERIFY A ZERO IS READ BACK
4480                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4481                                          ;;*************************************************************************
4482
4483                                  ;;*************
4484                                  ;             *
4485                                  ; TEST 100    *
4486                                  ;             *
4487                                  ;;*************
4488                                  ;;*************************************************************************
4489  016250  012737  000100  001226  TST100: MOV     #100,@#TSTNO
4490  016256  012737  016362  001216          MOV     #TST101,NEXT
4491  016264  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4492  016270  012705  000100                  MOV     #BIT6,R5        ;GET BIT 6
4493  016274  010513                          MOV     R5,(R3)         ;WRITE BIT 6 TO RECEIVER BUFFER REGISTER
4494  016276  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4495  016300  005005                          CLR     R5              ;SET "EXPECTED"
4496  016302  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4497  016304  001401                          BEQ     5$              ;BIT 6 IS OK
4498  016306  104003                          HLT     3               ;BIT  FAILED TO CLR
4499  016310  012705  000100          5$:     MOV     #BIT6,R5        ;RELOAD THE BIT
4500  016314  010513                          MOV     R5,(R3)         ;WRITE BIT 6 TO THE REG
4501  016316  052777  000400  163066          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4502  016324  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4503  016330  011304                          MOV     (R3),R4         ;GET BIT 6
4504  016332  032704  000100                  BIT     #BIT6,R4        ;TEST BIT 6 FOR RESET CLR
4505  016336  001401                          BEQ     7$              ;BIT 6 IS OK
4506  016340  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4507  016342  010513                  7$:     MOV     R5,(R3)         ;SET BIT 6
4508  016344  005013                          CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4509  016346  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4510  016350  005005                          CLR     R5              ;SET "EXPECTED"
4511  016352  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4512  016354  001401                          BEQ     10$             ;BIT 6 IS OK
4513  016356  104003                          HLT     3               ;BIT  FAILED TO CLEAR
```

```
4514  016360  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
4515
4516
4517                                          ;*************************** TEST 101 ****************************
4518                                          ;*RECEIVER BUFFER REGISTER BIT 7 READ ONLY DEVICE RESET AND CLEAR TEST
4519                                          ;*WRITE BIT 7 A ONE AND VERIFY A ZERO IS READ BACK
4520                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4521                                          ;;*************************************************************************
4522
4523                                     ;:************
4524                                     ;            *
4525                                     ; TEST 101
4526                                     ;            *
4527                                     ;:************
4528                                     ;;*************************************************************************
4529  016362  012737  000101  001226  TST101: MOV     #101,@#TSTNO
4530  016370  012737  016474  001216          MOV     #TST102,NEXT
4531  016376  013703  001406                 MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4532  016402  012705  000200                 MOV     #BIT7,R5        ;GET BIT 7
4533  016406  010513                          MOV     R5,(R3)         ;WRITE BIT 7 TO RECEIVER BUFFER REGISTER
4534  016410  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4535  016412  005005                          CLR     R5              ;SET "EXPECTED"
4536  016414  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4537  016416  001401                          BEQ     5$              ;BIT 7 IS OK
4538  016420  104003                          HLT     3               ;BIT  FAILED TO CLR
4539  016422  012705  000200          5$:     MOV     #BIT7,R5        ;RELOAD THE BIT
4540  016426  010513                          MOV     R5,(R3)         ;WRITE BIT 7 TO THE REG
4541  016430  052777  000400  162754          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4542  016436  004737  005044                 JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4543  016442  011304                          MOV     (R3),R4         ;GET BIT 7
4544  016444  032704  000200                 BIT     #BIT7,R4        ;TEST BIT 7 FOR RESET CLR
4545  016450  001401                          BEQ     7$              ;BIT 7 IS OK
4546  016452  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4547  016454  010513                  7$:     MOV     R5,(R3)         ;SET BIT 7
4548  016456  005013                          CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4549  016460  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4550  016462  005005                          CLR     R5              ;SET "EXPECTED"
4551  016464  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4552  016466  001401                          BEQ     10$             ;BIT 7 IS OK
4553  016470  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4554  016472  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
4555
4556
4557                                          ;*************************** TEST 102 ****************************
4558                                          ;*RECEIVER BUFFER REGISTER BIT 8 READ ONLY DEVICE RESET AND CLEAR TEST
4559                                          ;*WRITE BIT 8 A ONE AND VERIFY A ZERO IS READ BACK
4560                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4561                                          ;;*************************************************************************
4562
4563                                     ;:************
4564                                     ;            *
4565                                     ; TEST 102
4566                                     ;            *
4567                                     ;:************
4568                                     ;;*************************************************************************
4569  016474  012737  000102  001226  TST102: MOV     #102,@#TSTNO
```

```
4570  016502  012737  016606  001216          MOV     #TST103,NEXT
4571  016510  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4572  016514  012705  000400                  MOV     #BIT8,R5        ;GET BIT 8
4573  016520  010513                          MOV     R5,(R3)         ;WRITE BIT 8 TO RECEIVER BUFFER REGISTER
4574  016522  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4575  016524  005005                          CLR     R5              ;SET "EXPECTED"
4576  016526  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4577  016530  001401                          BEQ     5$              ;BIT 8 IS OK
4578  016532  104003                          HLT     3               ;BIT  FAILED TO CLR
4579  016534  012705  000400          5$:     MOV     #BIT8,R5        ;RELOAD THE BIT
4580  016540  010513                          MOV     R5,(R3)         ;WRITE BIT 8 TO THE REG
4581  016542  052777  000400  162642          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4582  016550  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4583  016554  011304                          MOV     (R3),R4         ;GET BIT 8
4584  016556  032704  000400                  BIT     #BIT8,R4        ;TEST BIT 8 FOR RESET CLR
4585  016562  001401                          BEQ     7$              ;BIT 8 IS OK
4586  016564  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4587  016566  010513                  7$:     MOV     R5,(R3)         ;SET BIT 8
4588  016570  005013                          CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4589  016572  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4590  016574  005005                          CLR     R5              ;SET "EXPECTED"
4591  016576  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4592  016600  001401                          BEQ     10$             ;BIT 8 IS OK
4593  016602  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4594  016604  104400                 10$:     SCOPE                   ;SCOPE THIS TEST
4595
4596
4597                                          ;*************************** TEST 103 ****************************
4598                                          ;*RECEIVER BUFFER REGISTER BIT 9 READ ONLY DEVICE RESET AND CLEAR TEST
4599                                          ;*WRITE BIT 9 A ONE AND VERIFY A ZERO IS READ BACk
4600                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4601                                          ;;************************************************************************
4602
4603                                  ;;*************
4604                                  ;           *
4605                                  ; TEST 103
4606                                  ;           *
4607                                  ;;*************
4608                                  ;;************************************************************************
4609  016606  012737  000103  001226  TST103: MOV     #103,@#TSTNO
4610  016614  012737  016720  001216          MOV     #TST104,NEXT
4611  016622  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4612  016626  012705  001000                  MOV     #BIT9,R5        ;GET BIT 9
4613  016632  010513                          MOV     R5,(R3)         ;WRITE BIT 9 TO RECEIVER BUFFER REGISTER
4614  016634  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4615  016636  005005                          CLR     R5              ;SET "EXPECTED"
4616  016640  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4617  016642  001401                          BEQ     5$              ;BIT 9 IS OK
4618  016644  104003                          HLT     3               ;BIT  FAILED TO CLR
4619  016646  012705  001000          5$:     MOV     #BIT9,R5        ;RELOAD THE BIT
4620  016652  010513                          MOV     R5,(R3)         ;WRITE BIT 9 TO THE REG
4621  016654  052777  000400  162530          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4622  016662  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4623  016666  011304                          MOV     (R3),R4         ;GET BIT 9
4624  016670  032704  001000                  BIT     #BIT9,R4        ;TEST BIT 9 FOR RESET CLR
4625  016674  001401                          BEQ     7$              ;BIT 9 IS OK
```

```
4626  016676  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4627  016700  010513                  7$:     MOV     R5,(R3)         ;SET BIT 9
4628  016702  005013                          CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4629  016704  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4630  016706  005005                          CLR     R5              ;SET "EXPECTED"
4631  016710  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4632  016712  001401                          BEQ     10$             ;BIT 9 IS OK
4633  016714  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4634  016716  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
4635
4636
4637                                          ;*************************** TEST 104 ****************************
4638                                          ;*RECEIVER BUFFER REGISTER BIT 10 READ ONLY DEVICE RESET AND CLEAR TEST
4639                                          ;*WRITE BIT 10 A ONE AND VERIFY A ZERO IS READ BACK
4640                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4641                                          ;;*************************************************************
4642
4643                                  ;;************
4644                                  ;            *
4645                                  ; TEST 104
4646                                  ;            *
4647                                  ;;************
4648                                  ;;*************************************************************************
4649  016720  012737  000104  001226  TST104: MOV     #104,@#TSTNO
4650  016726  012737  017032  001216          MOV     #TST10S,NEXT
4651  016734  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4652  016740  012705  002000                  MOV     #BIT10,R5       ;GET BIT 10
4653  016744  010513                          MOV     R5,(R3)         ;WRITE BIT 10 TO RECEIVER BUFFER REGISTER
4654  016746  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4655  016750  005005                          CLR     R5              ;SET "EXPECTED"
4656  016752  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4657  016754  001401                          BEQ     5$              ;BIT 10 IS OK
4658  016756  104003                          HLT     3               ;BIT  FAILED TO CLR
4659  016760  012705  002000          5$:     MOV     #BIT10,R5       ;RELOAD THE BIT
4660  016764  010513                          MOV     R5,(R3)         ;WRITE BIT 10 TO THE REG
4661  016766  052777  000400  162416          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4662  016774  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4663  017000  011304                          MOV     (R3),R4         ;GET BIT 10
4664  017002  032704  002000                  BIT     #BIT10,R4       ;TEST BIT 10 FOR RESET CLR
4665  017006  001401                          BEQ     7$              ;BIT 10 IS OK
4666  017010  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4667  017012  010513                  7$:     MOV     R5,(R3)         ;SET BIT 10
4668  017014  005013                          CLR      R3)            ;CLR RECEIVER BUFFER REGISTER
4669  017016  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4670  017020  005005                          CLR     R5              ;SET "EXPECTED"
4671  017022  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4672  017024  001401                          BEQ     10$             ;BIT 10 IS OK
4673  017026  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4674  017030  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
4675
4676
4677                                          ;*************************** TEST 105 ****************************
4678                                          ;*RECEIVER BUFFER REGISTER BIT 12 READ ONLY DEVICE RESET AND CLEAR TEST
4679                                          ;*WRITE BIT 12 A ONE AND VERIFY A ZERO IS READ BACK
4680                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4681                                          ;;*************************************************************
```

```
4682
4683                                    ;:************
4684                                    :           *
4685                                    ; TEST 105
4686                                    :           *
4687                                    ;:************
4688                                    ;:*************************************************************************
4689  017032  012737  000105  001226  TST105: MOV     #105,@#TSTNO
4690  017040  012737  017144  001216           MOV     #TST106,NEXT
4691  017046  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4692  017052  012705  010000                  MOV     #BIT12,R5       ;GET BIT 12
4693  017056  010513                           MOV     R5,(R3)         ;WRITE BIT 12 TO RECEIVER BUFFER REGISTER
4694  017060  011304                           MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4695  017062  005005                           CLR     R5              ;SET "EXPECTED"
4696  017064  020504                           CMP     R5,R4           ;R5=GOOD,R4= ?
4697  017066  001401                           BEQ     5$              ;BIT 12 IS OK
4698  017070  104003                           HLT     3               ;BIT  FAILED TO CLR
4699  017072  012705  010000           5$:     MOV     #BIT12,R5       ;RELOAD THE BIT
4700  017076  010513                           MOV     R5,(R3)         ;WRITE BIT 12 TO THE REG
4701  017100  052777  000400  162304          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4702  017106  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4703  017112  011304                           MOV     (R3),R4         ;GET BIT 12
4704  017114  032704  010000                  BIT     #BIT12,R4       ;TEST BIT 12 FOR RESET CLR
4705  017120  001401                           BEQ     7$              ;BIT 12 IS OK
4706  017122  104003                           HLT     3               ;BIT  FAILED TO CLEAR
4707  017124  010513                   7$:     MOV     R5,(R3)         ;SET BIT 12
4708  017126  005013                           CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4709  017130  011304                           MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4710  017132  005005                           CLR     R5              ;SET "EXPECTED"
4711  017134  020504                           CMP     R5,R4           ;R5=GOOD,R4= ?
4712  017136  001401                           BEQ     10$             ;BIT 12 IS OK
4713  017140  104003                           HLT     3               ;BIT  FAILED TO CLEAR
4714  017142  104400                   10$:    SCOPE                   ;SCOPE THIS TEST
4715
4716
4717                                    ;************************** TEST 106 **************************
4718                                    ;*RECEIVER BUFFER REGISTER BIT 14 READ ONLY DEVICE RESET AND CLEAR TEST
4719                                    ;*WRITE BIT 14 A ONE AND VERIFY A ZERO IS READ BACK
4720                                    ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4721                                    ;:*************************************************************
4722
4723                                    ;:************
4724                                    :           *
4725                                    ; TEST 106
4726                                    :           *
4727                                    ;:************
4728                                    ;:*************************************************************************
4729  017144  012737  000106  001226  TST106: MOV     #106,@#TSTNO
4730  017152  012737  017256  001216           MOV     #TST107,NEXT
4731  017160  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4732  017164  012705  040000                  MOV     #BIT14,R5       ;GET BIT 14
4733  017170  010513                           MOV     R5,(R3)         ;WRITE BIT 14 TO RECEIVER BUFFER REGISTER
4734  017172  011304                           MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4735  017174  005005                           CLR     R5              ;SET "EXPECTED"
4736  017176  020504                           CMP     R5,R4           ;R5=GOOD,R4= ?
4737  017200  001401                           BEQ     5$              ;BIT 14 IS OK
```

```
4738  017202  104003                          HLT     3               ;BIT  FAILED TO CLR
4739  017204  012705  040000          5$:     MOV     #BIT14,R5       ;RELOAD THE BIT
4740  017210  010513                          MOV     R5,(R3)         ;WRITE BIT 14 TO THE REG
4741  017212  052777  000400  162172          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4742  017220  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4743  017224  011304                          MOV     (R3),R4         ;GET BIT 14
4744  017226  032704  040000                  BIT     #BIT14,R4       ;TEST BIT 14 FOR RESET CLR
4745  017232  001401                          BEQ     7$              ;BIT 14 IS OK
4746  017234  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4747  017236  010513                  7$:     MOV     R5,(R3)         ;SET BIT 14
4748  017240  005013                          CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4749  017242  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4750  017244  005005                          CLR     R5              ;SET "EXPECTED"
4751  017246  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4752  017250  001401                          BEQ     10$             ;BIT 14 IS OK
4753  017252  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4754  017254  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
4755
4756
4757                                          ;************************** TEST 107 **************************
4758                                          ;*RECEIVER BUFFER REGISTER BIT 15 READ ONLY DEVICE RESET AND CLEAR TEST
4759                                          ;*WRITE BIT 15 P ONE AND VERIFY A ZERO IS READ BACK
4760                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4761                                          ;;**********************************************************************
4762
4763                                  ;;*************
4764                                  ;             *
4765                                  ; TEST 107
4766                                  ;             *
4767                                  ;;*************
4768                                  ;;**********************************************************************
4769  017256  012737  000107  001226  TST107: MOV     #107,@#TSTNO
4770  017264  012737  017370  001216          MOV     #TST110,NEXT
4771  017272  013703  001406                  MOV     RXDBUF,R3       ;GET THE RECEIVER BUFFER REGISTER
4772  017276  012705  100000                  MOV     #BIT15,R5       ;GET BIT 15
4773  017302  010513                          MOV     R5,(R3)         ;WRITE BIT 15 TO RECEIVER BUFFER REGISTER
4774  017304  011304                          MOV     (R3),R4         ;READ RECEIVER BUFFER REGISTER BACK
4775  017306  005005                          CLR     R5              ;SET "EXPECTED"
4776  017310  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4777  017312  001401                          BEQ     5$              ;BIT 15 IS OK
4778  017314  104003                          HLT     3               ;BIT  FAILED TO CLR
4779  017316  012705  100000          5$:     MOV     #BIT15,R5       ;RELOAD THE BIT
4780  017322  010513                          MOV     R5,(R3)         ;WRITE BIT 15 TO THE REG
4781  017324  052777  000400  162060          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4782  017332  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4783  017336  011304                          MOV     (R3),R4         ;GET BIT 15
4784  017340  032704  100000                  BIT     #BIT15,R4       ;TEST BIT 15 FOR RESET CLR
4785  017344  001401                          BEQ     7$              ;BIT 15 IS OK
4786  017346  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4787  017350  010513                  7$:     MOV     R5,(R3)         ;SET BIT 15
4788  017352  005013                          CLR     (R3)            ;CLR RECEIVER BUFFER REGISTER
4789  017354  011304                          MOV     (R3),R4         ;READ THE RECEIVER BUFFER REGISTER
4790  017356  005005                          CLR     R5              ;SET "EXPECTED"
4791  017360  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4792  017362  001401                          BEQ     10$             ;BIT 15 IS OK
4793  017364  104003                          HLT     3               ;BIT  FAILED TO CLEAR
```

```
 4794  017366  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
 4795
 4796                                          ;*************************** TEST 110 ****************************
 4797                                          ;*THIS TEST VERIFIES BIT7 OF THE  TRANSMITTER CONTROL REGISTER
 4798                                          ;*TEST THAT BIT 7 SETS AFTER A RESET AND MRESET
 4799                                          ;*VERIFY THAT WRITING THE LOW BYTE OF
 4800                                          ;*THE TRANSMITTER BUFFER REGISTER CLEARS BIT 7
 4801                                          ;:*************************************************************
 4802
 4803                                  ;:************
 4804                                  ;            *
 4805                                  ; TEST 110
 4806                                  ;            *
 4807                                  ;:************
 4808                                  ;:*************************************************************************
 4809  017370  012737  000110  001226  TST110: MOV     #110,@#TSTNO
 4810  017376  012737  017476  001216          MOV     #TST111,NEXT
 4811  017404  013702  001414                  MOV     TXDBUF,R2       ;LOAD SECOND REG
 4812  017410  013703  001412                  MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
 4813  017414  012705  000200                  MOV     #BIT7,R5        ;SET "EXPECTED"
 4814  017420  000005                          RESET
 4815  017422  011304                          MOV     (R3),R4         ;GET THE BIT
 4816  017424  020504                          CMP     R5,R4           ;R5=GOOD,R4=?
 4817  017426  001401                          BEQ     1$              ;ARE THEY THE SAME?
 4818  017430  104003                          HLT     3               ;NO--REPORT THE ERROR
 4819  017432  005005                  1$:     CLR     R5              ;SET "EXPECTED"
 4820  017434  005012                          CLR     (R2)            ;WRITE THE LOW BYTE  OF THE TXDBUF
 4821  017436  011304                          MOV     (R3),R4         ;READ BACK BIT 7
 4822  017440  020504                          CMP     R5,R4           ;R5=GOOD,R4=?
 4823  017442  001401                          BEQ     2$              ;ARE THEY THE SAME?
 4824  017444  104003                          HLT     3               ;NO-BIT 7 FAILED TO CLEAR
 4825  017446  012705  000200          2$:     MOV     #BIT7,R5        ;SET "EXPECTED"
 4826  017452  052777  000400  161732          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
 4827  017460  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
 4828  017464  011304                          MOV     (R3),R4         ;READ BACK BIT 7
 4829  017466  020504                          CMP     R5,R4           ;R5=GOOD,R4=?
 4830  017470  001401                          BEQ     3$              ;BRANCH IF BIT 7 OK
 4831  017472  104003                          HLT     3               ;BIT 7 FAILED TO SET AFTER A MRESET
 4832  017474  104400                  3$:     SCOPE                   ;SCOPE THIS TEST
 4833
 4834
 4835
 4836                                          ;*************************** TEST 111 ****************************
 4837                                          ;*TRANSMITTER CONTROL REGISTER BIT 9 READ ONLY DEVICE RESET AND CLEAR TEST
 4838                                          ;*WRITE BIT 9 A ONE AND VERIFY A ZERO IS READ BACK
 4839                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
 4840                                          ;:*************************************************************
 4841
 4842                                  ;:************
 4843                                  ;            *
 4844                                  ; TEST 111
 4845                                  ;            *
 4846                                  ;:************
 4847                                  ;:*************************************************************************
 4848  017476  012737  000111  001226  TST111: MOV     #111,@#TSTNO
 4849  017504  012737  017620  001216          MOV     #TST112,NEXT
```

```
4850  017512  013703  001412                  MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
4851  017516  012705  001000                  MOV     #BIT9,R5        ;GET BIT 9
4852  017522  010513                          MOV     R5,(R3)         ;WRITE BIT 9 TO TRANSMITTER CONTROL REGISTER
4853  017524  011304                          MOV     (R3),R4         ;READ TRANSMITTER CONTROL REGISTER BACK
4854  017526  005005                          CLR     R5              ;SET "EXPECTED"
4855  017530  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
4856  017534  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4857  017536  001401                          BEQ     5$              ;BIT 9 IS OK
4858  017540  104003                          HLT     3               ;BIT  FAILED TO CLR
4859  017542  012705  001000          5$:     MOV     #BIT9,R5        ;RELOAD THE BIT
4860  017546  010513                          MOV     R5,(R3)         ;WRITE BIT 9 TO THE REG
4861  017550  052777  000400  161634          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4862  017556  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4863  017562  011304                          MOV     (R3),R4         ;GET BIT 9
4864  017564  032704  001000                  BIT     #BIT9,R4        ;TEST BIT 9 FOR RESET CLR
4865  017570  001401                          BEQ     7$              ;BIT 9 IS OK
4866  017572  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4867  017574  010513                  7$:     MOV     R5,(R3)         ;SET BIT 9
4868  017576  005013                          CLR     (R3)            ;CLR TRANSMITTER CONTROL REGISTER
4869  017600  011304                          MOV     (R3),R4         ;READ THE TRANSMITTER CONTROL REGISTER
4870  017602  005005                          CLR     R5              ;SET "EXPECTED"
4871  017604  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
4872  017610  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4873  017612  001401                          BEQ     10$             ;BIT 9 IS OK
4874  017614  104003                          HLT     3               ;BIT  FAILED TO CLEAR
4875  017616  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
4876
4877
4878                                          ;************************* TEST 112 ****************************
4879                                          ;*TRANSMITTER CONTROL REGISTER BIT 14 READ ONLY DEVICE RESET AND CLEAR TEST
4880                                          ;*WRITE BIT 14 A ONE AND VERIFY A ZERO IS READ BACK
4881                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4882                                          ;:*************************************************************
4883
4884                                  ;:************
4885                                  ;            *
4886                                  ; TEST 112
4887                                  ;            *
4888                                  ;:************
4889                                  ;:*************************************************************************
4890  017620  012737  000112  001226  TST112: MOV     #112,@#TSTNO
4891  017626  012737  017742  001216          MOV     #TST113,NEXT
4892  017634  013703  001412                  MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
4893  017640  012705  040000                  MOV     #BIT14,R5       ;GET BIT 14
4894  017644  010513                          MOV     R5,(R3)         ;WRITE BIT 14 TO TRANSMITTER CONTROL REGISTER
4895  017646  011304                          MOV     (R3),R4         ;READ TRANSMITTER CONTROL REGISTER BACK
4896  017650  005005                          CLR     R5              ;SET "EXPECTED"
4897  017652  042704  000200                  BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
4898  017656  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
4899  017660  001401                          BEQ     5$              ;BIT 14 IS OK
4900  017662  104003                          HLT     3               ;BIT  FAILED TO CLR
4901  017664  012705  040000          5$:     MOV     #BIT14,R5       ;RELOAD THE BIT
4902  017670  010513                          MOV     R5,(R3)         ;WRITE BIT 14 TO THE REG
4903  017672  052777  000400  161512          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4904  017700  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4905  017704  011304                          MOV     (R3),R4         ;GET BIT 14
```

```
4906  017706  032704  040000            BIT     #BIT14,R4       ;TEST BIT 14 FOR RESET CLR
4907  017712  001401                    BEQ     7$              ;BIT 14 IS OK
4908  017714  104003                    HLT     3               ;BIT  FAILED TO CLEAR
4909  017716  010513            7$:     MOV     R5,(R3)         ;SET BIT 14
4910  017720  005013                    CLR     (R3)            ;CLR TRANSMITTER CONTROL REGISTER
4911  017722  011304                    MOV     (R3),R4         ;READ THE TRANSMITTER CONTROL REGISTER
4912  017724  005005                    CLR     R5              ;SET "EXPECTED"
4913  017726  042704  000200            BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
4914  017732  020504                    CMP     R5,R4           ;R5=GOOD,R4= ?
4915  017734  001401                    BEQ     10$             ;BIT 14 IS OK
4916  017736  104003                    HLT     3               ;BIT  FAILED TO CLEAR
4917  017740  104400            10$:    SCOPE                   ;SCOPE THIS TEST
4918
4919
4920                                    ;************************** TEST 113 **************************
4921                                    ;*TRANSMITTER CONTROL REGISTER BIT 15 READ ONLY DEVICE RESET AND CLEAR TEST
4922                                    ;*WRITE BIT 15 A ONE AND VERIFY A ZERO IS READ BACK
4923                                    ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
4924                                    ;:*************************************************************
4925
4926                                    ;:************
4927                                    ;            *
4928                                    ; TEST 113
4929                                    ;            *
4930                                    ;:************
4931                                    ;:**********.*************************************************
4932  017742  012737  000113  001226    TST113: MOV     #113,@#TSTNO
4933  017750  012737  020064  001216            MOV     #TST114,NEXT
4934  017756  013703  001412                    MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
4935  017762  012705  100000                    MOV     #BIT15,R5       ;GET BIT 15
4936  017766  010513                            MOV     R5,(R3)         ;WRITE BIT 15 TO TRANSMITTER CONTROL REGISTER
4937  017770  011304                            MOV     (R3),R4         ;READ TRANSMITTER CONTROL REGISTER BACK
4938  017772  005005                            CLR     R5              ;SET "EXPECTED"
4939  017774  042704  000200                    BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
4940  020000  020504                            CMP     R5,R4           ;R5=GOOD,R4= ?
4941  020002  001401                            BEQ     5$              ;BIT 15 IS OK
4942  020004  104003                            HLT     3               ;BIT  FAILED TO CLR
4943  020006  012705  100000            5$:     MOV     #BIT15,R5       ;RELOAD THE BIT
4944  020012  010513                            MOV     R5,(R3)         ;WRITE BIT 15 TO THE REG
4945  020014  052777  000400  161370            BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
4946  020022  004737  005044                    JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
4947  020026  011304                            MOV     (R3),R4         ;GET BIT 15
4948  020030  032704  100000                    BIT     #BIT15,R4       ;TEST BIT 15 FOR RESET CLR
4949  020034  001401                            BEQ     7$              ;BIT 15 IS OK
4950  020036  104003                            HLT     3               ;BIT  FAILED TO CLEAR
4951  020040  010513            7$:             MOV     R5,(R3)         ;SET BIT 15
4952  020042  005013                            CLR     (R3)            ;CLR TRANSMITTER CONTROL REGISTER
4953  020044  011304                            MOV     (R3),R4         ;READ THE TRANSMITTER CONTROL REGISTER
4954  020046  005005                            CLR     R5              ;SET "EXPECTED"
4955  020050  042704  000200                    BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
4956  020054  020504                            CMP     R5,R4           ;R5=GOOD,R4= ?
4957  020056  001401                            BEQ     10$             ;BIT 15 IS OK
4958  020060  104003                            HLT     3               ;BIT  FAILED TO CLEAR
4959  020062  104400            10$:    SCOPE                   ;SCOPE THIS TEST
4960
4961
```

```
                                        ;************************** TEST 114 **************************
                                        ;*TRANSMITTER BUFFER REGISTER BIT 12 READ ONLY DEVICE RESET AND CLEAR TEST
                                        ;*WRITE BIT 12 A ONE AND VERIFY A ZERO IS READ BACK
                                        ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
                                        ;;***********************************************************************

                                ;;*************
                                ;               *
                                ; TEST 114
                                ;               *
                                ;;*************
                                ;;***********************************************************************
020064  012737  000114  001226  TST114: MOV     #114,@#TSTNO
020072  012737  020176  001216          MOV     #TST115,NEXT
020100  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
020104  012705  010000                  MOV     #BIT12,R5       ;GET BIT 12
020110  010513                          MOV     R5,(R3)         ;WRITE BIT 12 TO TRANSMITTER BUFFER REGISTER
020112  011304                          MOV     (R3),R4         ;READ TRANSMITTER BUFFER REGISTER BACK
020114  005005                          CLR     R5              ;SET "EXPECTED"
020116  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
020120  001401                          BEQ     5$              ;BIT 12 IS OK
020122  104003                          HLT     3               ;BIT  FAILED TO CLR
020124  012705  010000          5$:     MOV     #BIT12,R5       ;RELOAD THE BIT
020130  010513                          MOV     R5,(R3)         ;WRITE BIT 12 TO THE REG
020132  052777  000400  161252          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
020140  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
020144  011304                          MOV     (R3),R4         ;GET BIT 12
020146  032704  010000                  BIT     #BIT12,R4       ;TEST BIT 12 FOR RESET CLR
020152  001401                          BEQ     7$              ;BIT 12 IS OK
020154  104003                          HLT     3               ;BIT  FAILED TO CLEAR
020156  010513                  7$:     MOV     R5,(R3)         ;SET BIT 12
020160  005013                          CLR     (R3)            ;CLR TRANSMITTER BUFFER REGISTER
020162  011304                          MOV     (R3),R4         ;READ THE TRANSMITTER BUFFER REGISTER
020164  005005                          CLR     R5              ;SET "EXPECTED"
020166  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
020170  001401                          BEQ     10$             ;BIT 12 IS OK
020172  104003                          HLT     3               ;BIT  FAILED TO CLEAR
020174  104400                  10$:    SCOPE                   ;SCOPE THIS TEST


                                        ;************************** TEST 115 **************************
                                        ;*TRANSMITTER BUFFER REGISTER BIT 13 READ ONLY DEVICE RESET AND CLEAR TEST
                                        ;*WRITE BIT 13 A ONE AND VERIFY A ZERO IS READ BACK
                                        ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
                                        ;;***********************************************************************

                                ;;*************
                                ;               *
                                ; TEST 115
                                ;               *
                                ;;*************
                                ;;***********************************************************************
020176  012737  000115  001226  TST115: MOV     #115,@#TSTNO
020204  012737  020310  001216          MOV     #TST116,NEXT
020212  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
020216  012705  020000                  MOV     #BIT13,R5       ;GET BIT 13
```

```
5018  020222  010513                          MOV     R5,(R3)         ;WRITE BIT 13 TO TRANSMITTER BUFFER REGISTER
5019  020224  011304                          MOV     (R3),R4         ;READ TRANSMITTER BUFFER REGISTER BACK
5020  020226  005005                          CLR     R5              ;SET "EXPECTED"
5021  020230  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
5022  020232  001401                          BEQ     5$              ;BIT 13 IS OK
5023  020234  104003                          HLT     3               ;BIT  FAILED TO CLR
5024  020236  012705  020000          5$:     MOV     #BIT13,R5       ;RELOAD THE BIT
5025  020242  010513                          MOV     R5,(R3)         ;WRITE BIT 13 TO THE REG
5026  020244  052777  000400  161140        BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5027  020252  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5028  020256  011304                          MOV     (R3),R4         ;GET BIT 13
5029  020260  032704  020000                  BIT     #BIT13,R4       ;TEST BIT 13 FOR RESET CLR
5030  020264  001401                          BEQ     7$              ;BIT 13 IS OK
5031  020266  104003                          HLT     3               ;BIT  FAILED TO CLEAR
5032  020270  010513                  7$:     MOV     R5,(R3)         ;SET BIT 13
5033  020272  005013                          CLR     (R3)            ;CLR TRANSMITTER BUFFER REGISTER
5034  020274  011304                          MOV     (R3),R4         ;READ THE TRANSMITTER BUFFER REGISTER
5035  020276  005005                          CLR     R5              ;SET "EXPECTED"
5036  020300  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
5037  020302  001401                          BEQ     10$             ;BIT 13 IS OK
5038  020304  104003                          HLT     3               ;BIT  FAILED TO CLEAR
5039  020306  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
5040
5041                                          ;************************** TEST 116 ****************************
5042                                          ;*TRANSMITTER BUFFER REGISTER BIT 14 READ ONLY DEVICE RESET AND CLEAR TEST
5043                                          ;*WRITE BIT 14 A ONE AND VERIFY A ZERO IS READ BACK
5044                                          ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
5045                                          ;:*****************************************************************
5046
5047
5048                                          ;:************
5049                                          ;            *
5050                                          ; TEST 116
5051                                          ;            *
5052                                          ;:************
5053                                          ;:*****************************************************************
5054  020310  012737  000116  001226  TST116: MOV     #116,@#TSTNO
5055  020316  012737  020422  001216          MOV     #TST117,NEXT
5056  020324  013703  001414                  MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
5057  020330  012705  040000                  MOV     #BIT14,R5       ;GET BIT 14
5058  020334  010513                          MOV     R5,(R3)         ;WRITE BIT 14 TO TRANSMITTER BUFFER REGISTER
5059  020336  011304                          MOV     (R3),R4         ;READ TRANSMITTER BUFFER REGISTER BACK
5060  020340  005005                          CLR     R5              ;SET "EXPECTED"
5061  020342  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
5062  020344  001401                          BEQ     5$              ;BIT 14 IS OK
5063  020346  104003                          HLT     3               ;BIT  FAILED TO CLR
5064  020350  012705  040000          5$:     MOV     #BIT14,R5       ;RELOAD THE BIT
5065  020354  010513                          MOV     R5,(R3)         ;WRITE BIT 14 TO THE REG
5066  020356  052777  000400  161026        BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5067  020364  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5068  020370  011304                          MOV     (R3),R4         ;GET BIT 14
5069  020372  032704  040000                  BIT     #BIT14,R4       ;TEST BIT 14 FOR RESET CLR
5070  020376  001401                          BEQ     7$              ;BIT 14 IS OK
5071  020400  104003                          HLT     3               ;BIT  FAILED TO CLEAR
5072  020402  010513                  7$:     MOV     R5,(R3)         ;SET BIT 14
5073  020404  005013                          CLR     (R3)            ;CLR TRANSMITTER BUFFER REGISTER
```

```
5074  020406  011304                        MOV     (R3),R4         ;READ THE TRANSMITTER BUFFER REGISTER
5075  020410  005005                        CLR     R5              ;SET "EXPECTED"
5076  020412  020504                        CMP     R5,R4           ;R5=GOOD,R4= ?
5077  020414  001401                        BEQ     10$             ;BIT 14 IS OK
5078  020416  104003                        HLT     3               ;BIT  FAILED TO CLEAR
5079  020420  104400                10$:    SCOPE                   ;SCOPE THIS TEST
5080
5081
5082                                        ;*************************** TEST 117 ****************************
5083                                        ;*TRANSMITTER BUFFER REGISTER BIT 15 READ ONLY DEVICE RESET AND CLEAR TEST
5084                                        ;*WRITE BIT 15 A ONE AND VERIFY A ZERO IS READ BACK
5085                                        ;*REPEAT FOR DEVICE RESET AND CLR INSTRUCTIONS.
5086                                        ;;*************************************************************
5087
5088                                    ;;************
5089                                    ;            *
5090                                    ; TEST 117
5091                                    ;            *
5092                                    ;;************
5093                                    ;;*************************************************************
5094  020422  012737  000117  001226 TST117: MOV     #117,@#TSTNO
5095  020430  012737  020534  001216        MOV     #TST120,NEXT
5096  020436  013703  001414                MOV     TXDBUF,R3       ;GET THE TRANSMITTER BUFFER REGISTER
5097  020442  012705  100000                MOV     #BIT15,R5       ;GET BIT 15
5098  020446  010513                        MOV     R5,(R3)         ;WRITE BIT 15 TO TRANSMITTER BUFFER REGISTER
5099  020450  011304                        MOV     (R3),R4         ;READ TRANSMITTER BUFFFR REGISTER BACK
5100  020452  005005                        CLR     R5              ;SET "EXPECTED"
5101  020454  020504                        CMP     R5,R4           ;R5=GOOD,R4= ?
5102  020456  001401                        BEQ     5$              ;BIT 15 IS OK
5103  020460  104003                        HLT     3               ;BIT  FAILED TO CLR
5104  020462  012705  100000         5$:    MOV     #BIT15,R5       ;RELOAD THE BIT
5105  020466  010513                        MOV     R5,(R3)         ;WRITE BIT 15 TO THE REG
5106  020470  052777  000400  160714        BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5107  020476  004737  005044                JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5108  020502  011304                        MOV     (R3),R4         ;GET BIT 15
5109  020504  032704  100000                BIT     #BIT15,R4       ;TEST BIT 15 FOR RESET CLR
5110  020510  001401                        BEQ     7$              ;BIT 15 IS OK
5111  020512  104003                        HLT     3               ;BIT  FAILED TO CLEAR
5112  020514  010513                 7$:    MOV     R5,(R3)         ;SET BIT 15
5113  020516  005013                        CLR     (R3)            ;CLR TRANSMITTER BUFFER REGISTER
5114  020520  011304                        MOV     (R3),R4         ;READ THE TRANSMITTER BUFFER REGISTER
5115  020522  005005                        CLR     R5              ;SET "EXPECTED"
5116  020524  020504                        CMP     R5,R4           ;R5=GOOD,R4= ?
5117  020526  001401                        BEQ     10$             ;BIT 15 IS OK
5118  020530  104003                        HLT     3               ;BIT  FAILED TO CLEAR
5119  020532  10440C                10$:    SCOPE                   ;SCOPE THIS TEST
5120
5121
5122                                        ;*************************** TEST 120 ****************************
5123                                        ;*PARAMETER STATUS REGISTER BIT 9 WRITE ONLY DEVICE RESET AND CLEAR TEST
5124                                        ;*TEST THAT BIT 9 CANNOT BE WRITTEN AND READ
5125                                        ;*BACK THE SAME.  READ BIT 9,COMPLEMENT IT AND WRITE IT
5126                                        ;*VERIFYING THAT IT  DID NOT CHANGE. WRITE BIT 9,
5127                                        ;*DO A DEVICE RESET AND VERIFY THE BIT WAS  CLEARED.
5128                                        ;*REPEAT FOR A CLR INSTRUCTION.
5129                                        ;;*************************************************************
```

```
5130
5131                                    ;:************
5132                                    ;            *
5133                                    ; TEST 120
5134                                    ;            *
5135                                    ;:************
5136                                    ;:*************************************************************************
5137  020534  012737  000120  001226  TST120: MOV     #120,@#TSTNO
5138  020542  012737  020662  001216           MOV     #TST121,NEXT
5139  020550  013703  001410                  MOV     PARCSR,R3       ;GET THE PARAMETER STATUS REGISTER
5140  020554  012705  001000                  MOV     #BIT9,R5        ;GET BIT 9
5141  020560  011305                          MOV     (R3),R5         ;READ THE REGISTER
5142  020562  010504                          MOV     R5,R4           ;SAVE THE BIT
5143  020564  032704  001000                  BIT     #BIT9,R4        ;CHECK BIT 9
5144  020570  001401                          BEQ     1$              ;BIT 9 IS OK
5145  020572  104003                          HLT     3               ;BIT  FAILED TO CLEAR
5146  020574  005104                  1$:     COM     R4              ;REVERSE THE BIT
5147  020576  010413                          MOV     R4,(R3)         ;WRITE THE BIT TO THE REG
5148  020600  011304                          MOV     (R3),R4         ;READ IT BACK
5149  020602  020504                          CMP     R5,R4           ;R5=GOOD, R4= ?
5150  020604  001401                          BEQ     3$              ;BR IF OK
5151  020606  104003                          HLT     3               ;COMPARE ERROR
5152  020610  012705  001000          3$:     MOV     #BIT9,R5        ;LOAD THE BIT
5153  020614  010513                          MOV     R5,(R3)         ;WRITE THE BIT TO THE REG
5154  020616  052777  000400  160566          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5155  020624  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5156  020630  011304                          MOV     (R3),R4         ;GET BIT 9
5157  020632  032704  001000                  BIT     #BIT9,R4        ;TEST BIT 9 FOR 0 AFTER RESET
5158  020636  001401                          BEQ     4$              ;BIT 9 IS OK
5159  020640  104003                          HLT     3               ;BIT  IS NOT A 0
5160  020642  010513                  4$:     MOV     R5,(R3)         ;WRITE THE BIT TO THE REG
5161  020644  005013                          CLR     (R3)            ;CLR THE PARAMETER STATUS REGISTER
5162  020646  011304                          MOV     (R3),R4         ;GET THE PARAMETER STATUS REGISTER
5163  020650  005005                          CLR     R5              ;SET "EXPECTED"
5164  020652  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
5165  020654  001401                          BEQ     10$             ;BIT 9 IS OK
5166  020656  104003                          HLT     3               ;BIT  FAILED TO CLEAR
5167  020660  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
5168
5169
5170                                          ;*************************** TEST 121 ****************************
5171                                          ;*PARAMETER STATUS REGISTER BIT 12 WRITE ONLY DEVICE RESET AND CLEAR TEST
5172                                          ;*TEST THAT BIT 12 CANNOT BE WRITTEN AND READ
5173                                          ;*BACK THE SAME.  READ BIT 12,COMPLEMENT IT AND WRITE IT
5174                                          ;*VERIFYING THAT IT  DID NOT CHANGE. WRITE BIT 12,
5175                                          ;*DO A DEVICE RESET AND VERIFY THE BIT WAS  CLEARED.
5176                                          ;*REPEAT FOR A CLR INSTRUCTION.
5177                                          ;:*************************************************************
5178
5179                                    ;:************
5180                                    ;            *
5181                                    ; TEST 121
5182                                    ;            *
5183                                    ;:************
5184                                    ;:*************************************************************************
5185  020662  012737  000121  001226  TST121: MOV     #121,@#TSTNO
```

```
5186  020670  012737  021010  001216          MOV     #TST122,NEXT
5187  020676  013703  001410                  MOV     PARCSR,R3       ;GET THE PARAMETER STATUS REGISTER
5188  020702  012705  010000                  MOV     #BIT12,R5       ;GET BIT 12
5189  020706  011305                          MOV     (R3),R5         ;READ THE REGISTER
5190  020710  010504                          MOV     R5,R4           ;SAVE THE BIT
5191  020712  032704  010000                  BIT     #BIT12,R4       ;CHECK BIT 12
5192  020716  001401                          BEQ     1$              ;BIT 12 IS OK
5193  020720  104003                          HLT     3               ;BIT  FAILED TO CLEAR
5194  020722  005104                  1$:     COM     R4              ;REVERSE THE BIT
5195  020724  010413                          MOV     R4,(R3)         ;WRITE THE BIT TO THE REG
5196  020726  011304                          MOV     (R3),R4         ;READ IT BACK
5197  020730  020504                          CMP     R5,R4           ;R5=GOOD, R4= ?
5198  020732  001401                          BEQ     3$              ;BR IF OK
5199  020734  104003                          HLT     3               ;COMPARE ERROR
5200  020736  012705  010000          3$:     MOV     #BIT12,R5       ;LOAD THE BIT
5201  020742  010513                          MOV     R5,(R3)         ;WRITE THE BIT TO THE REG
5202  020744  052777  000400  160440          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5203  020752  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5204  020756  011304                          MOV     (R3),R4         ;GET BIT 12
5205  020760  032704  010000                  BIT     #BIT12,R4       ;TEST BIT 12 FOR 0 AFTER RESET
5206  020764  001401                          BEQ     4$              ;BIT 12 IS OK
5207  020766  104003                          HLT     3               ;BIT  IS NOT A 0
5208  020770  010513                  4$:     MOV     R5,(R3)         ;WRITE THE BIT TO THE REG
5209  020772  005013                          CLR     (R3)            ;CLR THE PARAMETER STATUS REGISTER
5210  020774  011304                          MOV     (R3),R4         ;GET THE PARAMETER STATUS REGISTER
5211  020776  005005                          CLR     R5              ;SET "EXPECTED"
5212  021000  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
5213  021002  001401                          BEQ     10$             ;BIT 12 IS OK
5214  021004  104003                          HLT     3               ;BIT  FAILED TO CLEAR
5215  021006  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
5216
5217
5218                                          ;************************* TEST 122 ****************************
5219                                          ;*PARAMETER STATUS REGISTER BIT 15 WRITE ONLY DEVICE RESET AND CLEAR TEST
5220                                          ;*TEST THAT BIT 15 CANNOT BE WRITTEN AND READ
5221                                          ;*BACK THE SAME.  READ BIT 15,COMPLEMENT IT AND WRITE IT
5222                                          ;*VERIFYING THAT IT  DID NOT CHANGE. WRITE BIT 15,
5223                                          ;*DO A DEVICE RESET AND VERIFY THE BIT WAS  CLEARED.
5224                                          ;*REPEAT FOR A CLR INSTRUCTION.
5225                                          ;;**************************************************************
5226
5227                                  ;;*************
5228                                  ;*           *
5229                                  ; TEST 122
5230                                  ;*           *
5231                                  ;;*************
5232                                  ;;**************************************************************************
5233  021010  012737  000122  001226  TST122: MOV     #122,@#TSTNO
5234  021016  012737  021136  001216          MOV     #TST123,NEXT
5235  021024  013703  001410                  MOV     PARCSR,R3       ;GET THE PARAMETER STATUS REGISTER
5236  021030  012705  100000                  MOV     #BIT15,R5       ;GET BIT 15
5237  021034  011305                          MOV     (R3),R5         ;READ THE REGISTER
5238  021036  010504                          MOV     R5,R4           ;SAVE THE BIT
5239  021040  032704  100000                  BIT     #BIT15,R4       ;CHECK BIT 15
5240  021044  001401                          BEQ     1$              ;BIT 15 IS OK
5241  021046  104003                          HLT     3               ;BIT  FAILED TO CLEAR
```

```
5242  021050  005104                  1$:     COM     R4              ;REVERSE THE BIT
5243  021052  010413                          MOV     R4,(R3)         ;WRITE THE BIT TO THE REG
5244  021054  011304                          MOV     (R3),R4         ;READ IT BACK
5245  021056  020504                          CMP     R5,R4           ;R5=GOOD, R4= ?
5246  021060  001401                          BEQ     3$              ;BR IF OK
5247  021062  104003                          HLT     3               ;COMPARE ERROR
5248  021064  012705  100000          3$:     MOV     #BIT15,R5       ;LOAD THE BIT
5249  021070  010513                          MOV     R5,(R3)         ;WRITE THE BIT TO THE REG
5250  021072  052777  000400  160312          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5251  021100  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5252  021104  011304                          MOV     (R3),R4         ;GET BIT 15
5253  021106  032704  100000                  BIT     #BIT15,R4       ;TEST BIT 15 FOR 0 AFTER RESET
5254  021112  001401                          BEQ     4$              ;BIT 15 IS OK
5255  021114  104003                          HLT     3               ;BIT  IS NOT A 0
5256  021116  010513                  4$:     MOV     R5,(R3)         ;WRITE THE BIT TO THE REG
5257  021120  005013                          CLR     (R3)            ;CLR THE PARAMETER STATUS REGISTER
5258  021122  011304                          MOV     (R3),R4         ;GET THE PARAMETER STATUS REGISTER
5259  021124  005005                          CLR     R5              ;SET "EXPECTED"
5260  021126  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
5261  021130  001401                          BEQ     10$             ;BIT 15 IS OK
5262  021132  104003                          HLT     3               ;BIT  FAILED TO CLEAR
5263  021134  104400                  10$:    SCOPE                   ;SCOPE THIS TEST
5264
5265
5266                                          ;************************** TEST 123 ****************************
5267                                          ;*TRANSMITTER CONTROL REGISTER BIT 8 WRITE ONLY DEVICE RESET AND CLEAR TEST
5268                                          ;*TEST THAT BIT 8 CANNOT BE WRITTEN AND READ
5269                                          ;*BACK THE SAME.  READ BIT 8,COMPLEMENT IT AND WRITE IT
5270                                          ;*VERIFYING THAT IT  DID NOT CHANGE. WRITE BIT 8,
5271                                          ;*DO A DEVICE RESET AND VERIFY THE BIT WAS  CLEARED.
5272                                          ;*REPEAT FOR A CLR INSTRUCTION.
5273                                          ;;*************************************************************
5274
5275                                          ;;************
5276                                          ;            *
5277                                          ; TEST 123
5278                                          ;            *
5279                                          ;;************
5280                                          ;;*************************************************************
5281  021136  012737  000123  001226  TST123: MOV     #123,@#TSTNO
5282  021144  012737  021270  001216          MOV     #TST124,NEXT
5283  021152  013703  001412                  MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
5284  021156  012705  000200                  MOV     #BIT7,R5        ;SET EXPECTED BIT
5285  021162  011304                          MOV     (R3),R4         ;READ THE REGISTER
5286  021164  032704  000400                  BIT     #BIT8,R4        ;CHECK BIT 8
5287  021170  001401                          BEQ     1$              ;BIT 8 IS OK
5288  021172  104003                          HLT     3               ;BIT  FAILED TO CLEAR
5289  021174  005104                  1$:     COM     R4              ;REVERSE THE BIT
5290  021176  010413                          MOV     R4,(R3)         ;WRITE THE BIT TO THE REG
5291  021200  004737  005044                  JSR     PC,SMALL        ;WAIT FOR DEVICE RESET TO FINISH
5292  021204  011304                          MOV     (R3),R4         ;READ IT BACK
5293  021206  020504                          CMP     R5,R4           ;R5=GOOD, R4= ?
5294  021210  001401                          BEQ     3$              ;BR IF OK
5295  021212  104003                          HLT     3               ;COMPARE ERROR
5296  021214  012704  000400          3$:     MOV     #BIT8,R4        ;LOAD THE BIT
5297  021220  010413                          MOV     R4,(R3)         ;WRITE THE BIT TO THE REG
```

```
5298  021222  052777  000400  160162          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5299  021230  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5300  021234  011304                          MOV     (R3),R4         ;GET BIT 8
5301  021236  032704  000400                  BIT     #BIT8,R4        ;TEST BIT 8 FOR 0 AFTER RESET
5302  021242  001401                          BEQ     4$              ;BIT 8 IS OK
5303  021244  104003                          HLT     3               ;BIT  IS NOT A 0
5304  021246  012704  000400          4$:     MOV     #BIT8,R4        ;LOAD THE BIT
5305  021252  010413                          MOV     R4,(R3)         ;WRITE THE BIT TO THE REG
5306  021254  005013                          CLR     (R3)            ;CLR THE TRANSMITTER CONTROL REGISTER
5307  021256  011304                          MOV     (R3),R4         ;GET THE TRANSMITTER CONTROL REGISTER
5308  021260  020504                          CMP     R5,R4           ;R5=GOOD,R4= ?
5309  021262  001401                          BEQ     10$             ;BIT 8 IS OK
5310  021264  104003                          HLT     3               ;BIT  FAILED TO CLEAR
5311  021266  104400          10$:    SCOPE                           ;SCOPE THIS TEST
5312
5313
5314                                  ;************************** TEST 124 ****************************
5315                                  ;*RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
5316                                  ;*SET BIT8, VERIFY BIT8 WAS SET.
5317                                  ;*CLEAR BIT8, VERIFY BIT8 WAS CLEARED.
5318                                  ;;*************************************************************
5319
5320                          ;;************
5321                          ;           *
5322                          ; TEST 124
5323                          ;           *
5324                          ;;************
5325                          ;;*************************************************************************
5326  021270  012737  000124  001226  TST124: MOV     #124,@#TSTNO
5327  021276  012737  021344  001216          MOV     #TST125,NEXT
5328  021304  013703  001404                  MOV     RXCSR,R3        ;SET REGISTER TO BE TESTED.
5329  021310  012705  000400                  MOV     #BIT8,R5        ;SET "EXPECTED ".
5330  021314  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
5331  021316  011304                          MOV     (R3),R4         ;READ THE REGISTER.
5332  021320  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
5333  021322  001401                          BEQ     1$              ;ARE THEY THE SAME?
5334  021324  104003                          HLT     3               ;COMPARISON ERROR.
5335  021326  040513          1$:     BIC     R5,(R3)         ;CLEAR BIT8
5336  021330  011304                          MOV     (R3),R4         ;READ THE REGISTER.
5337  021332  005005                          CLR     R5              ;SET "EXPECTED"
5338  021334  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
5339  021336  001401                          BEQ     2$              ;BR IF OK
5340  021340  104003                          HLT     3               ;COMPARISON ERROR
5341  021342  104400          2$:     SCOPE                           ;SCOPE THIS TEST
5342
5343
5344                                  ;************************** TEST 125 ****************************
5345                                  ;*RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
5346                                  ;*SET BIT6, VERIFY BIT6 WAS SET.
5347                                  ;*CLEAR BIT6, VERIFY BIT6 WAS CLEARED.
5348                                  ;;*************************************************************
5349
5350                          ;;************
5351                          ;           *
5352                          ; TEST 125
5353                          ;           *
```

```
5354                                    ;::************
5355                                    ;::*************************************************************************
5356  021344  012737  000125  001226  TST125: MOV     #125,@#TSTNO
5357  021352  012737  021420  001216           MOV     #TST126,NEXT
5358  021360  013703  001404                  MOV     RXCSR,R3        ;SET REGISTER TO BE TESTED.
5359  021364  012705  000100                  MOV     #BIT6,R5        ;SET "EXPECTED ".
5360  021370  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
5361  021372  011304                          MOV     (R3),R4         ;READ THE REGISTER.
5362  021374  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
5363  021376  001401                          BEQ     1$              ;ARE THEY THE SAME?
5364  021400  104003                          HLT     3               ;COMPARISON ERROR.
5365  021402  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT6
5366  021404  011304                          MOV     (R3),R4         ;READ THE REGISTER.
5367  021406  005005                          CLR     R5              ;SET "EXPECTED"
5368  021410  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
5369  021412  001401                          BEQ     2$              ;BR IF OK
5370  021414  104003                          HLT     3               ;COMPARISON ERROR
5371  021416  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
5372
5373
5374                                          ;************************** TEST 126 **************************
5375                                          ;*RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
5376                                          ;*SET BIT5, VERIFY BIT5 WAS SET.
5377                                          ;*CLEAR BIT5, VERIFY BIT5 WAS CLEARED.
5378                                          ;::*********************************************************************
5379
5380                                    ;::************
5381                                    ;               *
5382                                    ; TEST 126
5383                                    ;               *
5384                                    ;::************
5385                                    ;::*************************************************************************
5386  021420  012737  000126  001226  TST126: MOV     #126,@#TSTNO
5387  021426  012737  021474  001216           MOV     #TST127,NEXT
5388  021434  013703  001404                  MOV     RXCSR,R3        ;SET REGISTER TO BE TESTED.
5389  021440  012705  000040                  MOV     #BIT5,R5        ;SET "EXPECTED ".
5390  021444  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
5391  021446  011304                          MOV     (R3),R4         ;READ THE REGISTER.
5392  021450  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
5393  021452  001401                          BEQ     1$              ;ARE THEY THE SAME?
5394  021454  104003                          HLT     3               ;COMPARISON ERROR.
5395  021456  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT5
5396  021460  011304                          MOV     (R3),R4         ;READ THE REGISTER.
5397  021462  005005                          CLR     R5              ;SET "EXPECTED"
5398  021464  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
5399  021466  001401                          BEQ     2$              ;BR IF OK
5400  021470  104003                          HLT     3               ;COMPARISON ERROR
5401  021472  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
5402
5403
5404                                          ;************************** TEST 127 **************************
5405                                          ;*RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
5406                                          ;*SET BIT4, VERIFY BIT4 WAS SET.
5407                                          ;*CLEAR BIT4, VERIFY BIT4 WAS CLEARED.
5408                                          ;::*********************************************************************
5409
```

```
                                        ;::************
                                        ;             *
                                        ; TEST 127
                                        ;             *
                                        ;::************
                                        ;::*****************************************************************************
021474  012737  000127  001226  TST127: MOV     #127,@#TSTNO
021502  012737  021550  001216           MOV     #TST130,NEXT
021510  013703  001404                   MOV     RXCSR,R3        ;SET REGISTER TO BE TESTED.
021514  012705  000020                   MOV     #BIT4,R5        ;SET "EXPECTED ".
021520  010513                           MOV     R5,(R3)         ;WRITE THE REGISTER.
021522  011304                           MOV     (R3),R4         ;READ THE REGISTER.
021524  020504                           CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
021526  001401                           BEQ     1$              ;ARE THEY THE SAME?
021530  104003                           HLT     3               ;COMPARISON ERROR.
021532  040513                  1$:      BIC     R5,(R3)         ;CLEAR BIT4
021534  011304                           MOV     (R3),R4         ;READ THE REGISTER.
021536  005005                           CLR     R5              ;SET "EXPECTED"
021540  020504                           CMP     R5,R4           ;R5=GOOD; R4=?
021542  001401                           BEQ     2$              ;BR IF OK
021544  104003                           HLT     3               ;COMPARISON ERROR
021546  104400                  2$:      SCOPE                   ;SCOPE THIS TEST

                                        ;*************************** TEST 130 ****************************
                                        ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT TEST.
                                        ;*SET BIT6, VERIFY BIT6 WAS SET.
                                        ;*CLEAR BIT6, VERIFY BIT6 WAS CLEARED.
                                        ;::*****************************************************************************
                                        ;::************
                                        ;             *
                                        ; TEST 130
                                        ;             *
                                        ;::************
                                        ;::*****************************************************************************
021550  012737  000130  001226  TST130: MOV     #130,@#TSTNO
021556  012737  021634  001216           MOV     #TST131,NEXT
021564  013703  001412                   MOV     TXCSR,R3        ;SET REGISTER TO BE TESTED.
021570  012705  000100                   MOV     #BIT6,R5        ;SET "EXPECTED ".
021574  010513                           MOV     R5,(R3)         ;WRITE THE REGISTER.
021576  011304                           MOV     (R3),R4         ;READ THE REGISTER.
021600  042704  000200                   BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
021604  020504                           CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
021606  001401                           BEQ     1$              ;ARE THEY THE SAME?
021610  104003                           HLT     3               ;COMPARISON ERROR.
021612  040513                  1$:      BIC     R5,(R3)         ;CLEAR BIT6
021614  011304                           MOV     (R3),R4         ;READ THE REGISTER.
021616  042704  000200                   BIC     #BIT7,R4        ;CLEAR UNWANTED BITS
021622  005005                           CLR     R5              ;SET "EXPECTED"
021624  020504                           CMP     R5,R4           ;R5=GOOD; R4=?
021626  001401                           BEQ     2$              ;BR IF OK
021630  104003                           HLT     3               ;COMPARISON ERROR
021632  104400                  2$:      SCOPE                   ;SCOPE THIS TEST
```

```
                                        ;*************************** TEST 131 ****************************
                                        ;*RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
                                        ;*SET BIT1, VERIFY BIT1 WAS SET.
                                        ;*CLEAR BIT1, VERIFY BIT1 WAS CLEARED.
                                        ;;*************************************************************

                                ;;*************
                                ;               *
                                ; TEST 131
                                ;               *
                                ;;*************
                                ;;*************************************************************************
021634  012737  000131  001226  TST131: MOV     #131,@#TSTNO
021642  012737  021720  001216          MOV     #TST132,NEXT
021650  013703  001404                  MOV     RXCSR,R3        ;SET REGISTER TO BE TESTED.
021654  012705  000002                  MOV     #BIT1,R5        ;SET "EXPECTED ".
021660  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
021662  011304                          MOV     (R3),R4         ;READ THE REGISTER.
021664  042704  177001                  BIC     #177001,R4      ;CLEAR UNWANTED BITS
021670  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
021672  001401                          BEQ     1$              ;ARE THEY THE SAME?
021674  104003                          HLT     3               ;COMPARISON ERROR.
021676  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT1
021700  011304                          MOV     (R3),R4         ;READ THE REGISTER.
021702  042704  177001                  BIC     #177001,R4      ;CLEAR UNWANTED BITS
021706  005005                          CLR     R5              ;SET "EXPECTED"
021710  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
021712  001401                          BEQ     2$              ;BR IF OK
021714  104003                          HLT     3               ;COMPARISON ERROR
021716  104400                  2$:     SCOPE                   ;SCOPE THIS TEST

                                        ;*************************** TEST 132 ****************************
                                        ;*RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
                                        ;*SET BIT2, VERIFY BIT2 WAS SET.
                                        ;*CLEAR BIT2, VERIFY BIT2 WAS CLEARED.
                                        ;;*************************************************************

                                ;;*************
                                ;               *
                                ; TEST 132
                                ;               *
                                ;;*************
                                ;;*************************************************************************
021720  012737  000132  001226  TST132: MOV     #132,@#TSTNO
021726  012737  022004  001216          MOV     #TST133,NEXT
021734  013703  001404                  MOV     RXCSR,R3        ;SET REGISTER TO BE TESTED.
021740  012705  000004                  MOV     #BIT2,R5        ;SET "EXPECTED ".
021744  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
021746  011304                          MOV     (R3),R4         ;READ THE REGISTER.
021750  042704  177001                  BIC     #177001,R4      ;CLEAR UNWANTED BITS
021754  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
021756  001401                          BEQ     1$              ;ARE THEY THE SAME?
021760  104003                          HLT     3               ;COMPARISON ERROR.
021762  040513                  1$:     BIC     R5,(R3)         ;CLEAR BIT2
021764  011304                          MOV     (R3),R4         ;READ THE REGISTER.
```

```
5522  021766  042704  177001              BIC     #177001,R4      ;CLEAR UNWANTED BITS
5523  021772  005005                      CLR     R5              ;SET "EXPECTED"
5524  021774  020504                      CMP     R5,R4           ;R5=GOOD; R4=?
5525  021776  001401                      BEQ     2$              ;BR IF OK
5526  022000  104003                      HLT     3               ;COMPARISON ERROR
5527  022002  104400              2$:     SCOPE                   ;SCOPE THIS TEST
5528
5529
5530                                      ;************************** TEST 133 **************************
5531                                      ;*RECEIVER CONTROL REGISTER READ/WRITE BIT TEST.
5532                                      ;*SET BIT3, VERIFY BIT3 WAS SET.
5533                                      ;*CLEAR BIT3, VERIFY BIT3 WAS CLEARED.
5534                                      ;;**********************************************************
5535
5536                                  ;:*************
5537                                  ;             *
5538                                  ; TEST 133
5539                                  ;             *
5540                                  ;:*************
5541                                  ;:**************************************************************
5542  022004  012737  000133  001226  TST133: MOV     #133,@#TSTNO
5543  022012  012737  022070  001216          MOV     #TST134,NEXT
5544  022020  013703  001404                  MOV     RXCSR,R3        ;SET REGISTER TO BE TESTED.
5545  022024  012705  000010                  MOV     #BIT3,R5        ;SET "EXPECTED".
5546  022030  010513                          MOV     R5,(R3)         ;WRITE THE REGISTER.
5547  022032  011304                          MOV     (R3),R4         ;READ THE REGISTER.
5548  022034  042704  177001                  BIC     #177001,R4      ;CLEAR UNWANTED BITS
5549  022040  020504                          CMP     R5,R4           ;R5=GOOD; R4=UNKNOWN.
5550  022042  001401                          BEQ     1$              ;ARE THEY THE SAME?
5551  022044  104003                          HLT     3               ;COMPARISON ERROR.
5552  022046  040513              1$:         BIC     R5,(R3)         ;CLEAR BIT3
5553  022050  011304                          MOV     (R3),R4         ;READ THE REGISTER.
5554  022052  042704  177001                  BIC     #177001,R4      ;CLEAR UNWANTED BITS
5555  022056  005005                          CLR     R5              ;SET "EXPECTED"
5556  022060  020504                          CMP     R5,R4           ;R5=GOOD; R4=?
5557  022062  001401                          BEQ     2$              ;BR IF OK
5558  022064  104003                          HLT     3               ;COMPARISON ERROR
5559  022066  104400              2$:         SCOPE                   ;SCOPE THIS TEST
5560
5561
5562                                      ;************************** TEST 134 **************************
5563                                      ;*RECEIVER CONTROL REGISTER READ/WRITE BIT 6 RESET AND CLEAR TEST
5564                                      ;*WRITE BIT 6,AND TEST THAT IT WILL BE CLEARED AFTER A
5565                                      ;*DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION
5566                                      ;;**********************************************************
5567
5568                                  ;:*************
5569                                  ;             *
5570                                  ; TEST 134
5571                                  ;             *
5572                                  ;:*************
5573                                  ;:**************************************************************
5574  022070  012737  000134  001226  TST134: MOV     #134,@#TSTNO
5575  022076  012737  022164  001216          MOV     #TST135,NEXT
5576  022104  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
5577  022110  052713  000100                  BIS     #BIT6,(R3)      ;SET BIT 6 AT RECEIVER CONTROL REGISTER
```

```
5578  022114  005005                          CLR     R5              ;SET "EXPECTED"
5579  022116  052777  000400  157266          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5580  022124  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5581  022130  032713  000100                  BIT     #BIT6,(R3)      ;TEST BIT 6
5582  022134  001402                          BEQ     1$              ;BIT 6 IS CLEARED
5583  022136  011304                          MOV     (R3),R4         ;LOAD "FOUND"
5584  022140  104003                          HLT     3               ;BIT 6 IS SET AND SHOULDN'T BE
5585  022142  052713  000100          1$:     BIS     #BIT6,(R3)      ;SET BIT 6 AGAIN
5586  022146  005013                          CLR     (R3)            ;CLEAR THE RECEIVER CONTROL REGISTER
5587  022150  032713  000100                  BIT     #BIT6,(R3)      ;TEST TO SEE IF BIT 6 CLEARED
5588  022154  001402                          BEQ     2$              ;BIT 6 IS OK
5589  022156  011304                          MOV     (R3),R4         ;LOAD "FOUND"
5590  022160  104003                          HLT     3               ;BIT 6 FAILED TO CLEAR
5591  022162  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
5592
5593
5594                                          ;*************************** TEST 135 ****************************
5595                                          ;*RECEIVER CONTROL REGISTER READ/WRITE BIT 5 RESET AND CLEAR TEST
5596                                          ;*WRITE BIT 5,AND TEST THAT IT WILL BE CLEARED AFTER A
5597                                          ;*DEVICE RESET AND RECEIVER CONTROL REGISTER CLR INSTRUCTION
5598                                          ;;*************************************************************
5599
5600                                  ;;*************
5601                                  ;             *
5602                                  ; TEST 135
5603                                  ;             *
5604                                  ;;*************
5605                                  ;;*************************************************************************
5606  022164  012737  000135  001226  TST135: MOV     #135,@#TSTNO
5607  022172  012737  022260  001216          MOV     #TST136,NEXT
5608  022200  013703  001404                  MOV     RXCSR,R3        ;GET THE RECEIVER CONTROL REGISTER
5609  022204  052713  000040                  BIS     #BIT5,(R3)      ;SET BIT 5 AT RECEIVER CONTROL REGISTER
5610  022210  005005                          CLR     R5              ;SET "EXPECTED"
5611  022212  052777  000400  157172          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5612  022220  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5613  022224  032713  000040                  BIT     #BIT5,(R3)      ;TEST BIT 5
5614  022230  001402                          BEQ     1$              ;BIT 5 IS CLEARED
5615  022232  011304                          MOV     (R3),R4         ;LOAD "FOUND"
5616  022234  104003                          HLT     3               ;BIT 5 IS SET AND SHOULDN'T BE
5617  022236  052713  000040          1$:     BIS     #BIT5,(R3)      ;SET BIT 5 AGAIN
5618  022242  005013                          CLR     (R3)            ;CLEAR THE RECEIVER CONTROL REGISTER
5619  022244  032713  000040                  BIT     #BIT5,(R3)      ;TEST TO SEE IF BIT 5 CLEARED
5620  022250  001402                          BEQ     2$              ;BIT 5 IS OK
5621  022252  011304                          MOV     (R3),R4         ;LOAD "FOUND"
5622  022254  104003                          HLT     3               ;BIT 5 FAILED TO CLEAR
5623  022256  104400                  2$:     SCOPE                   ;SCOPE THIS TEST
5624
5625
5626                                          ;*************************** TEST 136 ****************************
5627                                          ;*TRANSMITTER CONTROL REGISTER READ/WRITE BIT 6 RESET AND CLEAR TEST
5628                                          ;*WRITE BIT 6,AND TEST THAT IT WILL BE CLEARED AFTER A
5629                                          ;*DEVICE RESET AND TRANSMITTER CONTROL REGISTER CLR INSTRUCTION
5630                                          ;;*************************************************************
5631
5632                                  ;;*************
5633                                  ;             *
```

```
5634                                    ; TEST 136
5635                                    ;              *
5636                                    ;:*************
5637                                    ;:*****************************************************************************
5638  022260  012737  000136  001226  TST136: MOV     #136,@#TSTNO
5639  022266  012737  022354  001216          MOV     #TST137,NEXT
5640  022274  013703  001412                MOV     TXCSR,R3        ;GET THE TRANSMITTER CONTROL REGISTER
5641  022300  052713  000100                BIS     #BIT6,(R3)      ;SET BIT 6 AT TRANSMITTER CONTROL REGISTER
5642  022304  005005                        CLR     R5              ;SET "EXPECTED"
5643  022306  052777  000400  157076        BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5644  022314  004737  005044                JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5645  022320  032713  000100                BIT     #BIT6,(R3)      ;TEST BIT 6
5646  022324  001402                        BEQ     1$              ;BIT 6 IS CLEARED
5647  022326  011304                        MOV     (R3),R4         ;LOAD "FOUND"
5648  022330  104003                        HLT     3               ;BIT 6 IS SET AND SHOULDN'T BE
5649  022332  052713  000100          1$:   BIS     #BIT6,(R3)      ;SET BIT 6 AGAIN
5650  022336  005013                        CLR     (R3)            ;CLEAR THE TRANSMITTER CONTROL REGISTER
5651  022340  032713  000100                BIT     #BIT6,(R3)      ;TEST TO SEE IF BIT 6 CLEARED
5652  022344  001402                        BEQ     2$              ;BIT 6 IS OK
5653  022346  011304                        MOV     (R3),R4         ;LOAD "FOUND"
5654  022350  104003                        HLT     3               ;BIT 6 FAILED TO CLEAR
5655  022352  104400                  2$:   SCOPE                   ;SCOPE THIS TEST
5656
5657
5658                                        ;************************** TEST 137 ****************************
5659                                        ;*THIS TEST CHECKS THE MAINTENANCE CLOCK
5660                                        ;*USED THROUGHOUT THE REMAINING DIAGNOSTICS
5661                                        ;:*****************************************************************
5662
5663                                        ;:*************
5664                                        ;              *
5665                                        ; TEST 137
5666                                        ;              *
5667                                        ;:*************
5668                                        ;:*****************************************************************************
5669  022354  012737  000137  001226  TST137: MOV     #137,@#TSTNO
5670  022362  012737  022510  001216          MOV     #TST140,NEXT
5671  022370  052777  000400  157014        BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5672  022376  004737  005044                JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5673  022402  005037  001236                CLR     TEMP1           ;TEST TIM SETUP
5674  022406  052777  004000  156776        BIS     #SYSTST,@TXCSR  ;ENTER SYSTEM TST MODE TO TURN ON CLOCK
5675  022414  032777  004000  156772  1$:   BIT     #TIMER,@TXDBUF  ;CHECK THE CLOCK BIT
5676  022422  001407                        BEQ     2$              ;BR IF OFF
5677  022424  005237  001236                INC     TEMP1           ;INC WAIT LOOP
5678  022430  022737  177777  001236        CMP     #-1,TEMP1       ;CHECK FOR LOOP TO BE DONE
5679  022436  001366                        BNE     1$              ;BR IF MORE TIME TO WAIT
5680  022440  104000                        HLT                     ;TIMER CLOCK BIT FAILED TO CLEAR
5681  022442  005037  001236          2$:   CLR     TEMP1           ;SECOND HALF SETUP
5682  022446  032777  004000  156740  3$:   BIT     #TIMER,@TXDBUF  ;CHECK THE CLOCK BIT
5683  022454  001007                        BNE     4$              ;BR IF ON
5684  022456  005237  001236                INC     TEMP1           ;INC THE WAIT LOOP
5685  022462  022737  177777  001236        CMP     #-1,TEMP1       ;CHECK FOR LOPP DONE
5686  022470  001366                        BNE     3$              ;BR IF MORE TIME TO WAIT
5687  022472  104000                        HLT                     ;TIMER BIT FAILED TO SET
5688  022474                          4$:
5689  022474  052777  000400  156710        BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
```

```
5690  022502  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5691  022506  104400                          SCOPE                   ;SCOPE THIS TEST
5692
5693
5694                                          ;*************************** TEST 140 ****************************
5695                                          ;*THIS TEST WILL PERFORM STATIC TRANSMITTER FUNCTIONS
5696                                          ;*IN MAINTENANCE MODE.  IT WILL PROVE THE INTERACTION
5697                                          ;*OF SEND, DONE AND TSOM.
5698                                          ;*****************************************************************
5699
5700                                  ;*************
5701                                  ;             *
5702                                  ; TEST 140    *
5703                                  ;             *
5704                                  ;*************
5705                                  ;*****************************************************************
5706
5707  022510  012737  000140  001226  TST140: MOV     #140,@#TSTNO
5708  022516  012737  022664  001216          MOV     #TST141,NEXT
5709  022524  013702  001414                  MOV     TXDBUF,R2       ;LOAD TX BUFFER
5710  022530  013703  001412                  MOV     TXCSR,R3        ;LOAD TX CONTROL REGISTER
5711  022534  005012                          CLR     (R2)            ;CLEAR BUFFER
5712  022536  052777  000400  156646          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5713  022544  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5714  022550  052777  001000  156632          BIS     #CRCEN,@PARCSR  ;TURN OFF CRC
5715  022556  052713  014000                  BIS     #MMODE,(R3)     ;ENTER M/M - PROGRAM NOW CLOCKING
5716  022562  052713  000020                  BIS     #SEND,(R3)      ;ASSERT SEND
5717  022566  052705  014220                  BIS     #SEND!TXDONE!MMODE,R5   ;SET "EXPECTED"
5718  022572  011304                          MOV     (R3),R4         ;READ TX CONTROL REGISTER
5719  022574  020504                          CMP     R5,R4           ;ARE THEY EQUAL?
5720  022576  001401                          BEQ     1$              ;BR IF YES
5721  022600  104003                          HLT     3               ;NO
5722  022602  005005                  1$:     CLR     R5              ;"SET EXPECTED"
5723  022604  011204                          MOV     (R2),R4         ;READ TX BUFFER
5724  022606  042704  170000                  BIC     #170000,R4      ;MASK
5725  022612  020504                          CMP     R5,R4           ;R5=GOOD, R4=?
5726  022614  001405                          BEQ     2$              ;ARE THEY EQUAL?
5727  022616  013703  001414                  MOV     TXDBUF,R3       ;ERROR MESSAGE SETUP
5728  022622  104003                          HLT     3               ;NO
5729  022624  013703  001412                  MOV     TXCSR,R3        ;RETURN TO NORMAL
5730  022630  010512                  2$:     MOV     R5,(R2)         ;LOAD BUFFER
5731  022632  032713  000200                  BIT     #TXDONE,(R3)    ;TEST DONE
5732  022636  001401                          BEQ     3$
5733  022640  104005                          HLT     5               ;BIT FAILED TO CLR
5734  022642  052712  000400          3$:     BIS     #TSOM,(R2)      ;SET TSOM
5735  022646  104412  000017                  PKCLK   ,15.
5736  022652  032713  000200                  BIT     #TXDONE,(R3)    ;TEST DONE
5737  022656  001001                          BNE     4$
5738  022660  104006                          HLT     6               ;BIT FAILED TO SET
5739  022662  104400                  4$:     SCOPE                   ;SCOPE THIS TEST
5740
5741
5742                                          ;*************************** TEST 141 ****************************
5743                                          ;*THIS TEST CHECKS THE STATIC FUNCTIONS OF THE TRANSMITTER
5744                                          ;*IN MAINTENANCE MODE.  THIS TEST PROVES THE INTERACTION OF
5745                                          ;*TXDBUF, TXACT, TSOM,TRANSMITTED DATA AND DONE.
```

```
                                        ;::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
                                        ;::::::::::::
                                        ;            *
                                        ; TEST 141
                                        ;            *
                                        ;::::::::::::
                                        ;::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
022664  012737  000141  001226  TST141: MOV     #141,@#TSTNO
022672  012737  023200  001216          MOV     #TST142,NEXT
022700  013703  001412                  MOV     TXCSR,R3        ;LOAD CONTROL REGISTER
022704  013702  001414                  MOV     TXDBUF,R2       ;LOAD BUFFER
022710  052777  000400  156474          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
022716  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
022722  005012                          CLR     (R2)            ;RESET TXDONE
022724  052713  014000                  BIS     #MMODE,(R3)     ;ENTER M/M--PROGRAM CLOCKING
022730  052777  001000  156452          BIS     #CRCEN,@PARCSR  ;TURN OFF CRC
022736  052713  000020                  BIS     #SEND,(R3)      ;SET SEND
022742  005005                          CLR     R5              ;SET "EXPECTED".
022744  010512                          MOV     R5,(R2)         ;LOAD TX BUFFER
022746  012712  000400                  MOV     #TSOM,(R2)      ;TURN ON TSOM
022752  104412  000006                  PKCLK   ,6.             ;SYNC UP DUP
022756  052777  020000  156426          BIS     #CLK,@TXCSR     ;POKE CLOCK UP
022764  032713  001000                  BIT     #TXACT,(R3)     ;IS TXACT HIGH?
022770  001001                          BNE     1$              ;BR IF SET
022772  104007                          HLT     7               ;TXACT FAILED TO SET
022774                          1$:
022774  042777  020000  156410          BIC     #CLK,@TXCSR             ;POKE CLOCK DOWN
023002  104412  000015                  PKCLK   ,13.            ;PUSH OUT DATA
023006  032713  000020                  BIT     #SEND,(R3)      ;CHECK SEND
023012  001001                          BNE     2$              ;BR IF YES
023014  104017                          HLT     17              ;BIT FAILED TO SET
023016  012737  023016  001220  2$:     MOV     #2$,LOCK        ;SETUPFOR SW 09
023024  012705  000400                  MOV     #TSOM,R5        ;SET "EXPECTED".
023030  011204                          MOV     (R2),R4         ;GET THE BUFFER REG
023032  042704  170000                  BIC     #170000,R4      ;MASK CRC BITS
023036  020504                          CMP     R5,R4           ;R5=GOOD, R4=?
023040  001406                          BEQ     3$              ;BR IF A MATCH
023042  013703  001414                  MOV     TXDBUF,R3       ;ERROR MESSAGE SETUP
023046  104003                          HLT     3               ;BIT FAILED TO SET
023050  104401                          SCOP1                   ;SW09=1?
023052  012703  001412                  MOV     #TXCSR,R3       ;LOAD TRANSMITTER CSR
023056  012737  023056  001220  3$:     MOV     #3$,LOCK        ;SW09 SETUP
023064  042712  000400                  BIC     #TSOM,(R2)      ;CLR TSOM
023070  032713  000200                  BIT     #TXDONE,(R3)    ;TEST DONE
023074  001402                          BEQ     4$              ;BR IF CLEAR
023076  104005                          HLT     5               ;DONE BIT IS SET AND SHOULD BE CLEARED
023100  104401                          SCOP1                   ;SW09=1?
023102  012737  023102  001220  4$:     MOV     #4$,LOCK        ;SW09 SETUP
023110  032712  000400                  BIT     #TSOM,(R2)      ;TEST TSOM
023114  001402                          BEQ     5$              ;BR IF CLEAR
023116  104010                          HLT     10              ;BIT FAILED TO CLR
023120  104401                          SCOP1                   ;SW09=1?
023122  042713  000020          5$:     BIC     #SEND,(R3)      ;TURN OFF SEND
023126  104412  000020                  PKCLK   ,16.            ;POKE 8 BITS
023132  032713  000200                  BIT     #TXDONE,(R3)    ;CHECK DONE
```

```
5802  023136  001401                        BEQ     .+4             ;BR IF OFF
5803  023140  104017                        HLT     17              ;DONE SET AND SHOULDN'T BE
5804  023142  104412  000002                PKCLK   2               ;POKE ONE FULL CLOCK
5805  023146  032713  001000                BIT     #TXACT,(R3)     ;CHECK ACTIVE
5806  023152  001401                        BEQ     6$              ;BR IF OFF
5807  023154  104011                        HLT     11              ;ACTIVE SETS AND SHOULDN'T BE
5808  023156  032713  000200        6$:     BIT     #TXDONE,(R3)    ;IS DONE UP?
5809  023162  001001                        BNE     7$              ;BR IF YES
5810  023164  104006                        HLT     6               ;NO-REPORT ERROR
5811  023166  032713  040000        7$:     BIT     #MTDATA,(R3)    ;CHECK DATA OUT
5812  023172  001401                        BEQ     10$             ;BR IF OFF
5813  023174  104012                        HLT     12              ;DATA SET SHOULD BE CLEAR
5814  023176  104400                10$:    SCOPE                   ;SCOPE THIS TEST
5815
5816
5817                                        ;************************** TEST 142 ****************************
5818                                        ;*THIS TEST VERIFIES THAT THE DEVICE IDLES FLAGS
5819                                        ;*IDLE A MINIMUM OF 64. FLAGS.
5820                                        ;:************************************************************
5821
5822                                    ;:************
5823                                    ;            *
5824                                    ; TEST 142
5825                                    ;            *
5826                                    ;:************
5827                                    ;:************************************************************************
5828  023200  012737  000142  001226    TST142: MOV     #142,@#TSTNO
5829  023206  012737  023270  001216            MOV     #TST143,NEXT
5830  023214  013702  001414                    MOV     TXDBUF,R2       ;LOAD TX BUFFER
5831  023220  013703  001412                    MOV     TXCSR,R3        ;LOAD TX CONTROL REGISTER
5832  023224  052777  000400  156160            BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5833  023232  004737  005044                    JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5834  023236  005012                            CLR     (R2)            ;RESET TXDONE
5835  023240  052713  014000                    BIS     #MMODE,(R3)     ;ENTER M/MODE
5836  023244  052713  000020                    BIS     #SEND,(R3)      ;SET SEND
5837  023250  052712  000400                    BIS     #TSOM,(R2)      ;TURN ON START OF MSG
5838  023254  104412  000004                    PKCLK   4.              ;SYNC UP DUP
5839  023260  004137  007052                    JSR     R1,FLG          ;SEND 64. FLAGS
5840  023264  000100                            64.                     ;64. FLAGS
5841  023266  104400                            SCOPE                   ;SCOPE THIS TEST
5842
5843                                        ;************************** TEST 143 ****************************
5844                                        ;*THIS TEST PUSHES DATA THRU THE TRANSMITTER
5845                                        ;*IN MAINTENANCE MODE. THE TEST SENDS A FLAG,
5846                                        ;*AND TWO ALTERNATING ONES AND ZEROES CHARACTERS,
5847                                        ;*AN ALL ZEROES CHARACTER AND AN ALL ONES
5848                                        ;*CHARACTER TO VERIFY THE BIT STUFF CAPABILITY OF
5849                                        ;*THE DUP WITHOUT A CRC CHECK.THE TEST ROTATES
5850                                        ;*THE BITS THRU,SAMPLING THE DATA ON A BIT-BY-BIT
5851                                        ;*BASIS,LSB FIRST. IT STORES THE BIT IN THE MSB OF
5852                                        ;*THE SAVE LOCATION,COMPARES AND ROTATES RIGHT UNTIL
5853                                        ;*THE CHARACTER IS ASSEMBLED.
5854                                        ;:************************************************************
5855
5856                                    ;:************
5857                                    ;            *
```

```
                                        ; TEST 143
                                        ;          *
                                        ;:*********
                                        ;:*************************************************************************
023270  012737  000143  001226  TST143: MOV     #143,@#TSTNO
023276  012737  023644  001216          MOV     #TST144,NEXT
023304  013703  001412                  MOV     TXCSR,R3        ;LOAD TRANSMITTER CONTROL REGISTER
023310  013702  001414                  MOV     TXDBUF,R2       ;LOAD TRANSMITTER BUFFER
023314  052777  000400  156070          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
023322  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
023326  005012                          CLR     (R2)            ;RESET TXDONE
023330  052713  014000                  BIS     #MMODE,(R3)     ;ENTER MINT MODE--PROGRAM CLOCKING
023334  052777  001000  156046          BIS     #CRCEN,@PARCSR  ;TURN OFF CRC
023342  052713  000020                  BIS     #SEND,(R3)      ;ASSERT SEND
023346  052712  000400                  BIS     #TSOM,(R2)
023352  005037  001236                  CLR     TEMP1           ;CLEAR
023356  005037  001240                  CLR     TEMP2           ;CLEAR
023362  005037  001242                  CLR     TEMP3           ;CLEAR
023366  005037  001244                  CLR     TEMP4           ;CLEAR
023372  012704  000004                  MOV     #4,R4           ;LOAD THE # OF CHARACTERS TO DO
023376  012705  023632                  MOV     #TBL1,R5        ;LOAD THE TABLE POINTER
023402  012701  023632                  MOV     #TBL1,R1        ;DITTO
023406  011137  001246                  MOV     (R1),TEMP5      ;DITTO
023412  104412  000010                  PKCLK   .8.             ;START A FLAG
023416  012512                          MOV     (R5)+,(R2)      ;LOAD THE FIRST CHARACTER AND CLR TSOM
023420  104412  000014                  PKCLK   .12.            ;FINISH THE FLAG
023424  104412  000002          1$:     PKCLK   .2              ;PUSH OUT A BIT
023430  000241                          CLC                     ;PUT CARRY IN A KNOWN STATE
023432  032713  040000                  BIT     #MTDATA,(R3)    ;TEST THE BIT
023436  001401                          BEQ     2$              ;BR IF CLEAR
023440  000261                          SEC                     ;SET CARRY FOR SOFTWARE
023442  106037  001236          2$:     RORB    TEMP1           ;STORE THE BIT
023446  000241                          CLC                     ;PUT CARRY IN A KNOWN STATE
023450  106037  001246                  RORB    TEMP5           ;CHECK TO SEE WHAT THE BIT SHOULD BE
023454  103006                          BCC     3$              ;BR IF CLEAR
023456  000261                          SEC                     ;SET CARRY FOR SOFTWARE
023460  106037  001242                  RORB    TEMP3           ;SHIFT IN A ONE
023464  005237  001240                  INC     TEMP2           ;INC    ONES COUNT
023470  000404                          BR      4$              ;JUMP OVER
023472  106037  001242          3$:     RORB    TEMP3           ;LOAD A ZERO
023476  005037  001240                  CLR     TEMP2           ;CLEAR ONES COUNT
023502  023737  001236  001242  4$:     CMP     TEMP1,TEMP3     ;DOES HARDWARE = SOFTWARE
023510  001401                          BEQ     5$              ;BR IF YES
023512  104004                          HLT     4               ;HARDWARE AND SOFTWARE DON'T MATCH
                                                                ;R1 HOLDS THE ADRS OF THE OUTPUT CHAR
                                                                ;TEMP1,BIT7 = HARDWARE FOUND
                                                                ;TEMP3,BIT7 = SOFTWARE CALCULATED
                                                                ;TEMP4 GIVE THE BIT POSITION OUTPUT

023514  005704          5$:     TST     R4              ;CHECK FOR LAST CHAR
023516  001003                  BNE     6$              ;BR IF NOT DONE
023520  052712  001000          BIS     #TEOM,(R2)      ;LOAD END OF MESAGE
023524  000405                  BR      7$              ;FINISH TEST
023526  105777  155660  6$:     TSTB    @TXCSR          ;CHECK TO SEE IF EREADY FOR NEXT CHAR
023532  100002                  BPL     7$              ;BR IF NO
```

```
5914  023534  012512                          MOV     (R5)+,(R2)      ;LOAD NEXT CHAR
5915  023536  005304                          DEC     R4              ;LOWER CHAR COUNT
5916  023540  022737  000005  001240  7$:     CMP     #5,TEMP2        ;CHECK FOR STUFFED ZERO
5917  023546  001006                          BNE     10$             ;BR IF NO
5918  023550  104412  000002                  PKCLK   ,2              ;PUSH OUT THE STUFFED ZERO
5919  023554  032713  040000                  BIT     #MTDATA,(R3)    ;CHECK IT
5920  023560  001401                          BEQ     10$             ;BR IF OK
5921  023562  104021                          HLT     21              ;FAILED TO BIT-STUFF!!
5922  023564  005237  001244          10$:    INC     TEMP4           ;INC BIT COUNTER
5923  023570  022737  000010  001244          CMP     #8.,TEMP4       ;ARE WE DONE WITH THIS CHAR?
5924  023576  001312                          BNE     1$              ;BR IF MORE TO GO
5925  023600  005037  001236                  CLR     TEMP1           ;CLEAR OUT HARDWARE SAVE
5926  023604  005037  001242                  CLR     TEMP3           ;CLEAR OUT SOFTWARE SAVE
5927  023610  005037  001244                  CLR     TEMP4           ;CLEAR OU BIT COUNTER
5928  023614  005721                          TST     (R1)+           ;POP TBL POINTER
5929  023616  011137  001246                  MOV     (R1),TEMP5      ;
5930  023622  032712  001000                  BIT     #TEOM,(R2)      ;CHECK FOR END OF TEST
5931  023626  001676                          BEQ     1$              ;BR IF MORE TO GO
5932  023630  104400                          SCOPE                   ;SCOPE THIS TEST
5933
5934  023632  000252                  TBL1:   .WORD   252             ;THE FIRST FOUR CHARACTERS
5935  023634  000000                          .WORD   000             ;OF THIS TABLE ARE OUTPUT.
5936  023636  000125                          .WORD   125             ;THE LAST CHARACTER IS
5937  023640  000377                          .WORD   377             ;A PAD.
5938  023642  000000                          .WORD   000
5939
5940                                  ;R1 = SOFTWARE TABLE POINTER
5941                                  ;R2 = TXDBUF
5942                                  ;R3 = TXCSR
5943                                  ;R4 = CHAR COUNTER
5944                                  ;R5 = HARDWARE TABLE POINTER
5945                                  ;TEMP1 = HARDWARE BIT
5946                                  ;TEMP2 = 1'S COUNT
5947                                  ;TEMP3 = SOFTWARE BIT
5948                                  ;TEMP4 = CHARACTER BIT COUNTER
5949                                  ;TEMP5 = SOFTWARE BYTE
5950
5951
5952                                          ;************************** TEST 144 **************************
5953                                          ;*THIS TEST VERIFIES THE ABORT SEQUENCE AND
5954                                          ;*NORMAL DATA SEQUENCE OF FLAG, DATA, FLAG
5955                                          ;*FOLLOWED BY IDLE LINE. THIS TEST ALSO PROVES
5956                                          ;*THE FUNCTIONING OF ACTIVE, TSOM, TEOM, SEND AND DONE.
5957                                          ;*WITHOUT JSING A CRC CHECK.
5958                                          ;;**********************************************************
5959
5960                                  ;;*************
5961                                  ;
5962                                  ; TEST 144
5963                                  ;
5964                                  ;;*************
5965                                  ;;**********************************************************************
5966  023644  012737  000144  001226  TST144: MOV     #144,@#TSTNO
5967  023652  012737  024132  001216          MOV     #TST145,NEXT
5968  023660  013703  001412                  MOV     TXCSR,R3        ;LOAD CONTROL REGISTER
5969  023664  013702  001414                  MOV     TXDBUF,R2       ;LOAD TRANSMITTER BUFFER
```

```
5970  023670  052777  000400  155514          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
5971  023676  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
5972  023702  005012                          CLR     (R2)            ;RESET TXDONE
5973  023704  052713  014000                  BIS     #MMODE,(R3)     ;ENTER M/MODE
5974  023710  052777  001000  155472          BIS     #CRCEN,@PARCSR  ;SHUT OFF CRC
5975  023716  052713  000020                  BIS     #SEND,(R3)      ;SET SEND
5976  023722  052712  000400                  BIS     #TSOM,(R2)      ;TURN ON START OF MSG
5977  023726  104412  000010                  PKCLK   ,8.             ;SYNC UP DUP AND START PUSHING OUT A FLAG
5978  023732  052712  002000                  BIS     #TABORT,(R2)    ;SET ABORT
5979  023736  104412  000014                  PKCLK   ,12.            ;PUSH OUT THE ABORT
5980  023742  005037  001236                  CLR     TEMP1           ;CLEAR HOLD
5981  023746  104412  000002          1$:     PKCLK   ,2              ;PUSH A BIT
5982  023752  032713  040000                  BIT     #MTDATA,(R3)    ;CHECK DATA
5983  023756  001001                          BNE     2$              ;BR IF SET
5984  023760  104013                          HLT     13              ;DATA FAILED TO SET
5985  023762  005237  001236          2$:     INC     TEMP1           ;INC HOLD
5986  023766  022737  000002  001236          CMP     #2,TEMP1        ;CHECK FOR FINISH
5987  023774  001364                          BNE     1$              ;BR IF NO
5988  023776  032713  001000                  BIT     #TXACT,(R3)     ;TEST TRANSMITTER ACTIVE
5989  024002  001001                          BNE     64$             ;BR IF ACTIVE SET
5990  024004  104007                          HLT     7               ;ACTIVE IS RESET AND SHOULDN'T BE
5991  024006                          64$:
5992  024006  042712  002000                  BIC     #TABORT,(R2)    ;CLEAR ABORT
5993  024012  005037  001236                  CLR     TEMP1           ;CLEAR HOLD
5994  024016  104412  000002          3$:     PKCLK,2                 ;PUSH A BIT
5995  024022  032713  040000                  BIT     #MTDATA,(R3)    ;CHECK DATA
5996  024026  001001                          BNE     4$              ;BR IF SET
5997  024030  104013                          HLT     13              ;DATA OUT FAILED TO SET
5998  024032  005237  001236          4$:     INC     TEMP1           ;INC # TO DO
5999  024036  022737  000006  001236          CMP     #6,TEMP1        ;AND CHECK IT
6000  024044  001364                          BNE     3$              ;BR IF MORE TO GO
6001  024046  104412  000006                  PKCLK   ,6.             ;POKE CLOCK
6002  024052  012712  000252                  MOV     #252,(R2)       ;CLEAR TSOM AND LOAD DATA
6003  024056  104412  000024                  PKCLK   ,20.            ;FINISH THE FLAG AND DATA
6004  024062  052712  001000                  BIS     #TEOM,(R2)      ;SET TEOM
6005  024066  104412  000006                  PKCLK   ,6              ;POKE CLOCK
6006  024072  004137  007052                  JSR     R1,FLG          ;SEND THREE FLAGS
6007  024076  000003                          3                       ;DITTO
6008  024100  042713  000020                  BIC     #SEND,(R3)      ;CLEAR SEND
6009  024104  104412  000004                  PKCLK   ,4              ;POKE CLOCK
6010  024110  032713  040000                  BIT     #MTDATA,(R3)    ;TEST TXDAT
6011  024114  001001                          BNE     65$             ;MARK OUT
6012  024116  104013                          HLT     13              ;TXDAT A SPACE - SHOULD BE 1
6013  024120  032713  000200          65$:    BIT     #TXDONE,(R3)    ;IS DONE UP
6014  024124  001001                          BNE     66$             ;YES
6015  024126  104006                          HLT     6               ;NO - BUT IT SHOULD BE.
6016  024130  104400                  66$:    SCOPE                   ;SCOPE THIS TEST
6017
6018
6019
6020                                          ;*************************** TEST 145 ****************************
6021                                          ;*THIS TEST VERIFIES THAT A DATA
6022                                          ;*UNDER RUN CONDITION WILL CAUSE
6023                                          ;*THE TRANSMITTER DATA LATE BIT TO SET
6024                                          ;*AND THAT THE DEVICE WILL ABORT
6025                                          ;*UNTIL SEND IS CLEARED WHEN THE OUTPUT GOES TO A SPACE
```

```
6026                                    ;;*************************************************************************
6027                                    ;;*************
6028                                    ;               *
6029                                    ; TEST 145
6030                                    ;               *
6031                                    ;;*************
6032                                    ;;*************************************************************************
6033  024132  012737  000145  001226  TST145: MOV     #145,@#TSTNO
6034  024140  012737  024334  001216          MOV     #TST146,NEXT
6035  024146  013703  001412                 MOV     TXCSR,R3        ;LOAD TRANSMITTER CONTROL REGISTER
6036  024152  013702  001414                 MOV     TXDBUF,R2       ;LOAD TRANSMITTER BUFFER
6037  024156  052777  000400  155226         BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
6038  024164  004737  005044                 JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
6039  024170  005012                         CLR     (R2)            ;RESET TXDONE
6040  024172  052713  014000                 BIS     #MMODE,(R3)     ;ENTER MINT MODE--PROGRAM CLOCKING
6041  024176  052777  001000  155204         BIS     #CRCEN,@PARCSR  ;TURN OFF CRC
6042  024204  052713  000020                 BIS     #SEND,(R3)      ;ASSERT SEND
6043  024210  052712  000400                 BIS     #TSOM,(R2)
6044  024214  104412  000006                 PKCLK   ,6.             ;START OUTPUTING FLAG
6045  024220  012777  000000  155166         MOV     #0,@TXDBUF      ;CLEAR TSOM-LOAD BUFFER WITH ZEROES
6046  024226  104412  000016                 PKCLK   ,14.            ;FINISH FLAG
6047  024232  104412  000017                 PKCLK   ,15.            ;OUTPUT UP TO 7 1/2 CLOCK TICKS
6048  024236  032777  100000  155146         BIT     #TXDLAT,@TXCSR  ;MAKE SURE DNA IS NOT SET
6049  024244  001401                         BEQ     1$              ;BR IF CLEARED
6050  024246  104014                         HLT     14              ;BIT IS SET TOO SOON
6051  024250  104412  000002          1$:    PKCLK   ,2              ;FINISH LAST CLOCK
6052  024254  032777  100000  155130         BIT     #TXDLAT,@TXCSR  ;NOW CHECK DNA
6053  024262  001001                         BNE     2$              ;BRANCH IF SET
6054  024264  104015                         HLT     15              ;BIT IS CLEARED AND SHOULD BE SET
6055  024266                          2$:
6056  024266  042777  000020  155116         BIC     #SEND,@TXCSR    ;TURN OFF SEND
6057  024274  104412  000020                 PKCLK   ,16.            ;PUSH 8 BITS
6058  024300  012737  000010  001236         MOV     #8.,TEMP1       ;SETUP FOR IDLE LINE
6059  024306  104412  000002          3$:    PKCLK   ,2              ;OUTPUT BIT
6060  024312  032777  040000  155072         BIT     #MTDATA,@TXCSR  ;CHECK IT
6061  024320  001401                         BEQ     4$              ;BRANCH IF ZERO
6062  024322  104012                         HLT     12              ;BIT IS A 1
6063  024324  005337  001236          4$:    DEC     TEMP1           ;LOWER COUNT
6064  024330  001366                         BNE     3$              ;NOT DONE? - GO BACK
6065  024332  104400                         SCOPE                   ;SCOPE THIS TEST
6066
6067
6068                                    ;************************** TEST 146 **************************
6069                                    ;*THIS TEST VERIFIES THAT DROPPING OF
6070                                    ;*SEND BEFORE SETTING TEOM CAUSES
6071                                    ;*A SPACE TO BE OUTPUT AFTER COMPLETION OF
6072                                    ;*A CHARACTER WITH AND WITHOUT BIT STUFF.
6073                                    ;;*************************************************************
6074
6075                                    ;;*************
6076                                    ;               *
6077                                    ; TEST 146
6078                                    ;               *
6079                                    ;;*************
6080                                    ;;*************************************************************************
6081  024334  012737  000146  001226  TST146: MOV     #146,@#TSTNO
```

```
6082  024342  012737  024706  001216          MOV     #TST147,NEXT
6083  024350  013703  001412                  MOV     TXCSR,R3        ;LOAD TRANSMITTER CONTROL REGISTER
6084  024354  013702  001414                  MOV     TXDBUF,R2       ;LOAD TRANSMITTER BUFFER
6085  024360  052777  000400  155024          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
6086  024366  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
6087  024372  005012                          CLR     (R2)            ;RESET TXDONE
6088  024374  052713  014000                  BIS     #MMODE,(R3)     ;ENTER MINT MODE--PROGRAM CLOCKING
6089  024400  052777  001000  155002          BIS     #CRCEN,@PARCSR  ;TURN OFF CRC
6090  024406  052713  000020                  BIS     #SEND,(R3)      ;ASSERT SEND
6091  024412  052712  000400                  BIS     #TSOM,(R2)
6092  024416  104412  000006                  PKCLK   ,6.             ;PUSH 2 BITS
6093  024422  012777  000252  154764          MOV     #252,@TXDBUF    ;LOAD DATA
6094  024430  104412  000014                  PKCLK   ,12.            ;PUSH 6 BITS
6095  024434  042777  000020  154750          BIC     #SEND,@TXCSR    ;TURN OFF TRANSMITTER
6096  024442  104412  000002                  PKCLK   ,2              ;POKE A FULL CLOCK
6097  024446  012737  000010  001236          MOV     #8.,TEMP1       ;LOAD TEMP1
6098  024454  104412  000012          1$:     PKCLK   ,10.            ;PUSH 5 BITS
6099  024460  032777  040000  154724          BIT     #MTDATA,@TXCSR  ;CHECK DATA
6100  024466  001401                          BEQ     2$              ;BR IF CLEAR
6101  024470  104012                          HLT     12              ;DATA IS SET - SHOULD BE CLEAR
6102  024472  005337  001236          2$:     DEC     TEMP1           ;LOWER THE # OF TIMES TO REPEAT
6103  024476  001366                          BNE     1$              ;BR IF TO GOE MSG
6104  024500  032777  001000  154704          BIT     #TXACT,@TXCSR   ;CHECK ACTIVE
6105  024506  001401                          BEQ     7$              ;BR IF OFF
6106  024510  104011                          HLT     11              ;ACTIVE FAILED TO CLEAR
6107  024512                          7$:
6108  024512  052777  000400  154672          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
6109  024520  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
6110  024524  013703  001412                  MOV     TXCSR,R3        ;LOAD TRANSMITTER CONTROL REGISTER
6111  024530  013702  001414                  MOV     TXDBUF,R2       ;LOAD TRANSMITTER BUFFER
6112  024534  052777  000400  154650          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
6113  024542  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
6114  024546  005012                          CLR     (R2)            ;RESET TXDONE
6115  024550  052713  014000                  BIS     #MMODE,(R3)     ;ENTER MINT MODE--PROGRAM CLOCKING
6116  024554  052777  001000  154626          BIS     #CRCEN,@PARCSR  ;TURN OFF CRC
6117  024562  052713  000020                  BIS     #SEND,(R3)      ;ASSERT SEND
6118  024566  052712  000400                  BIS     #TSOM,(R2)
6119  024572  104412  000006                  PKCLK   ,6.             ;PUSH TWO BITS
6120  024576  012777  000177  154610          MOV     #177,@TXDBUF    ;LOAD DATA
6121  024604  104412  000020                  PKCLK   ,16.            ;PUSH 8 BITS
6122  024610  042777  000020  154574          BIC     #SEND,@TXCSR    ;TURN OFF TRANSMITTER
6123  024616  104412  000014                  PKCLK   ,12.            ;PUSH 6 BITS
6124  024622  032777  040000  154562          BIT     #MTDATA,@TXCSR  ;CHECK DATA OUT FOR BIT STUFF FUNCTION
6125  024630  001001                          BNE     3$              ;BR IF SET
6126  024632  104013                          HLT     13              ;BIT IS A 0, SHOULD BE A 1 - DEVICE
6127                                                                  ;FAILED TO BIT STUFF
6128  024634  104412  000004          3$:     PKCLK   ,4              ;PUSH 2 BITS
6129  024640  012737  000010  001236          MOV     #8.,TEMP1       ;LOAD TEMP1
6130  024646  104412  000002          4$:     PKCLK   ,2              ;PUSH A BIT
6131  024652  032777  040000  154532          BIT     #MTDATA,@TXCSR  ;CHECK DATA
6132  024660  001401                          BEQ     5$              ;BR IF OFF
6133  024662  104012                          HLT     12              ;DATA WINDOW SET - SHOULD BE CLEAR
6134  024664  005337  001236          5$:     DEC     TEMP1           ;LOWER THE # OF TIMES TO CHECK
6135  024670  001366                          BNE     4$              ;BR IF MORE TOGO
6136  024672  032777  001000  154512          BIT     #TXACT,@TXCSR   ;CHECK ACTIVE
6137  024700  001401                          BEQ     6$              ;BR IF OFF
```

```
6138  024702  104011                        HLT     11              ;ACTIVE FAILED TO CLEAR
6139  024704  104400                6$:     SCOPE                   ;SCOPE THIS TEST
6140
6141                                        ;*************************** TEST 147 ****************************
6142                                        ;*THIS TEST VERIFIES THAT A DATA UNDERRUN
6143                                        ;*CONDITION WILL CAUSE THE TRANSMITTER DATA
6144                                        ;*LATE BIT TO SET AND THAT THE DUP WILL GO TO
6145                                        ;*A MARK OUTPUT
6146                                        ;;*************************************************************
6147
6148                                ;;*************
6149                                ;               *
6150                                ; TEST 147
6151                                ;               *
6152                                ;;*************
6153                                ;;*************************************************************************
6154  024706  012737  000147  001226 TST147: MOV     #147,@#TSTNO
6155  024714  012737  025102  001216         MOV     #TST150,NEXT
6156  024722  013703  001412                MOV     TXCSR,R3        ;LOAD TRANSMITTER CONTROL REGISTER
6157  024726  013702  001414                MOV     TXDBUF,R2       ;LOAD TRANSMITTER BUFFER
6158  024732  052777  000400  154452        BIS     #MRESE,@TXCSR   ;RESET THE DEVICE
6159  024740  004737  005044                JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
6160  024744  005012                        CLR     (R2)            ;RESET TXDONE
6161  024746  052713  014000                BIS     #MMODE,(R3)     ;ENTER MINT MODE--PROGRAM CLOCKING
6162  024752  052777  001000  154430        BIS     #CRCEN,@PARCSR  ;TURN OFF CRC
6163  024760  052713  000020                BIS     #SEND,(R3)      ;ASSERT SEND
6164  024764  052712  000400                BIS     #TSOM,(R2)
6165  024770  104412  000006                PKCLK   ,6.             ;START FLAG
6166  024774  112777  000176  154412        MOVB    #176,@TXDBUF    ;LOAD CHARACTER TO BE BIT-STUFFED
6167  025002  042777  000400  154404        BIC     #TSOM,@TXDBUF   ;SHUT OFF TSOM
6168  025010  104412  000016                PKCLK   ,14.            ;FINISH CLOCKING FLAG
6169  025014  104412  000017                PKCLK   ,15.            ;CLOCK TO WITHIN 1 1/2 CLOCKS
6170  025020  032777  100000  154364        BIT     #TXDLAT,@TXCSR  ;CHECK DATA LATE
6171  025026  001401                        BEQ     1$              ;BRANCH IF CLEAR
6172  025030  104014                        HLT     14              ;BIT IS SET TOO SOON
6173  025032  104412  000002         1$:    PKCLK   2               ;CLOCK TO WITHIN A HALF-CLOCK OF DNA
6174  025036  032777  100000  154346        BIT     #TXDLAT,@TXCSR  ;CHECK DNA
6175  025044  001401                        BEQ     2$              ;BR IF OFF
6176  025046  104014                        HLT     14              ;BIT SET TOO SOON, DEVICE FAILED TO BIT-STUFF
6177  025050  104412  000004         2$:    PKCLK   4               ;FINISH CHARACTER
6178  025054  032777  100000  154330        BIT     #TXDLAT,@TXCSR  ;CHECK DNA
6179  025062  001001                        BNE     3$              ;BRANCH IF SET
6180  025064  104016                        HLT     16              ;BIT SHOULD BE SET AND IS CLEARED
6181  025066  032777  040000  154316  3$:   BIT     #MTDATA,@TXCSR  ;CHECK DATA
6182  025074  001001                        BNE     4$              ;BR IF SET
6183  025076  104013                        HLT     13              ;DATA WAS CLEAR - SHOULD BE SET
6184  025100  104400                4$:     SCOPE                   ;SCOPE THIS TEST
6185
6186
6187                                        ;*************************** TEST 150 ****************************
6188                                        ;*THIS TEST VERIFIES THAT SETTING TEOM
6189                                        ;*AND TSOM AT THE SAME TIME WILL HOLD
6190                                        ;*ACTIVE UP AND SEND A FLAG
6191                                        ;;*************************************************************
6192
6193                                ;;*************
```

```
6194                                    ;            *
6195                                    ; TEST 150
6196                                    ;            *
6197                                    ;:************
6198                                    ;:*************************************************************************
6199  025102  012737  000150  001226  TST150: MOV     #150,@#TSTNO
6200  025110  012737  025236  001216          MOV     #TST151,NEXT
6201  025116  013703  001412                 MOV     TXCSR,R3        ;LOAD TRANSMITTER CONTROL REGISTER
6202  025122  013702  001414                 MOV     TXDBUF,R2       ;LOAD TRANSMITTER BUFFER
6203  025126  052777  000400  154256         BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
6204  025134  004737  005044                 JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
6205  025140  005012                         CLR     (R2)            ;RESET TXDONE
6206  025142  052713  014000                 BIS     #MMODE,(R3)     ;ENTER MINT MODE--PROGRAM CLOCKING
6207  025146  052777  001000  154234         BIS     #CRCEN,@PARCSR  ;TURN OFF CRC
6208  025154  052713  000020                 BIS     #SEND,(R3)      ;ASSERT SEND
6209  025160  052712  000400                 BIS     #TSOM,(R2)
6210  025164  104412  000006                 PKCLK   ,6.             ;SYNC UP DUP
6211  025170  012777  000252  154216         MOV     #252,@TXDBUF    ;LOAD DATA
6212  025176  104412  000032                 PKCLK   ,26.            ;FINISH THE FLAG AND DATA
6213  025202  052777  001400  154204         BIS     #TEOM!#TSOM,@TXDBUF     ;TURN ON START AND END OF MSG
6214  025210  104412  000004                 PKCLK   ,4              ;PUSH 2 BITS
6215  025214  032777  001000  154170         BIT     #TXACT,@TXCSR   ;TEST ACTIVE
6216  025222  001001                         BNE     1$              ;BR IF SET
6217  025224  104007                         HLT     7               ;ACTIVE SHOULD BE SET AND IS CLEAR
6218  025226  004137  007052          1$:    JSR     R1,FLG          ;PUSH OUT A FLAG
6219  025232  000001                         1                       ;ONE FLAG
6220  025234  104400                         SCOPE                   ;SCOPE THIS TEST
6221
6222                                    ;*************************** TEST 151 ****************************
6223                                    ;*TEST OF THE BCC OPERATION USING
6224                                    ;*CRC.CCITT FOR THE POLYNOMINAL. SPECIFIC
6225                                    ;*DATA PATTERNS ARE USED TO ISOLATE FAULTS.
6226                                    ;:*************************************************************
6227
6228                                    ;:************
6229                                    ;            *
6230                                    ; TEST 151
6231                                    ;            *
6232                                    ;:************
6233                                    ;:*************************************************************************
6234  025236  012737  000151  001226  TST151: MOV     #151,@#TSTNO
6235  025244  012737  026306  001216          MOV     #TST152,NEXT
6236  025252  012737  102010  026300          MOV     #CRC.CCITT,XPOLY        ;SET THE POLYNOMINAL
6237  025260  012737  026116  001244          MOV     #5$,TEMP4       ;GET THE TABLE POINTER
6238  025266  012737  025274  001220          MOV     #1$,LOCK        ;SETUP FOR SW09=1
6239  025274                          1$:
6240  025274  052777  000400  154110         BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
6241  025302  004737  005044                 JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
6242  025306  005077  154102                 CLR     @TXDBUF         ;RESET TXDONE
6243  025312  052777  014000  154072         BIS  #MMODE,@TXCSR      ;ENTER M/MODE
6244  025320  052777  000020  154064         BIS     #SEND,@TXCSR    ;TURNON DUP
6245  025326  052777  000400  154060         BIS     #TSOM,@TXDBUF   ;TURN ON START OF MSG
6246  025334  104412  000006                 PKCLK   ,6.             ;SYNC UP DUP
6247  025340  017777  153700  154046         MOV     @TEMP4,@TXDBUF  ;LOAD DATA
6248  025346  104412  000020                 PKCLK   ,16.            ;PUSH 8 BITS
6249  025352  005037  026304                 CLR     CALBCC          ;CLEAR OUT OLD BCC
```

```
6250  025356  004537  026126              JSR     R5,SIMBCC       ;CALCULATE A SOFTWARE BCC
6251  025362  000010                      8.                      ;BASED
6252  025364  000000                      0                       ;ON THESE
6253  025366  177777                      -1                      ;PARAMETERS
6254  025370  012777  001000  154016      MOV     #TEOM,@TXDBUF   ;CLEAR TSOM, SET TEOM
6255  025376  104412  000016              PKCLK   ,14.            ;PUSHOUT DATA
6256  025402  004137  006750              JSR     R1,ACC          ;DO A BCC CALCULATION (HDWR)
6257  025406  100000                      BIT15                   ;FOR 16 BITS
6258  025410  013705  026304              MOV     CALBCC,R5       ;MOV SOFTWARE BCC TO EXPECTED
6259  025414  005105                      COM     R5              ;INVERT IT
6260  025416  020504                      CMP     R5,R4           ;DOES EXPECTED=FOUND?
6261  025420  001401                      BEQ     64$             ;BR IF OK
6262  025422  104020                      HLT     20              ;BCC CALCULATION ERROR
6263                                                              ;TO DEBUG CRC USE THE CRC
6264                                                              ;DEBUG AID TEST.  SEE FRONT OF THE
6265                                                              ;LISTING FOR TEST LOCATION
6266  025424  104401              64$:    SCOP1                   ;SW09=1?
6267  025426  062737  000002  001244      ADD     #2,TEMP4        ;MOVE THE TABLE POINTER
6268  025434  012737  025442  001220      MOV     #2$,LOCK        ;SETUP FOR SW09=1
6269  025442                      2$:
6270  025442  052777  000400  153742      BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
6271  025450  004737  005044              JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
6272  025454  005077  153734              CLR     @TXDBUF         ;RESET TXDONE
6273  025460  052777  014000  153724      BIS #MMODE,@TXCSR       ;ENTER M/MODE
6274  025466  052777  000020  153716      BIS     #SEND,@TXCSR    ;TURNON DUP
6275  025474  052777  000400  153712      BIS     #TSOM,@TXDBUF   ;TURN ON START OF MSG
6276  025502  104412  000006              PKCLK   ,6.             ;SYNC UP DUP
6277  025506  017777  153532  153700      MOV     @TEMP4,@TXDBUF  ;LOAD DATA
6278  025514  104412  000020              PKCLK   ,16.            ;PUSH 8 BITS
6279  025520  005037  026304              CLR     CALBCC          ;CLEAR OUT OLD BCC
6280  025524  004537  026126              JSR     R5,SIMBCC       ;CALCULATE A SOFTWARE BCC
6281  025530  000010                      8.                      ;BASED
6282  025532  000252                      ^B<10101010>            ;ON THESE
6283  025534  177777                      -1                      ;PARAMETERS
6284  025536  012777  001000  153650      MOV     #TEOM,@TXDBUF   ;CLEAR TSOM, SET TEOM
6285  025544  104412  000016              PKCLK   ,14.            ;PUSHOUT DATA
6286  025550  004137  006750              JSR     R1,ACC          ;DO A BCC CALCULATION (HDWR)
6287  025554  100000                      BIT15                   ;FOR 16 BITS
6288  025556  013705  026304              MOV     CALBCC,R5       ;MOV SOFTWARE BCC TO EXPECTED
6289  025562  005105                      COM     R5              ;INVERT IT
6290  025564  020504                      CMP     R5,R4           ;DOES EXPECTED=FOUND?
6291  025566  001401                      BEQ     65$             ;BR IF OK
6292  025570  104020                      HLT     20              ;BCC CALCULATION ERROR
6293                                                              ;TO DEBUG CRC USE THE CRC
6294                                                              ;DEBUG AID TEST.  SEE FRONT OF THE
6295                                                              ;LISTING FOR TEST LOCATION
6296  025572  104401              65$:    SCOP1                   ;SW09=1?
6297  025574  062737  000002  001244      ADD     #2,TEMP4        ;MOVE THE TABLE POINTER
6298  025602  012737  025610  001220      MOV     #3$,LOCK        ;SETUP FOR SW09=1
6299  025610                      3$:
6300  025610  052777  000400  153574      BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
6301  025616  004737  005044              JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
6302  025622  005077  153566              CLR     @TXDBUF         ;RESET TXDONE
6303  025626  052777  014000  153556      BIS #MMODE,@TXCSR       ;ENTER M/MODE
6304  025634  052777  000020  153550      BIS     #SEND,@TXCSR    ;TURNON DUP
6305  025642  052777  000400  153544      BIS     #TSOM,@TXDBUF   ;TURN ON START OF MSG
```

```
6306  025650  104412  000006                  PKCLK   6.              ;SYNC UP DUP
6307  025654  017777  153364  153532          MOV     @TEMP4,@TXDBUF  ;LOAD DATA
6308  025662  104412  000020                  PKCLK   16.             ;PUSH 8 BITS
6309  025666  005037  026304                  CLR     CALBCC          ;CLEAR OUT OLD BCC
6310  025672  004537  026126                  JSR     R5,SIMBCC       ;CALCULATE A SOFTWARE BCC
6311  025676  000010                          8.                      ;BASED
6312  025700  000125                          ^B<01010101>                    ;ON THESE
6313  025702  177777                          -1                      ;PARAMETERS
6314  025704  012777  001000  153502          MOV     #TEOM,@TXDBUF   ;CLEAR TSOM, SET TEOM
6315  025712  104412  000016                  PKCLK   14.             ;PUSHOUT DATA
6316  025716  004137  006750                  JSR     R1,ACC          ;DO A BCC CALCULATION (HDWR)
6317  025722  100000                          BIT15                   ;FOR 16 BITS
6318  025724  013705  026304                  MOV     CALBCC,R5       ;MOV SOFTWARE BCC TO EXPECTED
6319  025730  005105                          COM     R5              ;INVERT IT
6320  025732  020504                          CMP     R5,R4           ;DOES EXPECTED=FOUND?
6321  025734  001401                          BEQ     66$             ;BR IF OK
6322  025736  104020                          HLT     20              ;BCC CALCULATION ERROR
6323                                                                  ;TO DEBUG CRC USE THE CRC
6324                                                                  ;DEBUG AID TEST.  SEE FRONT OF THE
6325                                                                  ;LISTING FOR TEST LOCATION
6326  025740  104401                  66$:    SCOP1                   ;SW09=1?
6327  025742  062737  000002  001244          ADD     #2,TEMP4        ;MOVE THE TABLE POINTER
6328  025750  012737  025756  001220          MOV     #4$,LOCK        ;SETUP FOR SW09=1
6329  025756                          4$:
6330  025756  052777  000100  153426          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
6331  025764  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
6332  025770  005077  153420                  CLR     @TXDBUF         ;RESET TXDONE
6333  025774  052777  014000  153410          BIS #MMODE,@TXCSR       ;ENTER M/MODE
6334  026002  052777  000020  153402          BIS     #SEND,@TXCSR    ;TURNON DUP
6335  026010  052777  000400  153376          BIS     #TSOM,@TXDBUF   ;TURN ON START OF MSG
6336  026016  104412  000006                  PKCLK   6.              ;SYNC UP DUP
6337  026022  017777  153216  153364          MOV     @TEMP4,@TXDBUF  ;LOAD DATA
6338  026030  104412  000020                  PKCLK   16.             ;PUSH 8 BITS
6339  026034  005037  026304                  CLR     CALBCC          ;CLEAR OUT OLD BCC
6340  026040  004537  026126                  JSR     R5,SIMBCC       ;CALCULATE A SOFTWARE BCC
6341  026044  000010                          8.                      ;BASED
6342  026046  000377                          ^B<11111111>                    ;ON THESE
6343  026050  177777                          -1                      ;PARAMETERS
6344  026052  012777  001000  153334          MOV     #TEOM,@TXDBUF   ;CLEAR TSOM, SET TEOM
6345  026060  104412  000020                  PKCLK   16.             ;PUSH 8 BITS
6346  026064  004137  006750                  JSR     R1,ACC          ;DO A BCC CALCULATION (HDWR)
6347  026070  100000                          BIT15                   ;FOR 16 BITS
6348  026072  012705  157400                  MOV     #157400,R5      ;LOAD THE BCC
6349  026076  020504                          CMP     R5,R4           ;DOES EXPECTED=FOUND?
6350  026100  001401                          BEQ     67$             ;BR IF OK
6351  026102  104020                          HLT     20              ;BCC CALCULATION ERROR
6352                                                                  ;TO DEBUG CRC USE THE CRC
6353                                                                  ;DEBUG AID TEST.  SEE FRONT OF THE
6354                                                                  ;LISTING FOR TEST LOCATION
6355  026104  104401                  67$:    SCOP1                   ;SW09=1?
6356  026106  062737  000002  001244          ADD     #2,TEMP4        ;MOVE THE TABLE POINTER
6357  026114  104400                          SCOPE                   ;SCOPE THIS TEST
6358  026116  000000                  5$:     .WORD   0
6359  026120  000252                          .WORD   252
6360  026122  000125                          .WORD   125
6361  026124  000377                          .WORD   377
```

```
6362  026126  010046                    SIMBCC: MOV     R0,-(SP)
6363  026130  010146                            MOV     R1,-(SP)
6364  026132  010246                            MOV     R2,-(SP)
6365  026134  012537  001236                    MOV     (R5)+,TEMP1
6366  026140  012537  001240                    MOV     (R5)+,TEMP2
6367  026144  012537  001242                    MOV     (R5)+,TEMP3
6368  026150  005037  026302            1$:     CLR     BCCFBK
6369  026154  013700  001242                    MOV     TEMP3,R0
6370  026160  006037  001240                    ROR     TEMP2
6371  026164  005500                            ADC     R0
6372  026166  032700  000001                    BIT     #BIT0,R0
6373  026172  001402                            BEQ     2$
6374  026174  005137  026302                    COM     BCCFBK
6375  026200  013700  026300            2$:     MOV     XPOLY,R0
6376  026204  005100                            COM     R0
6377  026206  040037  026302                    BIC     R0,BCCFBK
6378  026212  000241                            CLC
6379  026214  006037  001242                    ROR     TEMP3
6380  026220  013700  026302                    MOV     BCCFBK,R0
6381  026224  013701  001242                    MOV     TEMP3,R1
6382  026230  010102                            MOV     R1,R2
6383  026232  040100                            BIC     R1,R0
6384  026234  043702  026302                    BIC     BCCFBK,R2
6385  026240  050200                            BIS     R2,R0
6386  026242  043737  026300  001242            BIC     XPOLY,TEMP3
6387  026250  050037  001242                    BIS     R0,TEMP3
6388  026254  005337  001236                    DEC     TEMP1
6389  026260  001333                            BNE     1$
6390  026262  013737  001242  026304            MOV     TEMP3,CALBCC
6391  026270  012602                            MOV     (SP)+,R2
6392  026272  012601                            MOV     (SP)+,R1
6393  026274  012600                            MOV     (SP)+,R0
6394  026276  000205                            RTS     R5
6395  026300  000000                    XPOLY:  0
6396  026302  000000                    BCCFBK: 0
6397  026304  000000                    CALBCC: 0
6398          120001                    CRC16=120001
6399          102010                    CRC.CCITT=102010
6400
6401
6402                                            ;************************** TEST 152 ****************************
6403                                            ;*THIS TEST IS AN AID FOR DEBUGGING CRC
6404                                            ;*ERRORS.  A CHARACTER IS LOADED INTO THE
6405                                            ;*DUP AND PUSHED OUT BIT BY BIT WHILE
6406                                            ;*ALLOWING THE OPERATOR TO MONITOR THE CRC
6407                                            ;*CHARACTER AS IT IS GENERATED.  THE DATA CHARACTER
6408                                            ;*CAN ALSO BE CHANGED BY THE OPERATOR.
6409                                            ;*PUT SW09=1 TO LOCK ON BITS.  TO CONTINUE HIT
6410                                            ;*ANY KEY ON THE TTY. AFTER 16 TIMES PUT DOWN SW09 TO LEAVE
6411                                            ;NOTE:  REMEMBER--IN SDLC A ONE IS A LOGIC LOW IN
6412                                            ;THE CRC GENERATOR.
6413                                            ;;*************************************************************
6414                                    ;;*************
6415                                    ;           *
6416                                    ; TEST 152
6417                                    ;           *
```

```
                                        ;::::::::::::
                                        ;:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
026306  012737  000152  001226  TST152: MOV     #152,@#TSTNO
026314  012737  002764  001216          MOV     #.EOP,NEXT
026322  052777  000400  153062          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
026330  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
026334  012737  102010  026300          MOV     #CRC.CCITT,XPOLY        ;LOAD THE POLYNOMIAL
026342  012737  000125  026510          MOV     #125,3$         ;LOAD DATA TO SOFTWARE BCC-CHANGE CHARACTER HERE
026350  013737  026510  001252          MOV     3$,SAVR1
026356  012737  177777  026304          MOV     #-1,CALBCC              ;CLEAR FOR SOFTWARE BCC
026364  013737  026304  026512          MOV     CALBCC,4$
026372  005037  001242                  CLR     TEMP3           ;CLR SOFTWARE MEMORY
026376  005037  001244                  CLR     TEMP4           ;CLEAR BIT COUNTER
026402  005037  001246                  CLR     TEMP5
026406  005077  153002                  CLR     @TXDBUF         ;RESET TXDONE
026412  052777  014000  152772          BIS     #MMODE,@TXCSR   ;ENTER MAINT MODE-PROGRAM CLOCKING
026420  052777  000020  152756          BIS     #RCVEN,@RXCSR   ;TURN ON RECEIVER
026426  052777  000020  152756          BIS     #SEND,@TXCSR    ;TURN ON TRANSMITTER
026434  012777  000400  152752          MOV     #TSOM,@TXDBUF   ;
026442  104412  000044                  PKCLK   ,36.            ;PUSH OUT 2
026446  013777  026510  152740          MOV     3$,@TXDBUF      ;LOAD DATA
026454  104412  000020                  PKCLK   ,16.            ;PUSH OUT ANOTHER
026460  104412  000002          1$:     PKCLK   ,2              ;PUSH OUT A BIT
026464  013737  001244  001254          MOV     TEMP4,SAVR2     ;SET UP TO TYPE
026472  005237  001242                  INC     TEMP3           ;UPDATE THE COUNT
026476  005237  001244                  INC     TEMP4           ;UPDATE BIT COUNTER
026502  004537  026126          2$:     JSR     R5,SIMBCC       ;CALCULATE SOFTWARE BCC BASED ON THESE PARAMETERS
026506  000001                          1                       ;SHIFTS
026510  000000          3$:     .WORD 0                         ;DATA
026512  000000          4$:     .WORD 0                         ;PREVIOUS BCC
026514  004737  026612                  JSR     PC,5$           ;CHECK TO SEE IF WE SHOULD WAIT FOR SCOPING
026520  000241                          CLC                     ;CLEAR FOR NEXT ROTATE
026522  106037  026510                  RORB    3$              ;SET UP THE NEXT BIT
026526  013737  026304  026512          MOV     CALBCC,4$       ;FOR THE SOFTWARE BCC
026534  022737  000006  001244          CMP     #6,TEMP4
026542  001002                          BNE     .+6
026544  005077  152644                  CLR     @TXDBUF
026550  022737  000014  001242          CMP     #12.,TEMP3
026556  001003                          BNE     12$
026560  012777  001000  152626          MOV     #TEOM,@TXDBUF
026566  022737  000020  001244  12$:    CMP     #16.,TEMP4      ;ALL DONE WITH THE CHARACTER?
                                                                ;INCREASE THE COMPARE NUMBER TO
                                                                ;ALLOW CRC TO BE OUTPUT
026574  001331                          BNE     1$              ;BR IF MORE TO GO
026576  052777  000400  152606          BIS     #MRESET,@TXCSR  ;RESET THE DEVICE
026604  004737  005044                  JSR     PC,SMALL        ;WAIT FOR RESET TO FINISH
026610  104400                          SCOPE                   ;SCOPE THIS TEST

026612  032777  001000  152362  5$:     BIT     #SW09,@SWR      ;SW09=1?
026620  001432                          BEQ     6$              ;BR IF NO
026622  013704  026304                  MOV     CALBCC,R4       ;THE DATA CHARACTER IS
026626  012737  000001  001256          MOV     #1,SAVR3        ;FOLLOWED BY A ZERO CHARACTER.THE
026634  000241                          CLC                     ;DATA BIT IN CRC SHOWS WHICH BIT
026636  006004                  11$:    ROR     R4              ;OF THE TWO CHARACTERS IS BEING
026640  006137  001256                  ROL     SAVR3           ;GENERATED
026644  103374                          BCC     11$
```

```
 6474  026646  105737  001246                    TSTB    TEMPS
 6475  026652  001006                            BNE     10$
 6476  026654  104402  027407                    TYPE    .EM1            ;TYPE MSG
 6477  026660  104402  027436                    TYPE    .MH1            ;TYPE HEADER
 6478  026664  105137  001246                    COMB    TEMPS
 6479  026670  104410                    10$:    CONVRT
 6480  026672  030100                            DT1
 6481  026674  105777  152304            7$:     TSTB    @TKCSR          ;CHECK TTY DONE--GO SCOPE THE CRC GENERATOR
 6482  026700  100375                            BPL     7$              ;BR IF NOT YET
 6483  026702  017701  152300                    MOV     @TKDBR,R1       ;READ THE BUFFER
 6484  026706  000207                    6$:     RTS     PC              ;RETURN
 6485
 6486
```

```
6487
(1)  026710  005015  051124  047101  EM2:     ASCIZ <15><12>/TRANSMITTER  /
(1)  026730  005015  042522  042503  EM3:    .ASCIZ <15><12>/RECEIVER  /
(1)  026745     015  052412  042516  EM4:    .ASCIZ <15><12>/UNEXPECTED INTERRUPT/
(1)  026774  005015  051120  046511  EM6:    .ASCIZ <15><12>↑PRIMARY REGISTER TEST ↑
(1)  027025     377  051124  047101  EM10:   .ASCIZ  <377>/TRANSMITTER DATA ERROR. /
(1)  027057     377  044103  041505  EM11:   .ASCIZ  <377>/CHECK CARRY AND OUTBIT.   /
(1)  027113     104  047117  020105  EM12:   .ASCIZ  /DONE BIT  /
(1)  027126  040506  046111  042105  EM13:   .ASCIZ  /FAILED TO CLEAR /
(1)  027147     106  044501  042514  EM14:   .ASCIZ  /FAILED TO SET /
(1)  027166  041501  044524  042526  EM15:   .ASCIZ  /ACTIVE BIT  /
(1)  027203     124  047523  020115  EM16:   .ASCIZ  /TSOM BIT  /
(1)  027216  052517  050124  052125  EM17:   .ASCIZ  /OUTPUT DATA  /
(1)  027234  042523  020124  047524  EM22:   .ASCIZ  /SET TOO SOON /
(1)  027252  052377  042130  040514  EM23:   .ASCIZ  <377>/TXDLAT BIT  /
(1)  027270  041777  041522  041440  EM24:   .ASCIZ  <377>/CRC CALCULATION ERROR /
(1)  027320  047111  042524  052522  DH2:    .ASCIZ /INTERUPTED UNEXPECTEDLY./
(1)  027351     377  054105  042520  DH6:    .ASCIZ <377>/EXPECTED   FOUND    PRI REG /
(1)  027407     377  051103  020103  EM1:    .ASCIZ  <377>/CRC GENERATOR STATUS /
(1)  027436  042377  052101  020101  MH1:    .ASCIZ  <377>/DATA CHAR     DATA BIT IN CRC GEN.     CRC FOR THIS BIT  /
(1)  027531     106  044501  042514  DH7:    .ASCIZ  /FAILED TO BIT-STUFF. /
(1)  027557     015  050012  044522  EM5:    .ASCIZ  <15><12>/PRIMARY REGISTER ADDRESSING TIME-OUT /
(1)  027627     015  050012  044522  EM7:    .ASCIZ <15><12>/PRIMARY REGISTER RESET TEST  /
(1)  027667     377  042522  044507  DH5:    .ASCIZ  <377>/REGISTER REFERENCED       TRAPPED FROM /
(1)                                  .EVEN
(1)  027732                          .ERRTAB:
(1)  027732  000000                        0
(1)
(1)  027734  000000                        0
(1)  027736  000000                        0
(1)  027740  027557                        EM5
(1)  027742  027667                        DH5      ;HALT 1
(1)  027744  030116                        DT5
(1)
(1)
(1)  027746  027627                        EM7
(1)  027750  027351                        DH6      ;HALT 2
(1)  027752  030130                        DT6
(1)
(1)  027754  026774                        EM6
(1)  027756  027351                        DH6      ;HALT 3
(1)  027760  030130                        DT6
(1)
(1)  027762  027025                        EM10
(1)  027764  000000                        0        ;HALT 4
(1)  027766  000000                        0
(1)
(1)
(1)  027770  027113                        EM12
(1)  027772  027126                        EM13     ;HALT 5
(1)  027774  000000                        0
(1)
(1)  027776  027113                        EM12
(1)  030000  027147                        EM14     ;HALT 6
(1)  030002  000000                        0
(1)
```

```
(1)  030004  027166                          EM15
(1)  030006  027147                          EM14    ;HALT 7
(1)  030010  000000                          0
(1)
(1)  030012  027203                          EM16
(1)  030014  027126                          EM13    ;HALT 10
(1)  030016  000000                          0
(1)
(1)  030020  027166                          EM15
(1)  030022  027126                          EM13    ;HALT 11
(1)  030024  000000                          0
(1)
(1)  030026  027216                          EM17
(1)  030030  027126                          EM13    ;HALT 12
(1)  030032  000000                          0
(1)
(1)  030034  027216                          EM17
(1)  030036  027147                          EM14    ;HALT 13
(1)  030040  000000                          0
(1)
(1)  030042  027252                          EM23
(1)  030044  027234                          EM22    ;HALT 14
(1)  030046  000000                          0
(1)
(1)  030050  027252                          EM23
(1)  030052  027147                          EM14    ;HALT 15
(1)  030054  000000                          0
(1)
(1)  030056  027252                          EM23
(1)  030060  027126                          EM13    ;HALT 16
(1)  030062  000000                          0
(1)
(1)  030064  027113                          EM12
(1)  030066  027234                          EM22    ;HALT 17
(1)  030070  000000                          0
(1)
(1)  030072  027270                          EM24
(1)  030074  027351                          DH6     ;HALT 20
(1)  030076  030130                          DT6
(1)
(1)  030100  000003                  DT1:    3
(1)  030102     006      021                 .BYTE   6,17.
(1)  030104  001252                          SAVR1
(1)  030106     006      017                 .BYTE   6,15.
(1)  030110  001254                          SAVR2
(1)  030112     006      002                 .BYTE   6,2
(1)  030114  001256                          SAVR3
(1)  030116  000002                  DT5:    2
(1)  030120     006      017                 .BYTE   6,15.
(1)  030122  001252                          SAVR1
(1)  030124     006      002                 .BYTE   6,2
(1)  030126  001254                          SAVR2
(1)  030130  000003                  DT6:    3
(1)  030132     006      004                 .BYTE   6,4
(1)  030134  001262                          SAVR5
(1)  030136     006      002                 .BYTE   6,2
```

```
(1)  030140  001260                          SAVR4
(1)  030142     006      002                 .BYTE   6,2
(1)  030144  001256                          SAVR3
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)
(1)  030146                                  CORMAX:
                030001                       .END
```

```
ABRT    007000          2352#
ACC     006750          2343#  6256   6286   6316   6346
ADRCNT= 003735          1932*  1968*  1977#
BCCFBK  026302          6368*  6374*  6377*  6380   6384   6396#
BINWRD  004240          2019*  2020   2057#
BITW  = 002000          1409#
BIT0  = 000001          1271#  1391   2802   3484   3488   3492   3494   3892   3899   3904   4252   4259   4264
                        6372
BIT1  = 000002          1270#  1390   2832   3132   3136   3140   3142   3516   3520   3524   3526   4292   4299
                        4304   5481
BIT10 = 002000          1261#  1381   1396   1409   1429   2674   3102   3356   3360   3364   3366   3804   3808
                        3812   3814   4012   4019   4024   4652   4659   4664
BIT11 = 004000          1260#  1380   1408   1428   2706   3388   3392   3396   3398   4052   4059   4064
BIT12 = 010000          1259#  1379   1395   1402   1407   1427   2738   3420   3424   3428   3430   4092   4099
                        4104   4692   4699   4704   4977   4984   4989   5188   5191   5200   5205
BIT13 = 020000          1258#  1378   1406   1426   2770   3452   3456   3460   3462   4132   4139   4144   5017
                        5024   5029
BIT14 = 040000          1257#  1377   1394   1405   1425   1691   1839   4172   4179   4184   4732   4739   4744
                        4893   4901   4906   5057   5064   5069
BIT15 = 100000          1256#  1376   1393   1400   1404   1424   1688   4212   4219   4224   4772   4779   4784
                        4935   4943   4948   5097   5104   5109   5236   5239   5248   5253   6257   6287   6317
                        6347
BIT2  = 000004          1269#  1389   1777   2862   3164   3168   3172   3174   3548   3552   3556   3558   4332
                        4339   4344   5513
BIT3  = 000010          1268#  1388   1415   2610   2892   3196   3200   3204   3206   3292   3296   3300   3302
                        3580   3584   3588   3590   4372   4379   4384   5545
BIT4  = 000020          1267#  1387   1414   2642   2922   3228   3232   3236   3238   3324   3328   3332   3334
                        3612   3616   3620   3622   4412   4419   4424   5419
BIT5  = 000040          1266#  1386   2952   3644   3648   3652   3654   4452   4459   4464   5389   5609   5613
                        5617   5619
BIT6  = 000100          1265#  1385   1413   2982   3676   3680   3684   3686   4492   4499   4504   5359   5449
                        5577   5581   5585   5587   5641   5645   5649   5651
BIT7  = 000200          1264#  1384   1412   2439   2440   2456   2563   2613   2619   2645   2651   2677   2683
                        2709   2715   2741   2747   2773   2779   3012   3708   3712   3716   3718   3932   3939
                        3944   4532   4539   4544   4813   4825   4855   4871   4897   4913   4939   4955   5224
                        5452   5458
BIT8  = 000400          1263#  1383   1398   1411   1431   3042   3260   3264   3268   3270   3740   3744   3748
                        3750   4572   4579   4584   5286   5296   5301   5304   5329
BIT9  = 001000          1262#  1382   1397   1401   1410   1430   3072   3772   3776   3780   3782   3972   3979
                        3984   4612   4619   4624   4851   4859   4864   5140   5143   5152   5157
BPW     003306          1783   1861#
BRX     003310          1784   1862#
CALBCC  026304          6249*  6258   6279*  6288   6309*  6318   6339*  6390*  6397#  6427*  6428   6451   6463
CARDET= 010000          1379#
CHRCNT  004236          2017*  2021   2037*  2055#  2056
CLK   = 020000          1406#  2187   2190   5768   5773
CLK.A   001433          1517#  2303
CLRVEC  005104          2212#
CNVRT = 104411          1484#  1800   1802   1804   1806   2130   2132
CONVRT= 104410          1482#  1736   2146   6479
CORMAX  030146          6487#
CRCEN = 001000          1401#  5714   5762   5870   5974   6041   6089   6116   6162   6207
CRCERR= 010000          1395#
CRC.CC= 102010          6236   6399#  6424
CRC16 = 120001          6398#
CREAM   001316          1371#  1598*  2257*  2258   2260*  2265   2266*  2267   2270*
```

```
CTS    = 020000          1378#
CYCLE    006226          1785    1820    1821    2247#
DATA     001324          1438#
DATABP   004612          2119*   2122    2144    2147#
DATAHD   004600          2118*   2140    2143#
DECMOD= 100000           1400#
DEVADR   003732          1930*   1965    1975#
DH2      027320          6487#
DH5      027667          6487#
DH6      027351          6487#
DH7      027531          6487#
DISPLA   001200          1315#   1611*   1797*
DISPRE   000174          1303#   1611
DSCA   = 100000          1376#
DSCB   = 000001          1391#
DSINTE= 000040           1386#
DSR    = 001000          1382#
DTR    = 000002          1390#
DT1      030100          6480    6487#
DT5      030116          6487#
DT6      030130          6487#
DUPACT   001310          1365#   1701*   1725*   1758*   1759    2247    2252
DUPCR0   001500          1526#   1640    1704
DUPCR1   001506          1530#   1705
DUPCR2   001514          1534#
DUPCR3   001522          1538#
DUPCR4   001530          1542#
DUPCR5   001536          1546#
DUPCR6   001544          1550#
DUPCR7   001552          1554#
DUPNUM   001311          1366#   1594    1693*   1726*   1810
DUPRPS   001376          1496#   2293*   2294*   2295
DUPRVC   001374          1495#   1827    2272*   2293
DUPSEC   001416          1504#   2279*
DUPTPS   001402          1498#   2297*   2298*
DUPTR0   001502          1527#   1648
DUPTR1   001510          1531#
DUPTR2   001516          1535#
DUPTR3   001524          1539#
DUPTR4   001532          1543#
DUPTR5   001540          1547#
DUPTR6   001546          1551#
DUPTR7   001554          1555#
DUPTVC   001400          1497#   2295*   2296*   2297
DUP.EN   001560          1558#   1633    1717    2258    2267
DUP.MA   001500          1371    1525#   1598    1631    1715    1719    1733    2260    2270
DUP0.A   001504          1528#   1688*   1691*
DUP1.A   001512          1532#
DUP2.A   001520          1536#
DUP3.A   001526          1540#
DUP4.A   001534          1544#
DUP5.A   001542          1548#
DUP6.A   001550          1552#
DUP7.A   001556          1556#
EM1      027407          6476    6487#
EM:0     027025          6487#
```

```
EM11    027057          6487#
EM12    027113          6487#
EM13    027126          6487#
EM14    027147          6487#
EM15    027166          6487#
EM16    027203          6487#
EM17    027216          6487#
EM2     026710          6487#
EM22    027234          6487#
EM23    027252          6487#
EM24    027270          6487#
EM3     026730          6487#
EM4     026745          6487#
EM5     027557          6487#
EM6     026774          6487#
EM7     027627          6487#
ERRCNT  001232          1336#   1600*   1833    2169*
ERRFLG  001341          1451#   1596*   1795*   1853*   2107*   2120    2134*
ERRMSG  004566          2117*   2135    2138#
ERTABO  004772          2132    2178#
EXIT  = 000000          1272#
EXITER  004726          2152    2169#
FIX.00  006744          2304    2338#
FLAG    001332          1441#   1629*   1742    1745*
FLG     007052          2365#   5839    6006    6218
HALTS   004616          2103    2149#
HDXEN = 000010          1415#
HILIM   003730          1929*   1956    1974#
HUPPSR  001420          1505#   2283*   2284
HUPRBF  001422          1506#   2281*   2282*   2283
HUPRCR  001424          1507#   2275*   2276*   2277
HUPTBF  001426          1508#   2290*   2291*
HUPTCR  001430          1509#   2286*   2287*   2288
ICOUNT  001222          1332#   1851    1856*
INBUF   006162          1899    1935    2066*   2067    2071    2234#
INIFLG  001340          1450#   1615    1621*   1630*
INSTER= 104404          1474#   1950    2076
INSTR = 104403          1472#   1635    1643    1651    1660    1668    1672    1678    1682    2310
INSTR2  003530          1906    1918#
LIMITS  003656          1945    1956#
LOBITS  003734          1931*   1960    1976#   1977
LOCK    001220          1331#   1855*   1869    1871    2126    2399*   2431*   2452*   5778*   5788*   5794*   6238*   6269*
                        6298*   6328#
LOKFLG  001342          1452#
LOLIM   003726          1928*   1958    1973#
LPCNT   001224          1333#   1850*   1851    1854*
LSTERR  001234          1337#   1601*   1794*   1837*   2104    2106*
LOO.OO  001434          1519#   2273*   2301
MASK.A  001432          1515#   2302
MASTEK  005412          2128    2223#
MCRLF   005136          1884    2008    2124    2125    2133    2223#   2309    2326
MCSR    005474          1636    2223#
MCSRX   005342          1799    2223#
MDATA   006120          2035    2045    2232#
MEPASS  005157          1798    2223#
MERRPC  005467          2131    2223#
```

```
MERRX   005367          1805    2223#
MERR2   005206          2223#   2249
MERR3   005255          1755    2223#
MEXT  = 010000          1419#   2159
MH1     027436          6477    6487#
MIND    001330          1440#
MJMPR   005526          1669    2223#
MLOCK   005313          1779    2223#
MMODE = 014000          1418#   5715    5717    5761    5835    5869    5973    6040    6088    6115    6161    6206    6243
                        6273    6303    6333    6433
MMODEA= 004000          1408#
MMODEB= 010000          1407#
MNEW    005414          1750    2223#
MPAR    005703          1652    2223#
MPASSX  005356          1803    2223#
MPOWER  005141          2208    2223#
MQM     005132          1916    2223#   2332
MR      005203          1786    2223#   2327
MRESET= 000400          1272#   1411#   3134    3166    3198    3230    3262    3294    3326    3358    3390    3422    3454
                        3486    3518    3550    3582    3614    3646    3678    3710    3742    3774    3806    3901    3941
                        3981    4021    4061    4101    4141    4181    4221    4261    4301    4341    4381    4421    4461
                        4501    4541    4581    4621    4661    4701    4741    4781    4826    4861    4903    4945    4986
                        5026    5066    5106    5154    5202    5250    5298    5579    5611    5643    5671    5689    5712
                        5758    5832    5866    5970    6037    6085    6108    6112    6158    6203    6240    6270    6300
                        6330    6422    6462
MSRJM   005762          1683    2223#
MSTJM   005727          1679    2223#
MTCN    005601          1673    2223#
MTDATA= 040000          1405#   2346    2355    2367    2372    2379    5811    5886    5919    5982    5995    6010    6060
                        6099    6124    6131    6181
MTITLE  001000          1310#   1620
MTOTAL  005650          1661    2223#
MTSTN   005400          2129    2223#   2311
MTSTPC  005301          2223#
MVEC    005513          1644    2223#
MVECX   005350          1801    2223#
NEXT    001216          1330#   1857    2174    2398*   2430*   2484*   2508*   2532*   2557*   2583*   2608*   2640*   2672*
                        2704*   2736*   2768*   2800*   2830*   2860*   2890*   2920*   2950*   2980*   3010*   3040*   3070*
                        3100*   3130*   3162*   3194*   3226*   3258*   3290*   3322*   3354*   3386*   3418*   3450*   3482*
                        3514*   3546*   3578*   3610*   3642*   3674*   3706*   3738*   3770*   3802*   3835*   3863*   3890*
                        3930*   3970*   4010*   4050*   4090*   4130*   4170*   4210*   4250*   4290*   4330*   4370*   4410*
                        4450*   4490*   4530*   4570*   4610*   4650*   4690*   4730*   4770*   4810*   4849*   4891*   4933*
                        4975*   5015*   5055*   5095*   5138*   5186*   5234*   5282*   5327*   5357*   5387*   5417*   5447*
                        5479*   5511*   5543*   5575*   5607*   5639*   5670*   5708*   5755*   5829*   5863*   5967*   6034*
                        6082*   6155*   6200*   6235*   6421*
OPCLRJ  001323          1437#   1671    1686    1730*
OVRRUN= 040000          1394#
PARAM = 104405          1476#   1637    1645    1653    1662    2312
PARAM1  003576          1934#   1951
PARBIT= 000000          1272#
PARCSR  001410          1501#   2280*   2534    3864    5139    5187    5235    5714*   5762*   5870*   5974*   6041*   6089*
                        6116*   6162*   6207*
PARERR  003652          1937    1939    1941    1950#   1957    1959    1961
PASCNT  001230          1335#   1595*   1796*   1797    1830
PERFOR= 000000          1272#
PKCLK = 104412          1486#   2344    2354    2366    2371    2378    5735    5767    5774    5800    5804    5838    5881
```

```
                        5883    5884    5918    5977    5979    5981    5994    6001    6003    6005    6009    6044    6046
                        6047    6051    6057    6059    6092    6094    6096    6098    6119    6121    6123    6128    513C
                        6165    6168    6169    6173    6177    6210    6212    6214    6246    6248    6255    6276    6278
                        6285    6306    6308    6315    6336    6338    6345    6437    6439    6440
POPRO = 012600          1250#   2168
POP1SP= 005726          1248#
POP2SP= 022626          1252#   1859
PRIRTY  001320          1435#   1659*   1723*
PRISEC= 010000          1402#
PS    = 177776          1241#   1591*   1773*
PUSHRO= 010046          1249#   2153
PUSH1S= 005746          1247#
PUSH2S= 024646          1251#
PWRUP   005060          2202    2204#
QV.FLG  001343          1453#   1597*   1809*   1848
RABORT= 002000          1396#
RCRCIN= 040000          1425#
RCRC7T= 100000          1424#
RCVEN = 000020          1387#   6434
RECACT= 004000          1380#
REOM  = 001000          1397#
RESREG  004614          2145    2148#
RESTRT  003116          1808    1812    1820#
RESOS = 104407          1480#   2148
RETURN  001214          1329#   1603*   1785*   1787    1820*   1857*   1860    2174*   2176    2209    2325*   2335*   2336
RING  = 040000          1377#
RINTEN= 000100          1385#
RSOM  = 000400          1398#
RTS   = 000004          1389#
RUN     001314          1369#   1599*   2252    2255*   2256*   2263*   2264*
RXCSR   001404          1499#   1824    2271*   2275    2401    2433    2451    2486    3131    3163    3195    3227    3259
                        3891    3931    3971    4011    4051    4091    4131    4171    4211    5328    5358    5388    5418
                        5480    5512    5544    5576    5608    6434*
RXDBUF  001406          1500#   2277*   2278*   2279    2280    2281    2510    3836    4251    4291    4331    4371    4411
                        4451    4491    4531    4571    4611    4651    4691    4731    4771
RXDERR= 100000          1393#
RXDONE= 000200          1384#
SAVACT  001312          1367#   1692*   1698*   1700    1701    1727*   1753
SAVNUM  001313          1368#   1594*   1694*   1728*   1807*   1810*
SAVPC   001266          1354#   1982*   2180
SAVPCA  001306          1363#
SAVR0   001250          1347#   1991*   1996
SAVR0A  001270          1356#
SAVR1   001252          1348#   1990*   1997    6426*   6487
SAVR1A  001272          1357#
SAVR2   001254          1349#   1989*   1998    6441*   6487
SAVR2A  001274          1358#
SAVR3   001256          1350#   1988*   1999    6469*   6472*   6487
SAVR3A  001276          1359#
SAVR4   001260          1351#   1987*   2000    6487
SAVR4A  001300          1360#
SAVR5   001262          1352#   1986*   2001    6487
SAVR5A  001302          1361#
SAVSP   001264          1353#
SAVSPA  001304          1362#
SAVOS = 104406          1478#   2108
```

```
SCOPE = 104400          1466* 2411  2464  2493  2516  2541  2567  2592  2624  2656  2688  2720  2752
                        2784  2814  2844  2874  2904  2934  2964  2994  3024  3054  3084  3114  3146
                        3178  3210  3242  3274  3306  3338  3370  3402  3434  3466  3498  3530  3562
                        3594  3626  3658  3690  3722  3754  3786  3818  3844  3874  3914  3954  3994
                        4034  4074  4114  4154  4194  4234  4274  4314  4354  4394  4434  4474  4514
                        4554  4594  4634  4674  4714  4754  4794  4832  4875  4917  4959  4999  5039
                        5079  5119  5167  5215  5263  5311  5341  5371  5401  5431  5463  5495  5527
                        5559  5591  5623  5655  5691  5739  5814  5841  5932  6016  6065  6139  6184
                        6220  6357  6464
SCOP1 = 104401          1468* 2405  2445  2460  5786  5793  5798  6266  6296  6326  6355
SEND  = 000020          1414* 5716  5717  5763  5775  5799  5836  5871  5975  6008  6042  6056  6090
                        6095  6117  6122  6163  6208  6244  6274  6304  6334  6435
SETFLG= 104413          1488* 1670  1674  1680  1684
SHIFTS  001326          1439*
SIMBCC  026126          6250  6280  6310  6340  6362* 6444
SMALL   005044          2197* 3135  3167  3199  3231  3263  3295  3327  3359  3391  3423  3455  3487
                        3519  3551  3583  3615  3647  3679  3711  3743  3775  3807  3902  3942  3982
                        4022  4062  4102  4142  4182  4222  4262  4302  4342  4382  4422  4462  4502
                        4542  4582  4622  4662  4702  4742  4792  4827  4962  4904  4946  4987  5027
                        5067  5107  5155  5203  5251  5291  5299  5580  5612  5644  5672  5690  5713
                        5759  5833  5867  5971  6038  6086  6109  6113  6159  6204  6241  6271  6301
                        6331  6423  6463
SPACNT= 004237          2018* 2039  2042* 2056*
SRD   = 002000          1381*
SRJMFL  001336          1443* 1685
STACK = 001150          1242* 1592  1774  2175  2205
STD   = 000010          1388*
STJMFL  001334          1442* 1681
STPSYN= 000400          1383*
SVOS    003744          1986*
SWR     001202          1316* 1608  1610* 1622  1626  1740  1748  1753  1758  1777  1939  1846  1867
                        1880  2097  2102  2151  2170  2172  2307  6466
SWREG   000176          1304* 1610
SW00  = 000001          1223* 1626  1740
SW01  = 000002          1222* 2307
SW02  = 000004          1221*
SW03  = 000010          1219* 1748
SW04  = 000020          1218*
SW05  = 000040          1217*
SW06  = 000100          1216*
SW07  = 000200          1215*
SW08  = 000400          1214* 2170
SW09  = 001000          1213* 1867  6466
SW10  = 002000          1212* 2172
SW11  = 004000          1211* 1846
SW12  = 010000          1210* 1880  2097
SW13  = 020000          1209* 2102
SW14  = 040000          1208*
SW15  = 100000          1207*
SYSTST= 004000          1420* 2159  5674
TABORT= 002000          1429* 5978  5992
TBL1    023632          5878  5879  5934*
TCNFLG  001322          1436* 1675  1676  1689  1729*
TCRCIN= 010000          1427*
TCRC7T= 020000          1426*
TEMP    006056          2022  2230*
```

```
TEMP1   001236          1342#  1665   1693   1694   1695*  1733*  1734   1738*  2225   5673*  5677*  5678   5681*
                        5684*  5685   5873*  5889*  5899   5925*  5980*  5985*  5986   5993*  5998*  5999   6058*
                        6063*  6097*  6102*  6129*  6134*  6365*  6388*
TEMP2   001240          1343#  1656   1659   1700*  1703*  1707*  1734*  2227   5874*  5995*  5898*  5916   6366*
                        6370*
TEMP3   001242          1344#  5875*  5894*  5897*  5899   5926*  6367*  6369   6379*  6381   6386*  6387*  6390
                        6429*  6442*  6455
TEMP4   001244          1345#  2353*  2361*  2365*  2382*  5876*  5922*  5923   5927*  6237*  6247   6267*  6277
                        6297*  6307   6327*  6337   6356*  6430*  6441   6443*  6452   6458
TEMP5   001246          1346#  2352*  2358*  2359   2370*  2375*  2376   5880*  5891*  5929*  6431*  6474   6478*
TEOM  = 001000          1430#  5910   5930   6004   6213   6254   6284   6314   6344   6457
TIMER = 004000          1428#  5675   5682
TKCSR   001204          1321#  1842   1901   6481
TKDBR   001206          1322#  1844   1903   1909   6483
TLAST = 026306          2330   6487#
TPCSR   001210          1323#  1885   1907   2099
TPDBR   001212          1324#  1887*  1909*  2101*
TRPOK   004330          2088#
TRP.PC  006224          2236#
TSOM  = 000400          1431#  5734   5766   5779   5789   5795   5837   5872   5976   6043   6091   6118   6164
                        6167   6209   6213   6245   6275   6305   6335   6436
TSTNO   001226          1334#  1602*  2183   2315   2321   2323   2397*  2429*  2483*  2507*  2531*  2556*  2582*
                        2607*  2639*  2671*  2703*  2735*  2767*  2799*  2829*  2859*  2889*  2919*  2949*  2979*
                        3009*  3039*  3069*  3099*  3129*  3161*  3193*  3225*  3257*  3289*  3321*  3353*  3385*
                        3417*  3449*  3481*  3513*  3545*  3577*  3609*  3641*  3673*  3705*  3737*  3769*  3801*
                        3834*  3862*  3889*  3929*  3969*  4009*  4049*  4089*  4129*  4169*  4209*  4249*  4289*
                        4329*  4369*  4409*  4449*  4489*  4529*  4569*  4609*  4649*  4689*  4729*  4769*  4809*
                        4848*  4890*  4932*  4974*  5014*  5054*  5094*  5137*  5185*  5233*  5281*  5326*  5356*
                        5386*  5416*  5446*  5478*  5510*  5542*  5574*  5606*  5638*  5669*  5707*  5754*  5828*
                        5862*  5966*  6033*  6081*  6154*  6199*  6234*  6420*
TST1    007160          2318   2335   2397#
TST10   007714          2583   2607#
TST100  016250          4450   4489#
TST101  016362          4490   4529#
TST102  016474          4530   4569#
TST103  016606          4570   4609#
TST104  016720          4610   4649#
TST105  017032          4650   4689#
TST106  017144          4690   4729#
TST107  017256          4730   4769#
TST11   010000          2608   2639#
TST110  017370          4770   4809#
TST111  017476          4810   4848#
TST112  017620          4849   4890#
TST113  017742          4891   4932#
TST114  020064          4933   4974#
TST115  020176          4975   5014#
TST116  020310          5015   5054#
TST117  020422          5055   5094#
TST12   010064          2640   2671#
TST120  020534          5095   5137#
TST121  020662          5138   5185#
TST122  021010          5186   5233#
TST123  021136          5234   5281#
TST124  021270          5282   5326#
TST125  021344          5327   5356#
```

```
TST126  021420          5357    5386#
TST127  021474          5387    5416#
TST13   010150          2672    2703#
TST130  021550          5417    5446#
TST131  021634          5447    5478#
TST132  021720          5479    5510#
TST133  022004          5511    5542#
TST134  022070          5543    5574#
TST135  022164          5575    5606#
TST136  022260          5607    5638#
TST137  022354          5639    5669#
TST14   010234          2704    2735#
TST140  022510          5670    5707#
TST141  022664          5708    5754#
TST142  023200          5755    5828#
TST143  023270          5829    5862#
TST144  023644          5863    5966#
TST145  024132          5967    6033#
TST146  024334          6034    6081#
TST147  024706          6082    6154#
TST15   010320          2736    2767#
TST150  025102          6155    6199#
TST151  025236          6200    6234#
TST152  026306          6235    6420#   6487
TST153= ****** U        6421
TST16   010404          2768    2799#
TST17   010460          2800    2829#
TST2    007264          2398    2429#
TST20   010534          2830    2859#
TST21   010610          2860    2889#
TST22   010664          2890    2919#
TST23   010740          2920    2949#
TST24   011014          2950    2979#
TST25   011070          2980    3009#
TST26   011144          3010    3039#
TST27   011220          3040    3069#
TST3    007442          2430    2483#
TST30   011274          3070    3099#
TST31   011350          3100    3129#
TST32   011444          3130    3161#
TST33   011540          3162    3193#
TST34   011634          3194    3225#
TST35   011730          3226    3257#
TST36   012024          3258    3289#
TST37   012120          3290    3321#
TST4    007504          2484    2507#
TST40   012214          3322    3353#
TST41   012310          3354    3385#
TST42   012404          3386    3417#
TST43   012500          3418    3449#
TST44   012574          3450    3481#
TST45   012670          3482    3513#
TST46   012764          3514    3545#
TST47   013060          3546    3577#
TST5    007542          2508    2531#
TST50   013154          3578    3609#
```

```
TST51   013250          3610    3641#
TST52   013344          3642    3673#
TST53   013440          3674    3705#
TST54   013534          3706    3737#
TST55   013630          3738    3769#
TST56   013724          3770    3801#
TST57   014020          3802    3834#
TST6    007604          2532    2556#
TST60   014060          3835    3862#
TST61   014122          3863    3889#
TST62   014234          3890    3929#
TST63   014346          3930    3969#
TST64   014460          3970    4009#
TST65   014572          4010    4049#
TST66   014704          4050    4089#
TST67   015016          4090    4129#
TST7    007652          2557    2582#
TST70   015130          4130    4169#
TST71   015242          4170    4209#
TST72   015354          4210    4249#
TST73   015466          4250    4289#
TST74   015600          4290    4329#
TST75   015712          4330    4369#
TST76   016024          4370    4409#
TST77   016136          4410    4449#
TTST    003174          1780*   1781*   1783*   1784*   1840#
TWOSYN= 000000          1272#
TXACT = 001000          1410#   5769    5805    5988    6104    6136    6215
TXCSR   001412          1502#   2159*   2187*   2190*   2284*   2285*   2286    2346    2355    2367    2372    2379    2559
                        2609    2641    2673    2705    2737    2769    3134*   3166*   3198*   3230*   3262*   3291    3294*
                        3323    3326*   3355    3358*   3387    3390*   3419    3422*   3451    3454*   3486*   3518*   3550*
                        3582*   3614*   3646*   3678*   3710*   3742*   3774*   3806*   3901*   3941*   3981*   4021*   4061*
                        4101*   4141*   4181*   4221*   4261*   4301*   4341*   4381*   4421*   4461*   4501*   4541*   4581*
                        4621*   4661*   4701*   4741*   4781*   4812    4826*   4850    4861*   4892    4903*   4934    4945*
                        4986*   5026*   5066*   5106*   5154*   5202*   5250*   5283    5298*   5448    5579*   5611*   5640
                        5643*   5671*   5674*   5689*   5710    5712*   5729    5756    5758*   5768*   5773*   5787    5831
                        5832*   5864    5866*   5912    5968    5970*   6035    6037*   6048    6052    6056*   6060    6083
                        6085*   6C  5*  6099    6104    6108*   6110    6112*   6122*   6124    6131    6136    6156    6158*
                        6170    6174    6178    6181    6201    6203*   6215    6240*   6243*   6244*   6270*   6273*   6274*
                        6300*   6303*   6304*   6330*   6333*   6334*   6422*   6433*   6435*   6462*
TXDBUF  001414          1503#   2288*   2289*   2290    2585    2801    2831    2861    2891    2921    2951    2981    3011
                        3041    3071    3101    3483    3515    3547    3579    3611    3643    3675    3707    3739    3771
                        3803    4811    4976    5016    5056    5096    5675    5682    5709    5727    5757    5784    5830
                        5865    5969    6036    6045*   6084    6093*   6111    6120*   6157    6166*   6167*   6202    6211*
                        6213*   6242*   6245*   6247*   6254*   6272*   6275*   6277*   6284*   6302*   6305*   6307*   6314*
                        6332*   6335*   6337*   6344*   6432*   6436*   6438*   6454*   6457*
TXDLAT= 100000          1404#   6048    6052    6170    6174    6178
TXDONE= 000200          1412#   5717    5731    5736    5790    5801    5808    6013
TXINTE= 000100          1413#
TYPDAT  004602          2123    2141    2144#
TYPE  = 104402          1470#   1620    1732    1750    1755    1779    1786    1798    1799    1801    1803    1805    1884
                        1897    1916    2008    2045    2124    2125    2128    2129    2131    2133    2137    2142    2207
                        2249    2309    2326    2327    2332    6476    6477
TYPMSG  004502          2121    2124#
USEP  = 000000          1417#
WRDCNT  004234          2016*   2046*   2054#
```

CROSS REFERENCE TABLE -- USER SYMBOLS

```
WRK0.F  004570          2136    2139#
XBX     004374          2098    2100    2102#
XCSR    003130          1800    1822#
XERR    003152          1806    1831#
XHEAD   006015          1732    2223#
XPASS   003144          1804    1828#
XPOLY   026300          6236#   6375    6386    6395#   6424#
XSTATQ  006044          1737    2223#
XTSTN   005000          2130    2181#
XVEC    003136          1802    1825#
$E    = 000154             1#  2398    2400#   2430    2432#   2484    2485#   2508    2509#   2532    2533#   2557    2558#
                        2583    2584#   2608    2609#   2640    2641#   2672    2673#   2704    2705#   2736    2737#   2768
                        2769#   2800    2801#   2830    2831#   2860    2861#   2890    2891#   2920    2921#   2950    2951#
                        2980    2981#   3010    3011#   3040    3041#   3070    3071#   3100    3101#   3130    3131#   3162
                        3163#   3194    3195#   3226    3227#   3258    3259#   3290    3291#   3322    3323#   3354    3355#
                        3386    3387#   3418    3419#   3450    3451#   3482    3483#   3514    3515#   3546    3547#   3578
                        3579#   3610    3611#   3642    3643#   3674    3675#   3706    3707#   3738    3739#   3770    3771#
                        3802    3803#   3835    3836#   3863    3864#   3890    3891#   3930    3931#   3970    3971#   4010
                        4011#   4050    4051#   4090    4091#   4130    4131#   4170    4171#   4210    4211#   4250    4251#
                        4290    4291#   4330    4331#   4370    4371#   4410    4411#   4450    4451#   4490    4491#   453?
                        4531#   4570    4571#   4610    4611#   4650    4651#   4690    4691#   4730    4731#   4770    477 
                        4810    4811#   4849    4850#   4891    4892#   4933    4934#   4975    4976#   5015    5016#   5055
                        5056#   5095    5096#   5138    5139#   5186    5187#   5234    5235#   5282    5283#   5327    5328#
                        5357    5358#   5387    5388#   5417    5418#   5447    5448#   5479    5480#   5511    5512#   5543
                        5544#   5575    5576#   5607    5608#   5639    5640#   5670    5671#   5708    5709#   5755    5756#
                        5829    5830#   5863    5864#   5967    5968#   6034    6035#   6082    6083#   6155    6156#   6200
                        6201#   6235    6236#   6421    6422#
$ENDAD  003104          1298    1618    1814#   2149
$N    = 000152             1#  2386    2391    2400#   2416    2423    2432#   2471    2477    2485#   2496    2501    2509#
                        2519    2525    2533#   2544    2550    2558#   2570    2576    2584#   2595    2601    2609#   2627
                        2633    2641#   2659    2665    2673#   2691    2697    2705#   2723    2729    2737#   2755    2761
                        2769#   2787    2793    2801#   2817    2823    2831#   2847    2853    2861#   2877    2883    2891#
                        2907    2913    2921#   2937    2943    2951#   2967    2973    2981#   2997    3003    3011#   3027
                        3033    3041#   3057    3063    3071#   3087    3093    3101#   3117    3123    3131#   3149    3155
                        3163#   3181    3187    3195#   3213    3219    3227#   3245    3251    3259#   3277    3283    3291#
                        3309    3315    3323#   3341    3347    3355#   3373    3379    3387#   3405    3411    3419#   3437
                        3443    3451#   3469    3475    3483#   3501    3507    3515#   3533    3539    3547#   3565    3571
                        3579#   3597    3603    3611#   3629    3635    3643#   3661    3667    3675#   3693    3699    3707#
                        3725    3731    3739#   3757    3763    3771#   3789    3795    3803#   3821    3828    3836#   3847
                        3856    3864#   3877    3883    3891#   3917    3923    3931#   3957    3963    3971#   3997    4003
                        4011#   4037    4043    4051#   4077    4083    4091#   4117    4123    4131#   4157    4163    4171#
                        4197    4203    4211#   4237    4243    4251#   4277    4283    4291#   4317    4323    4331#   4357
                        4363    4371#   4397    4403    4411#   4437    4443    4451#   4477    4483    4491#   4517    4523
                        4531#   4557    4563    4571#   4597    4603    4611#   4637    4643    4651#   4677    4683    4691#
                        4717    4723    4731#   4757    4763    4771#   4796    4803    4811#   4836    4842    4850#   4878
                        4884    4892#   4920    4926    4934#   4962    4968    4976#   5002    5008    5016#   5042    5048
                        5056#   5082    5088    5096#   5122    5131    5139#   5170    5179    5187#   5218    5227    5235#
                        5266    5275    5283#   5314    5320    5328#   5344    5350    5358#   5374    5380    5388#   5404
                        5410    5418#   5434    5440    5448#   5466    5472    5480#   5498    5504    5512#   5530    5536
                        5544#   5562    5568    5576#   5594    5600    5608#   5626    5632    5640#   5658    5663    5671#
                        5695    5701    5709#   5742    5748    5756#   5817    5822    5830#   5843    5856    5864#   5952
                        5960    5968#   6020    6027    6035#   6068    6075    6083#   6141    6148    6156#   6187    6193
                        6201#   6222    6228    6236#   6402    6414    6422#   6487#
$RAYO = 177777       1#  2386#   2387    2389#   2416#   2417    2421#   2471#   2472    2475#   2496#   2497    2499#
                        2519#   2520    2523#   2544#   2545    2548#   2570#   2571    2574#   2595#   2596    2599#   2627#
                        2628    2631#   2659#   2660    2663#   2691#   2692    2695#   2723#   2724    2727#   2755#   2756
```

```
                    2759#  2787#  2788   2791#  2817#  2818   2821#  2847#  2848   2851#  2877#  2878   2881#
                    2907#  2908   2911#  2937#  2938   2941#  2967#  2968   2971#  2997#  2998   3001#  3027#
                    3028   3031#  3057#  3058   3061#  3087#  3088   3091#  3117#  3118   3121#  3149#  3150
                    3153#  3181#  3182   3185#  3213#  3214   3217#  3245#  3246   3249#  3277#  3278   3281#
                    3309#  3310   3313#  3341#  3342   3345#  3373#  3374   3377#  3405#  3406   3409#  3437#
                    3438   3441#  3469#  3470   3473#  3501#  3502   3505#  3533#  3534   3537#  3565#  3566
                    3569#  3597#  3598   3601#  3629#  3630   3633#  3661#  3662   3665#  3693#  3694   3697#
                    3725#  3726   3729#  3757#  3758   3761#  3789#  3790   3793#  3821#  3822   3826#  3847#
                    3848   3854#  3877#  3878   3681#  3917#  3918   3921#  3957#  3958   3961#  3997#  3998
                    4001#  4037#  4038   4041#  4077#  4078   4081#  4117#  4118   4121#  4157#  4158   4161#
                    4197#  4198   4201#  4237#  4238   4241#  4277#  4278   4281#  4317#  4318   4321#  4357#
                    4358   4361#  4397#  4398   4401#  4437#  4438   4441#  4477#  4478   4481#  4517#  4518
                    4521#  4557#  4558   4561#  4597#  4598   4601#  4637#  4638   4641#  4677#  4678   4681#
                    4717#  4718   4721#  4757#  4758   4761#  4796#  4797   4801#  4836#  4837   4840#  4878#
                    4879   4882#  4920#  4921   4924#  4962#  4963   4966#  5002#  5003   5006#  5042#  5043
                    5046#  5082#  5083   5086#  5122#  5123   5129#  5170#  5171   5177#  5218#  5219   5225#
                    5266#  5267   5273#  5314#  5315   5318#  5344#  5345   5348#  5374#  5375   5378#  5404#
                    5405   5408#  5434#  5435   5438#  5466#  5467   5470#  5498#  5499   5502#  5530#  5531
                    5534#  5562#  5563   5566#  5594#  5595   5598#  5626#  5627   5630#  5658#  5659   5661#
                    5695#  5696   5699#  5742#  5743   5746#  5817#  5818   5820#  5843#  5844   5854#  5952#
                    5953   5958#  6020#  6021   6026#  6068#  6069   6073#  6141#  6142   6146#  6187#  6188
                    6191#  6222#  6223   6226#  6402#  6403   6413#
$Y      = 000014    1457#  1466   1468#  1470#  1472#  1474#  1476#  1478#  1480#  1482#  1484#  1486#  1488#
                    1490#
.       = 030146    1283#  1284   1287#  1294#  1299#  1302#  1305#  1309#  1311#  1365#  1366#  1367#  1368#
                    1369#  1370#  1436#  1437#  1442#  1443#  1524#  1526#  1527#  1528#  1530#  1531#  1532#
                    1534#  1535#  1536#  1538#  1539#  1540#  1542#  1543#  1544#  1546#  1547#  1548#  1550#
                    1551#  1552#  1554#  1555#  1556#  1627   1757   2223#  2231#  2233#  2235#  2236#  2251
                    2347   5802   6453
.BEGIN  002660      1773#
.CNVRT  004034      1485   2009#
.CONVR  004030      1483   2008#
.EOP    002764      1794#  6421
.ERRTA  027732      2116   6487#
.HLT    004350      1290   2097#
.INSTE  003516      1475   1914#
.INSTR  003412      1473   1893#
.INSTI  003432      1897#  1917
.MSG    003434      1895*  1898#
.PARAM  003536      1477   1925#
.PFAIL  005050      1288   1593   2202#  2206
.PKCLK  005006      1487   2184#
.RESOS  003776      1481   1996#
.SAVOS  003736      1479   1982#
.SCOPE  003160      1467   1837#
.SCOPI  003312      1469   1867#
.SETFL  004242      1489   2065#  2077
.START  001562      1306   1591#  1603
.TRPSR  004316      1292   2085#
.TRPTA  001344      1465#  2090
.TYPE   003336      1471   1877#
```

```
DUPEND      1#  1788
DUPFRN      1#  1190
HLT      1253#  2357  2369  2374  2381  2413  2443  2459  2492  2515  2540  2566  2591  2616  2623
         2648   2655  2680  2687  2712  2719  2744  2751  2776  2783  2807  2813  2837  2843  2867
         2873   2897  2903  2927  2933  2957  2963  2987  2993  3017  3023  3047  3053  3077  3083
         3107   3113  3139  3145  3171  3177  3203  3209  3235  3241  3267  3273  3299  3305  3331
         3337   3363  3369  3395  3401  3427  3433  3459  3465  3491  3497  3523  3529  3555  3561
         3587   3593  3619  3625  3651  3657  3683  3689  3715  3721  3747  3753  3779  3785  3811
         3817   3843  3873  3898  3906  3913  3938  3946  3953  3978  3986  3993  4018  4026  4033
         4058   4066  4073  4098  4106  4113  4138  4146  4153  4178  4186  4193  4218  4226  4233
         4258   4266  4273  4298  4306  4313  4338  4346  4353  4378  4386  4393  4418  4426  4433
         4458   4466  4473  4498  4506  4513  4538  4546  4553  4578  4586  4593  4618  4626  4633
         4658   4666  4673  4698  4706  4713  4738  4746  4753  4778  4786  4793  4818  4824  4831
         4858   4866  4874  4900  4908  4916  4942  4950  4958  4983  4991  4998  5023  5031  5038
         5063   5071  5078  5103  5111  5118  5145  5151  5159  5166  5193  5199  5207  5214  5241
         5247   5255  5262  5288  5295  5303  5310  5334  5340  5364  5370  5394  5400  5424  5430
         5455   5462  5487  5494  5519  5526  5551  5558  5584  5590  5616  5622  5648  5654  5680
         5687   5721  5728  5733  5738  5771  5777  5785  5792  5797  5803  5807  5810  5813  5901
         5921   5984  5990  5997  6012  6015  6050  6054  6062  6101  6106  6126  6133  6138  6172
         6176   6180  6183  6217  6262  6292  6322  6351
JOB         1#
$ABORT   1190#  2352
$ADDT1   1190#  2386
$ADDT2   1190#  2594  2626  2658  2690  2722  2754  2786  2816  2846  2876  2906  2936  2966  2996
         3026   3056  3086  5313  5343  5373  5403  5433  5465  5497  5529
$ADDT3   1190#  3820  3846
$ADDT4   1190#  3876  3916  3956  3996  4036  4076  4116  4156  4196  4236  4276  4316  4356  4396
         4436   4476  4516  4556  4596  4636  4676  4716  4756  4835  4877  4919  4961  5001  5041
         5081   5121  5169  5217
$ADDT5   1190#  3116  3148  3180  3212  3244  3276  3308  3340  3372  3404  3436  3468  3500  3532
         3564   3596  3628  3660  3692  3724  3756  3788  5561  5593  5625
$ADDT6   1190#  2470  2495  2518  2543  2569
$ADDT7   1190#  2416
$ADDT8   1190#  4796
$ADDT9   1190#  5265
$AUTOS      1#
$BCC     1190#  2343
$BUFFE      1#  2229
$CLKDN      1#  2190  5773
$CLKUP      1#  2187  5768
$CLRVE      1#  2210
$CYCLE      1#  2238
$DEC23   1190#  6401
$DEFAU      1#  1715
$DELY       1#
$EOP        1#  1788
$ERRTA      1#
$FINI       1#  6487
$FLAG    1190#  2365
$GETFL      1#  1668  1672  1678  1682
$GETPA      1#  1635  1643  1651  1660  2310
$HEADE      1#  1190
$MSG        1#  2223
$MSGA    6487#
$PFAIL      1#  2200
$POKCK      1#  2184
```

```
$QUEST      1#  1626
$RESET      1#  3134  3166  3198  3230  3262  3294  3326  3358  3390  3422  3454  3486  3518  3550
            3582  3614  3646  3678  3710  3742  3774  3806  3901  3941  3981  4021  4061  4101  4141
            4181  4221  4261  4301  4341  4381  4421  4461  4501  4541  4581  4621  4661  4701  4741
            4781  4826  4861  4903  4945  4986  5026  5066  5106  5154  5202  5250  5298  5573  5611
            5643  5671  5688  5712  5758  5832  5866  5970  6037  6085  6108  6112  6158  6203  6239
            6269  6299  6329  6422  6462
$SCOPE      1#  1834
$SCOP2      1#  1190#  6466
$SETFL      1#  2059
$SIMBC   1190#  6362
$SMALL      1#  2195
$TCRCX   1190#  6238  6268  6298  6328
$TIME    1190#  5658
$TRPDE      1#  1466  1468  1470  1472  1474  1476  1478  1480  1482  1484  1486  1488
$TSETU   1190#  5864  6035  6083  6110  6156  6201
$TSTN       1#  2391  2423  2477  2501  2525  2550  2576  2601  2633  2665  2697  2729  2761  2793
            2823  2853  2883  2913  2943  2973  3003  3033  3063  3093  3123  3155  3187  3219  3251
            3283  3315  3347  3379  3411  3443  3475  3507  3539  3571  3603  3635  3667  3699  3731
            3763  3795  3828  3856  3883  3923  3963  4003  4043  4083  4123  4163  4203  4243  4283
            4323  4363  4403  4443  4483  4523  4563  4603  4643  4683  4723  4763  4803  4842  4884
            4926  4968  5008  5048  5088  5131  5179  5227  5275  5320  5350  5380  5410  5440  5472
            5504  5536  5568  5600  5632  5663  5701  5748  5822  5856  5960  6027  6075  6148  6193
            6228  6414
$TT1     1190#  5694
$TT10    1190#  6019
$TT11    1190#  6067
$TT12    1190#  6140
$TT13    1190#  6186
$TT2     1190#  5741
$TT3     1190#  5816
$TT4     1190#  5842
$TT5     1190#  5951
$TXYCR   1190#  6222
$VARIQ      1#  1308
$WAIT       1#
$XZ         1#  2386  2389  2416  2421  2471  2475  2496  2499  2519  2523  2544  2548  2570  2574
            2595  2599  2627  2631  2659  2663  2691  2695  2723  2727  2755  2759  2787  2791  2817
            2821  2847  2851  2877  2881  2907  2911  2937  2941  2967  2971  2997  3001  3027  3031
            3057  3061  3087  3091  3117  3121  3149  3153  3181  3185  3213  3217  3245  3249  3277
            3281  3309  3313  3341  3345  3373  3377  3405  3409  3437  3441  3469  3473  3501  3505
            3533  3537  3565  3569  3597  3601  3629  3633  3661  3665  3693  3697  3725  3729  3757
            3761  3789  3793  3821  3826  3847  3854  3877  3881  3917  3921  3957  3961  3997  4001
            4037  4041  4077  4081  4117  4121  4157  4161  4197  4201  4237  4241  4277  4281  4317
            4321  4357  4361  4397  4401  4437  4441  4477  4481  4517  4521  4557  4561  4597  4601
            4637  4641  4677  4681  4717  4721  4757  4761  4796  4801  4836  4840  4878  4882  4920
            4924  4962  4966  5002  5006  5042  5046  5082  5086  5122  5129  5170  5177  5213  5225
            5266  5273  5314  5318  5344  5348  5374  5378  5404  5408  5434  5438  5466  5470  5498
            5502  5530  5534  5562  5566  5594  5598  5626  5630  5658  5661  5695  5699  5742  5746
            5817  5820  5843  5854  5952  5958  6020  6026  6068  6073  6141  6146  6197  6191  6222
            6226  6402  6413

. ABS.  030146    000
```

```
 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 DSKZ:DZDPBB,DSKZ:DZDPBB.SEQ/SOL/CRF/DOC/NL:TOC=DZDPBB.MAC,DZDPBB.P11
 RUN-TIME: 18 25 1 SECONDS
 RUN-TIME RATIO: 176/45=3.8
 CORE USED:  25K  (49 PAGES)

 DOCUMENT PAGES:  146
```