DV11

DV11 STAT LN CD TSTS

CZDVBC0

AH-8733C-MC

FICHE 1 OF 1

SEP 1979

digital

MADE IN USA

IDENTIFICATION

PRODUCT CODE:	AC-8732C-MC
PRODUCT NAME:	CZDVBC0 DV11 STAT LN CD TSTS
DATE RELEASED:	MARCH 1979
MAINTAINER:	DIAGNOSTICS
AUTHOR:	JOHN EGOLF, R.SOLER

1. ABSTRACT

The function of the DV11 diagnostics are to verify that the option operates according to specifications. The diagnostics verfiy that there are no malfunctions and the all operations of the DV11 are correct in its enviroment.
Parameters may be set to alert diagnostics as to the DV11 configuration by using the "TRIAL" program (CZDVE SA:210). All questions should be answered and then each diagnostic will "OVERLAY" these parameters which are stored in the "STATUS TABLE" (see section 8.4a). The alternative to "TRIAL" program is "AUTO SIZING" (see section 8.5).

CZDVB exercises all existing line cards in a static state (micro processor is NEVER TURNED ON). Transmitter and receiver flags,transmitter and receiver data,receiver syncing and char silo are tested. Most tests exercise a "group" of 4 lines at a time (00-03,04-07,08-11,12-15). For ease of trobleshooting; only one line card may be installed and by alerting the diagnostic as to which line cards are PHYSICALLY REMOVED (see section 8.4A) program will run any combination of line cards.

NOTE: Czdvbc0 has been enhanced to enable checking of parity logic, by enabling parity(odd/even) in any of the character lengths (5-8 bit). caution should be exercised in selecting sync characters however. ie. if parity selected is odd, make sure sync character selected contains odd number of bits. likewise when even parity selected be sure sync character contains even number of bits.

Currently there are six off line diagnostics that are to be run in sequence to insure that if an error should occur it will be detected at an early stage and insuring that diagnosis of error will be immediate to problem
NOTE: Additional diagnostics may be added in the future.

The six diagnostics are:
1. DZDVA [REV] Basis R/W test and ROM instruction exerciser.
2. czdvb [rev] DV11 STAT LN CD TSTS
3. CZDVC [REV] ROM TST PRT 1
4. DZDVD [REV] 'FREE RUNNING' Rom tests part 2.
5. CZDVE [REV] DV11 MODEM CNTRL
[TRIAL PROGRAM]
6. DZDVF [REV] Asynchronous line card tests.

2. REQUIREMENTS

2.1 EQUIPMENT

Any PDP11 family CPU (WITH MINIMUM 8K MEMORY)
ASR 33 (or equilivalent)
DV11-AA MUX CNTRL UNIT
AT LEAST ONE OF THE FOLLOWING
DV11-BA 8 LINE SYNC MODULES
DV11-BB 8 LINE ASYNC MODULES
DV11-BC 4 SYNC LINES, 4 ASYNC LINES

2.2 STORAGE

Program will use all 8K of memory except where ABL and BOOTSTRAP LOADER reside. Location 1500 thru 1736 are especially to be noted and to be untouched by operator after DV11 trial program has been executed; or after the 'AUTO SIZING' has been done.

3. LOADING PROCEEDURE

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
4k	17
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. STARTING PROCEEDURE

A. Set switch register to 000200
B. Depress 'LOAD ADDRESS' key and release
C. Set SWR to zero for 'AUTO SIZING' or leave
leave SWR bit 7=1 to use existing parameters set up by DV11 trial program or a previously run DV11 diagnostic that used the 'AUTO SIZING'.(section 7.2 and 8.4,8.5 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:

'MAP OF DV11 STATUS'	
1500	175000
1502	000300
1504	000226
1506	000062
1510	000226
1512	000062
1514	000226
1516	000062
1520	000226
1522	000062

The above is only an example! This would indicate the status table starting at add. 1500 in the program. THE STATUS TABLE MUST BE VERIFIED BY THE USER IF AUTO SIZING IS DONE. For information of status table see section 8.4 for help.

The program will type 'R' and proceed to run the diagnostic

4.1 CONTROL SWITCH SETTINGS

NOTE: If there is no read SWR (177570); SWR may be modified at Loc:176 or by hitting Control 'G' <^G> on console terminal.

SW 15	Set:	Halt on error
SW 14	Set:	Loop on current test
SW 13	Set:	Inhibit error print out
SW 12	Set:	Inhibit **ALL** type out/bell on error.
SW 11	Set:	Inhibit iterations. (quick pass)
SW 10	Set:	Escape to next test
SW 09	Set:	Loop with current data
SW 08	Set:	Catch error and loop on it
SW 07	Set:	Use previous status table. CLR-do AUTO SIZE.
SW 06	Set:	Reserved
SW 05	Set:	Reserved
SW 04	Set:	Reserved
SW 03	Set:	Reserved
SW 02	Set:	Lock on selected test
SW 01	Set:	Restart program at selected test
SW 00	Set:	Reselect DV11's desired active.

4.1.2 SWITCH REGISTER RESTRICTIONS

SW 00 RESELECT DV11'S DESIRED ACTIVE. please note that a message is typed out for setting the switch register equal to DV11's active. this means if the system has four DV11s; bits 00,01,02,03 will be set in loc 'DVACTV' from the switch register. Using this switch(SW00) alters that location;therefore if four DV11s are in the system ***DO NOT*** set switchs greater than SW 03 in the up position. this would be a fatal error. do not select more active DV11s than has been given information about in trial program.

METHOD: A: Load address 200
B: Start with SW 00=1
C: Program will type message
D: Set the binary number of DV11s desired active EXAMPLE: 1=1 DV11; 3=2 DV11; 7=3 DV11; 17=4 DV11 37=5 DV11 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 hass been made before trying to select a test that is not in the order of sequence the reason being is that the program has to clear areas and set up parameters. Also when a test is selected ALWAYS START AT THE VERY BEGINNING OF THAT TEST.

SW 09 LOOP ON CURRENT DATA: this switch will only work if call 'SCOP1' is in that test. The reason being that most tests deal with blocks of different data to be sent or received all at once thus in block data; one pattern cann't be singled out.

4.1.3 SWITCH REGISTER PRIORITYS

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).

SCOPE SWITCHES

1. SW 09 (if enabled by 'SCOP1') on an error; If an '*' is printed in front of the test no. (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); SW08 is best.
   (SW14=1,0, SW10=0, SW09=0, SW08=1). for intermittemt errors; SW14=1 will loop on test reguardless of error or not error.
   (SW14=1, SW10=0, SW09=0, SW08=1,0)
2. SW 14
3. SW 11

4.2 STARTING ADDRESS

starting address is at 000200 there are no other starting addresses for the DV11 diagnostics previously mentioned except for CZDVE which is: 000200 for the modem control and cable tests and 000210 for the manual parameter input program.

NOTE: If address 000042 is non-zero the program assumes it is under ACT11 or XXDP control and will act accordingly after *ALL* available DV11's are tested the program will return to 'XXDP' or 'ACT-11'.

5. OPERATING PROCEDURE

When program is initially started messages as described in section four will be printed.

and program will begin running the diagnostic

5.2 PROGRAM AND/OR OPERATOR ACTION

The typical approach should be

1. Halt on error (via SW 15=1) when ever 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 this way the EXACT FUNCTIONING of the test CAN BE INTERPEDITED.

6. 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 the error message which is to give the operator an indication of the error.

6.2 ERROR RECOVERY

If for some reason the DV11 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' (address 1224) for the number of the test that was running at the time of the catastrophic error. In this way the operator will have an idea as to what the DV11 was doing at the time of the error.

7. RESTRICTIONS

7.1 STARTING RESTRICTIONS

See section 4. (PLEASE)
Status table should be verified reguardless of how program was started.
Also it is important to use this listing along with the information printed on the TTY to completly isolate problems.

7.2 OPERATING RESTRICTIONS

DV11 trial program must be run prior to the first and only the first running of any DV11 diagnostic if ''AUTO SIZING'' is not used.
NOTE: If no program other than a DV11 diagnostic was loaded after DV11 trial or if core memory has not been changed;or if there is no DV11 configuration changes; the DV11 trial program need never be run again. However if any of the above have been violated the DV11 trial program must be run again before running the diagnostics NOTE: An alternative to the above is attempting the 'AUTO SIZING' when program is initially started with SW07=0.

7.3 HARDWARE CONFIGURATION RESTRICTIONS (SYNC LINE CARDS ONLY)

1. Hardware must be set to FULL DUPLEX
2. All lines of a particular line card must be configured the same.

8. MISCELLANEOUS

8.1 EXECUTION TIME

All DV11 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' untill all DV11's in system are tested. When the diagnostic has completed a pass the following is an example of the print out to be expected.

END PASS CZDVBC0 CSR: 175000 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.
NOTE: CZDVE (MODEM AND CABLE TEST) END PASS message is a large ''END'' typed out on tty. Please note that each character printed is actually and ''END PASS'' indication. This was used in place of ''BELL'' because if sw12=1 and an error occured the BELL may be mistaken for END PASS. The pass execution is so fast that the standard END PASS was too lenghtly. THEREFORE each char is an 'END PASS and the entire ''END'' is not required for acceptance.

8.4 KEY LOCATIONS

RETURN (1212) Contains the address where program will return when iteration count is reached or if loop on test is asserted.

NEXT (1214) Contains the address of the next test to be peformed.

TSTNO (1224) Contains the number of the test now being peformed.

RUN (1302) The bit in 'RUN' always points one past the DV11 currently being tested. EXAMPLE: (RUN) 1302/0000000001000000 Means that DV11 no.05 is the DV11 now running.

DVCR00-DVCR17
DVST00-DVST17
(1500)-(1736)

These locations contain the information needed to test up to 8 (decimal) DV11s sequentialy. they contain the CSR,VECTOR and STATUS concerning the configuration of each DV11.

DVACTV (1276) Each bit set in this location indicates that the associated DV11 will be tested in turn. EXAMPLE: (DVACTV) 1276/0000000000011111 means that DV11 no. 00,01,02,03,04 will be tested. EXAMPLE: (DVACTV) 1276/0000000000010C01 Means that DV11 no. 00,04 will be tested.

DVSCR (1356) Contains the receiver csr of the current DV11 under test.

L00.03 (1412)
L04.07 (1414)
L08.11 (1416)
L12.15 (1420) Contains the status of the current DV11 under test.

BIT 15 Set: Line card *NOT installed (AND WONT BE TESTED)
BIT 14 Set: Parity enabled
BIT 13 Set: Even parity selected
BIT 12 Set: One sync, =0: two syncs.
BIT 11 Set: Async line card, =0 Sync line card.
BIT 10 Set: Reserved
BIT 09 Set: Bits per char. (used with bit8)
BIT 08 Set: Bits per char. (used with bit9)

BIT09	BIT08	BITS PER CHAR.
0	0	8
0	1	7
1	0	6
1	1	5

BIT 07-00 SYNC "A" for specified line card. Bits 07-00 must be all zeros for testing Async line cards.

8.4A MORE ON THAT 'STATUS TABLE' (1500-1736)

```
        'MAP OF DV11 STATUS'
1500    175000
1502    000300
1504    000226
1506    000062
1510    000226
1512    000062
1514    004000
1516    000000
1520    004000
1522    000000
```

The above information will be repeated for each of up to 8 DV11's in the system(these will follow under this table). EXPLANATION:

1500 175000 This is the system control register for the 1st DV11 in the system.

1502 000300 This is vector 'A' for the first DV11 in the system.

1504 000226 This represents 'SYNC A' and the software status for the 1st line card in the 1st DV11. The bits are as follows:

```
BIT 15  Set:    Line card *NOT installed (AND WONT BE TESTED)
BIT 14  Set:    Parity enabled
BIT 13  Set:    Even parity selected
BIT 12  Set:    One sync, =0:  two syncs.
BIT 11  Set:    Async line card, =0 Sync line card.
BIT 10  Set:    Reserved
BIT 09   Set:   Bits per char.  (used with bit8)
BIT 08  Set:    Bits per char.  (used with bit9)
        BIT09 BIT08 BITS PER CHAR.
          0     0     8
          0     1     7
          1     0     6
          1     1     5
BIT 07-00       SYNC 'A' for specified line card.
```

1506 000062 This represents 'SYNC B' for the 1st line card.

1510 000226 This is 'SYNC A' and line status for the 2nd line card. (for bits defination see explanation for line card 1).

1512 000062 This is 'SYNC B' for the second line card.

1514 000226 This is 'SYNC A' and line status for the 3rd line card. (for bits defination see explanation for line card 1).

1516 000062 This is 'SYNC B' for line card no. 3.

1520 000226 This is 'SYNC A' and line status for the 4th line card. (for bits defination see explanation for line card 1).

1522 000062 This is SYNC B for the 4th line card.

The above is repeated for each DV11 in the system. The table is filled by AUTO SIZING or by the manual parameter input program as described previously. Also if desired by user; the locations may be altered by hand (toggled in) to suit the specific configuration.

8.5 *** METHOD OF AUTO SIZING ***

8.5.1 FINDING THE CONTROL STATUS REGISTER.

The program will start at address 175000 and start 'REFERENCEING' address. If a NON-EX MEMORY TRAP occures; the pointer (holding 175000) is updated by 10 and the above is repeated untill address 175400 is reached. If a 'SLAVE SYNC RESPONSE' was issued by the DV11 (or any other device) (no nxm trap)(and it (SEL0) was=0) ; pointer plus 12 (SEL12) is tested to contain 177777 (MUST BE EXACTLY 177777); if a trap is encountered or if SEL12 does not contain 177777 the above updating is performed. If SEL12 was equal to 177777 the pointer is stored away and the routine continues as above:
NOTE: If the program does not find your DV11; something is wrong and AUTO SIZING should not be done.

8.5.2 FINDING THE VECTOR

The vector area (address 300-776) is filled with the instruction IOT and '.+2' (next address). Bit7 and Bit6 (RX INTERUPT AND RX INTERUPT IE) are set into DVscr register; a delay is made and if no interupt occures (because of a bad DV11) the program assumes vector address 300 and the problem should be fixed in the diagnostic. Once the problem is fixed; the program should be re-setup again to get correct vector. If an interupt occured; the address to which the DV11 interupted to is picked up and reported as the vector. NOTE: if the vector reported is not the vector set up by you; there is a problem and AUTO SIZING should not be done.

8.5.3 PARAMETER ASSUMPTIONS.

Since too much hardware would need to be turned on to SIZE the rest of the parameters; the program must assume the remaining variations. The result if not to your specific configuration may be altered by hang (toggle in) is desired. In this way 95% of the parameter setup was done by the program and 5% by you.
THEREFORE:

1) ALL LINE CARDS(4) ARE ASSUMED TO BE INSTALLED.
   Set Bit15 of status map of any (appropriate) line cards missing
2) TWO SYNCS.
   Set Bit12 if you have a 4 line group set for 1 sync.
3) EIGHT BITS PER CHAR.
   Adjust bits 9 and bit 8 in status map for your correct config.
4) SYNCHRONOUS LINE CARDS INSTALLED
   Set bit11 of status map for Async line card and zero sync cards.
5) SYNC ''A''=226 AND SYNC ''B''=062

In all adjustments please refer to section 8.4a for greater detail.

DOCUMENT
****
CZDVBC LST
****

```
2       AC-8732C-MC/<377>/CZDVBC0 DV11 STAT LN CD TSTS
        COPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
        ------------------------------------------------------------------

1121    ROUTINE USED TO "AUTO SIZE" THE DV11
        CSR AND VECTOR.
        NOTE:   THE CSR MAY BE ANY WHERE IN THE FLOATING
                ADDRESS RANGE (175000:175400)
                AND THE VECTOR MAY BE ANY WHERE IN THE
                FLOATING VECTOR RANGE (300:770)


1214    ********** TEST 1 **********
        TEST THAT "TRANSMITTER FLAG WAITING"
        IS TRUE AND THAT "RECV FLAG WAITING" IS
        FALSE  AFTER AN INIT.
        THIS TEST WILL BE DONE FOR BOTH ASYNC AND SYNC LINE CARDS.

1280    ********** TEST 2 **********
        TEST THAT "MATCH DETECT" IS
        FALSE AFTER AN INIT.
        THIS TEST WILL BE DONE FOR BOTH ASYNC AND SYNC LINE CARDS.

1328    ********** TEST 3 **********
        TEST THAT MAINT BIT WINDOW IS CLEARED
         AFTER AN INIT.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

1378    ********** TEST 4 **********
        TEST THAT THE BIT WINDOW WILL
        STAY CLEARED WHEN MAINT INTERNAL

1381    MODE IS SELECTED BUT COND. STROBE IS
        NOT ASSERETED.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

1431    ********** TEST 5 **********
        TEST THAT THE BIT WINDOW WILL
        SET WHEN MAINT INTERNAL MODE IS SELECTED
        AND COND. STOBE IS ASSERTED.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

1485    ********** TEST 6 **********
        TEST THAT THE BIT WINDOW WILL BE CLEARED
        WHEN MAINT INTERNAL MODE IS SELECTED AND TX DSABLE
        IS ASSERTED.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

1539    ********** TEST 7 **********
        TEST THAT "MAINT DATA" WILL SHOW
        UP IN "MAINT BIT WINDOW".
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
```

```
1606    ********** TEST 10 **********
        TEST TO XMIT A BINARY COUNT PATTERN
        THRU THE USE OF THE BIT WINDOW.
        ONLY ONE LINE AT A TIME WILL BE EXERCISED.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

1724    VERIFY THAT SETTING TMARK BIT PUTS LINE AT MARK.


1746    ********** TEST 11 **********
        TEST TO CHECK THE IDLE CHARACTER
        FOR EACH LINE OF THE TRANSMITTER.
        THIS TEST USES ''SYNCA''.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

1852    ********** TEST 12 **********
        TEST TO CHECK THE IDLE CHARACTER
        FOR EACH LINE OF THE TRANSMITTER.
        THIS TEST USES ''SYNCB''.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

1962    ********** TEST 13 **********
        THIS TEST CHECKS ''RECEIVE CHAR SILO'' TO BE
        ALL ZERO'S WHEN ''DATA ENABLE'' IS NOT SET.
        EXPECTED DATA SHOULD BE LINE NUMBER ONLY
        DATA 0'S AND ERROR FLAGS 0.
        THIS TEST WILL BE DONE FOR BOTH ASYNC AND SYNC LINE CARDS.

2034    ********** TEST 14 **********
        THIS TEST CHECKS ''RECEIVER CHAR SILO''
        WHEN ''DATA ENABLE IS SET'' EXPECTED DATA S/B
        ALL 1'S FOR RX DATA, LINE NUMBER CORRECT,
        AND ERROR FLAGS =0.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

2111    ********** TEST 15 **********
        TEST THAT EACH RECEIVER WILL SET
        ''MATCH DETECT'' WHEN THE FIRST SYNC
        CHARACTER IS PUMPED INTO IT.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

2174    ********** TEST 16 **********
        TEST TO VERIFY THAT IF THE DV11 RECEIVER
        IS SET FOR ONE SYNC CHAR;
        ''MATCH DET'' *AND* ''CHAR FLAG'' ARE
        SET AFTER ONE SYNC IS PUSHED INTO THE RECEIVER
                        HOWEVER...
        IF THE DV11 RECEIVER IS SET FOR
        TWO SYNC CHARS....
        VERIFY THAT ''MATCH DET'' SETS ON THE FIRST SYNC
        AND VERIFY THAT ''MATCH DET'' *AND* ''CHAR FLAG''
        ARE SET ON THE SECOND SYNC.
        THIS TEST USES ''SYNC A''.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
```

```
2275    ************************ TEST 17 *********************************
        TEST TO VERIFY THAT IF THE DV11 RECEIVER

2277    IS SET FOR ONE SYNC CHAR;
        'MATCH DET'' *AND* ''CHAR FLAG'' ARE
        SET AFTER ONE SYNC IS PUSHED INTO THE RECEIVER
                    HOWEVER...
        IF THE DV11 RECEIVER IS SET FOR
        TWO SYNC CHARS....
        VERIFY THAT 'MATCH DET'' SETS ON THE FIRST SYNC
        AND VERIFY THAT 'MATCH DET'' *AND* ''CHAR FLAG''
        ARE SET ON THE SECOND SYNC.
        THIS TEST USES 'SYNC B''.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

2380    ************************ TEST 20 *********************************
        TEST TO FORCE RECEIVER OVERRUN.
        THIS TEST WILL PUSH INTO THE RECEIVER
        TWO FULL CHARS (SYNCS) AND ONE MORE CHAR MINUS
        ONE BIT. THE PROGRAM WILL VERIFY NO OVERRUN EXISTS
        THEN THE LAST BITS WILL BE PUSHED IN VERIFING
        THAT THE OVERRUN WAS GENERATED.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

2508    ************************ TEST 21 *********************************
        TEST OF RECEIVER DATA .
        THIS TEST RUNS A BINARY COUNT PATTERN THROUGH
        THE RECEIVER OF EACH LINE
        THROUGH THE USE OF MAINT. DATA BIT.
        THE TX IS NEVER ENABLED.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

2623    ************************ TEST 22 *********************************
        TEST OF RECEIVER PARITY LOGIC.
        THIS TEST RUNS PREDETERMINED DATA PATTERNS
        THROUGH THE RECEIVER OF EACH LINE, BY
        MEANS OF THE MAINTENACE DATA BIT. IF ODD
        PARITY IS SELECTED, AN EVEN DATA PATTERN
        IS GENERATED THROUGH THE RECEIVER WITH
        THE PARITY BIT CLEAR. THIS SHOULD CAUSE A
        RECEIVER PARITY ERROR. IF NOT, THEN WE CAN
        ASSUME THE PARITY CHECKING LOGIC IN THE
        RECEIVER IS DEFECTIVE. DATA IS STILL
        CHECKED TO INSURE INTEGERITY. EVEN PARITY
        WILL LIKEWISE BE TESTED BY GENERATING
        AN ODD DATA PATTERN. ALL CHARACTER LENGTHS
        MAY BE TESTED. THE TX IS NEVER ENABLED.
        THIS TEST WILL DE DONE FOR SYNC LINE CARDS ONLY.
```

```
2778    ********************** TEST 23 ********************************
        TEST OF RECEIVER DATA .
        THIS TEST RUNS A SET PATTERN THROUGH

2781    THE RECEIVER OF EACH LINE
        THROUGH THE USE OF THE TRANSMITTER.
        THIS TEST EXERCISES ALL LINES IN GROUPS OF 4.
        NOTE: SHOULD A DATA COMPARE ERROR OCCUR; THE PROGRAM
                REPORTS THE ERROR AS A RECEIVER DATA ERROR BASED
                ON THE TRANSMITTER HAS PREVIOUSLY BEEN CHECKED AND ASSUMED GOOD.
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

2940    ********************** TEST 24 ********************************
        TEST OF RECEIVER 'RE-SYNC''
        THIS TEST WILL SEND (BY BIT WINDOW) TWO SYNC CHARS AND
        THEN VERIFY THAT RX CHAR FLAG IS TRUE.
        THEN A 'RE-SYNC'' WILL BE ISSUED AND
        TWO NON-SYNC CHARS WIIL BE SENT INTO THE RX
        VERIFYING THAT THERE IS NO RX CHAR FLAG.
        NEXT TWO SYNC CHARS ARE AGAIN MOVED INTO THE RX
        VERIFING CHAR FLAG AND THE THE RX SOULD INDEED

2949                RE SYNC!
        THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.

3052    ********************** TEST 25 ********************************
        TEST TO VERIFY THAT SETTING RECEIVER ENABLE
        WILL SET RX FLAG AND MATCH DETECT.
        TEST WILL ALSO VERIFY THAT CLEARING RECEIVER
        ENABLE WILL CLEAR RX FLAG AND MATCH DETECT.
        THIS TEST WILL BE DONE FOR ASYNC LINE CARDS ONLY.

3140    ********************** TEST 26 ********************************
        TEST TO SET RECEIVER ENABLE.
        SET 'RX DATA ENABLE''.
        CLR 'RX DATA ENABLE''.
        AND EXPECT BOTH RX FLAG AND MATCH DETECT TO BE FALSE.
        THIS TEST WILL BE DONE FOR ASYNC LINE CARDS ONLY.

3211    ********************** TEST 27 ********************************
        TEST TO SET RECEIVER ENABLE.
        ISSUE A RESYNC  SIGNAL.
        AND EXPECT BOTH RX FLAG AND MATCH DETECT TO BE FALSE.
        THIS TEST WILL BE DONE FOR ASYNC LINE CARDS ONLY.
```

```
                        ;*AC-8732C-MC/<377>/CZDVBC0 DV11 STAT LN CD TSTS
                        ;*COPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
                        ;*-------------------------------------------------------------------

                                ;STARTING PROCEDURE
                                ;LOAD PROGRAM
                                ;LOAD ADDRESS 000200
                                ;PRESS START
                                ;PROGRAM WILL TYPE ''A 8732C-MC/<377>/CZDVBC0 DV11 STAT LN CD TSTS ''
                                ;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                ;=1, DO ''AUTO SIZING'' ON INITAL START UP.
  000100                SW06=100
  000040                SW05=40
  000020                SW04=20
  000010                SW03=10
  000004                SW02=4                  ;LOCK ON TEST SELECT
  000002                SW01=2                  ;RESTART PROGRAM AT SELECTED TEST
  000001                SW00=1                  ;RESELECT DV11 DESIRED ACTIVE
                                                ;NOTE: THIS MUST NOT EXCEED ORIGINAL COUNT
```

```
                        ;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
001200                  STACK=1200      ;START OF PROCESSOR STACK

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
                        ;--------------------------------
010000                  ALU=BIT12
020000                  RAM=BIT13
030000                  XFR=BIT13+BIT12
040000                  NPR=BIT14
050000                  S.C=BIT14+BIT12
060000                  BCC=BIT14+BIT13
070000                  BRB=BIT14+BIT13+BIT12
                        ;--------------------------------
```

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

            000000              .=0
                                        ;STANDARD INTERRUPT VECTORS
                                        ;-------------------------

            000024              .=24
000024  004402                          .PFAIL                  ;POWER FAIL HANDLER
000026  000340                          340                     ;SERVICE AT LEVEL 7
000030  004002                          .HLT                    ;ERROR HANDLER
000032  000340                          340                     ;SERVICE AT LEVEL 7
000034  003750                          .TRPSRV                 ;GENERAL HANDLER DISPATCH SERVICE
000036  000340                          340                     ;SERVICE AT LEVEL 7
            000040              .=40
000040  000001                          .BLKW 1                 ;SAVE FOR ACT-11 OR DDP2
000042  000001                          .BLKW 1                 ;RETURN ADDRESS IF UNDER ACT-11 OR DDP2
000044  000001                          .BLKW 1                 ;SAVE FOR ACT-11 OR DDP2
000046  002560                          LOGICAL                 ;FOR USE WITH ACT-11 OR DDP2

            000174              .=174
000174  000000              LIGHT:  0
            000176              .=176
000176  000000              SSWR:   0

            000200              .=200
000200  000137  001742              JMP     .START              ;GO TO START OF PROGRAM

            001000              .=1000
001000  005377  041501  034055  MTITLE: .ASCIZ  <377><12>/AC-8732C-MC/<377>/CZDVBC0 DV11 STAT LN CD TSTS /<377>
(2)
            001200              .=1200
001200                          LIGHTS:
001200  177570                          177570
001202  177570              SWR:        177570
                                        ;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
                                        ;--------------------------

001214  000000              RETURN: 0                           ;SCOPE ADDRESS FOR LOOP ON TEST
001216  000000              NEXT:   0                           ;ADDRESS OF NEXT TEST TO BE EXECUTED
001220  000000              LOCK:   0                           ;ADDRESS FOR LOCK ON CURRENT DATA
```

```
   138  001222  000003          ICOUNT: 3                       ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
   139  001224  000000          LPCNT:  0                       ;NUMBER OF ITEREATIONS COMPLETED
   140  001226  000000          TSTNO:  0                       ;NUMBER OF TEST IN PROGRESS
   141  001230  000000          PASCNT: 0                       ;NUMBER OF PASSES COMPLETED
   142  001232  000000          ERRCNT: 0                       ;TOTAL NUMBER OF ERRORS
   143  001234  000000          LSTERR: 0                       ;PC OF LAST ERROR CALL
   144
   145                                  ;PROGRAM VARIABLES
   146                                  ;-----------------
   147
   148  001236  000000          STAT:   0                       ;DV STATUS WORD STORAGE
   149  001240  000000          SYNCX:  0
   150  001242  000000          CLKX:   0
   151  001244  000000          MASKX:  0
   152  001246  000000          TEMP1:  0                       ;TEMPORARY STORAGE
   153  001250  000000          TEMP2:  0                       ;TEMPORARY STORAGE
   154  001252  000000          TEMP3:  0                       ;TEMPORARY STORAGE
   155  001254  000000          TEMP4:  0                       ;TEMPORARY STORAGE
   156  001256  000000          TEMP5:  0                       ;TEMPORARY STORAGE
   157  001260  000000          SAVR0:  0                       ;R0 STORAGE
   158  001262  000000          SAVR1:  0                       ;R1 STORAGE
   159  001264  000000          SAVR2:  0                       ;R2 STORAGE
   160  001266  000000          SAVR3:  0                       ;R3 STORAGE
   161  001270  000000          SAVR4:  0                       ;R4 STORAGE
   162  001272  000000          SAVR5:  0                       ;R5 STORAGE
   163  001274  000000          SAVSP:  0                       ;STACK POINTER STORAGE
   164  001276  000000          SAVPC:  0                       ;PROGRAM COUNTER STORAGE
   165  001300  000001          DVACTV: .BLKB 1                 ;DV11'S SELECTED ACTIVE.
   166  001301  000001          DVNUM:  .BLKB 1                 ;OCTAL NUMBER OF DV11'S.
   167  001302  000001          SAVACT: .BLKB 1                 ;ORIGINAL ACTV.  DEVICES.
   168  001303  000001          SAVNUM: .BLKB 1                 ;WORKABLE NUMBER.
   169  001304  000001          RUN:    .BLKB 1                 ;POINTER ONE PAST RUNNING DEVICE.
   170          001306          .EVEN
   171  001306  001500          CREAM:  DV.MAP                  ;TABLE POINTER.
```

```
                                        ;PROGRAM CONTROL FLAGS
                                        ;---------------------

001310  000     INIFLG: .BYTE   0                ;PROGRAM INITIALIZATION FLAG
001311  000     ERRFLG: .BYTE   0                ;ERROR OCCURED FLAG
001312  000     LOKFLG: .BYTE   0                ;LOCK ON CURRENT TEST FLAG
001313  000     QV.FLG: .BYTE   0                ;QUICK VERIFY FLAG.
                                                 ;ON FIRST PASS OF EACH DV11 ITERATIONS WILL BE SUPPRESSE

                .EVEN
000000          $Y=0

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

                                        ;:**************************************************************************
                                        ;--------------------------------------------------------------------------
001314          .TRPTAB:
104400          SCOPE=TRAP+0            ;CALL TO SCOPE LOOP AND ITERATION HANDLER
001314  002634          .SCOPE
104401          SCOP1=TRAP+1            ;CALL TO LOOP ON CURRENT DATA HANDLER
001316  003020          .SCOP1
104402          TYPE=TRAP+2             ;CALL TO TELETYPE OUTPUT ROUTINE
001320  003044          .TYPE
104403          INSTR=TRAP+3            ;CALL TO ASCII STRING INPUT ROUTINE
001322  003120          .INSTR
104404          INSTER=TRAP+4           ;CALL TO INPUT ERROR HANDLER
001324  003224          .INSTER
104405          PARAM=TRAP+5            ;CALL TO NUMERICAL DATA INPUT ROUTINE
001326  003244          .PARAM
104406          SAV05=TRAP+6            ;CALL TO REGISTER SAVE ROUTINE
001330  003444          .SAV05
104407          RES05=TRAP+7            ;CALL TO REGISTER RESTORE ROUTINE
001332  003504          .RES05
104410          CONVRT=TRAP+10          ;CALL TO DATA OUTPUT ROUTINE
001334  003536          .CONVRT
104411          CNVRT=TRAP+11           ;CALL TO DATA OUTPUT ROUNTINE WITHOUT CR/LF.
001336  003542          .CNVRT
104412          MSTCLR=TRAP+12          ;CALL TO ISUE A MASTER CLEAR
001340  004556          .MSTCLR
104413          RAMCLR=TRAP+13          ;CALL TO CLEAR THE RAMS
001342  004516          .RAMCLR
104414          DELAY=TRAP+14           ;CALL TO VARIABLE DELAY COUNTER
001344  004476          .DELAY
104415          ROMCLK=TRAP+15          ;CALL TO CLOCK ROM ONCE
001346  004566          .ROMCLK
104416          DATACLK=TRAP+16         ;CALL TO CLK DATA
001350  004576          .DATACLK

                                        ;--------------------------------------------------------------------------
                                        ;:**************************************************************************
```

```
                                        ;DV11 VECTOR AND REGISTER INDIRECT POINTERS

001352  000000          DVRVEC: 0               ;POINTER TO DV11 RECEIVER INTERRUPT VECTOR
001354  000000          DVRLVL: 0               ;POINTER TO DV11 RECEIVER INTERRUPT SERVICE PS
001356  000000          DVTVEC: 0               ;POINTER TO DV11 TRANSMITTER INTERRUPT VECTOR
001360  000000          DVTLVL: 0               ;POINTER TO DV11 TRANSMITTER INTERRUPT SERVICE PS
001362  000000          DVSCR:  0               ;POINTER TO DV11 SYSTEM CONTROL REGISTER
001364  000000          DVSCRH: 0               ;POINTER TO DV11 SYSTEM CONTROL REGISTER HIGH BYTE.
001366  000000          DVRIC:  0               ;POINTER TO DV11 NEXT RECEIVED CHARACTER REGISTER
001370  000000          DVLCR:  0               ;POINTER TO DV11 LINE PRAMETER REGISTER
001372  000000          DVSRS:  0               ;POINTER TO DV11 SECONDARY REGISTER SELECT REGISTER
001374  000000          DVSRSH: 0               ;POINTER TO DV11 SECONDARY REGISTER SELECT HIGH BYTE.
001376  000000          DVSRA:  0               ;POINTER TO DV11 SECONDARY REGISTER ACCESS REGISTER
001400  000000          DVSFR:  0               ;POINTER TO DV11 SPECIAL FUNCTIONS REGISTER
001402  000000          DVNSR:  0               ;POINTER TO DV11 NPR STATUS REGISTER
001404  000000          RESV16: 0               ;POINTER TO RESERVED REGISTER.


                                        ;DV11 CONTROL INDICATORS FOR CURRENT DV11 UNDER TEST
                                        ;--------------------------------------------------

001406  000000          MASK.A: .WORD 000       ;LAST CHAR TO TEST AND PARITY MASK FOR LINES 00-03
001410  000000          MASK.B: .WORD 000       ;LAST CHAR TO TEST AND PARITY MASK FOR LINES 04-07
001412  000000          MASK.C: .WORD 000       ;LAST CHAR TO TEST AND PARITY MASK FOR LINES 08-11
001414  000000          MASK.D: .WORD 000       ;LAST CHAR TO TEST AND PARITY MASK FOR LINES 12-15

001416      010         CLK.A:  .BYTE 8.        ;NUMBER OF CLOCKS NEEDED FOR ONE CHAR FOR LINES 00-03
001417      010         CLK.B:  .BYTE 8.        ;NUMBER OF CLOCKS NEEDED FOR ONE CHAR FOR LINES 04-07
001420      010         CLK.C:  .BYTE 8.        ;NUMBER OF CLOCKS NEEDED FOR ONE CHAR FOR LINES 08-11
001421      010         CLK.D:  .BYTE 8.        ;NUMBER OF CLOCKS NEEDED FOR ONE CHAR FOR LINES 12-15

001422  000000          L00.03: 000000          ;PARAMETERS FOR LINES 00-03
001424  000000          L04.07: 000000          ;PARAMETERS FOR LINES 04-07
001426  000000          L08.11: 000000          ;PARAMETERS FOR LINES 08-11
001430  000000          L12.15: 000000          ;PARAMETERS FOR LINES 12-15

001432  000000          SYNC2A: 000000          ;SYNC 2
001434  000000          SYNC2B: 000000          ;
001436  000000          SYNC2C: 000000          ;
001440  000000          SYNC2D: 000000          ;


                                        ;SUMMARY
                                        ;-------
                        ;       MASK.X          040     5 BITS PER CHAR.
                        ;                       100     6 BITS PER CHAR.
                        ;                       200     7 BITS PER CHAR.
                        ;                       400     8 BITS PER CHAR.

                        ;       CLK.X           005     5 BITS PER CHAR.
                        ;                       006     6 BITS PER CHAR.
                        ;                       007     7 BITS PER CHAR.
                        ;                       010     8 BITS PER CHAR.
                        ;       IF PARITY IS ENABLED; ADD PLUS ONE TO THE ABOVE "CLK.X"
                        ;       FOR EACH GROUP THAT PARITY IS ENABLED.
```

```
                                        ;DV11 STATUS TABLE AND ADDRESS ASSIGNMENTS
                                        ;-----------------------------------------

                001500          .=1500
001500                          DV.MAP:
001500  000001                  DVCR00: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 00
001502  000001                  DVTR00: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 00
001504  000001                  DV00.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 00
001506  000001                  SYNA00: .BLKW 1         ;SYNC TWO
001510  000001                  DV00.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 00
001512  000001                  SYNB00: .BLKW 1         ;SYNC TWO
001514  000001                  DV00.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 00
001516  000001                  SYNC00: .BLKW 1         ;SYNC TWO
001520  000001                  DV00.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 00
001522  000001                  SYND00: .BLKW 1         ;SYNC TWO

001524  000001                  DVCR01: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 01
001526  000001                  DVTR01: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 01
001530  000001                  DV01.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 01
001532  000001                  SYNA01: .BLKW 1         ;SYNC TWO
001534  000001                  DV01.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 01
001536  000001                  SYNB01: .BLKW 1         ;SYNC TWO
001540  000001                  DV01.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 01
001542  000001                  SYNC01: .BLKW 1         ;SYNC TWO
001544  000001                  DV01.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 01
001546  000001                  SYND01: .BLKW 1         ;SYNC TWO

001550  000001                  DVCR02: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 02
001552  000001                  DVTR02: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 02
001554  000001                  DV02.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 02
001556  000001                  SYNA02: .BLKW 1         ;SYNC TWO
001560  000001                  DV02.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 02
001562  000001                  SYNB02: .BLKW 1         ;SYNC TWO
001564  000001                  DV02.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 02
001566  000001                  SYNC02: .BLKW 1         ;SYNC TWO
001570  000001                  DV02.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 02
001572  000001                  SYND02: .BLKW 1         ;SYNC TWO

001574  000001                  DVCR03: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 03
001576  000001                  DVTR03: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 03
001600  000001                  DV03.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 03
001602  000001                  SYNA03: .BLKW 1         ;SYNC TWO
001604  000001                  DV03.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 03
001606  000001                  SYNB03: .BLKW 1         ;SYNC TWO
001610  000001                  DV03.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 03
001612  000001                  SYNC03: .BLKW 1         ;SYNC TWO
001614  000001                  DV03.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 03
001616  000001                  SYND03: .BLKW 1         ;SYNC TWO

001620  000001                  DVCR04: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 04
001622  000001                  DVTR04: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 04
001624  000001                  DV04.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 04
001626  000001                  SYNA04: .BLKW 1         ;SYNC TWO
001630  000001                  DV04.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 04
001632  000001                  SYNB04: .BLKW 1         ;SYNC TWO
001634  000001                  DV04.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 04
```

```
  334  001636  000001                SYNC04: .BLKW 1         ;SYNC TWO
  335  001640  000001                DV04.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 04
  336  001642  000001                SYND04: .BLKW 1         ;SYNC TWO
  337
  338  001644  000001                DVCR05: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 05
  339  001646  000001                DVTR05: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 05
  340  001650  000001                DV05.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 05
  341  001652  000001                SYNA05: .BLKW 1         ;SYNC TWO
  342  001654  000001                DV05.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 05
  343  001656  000001                SYNB05: .BLKW 1         ;SYNC TWO
  344  001660  000001                DV05.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 05
  345  001662  000001                SYNC05: .BLKW 1         ;SYNC TWO
  346  001664  000001                DV05.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 05
  347  001666  000001                SYND05: .BLKW 1         ;SYNC TWO
  348
  349  001670  000001                DVCR06: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 06
  350  001672  000001                DVTR06: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 06
  351  001674  000001                DV06.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 06
  352  001676  000001                SYNA06: .BLKW 1         ;SYNC TWO
  353  001700  000001                DV06.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 06
  354  001702  000001                SYNB06: .BLKW 1         ;SYNC TWO
  355  001704  000001                DV06.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 06
  356  001706  000001                SYNC06: .BLKW 1         ;SYNC TWO
  357  001710  000001                DV06.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 06
  358  001712  000001                SYND06: .BLKW 1         ;SYNC TWO
  359
  360  001714  000001                DVCR07: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 07
  361  001716  000001                DVTR07: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 07
  362  001720  000001                DV07.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 07
  363  001722  000001                SYNA07: .BLKW 1         ;SYNC TWO
  364  001724  000001                DV07.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 07
  365  001726  000001                SYNB07: .BLKW 1         ;SYNC TWO
  366  001730  000001                DV07.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 07
  367  001732  000001                SYNC07: .BLKW 1         ;SYNC TWO
  368  001734  000001                DV07.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 07
  369  001736  000001                SYND07: .BLKW 1         ;SYNC TWO
  370
  371  001740  000000                DV.END: 000000
  372
  373                                        ;PROGRAM INITIALIZATION
  374                                        ;LOCK OUT INTERRUPTS
  375                                        ;SET UP PROCESSOR STACK
  376                                        ;SET UP POWER FAIL VECTOR
  377                                        ;CLEAR PROGRAM CONTROL FLAGS AND COUNTS
  378                                        ;TYPE TITLE MESSAGE
  379
  380  001742  012737  000340  177776 .START: MOV     #340,PS                 ;LOCK OUT INTERRUPTS
  381  001750  012706  001200                 MOV     #STACK,SP               ;SET UP STACK
  382  001754  012737  004402  000024         MOV     #.PFAIL,@#24            ;SET UP POWER FAIL VECTOR
  383  001762  113737  001301  001303         MOVB    DVNUM,SAVNUM            ;SAVE NUMBER OF DEVICES IN SYSTEM.
  384  001770  005037  001230                 CLR     PASCNT                  ;CLEAR PASS COUNT
  385  001774  105037  001311                 CLRB    ERRFLG                  ;CLEAR ERROR FLAG
  386  002000  105037  001313                 CLRB    QV.FLG                  ;ZERO QUICK VERIFY FLAG
  387  002004  012737  001500  001306         MOV     #DV.MAP,CREAM           ;GET MAP POINTER.
  388  002012  112737  000001  001304         MOVB    #1,RUN                  ;POINT POINTER TO FIRST DEVICE.
  389  002020  005037  001232                 CLR     ERRCNT                  ;CLEAR ERROR COUNT
```

```
                                                CLR     LSTERR                  ;CLEAR LAST ERROR POINTER
                                                MOV     #1,TSTNO                ;SET UP FOR TEST 1
                                                MOV     #.START,RETURN          ;SET UP FOR POWER FAIL BEFORE
                                                                                ;TESTING STARTS
                                                TSTB    INIFLG                  ;HAS INITIALIZATION BEEN PERFORMED
                                                BNE     1$                      ;BR IF YES
                                                MOV     4,-(SP)                 ;
                                                MOV     6,-(SP)                 ;
                                                CLR     6                       ;
                                                MOV     #80$,4                  ;
                                                TST     @SWR                    ;
                                                NOP                             ;
                                                BR      81$                     ;
                                        80$:    CMP     (SP)+,(SP)+             ;
                                                MOV     #LIGHT,LIGHTS           ;
                                                MOV     #SSWR,SWR               ;
                                        81$:    MOV     (SP)+,6                 ;
                                                MOV     (SP)+,4                 ;
                                                TYPE    ,MTITLE                 ;TYPE TITLE MESSAGE
                                                COMB    INIFLG                  ;IF NOT SET FLAG AND DO
                                                TSTB    @SWR                    ;BIT7=1??
                                                BMI     16$                     ;BR IF NO AUTO SIZE
                                                JSR     PC,CSRMAP               ;GO DO THE AUTO SIZE
                                        16$:    TYPE    ,XHEAD          ;TYPE HEADER
                                                MOV     #DV.MAP,TEMP1           ;SET POINTER
                                        5$:     MOV     @TEMP1,TEMP2            ;SET DATA
                                                CMP     #177777,TEMP2           ;ALL DONE?
                                                BEQ     1$                      ;BR IF YES
                                                CONVRT                          ;
                                                XSTATQ                          ;
                                                ADD     #2,TEMP1                ;UPDATE POINTER
                                                BR      5$                      ;
                                        1$:     TST     @#42                    ;IS PROGRAM RUNNING UNDER MONITOR
                                                BNE     3$                      ;BR IF YES
                                                BIT     #SW00,@SWR              ;SELECT SPECIFIC DEVICES??
                                                BEQ     3$                      ;BR IF NO.
                                                TYPE    ,MNEW                   ;TYPE THE MESSAGE.
                                                CLR     R0                      ;ZERO DATA LIGHTS
                                                HALT                            ;WAIT FOR USER TO TELL WHAT DEVICES TO RUN
                                                CMPB    @SWR,SAVACT             ;IS THE NUMBER VALID?
                                                BLOS    2$                      ;BR IF NUMBER IS OK.
                                                TYPE    ,MERR3                  ;TELL USER OF INVALID NUMBER.
                                                HALT                            ;STOP EVERY THING.
                                                BR      .-2                     ;RESTART THE PROGRAM AGAIN.
                                        2$:     MOVB    @SWR,DVACTV             ;GET NEW DEVICE PATTERN
                                                MOVB    DVACTV,R0               ;SHOW THE USER WHAT HE SELECTED.
                                                BIC     #^C<377>,R0             ;USE ONLY LOW BYTE.
                                                HALT                            ;CONTINUE DYNAMIC SWITCHES.
                                        3$:     MOV     #300,R0                 ;PREPARE TO CLEAR THE FLOATING
                                                MOV     #302,R1                 ;VECTOR AREA. 300-776
                                        4$:     MOV     R1,(R0)+                ;START PUTTING 'PC+2 - HALT''
                                                CLR     (R1)+                   ;IN VECTOR AREA.
                                                CMP     (R0)+,(R1)+             ;POP POINTERS
                                                CMP     #1000,R0                ;ALL DONE??
                                                BNE     4$                      ;BR IF NO.
```

```
                                        ;TEST START AND RESTART
                                        ;----------------------

002332  012737  000340  177776  .BEGIN: MOV     #340,PS         ;LOCK OUT INTERRUPTS
002340  012706  001200                  MOV     #STACK,SP       ;SET UP STACK
002344  005737  000042                  TST     @#42            ;IS PROGRAM UNDER MONITOR CONTROL
002350  001023                          BNE     3$              ;BR IF YES
002352  032777  000004  176622          BIT     #BIT2,@SWR      ;CHECK FOR LOCK ON TEST
002360  001411                          BEQ     1$              ;BR IF NO LOCK DESIRED.
002362  104402  005301                  TYPE    ,MLOCK          ;TYPE LOCK SELECTED.
002366  012737  000240  002702          MOV     #NOP,TTST       ;ADJUST SCOPE ROUTINE.
002374  012737  000240  002704          MOV     #NOP,TTST+2     ;SET UP TO LOCK
002402  000406                          BR      2$              ;CONTINUE ALONG.
002404  013737  003014  002702  1$:     MOV     BRW,TTST        ;PREPARE NORMAL SCOPE ROUTINE
002412  013737  003016  002704          MOV     BRX,TTST+2      ;LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
002420                          2$:
002420  012737  005666  001214  3$:     MOV     #CYCLE,RETURN   ;START AT "CYCLE" FIND WHICH DEVICE TO TEST
002426  104402  005171          4$:     TYPE    ,MR             ;TYPE R
002432  000177  176556                  JMP     @RETURN         ;START TESTING
```

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

002436  000005                  .EOP:   RESET                   ;MAKE THE  WORLD CLEAN AGAIN.
002440  005037  001234                  CLR     LSTERR          ;CLEAR LAST ERROR PC
002444  105037  001311                  CLRB    ERRFLG          ;CLEAR ERROR FLAG
002450  005237  001230                  INC     PASCNT          ;UPDATE PASS COUNT
002454  013777  001230  176516          MOV     PASCNT,@LIGHTS  ;DISPLAY PASS COUNT
002462  104402  005145                  TYPE    ,MEPASS         ;TYPE END PASS
002466  104402  005330                  TYPE    ,MCSRX          ;TYPE CSR
002472  104411  002604                  CNVRT   ,XCSR           ;SHOW IT
002476  104402  005336                  TYPE    ,MVECX          ;TYPE VECTOR
002502  104411  002612                  CNVRT   ,XVEC           ;SHOW IT
002506  104402  005344                  TYPE    ,MPASSX         ;TYPE PASSES
002512  104411  002620                  CNVRT   ,XPASS          ;SHOW IT
002516  104402  005355                  TYPE    ,MERRX          ;TYPE ERRORS
002522  104411  002626                  CNVRT   ,XERR           ;SHOW IT
002526  105337  001303                  DECB    SAVNUM          ;ARE ALL DEVICES TESTED?
002532  001017                          BNE     RESTRT          ;BR IF NO.
002534  112737  000377  001313          MOVB    #377,QV.FLG     ;SET THE QUICK VERIFY FLAG.
002542  113737  001301  001303          MOVB    DVNUM,SAVNUM    ;RESTORE THE COUNT
002550  013701  000042                  MOV     @#42,R1         ;CHECK FOR ACT-11 OR DDP
002554  001406                          BEQ     RESTRT          ;IF NOT, CONTINUE TESTING
002556  000005                          RESET                   ;STOP THE SHOW--CLEAR THE WORLD
002560                          LOGICAL:
002560  004711                          JSR     PC,(R1)
002562  000240                          NOP
002564  000240                          NOP
002566  000240                          NOP
002570  000240                          NOP
002572  012737  005666  001214  RESTRT: MOV     #CYCLE,RETURN
002600  000137  005666                  JMP     CYCLE
002604  000001                  XCSR:   1
002606     006     002                  .BYTE   6,2
002610  001362                          DVSCR
002612  000001                  XVEC:   1
002614     003     002                  .BYTE   3,2
002616  001352                          DVRVEC
002620  000001                  XPASS:  1
002622     006     002                  .BYTE   6,2
002624  001230                          PASCNT
002626  000001                  XERR:   1
002630     006     002                  .BYTE   6,2
002632  001232                          ERRCNT

                                        ;SCOPE LOOP AND INTERATION HANDLER
                                        ;--------------------------------

002634                          .SCOPE:
002634  022737  177570  001202          CMP     #177570,SWR     ;IS THERE A REAL SWR?
002642  001411                          BEQ     64$             ;BR IF YES
002644  017746  176336                  MOV     @TKDBR,-(SP)    ;SAVE KEYBOARD CHAR
002650  042716  000200                  BIC     #BIT7,(SP)      ;CLEAR PARITY BIT
```

```
                                        CMPB    #7,(SP)         ;WAS IT CNTRL 'G' ?
                                        BNE     .+6             ;BR IF NO.
                                        JSR     PC,SERV.G       ;SERVICE ''CNTRL 'G'''.
                                64$:    CLR     LSTERR          ;CLEAR LAST ERROR PC.
                                        MOV     R0,(SP)         ;SAVE R0 ON THE STACK
                                        BIT     #BIT14,@SWR     ;''LOOP ON THIS TEST''?
                                TTST:   BEQ     1$              ;BR IF NO.  (IF LOCK SW01=1; THIS LOC =240)
                                        BR      3$              ;GOTO 3$    (IF LOCK SW01=1; THIS LOC =240)
                                        TSTB    @TKCSR          ;KEYBOAD DONE?
                                        BPL     3$              ;BR IF NO. (LOCK: HIT KEY TO GOTO NEXT TEST)
                                        MOV     @TKDBR,R0       ;CLEAR DONE BIT
                                        BR      2$              ;CONTINUE
                                1$:     BIT     #SW11,@SWR      ;DELETE ITERATION?  (QUICK PASS)
                                        BNE     2$              ;BR IF YES
                                        TSTB    QV.FLG          ;HAVE PASSES BEECOMPLETED?
                                        BEQ     2$              ;BR IF QUICK PASS.
                                        INC     LPCNT           ;UPDATE ITERATION COUNTER
                                        CMP     LPCNT,ICOUNT    ;ARE ALL ITERATIONS DONE??
                                        BNE     3$              ;BR IF NOT YET
                                2$:     CLRB    ERRFLG          ;PREPARE FOR NEW TEST
                                        CLR     LPCNT           ;START ICOUNTER AT 0
                                        CLR     LOCK
                                        MOV     #20,ICOUNT      ;RESET ITERATIONS
                                        MOV     NEXT,RETURN     ;GET NEXT TEST
                                3$:     MOV     (SP),R0         ;POP R0 OFF OF THE STACK
                                        POP2SP                  ;FAKE AN ''RTI''
                                        JMP     @RETURN         ;GO DO THE TEST
                                BRW:    1407
                                BRX:    437

                                        ;CHECK FOR FREEZE ON CURRENT DATA
                                        ;--------------------------------

                                .SCOP1: BIT     #SW09,@SWR      ;IS SW09=1(SET)?
                                        BEQ     1$              ;BR IF NOT SET.
                                        TST     LOCK
                                        BEQ     1$
                                        MOV     LOCK,(SP)       ;GOTO THE ADDRESS IN LOCK.
                                1$:     RTI                     ;GO BACK.

                                        ;TELETYPE OUTPUT ROUTINE
                                        ;-----------------------

                                .TYPE:  MOV     R5,-(SP)        ;SAVE R5 ON THE STACK.
                                        MOV     @2(SP),R5       ;GET ADDRESS OF MESSAGE.
                                        ADD     #2,2(SP)        ;POP OVER ADDRESS.
                                1$:     BIT     #SW12,@SWR      ;INHIBIT ALL PRINT OUT??
                                        BNE     3$              ;BR IF NO PRINT OUT WANTED (SW12=1)
                                        TSTB    (R5)            ;IS NUMBER MINUS? (MSB=1(BIT7))
                                        BPL     2$              ;BR IF NUMBER IS PLUS
                                        TYPE    ,MCRLF          ;TYPE A CR/LF!
                                2$:     TSTB    @TPCSR          ;TTY READY?
                                        BPL     2$              ;BR IF NO.
                                        MOVB    (R5)+,@TPDBR    ;PRINT CURRENT CHAR.
                                        BNE     1$              ;IF NOT ZERO KEEP PRINTING!
                                3$:     MOV     (SP)+,R5        ;END OF OUTPUT. RESTORE R5
```

```
                                        RTI                     ;GO HOME
                                        ;-----------------------
                                .INSTR: MOV     R3,-(SP)        ;SAVE R3 ON STACK
                                        MOV     R4,-(SP)        ;SAVE R4 ON STACK
                                        MOV     @4(SP),.MSG
                                        ADD     #2,4(SP)
                                .INST1: TYPE
                                .MSG:   0
                                        MOV     #INBUF,R4
                                        MOV     #7,R3
                                1$:     TSTB    @TKCSR
                                        BPL     1$
                                        MOVB    @TKDBR,(R4)
                                        BICB    #200,(R4)
                                        CMPB    (R4)+,#15
                                        BEQ     INSTR2
                                2$:     TSTB    @TPCSR
                                        BPL     2$
                                        MOV     @TKDBR,@TPDBR
                                        DEC     R3
                                        BNE     1$
                                        MOV     (SP)+,R4
                                        MOV     (SP)+,R3
                                .INSTE: TYPE    ,MQM
                                        MOV     R3,-(SP)
                                        MOV     R4,-(SP)
                                        BR      .INST1
                                INSTR2: MOV     (SP)+,R4        ;RESTORE R4
                                        MOV     (SP)+,R3        ;RESTORE R3
                                        RTI

                                        ;CONVERT ASCII STRING TO OCTAL
                                        ;----------------------------

                                .PARAM: MOV     R5,-(SP)
                                        MOV     R4,-(SP)
                                        MOV     4(SP),R5
                                        MOV     (R5)+,LOLIM
                                        MOV     (R5)+,HILIM
                                        MOV     (R5)+,DEVADR
                                        MOVB    (R5)+,LOBITS
                                        MOVB    (R5)+,ADRCNT
                                        MOV     R5,4(SP)
                                PARAM1: CLR     R5
                                        MOV     #INBUF,R4
                                        CMPB    #15,(R4)
                                        BEQ     PARERR
                                1$:     CMPB    (R4),#60
                                        BLT     PARERR
                                        CMPB    (R4),#67
                                        BGT     PARERR
                                        BICB    #60,(R4)
                                        BISB    (R4)+,R5
                                        CMPB    #15,(R4)
                                        BEQ     LIMITS
```

```
                        ASL     R5
                        ASL     R5
                        ASL     R5
                        BR      1$
PARERR:                 INSTER
                        BR      PARAM1

                        ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
                        ;--------------------------------------

LIMITS:                 CMP     R5,HILIM
                        BHI     PARERR
                        CMP     R5,LOLIM
                        BLO     PARERR
                        BITB    LOBITS,R5
                        BNE     PARERR

                        ;STORE NUMBER AT SPECIFIED ADDRESS

                        MOV     DEVADR,R4
1$:                     MOV     R5,(R4)+
                        ADD     #2,R5
                        DECB    ADRCNT
                        BNE     1$
                        MOV     (SP)+,R4
                        MOV     (SP)+,R5
                        RTI
LOLIM:                  0
HILIM:                  0
DEVADR:                 0
LOBITS:                 0
ADRCNT=LOBITS+1

                        ;SAVE PC OF TEST THAT FAILED AND R0-R5
                        ;--------------------------------------

.SAV05:                 MOV     4(SP),SAVPC     ;SAVE R7 (PC)

                        ;SAVE R0-R5

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

                        ;RESTORE R0-R5

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

```
                        RTI                             ;LEAVE

                        ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
                        ;-------------------------------------------------------

003536  104402  005104  .CONVR: TYPE    ,MCRLF
003542  010046          .CNVRT: MOV     R0,-(SP)
003544  010146                  MOV     R1,-(SP)
003546  010346                  MOV     R3,-(SP)
003550  010446                  MOV     R4,-(SP)
003552  010546                  MOV     R5,-(SP)
003554  017601  000012          MOV     @12(SP),R1
003560  062766  000002  000012  ADD     #2,12(SP)
003566  012137  003742          MOV     (R1)+,WRDCNT
003572  112137  003744  1$:     MOVB    (R1)+,CHRCNT
003576  112137  003745          MOVB    (R1)+,SPACNT
003602  013137  003746          MOV     @(R1)+,BINWRD
003606  013704  003746  2$:     MOV     BINWRD,R4
003612  113705  003744          MOVB    CHRCNT,R5
003616  012700  005562          MOV     #TEMP,R0
003622  010403          3$:     MOV     R4,R3
003624  042703  177770          BIC     #177770,R3
003630  062703  000060          ADD     #060,R3
003634  110320                  MOVB    R3,(R0)+
003636  000241                  CLC
003640  006004                  ROR     R4
003642  000241                  CLC
003644  006004                  ROR     R4
003646  000241                  CLC
003650  006004                  ROR     R4
003652  005305                  DEC     R5
003654  001362                  BNE     3$
003656  012703  005624          MOV     #MDATA,R3
003662  114023          4$:     MOVB    -(R0),(R3)+
003664  105337  003744          DECB    CHRCNT
003670  001374                  BNE     4$
003672  105737  003745          TSTB    SPACNT
003676  001405                  BEQ     6$
003700  112723  000040  5$:     MOVB    #040,(R3)+
003704  105337  003745          DECB    SPACNT
003710  001373                  BNE     5$
003712  105013          6$:     CLRB    (R3)
003714  104402  005624          TYPE    ,MDATA
003720  005337  003742          DEC     WRDCNT
003724  001322                  BNE     1$
003726  012605                  MOV     (SP)+,R5
003730  012604                  MOV     (SP)+,R4
003732  012603                  MOV     (SP)+,R3
003734  012601                  MOV     (SP)+,R1
003736  012600                  MOV     (SP)+,R0
003740  000002                  RTI
003742  000000  WRDCNT: 0
003744  000000  CHRCNT: 0
        003745  SPACNT=CHRCNT+1
003746  000000  BINWRD: 0
```

```
                                        ;TRAP DISPATCH SERVICE
                                        ;ARGUMENT OF TRAP IS EXTRACTED
                                        ;AND USED AS OFFSET TO OBTAIN POINTER
                                        ;TO SELECTED SUBROUTINE

003750  011646              .TRPSR: MOV     (SP),-(SP)          ;GET PC OF RETURN
003752  162716  000002              SUB     #2,(SP)             ;=PC OF TRAP
003756  017616  000000              MOV     @(SP),(SP)          ;GET TRP
003762  006316              TRPOK:  ASL     (SP)                ;MULTIPLY TRAP ARG BY 2
003764  042716  177001              BIC     #177001,(SP)        ;CLEAR UNWANTED BITS
003770  062716  001314              ADD     #.TRPTAB,(SP)       ;POINTER TO SUBROUTINE ADDRESS
003774  017616  000000              MOV     @(SP),(SP)          ;SUBROUTINE ADDRESS
004000  000136                      JMP     @(SP)+              ;GO TO SUBROUTINE

                                        ;ERROR HANDLER
                                        ;-------------

004002                      .HLT:
004002  022737  177570  001202      CMP     #177570,SWR     ;IS THERE A REAL SWR?
004010  001411              BEQ     64$             ;BR IF YES
004012  017746  175170              MOV     @TKDBR,-(SP)    ;SAVE KEYBOARD CHAR
004016  042716  000200              BIC     #BIT7,(SP)      ;CLEAR PARITY BIT
004022  122726  000007              CMPB    #7,(SP)+        ;WAS IT CNTRL 'G' ?
004026  001002              BNE     .+6             ;BR IF NO.
004030  004737  004640              JSR     PC,SERV.G       ;SERVICE "CNTRL 'G'".
004034  032777  010000  175140  64$:BIT     #SW12,@SWR      ;BELL ON ERROR?
004042  001406              BEQ     XBX             ;BR IF NO BELL
004044  105777  175140              TSTB    @TPCSR          ;TTY READY.
004050  100003              BPL     XBX             ;DON'T WAIT IF TTY NOT READY.
004052  112777  000207  175132      MOVB    #207,@TPDBR     ;PUSH A BELL AT THE TTY.
004060  032777  020000  175114  XBX:BIT     #SW13,@SWR      ;DELETE ERROR PRINT OUT?
004066  001105              BNE     HALTS           ;BR IF NO PRINT OUT WANTED.
004070  021637  001234              CMP     (SP),LSTERR     ;WAS THIS ERROR FOUND LAST TIME?
004074  001404              BEQ     1$              ;BR IF YES
004076  011637  001234              MOV     (SP),LSTERR     ;RECORD BEING HERE
004102  105037  001311              CLRB    ERRFLG          ;PREPARE HEADER
004106  104406              1$:     SAV05                   ;SAVE ALL PROC REGISTERS
004110  011605                      MOV     (SP),R5         ;GET THE PC OF ERROR
004112  162705  000002              SUB     #2,R5           ;GET ADDRESS OF TRAP CALL
004116  011504                      MOV     (R5),R4         ;GET HLT INSTRUCTION
004120  006304                      ASL     R4              ;MULT BY TWO
004122  061504                      ADD     (R5),R4         ;DOUBLE IT
004124  006304                      ASL     R4              ;MULT AGAIN
004126  042704  177001              BIC     #177001,R4      ;CLEAR JUNK
004132  062704  024230              ADD     #.ERRTAB,R4     ;GET POINTER
004136  012437  004252              MOV     (R4)+,ERRMSG    ;GET ERROR MESSAGE
004142  012437  004264              MOV     (R4)+,DATAHD    ;GET DATA HEADRER
004146  011437  004276              MOV     (R4),DATABP     ;GET DATA TABLE
004152  105737  001311              TSTB    ERRFLG          ;TYPE HEADREER
004156  001403              BEQ     TYPMSG          ;BR IF YES
004160  005737  004276              TST     DATABP          ;DOES DATA TABLE EXIST?
004164  001040              BNE     TYPDAT          ;BR IF YES.
004166  104402  005104      TYPMSG: TYPE    ,MCRLF
004172  104402  005104              TYPE    ,MCRLF
004176  005737  001220              TST     LOCK
```

```
                        BEQ     1$
                        TYPE    ,MASTEK
                1$:     TYPE    ,MTSTN
                        CNVRT   ,XTSTN          ;SHOW IT
                        TYPE    ,MERRPC         ;TYPE PC.
                        CNVRT   ,ERTAB0         ;SHOW IT
                        TYPE    ,MCRLF          ;GIVE A CR/LF
                        MOVB    #-1,ERRFLG      ;NO MORE HEADER UNLESS NO DATA TABLE.
                        TST     ERRMSG          ;IS THERE AN ERROR MESSAGE?
                        BEQ     WRK0.FM         ;BR IF NO.
                        TYPE                    ;TYPE
                ERRMSG: 0                       ;    ERROR MESSAGE
                WRK0.FM:                        ;
                        TST     DATAHD          ;DATA HEADER?
                        BEQ     TYPDAT          ;BR IF NO
                        TYPE                    ;TYPE
                DATAHD: 0                       ;    DATA HEADER
                TYPDAT: TST     DATABP          ;DATA TABLE?
                        BEQ     RESREG          ;BR IF NO.
                        CONVRT                  ;SHOW
                DATABP: 0                       ;    DATA TABLE
                RESREG: RES05                   ;RESTORE PROC REGISTERS
                HALTS:  TST     @SWR            ;HALT ON ERROR?
                        BPL     EXITER          ;BR IF NO HALT ON ERROR
                        PUSHR0                  ;SAVE R0
                        MOV     2(SP),R0        ;SHOW ERROR PC IN DATA LIGHTS
                        HALT                    ;HALT
                        POPR0                   ;GET R0
                EXITER: INC     ERRCNT          ;UPDATE ERROR COUNT
                        BIT     #SW08,@SWR      ;GOTO TOP OF TEST?
                        BNE     1$              ;BR IF YES
                        BIT     #SW10,@SWR      ;GOTO NEXT TEST?
                        BEQ     2$              ;BR IF NO
                        MOV     NEXT,RETURN     ;SET FOR NEXT TEST
                1$:     MOV     #STACK,SP       ;RESET SP
                        JMP     @RETURN         ;GOTO SPECIFIED TEST
                2$:     RTI                     ;RETURN
                ERTAB0: 1
                        .BYTE   6,2
                        SAVPC
                XTSTN:  1
                        .BYTE   3,2
                        TSTNO
                        ;ENTER HERE ON POWER FAILURE
                        ;--------------------------


                .PFAIL:
                        MOV     #RESTART,24             ;SET UP FOR POWER UP TRAP
                        HALT                            ;HALT ON POWER DOWN NORMAL
                        BR      .

                        ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED

                RESTAR:
                        MOV     #.PFAIL,24      ;SET UP FOR POWER FAILURE
```

```
                                        MOV     #STACK,SP       ;RESET THE STACK POINTER
                                        CLR     TEMP            ;READY FOR TIMMER
                                        INC     TEMP            ;PLUS ONE TO THE TIMER!
                                        BNE     .-4             ;BR IF MORE TO GO
                                        TYPE    ,MPFAIL         ;TYPE THE MESSAGE
                                        CNVRT   ,PFTAB          ;TELL WHAT TEST TO RETURN TO.
                                        CLRB    ERRFLG          ;START CLEAN
                                        CLR     LSTERR          ;'''''''''''''
                                        MSTCLR                  ;START CLEAN UP OF DEVICE
                                        RAMCLR                  ;CLEAR IT ALL!
                                        JMP     @RETURN         ;START DOING THAT TEST AGAIN.
                                PFTAB:  1
                                .BYTE   3,2
                                        TSTNO
                                .DELAY: MOV     R0,-(SP)
                                        MOV     1$,R0
                                        DEC     R0
                                        BNE     .-2
                                        MOV     (SP)+,R0
                                        RTI
                                1$:     30.

                                .RAMCLR:
                                        MOV     #MRESET,@DVSCR  ;ISSUE A MASTER CLEAR
                                        MOV     R1,-(SP)        ;SAVE R1 ON THE STACK
                                        MOV     R4,-(SP)        ;SAVE R4 ON THE STACK
                                        MOV     DVSRS,R1        ;GET SECONDARY SEL. REG.
                                        MOV     DVSRA,R4        ;GET SECONDARY REGISTER ACCESS REG.
                                1$:     CLR     (R4)            ;ZERO THE SECONDARY REGISTER.
                                        ADD     #^C<BIT11+BIT10+BIT9+BIT8+BIT3+BIT2+BIT1+BIT0>+BIT0,(R1)
                                        BNE     1$
                                        MOV     (SP)+,R4        ;RESTORE R4
                                        MOV     (SP)+,R1        ;RESTORE R1
                                        RTI

                                .MSTCLR:
                                        MOV     #MRESET,@DVSCR  ;ISSUE MASTER CLEAR.
                                        RTI

                                .ROMCLK:
                                        BIS     #BIT1,@DVSCR
                                        RTI

                                .DATACLK:
                                        MOV     R0,-(SP)
                                        CLR     R0
                                        BIS     #BIT8,@DVLCR
                                1$:     MOV     @DVLCR,3$
                                        RORB    3$+1
                                        BCC     2$
                                        INC     R0
                                        BNE     1$
                                        HLT     0
                                2$:     MOV     (SP)+,R0
                                        RTI
                                3$:     .BLKW 1
```

```
913
914  004640  032777  004000  174336  SERV.G: BIT     #4000,@TKCSR    ;RX BUSY?
915  004646  001374                          BNE     SERV.G          ;BR IF YES
916  004650  017737  174326  005072          MOV     @SWR,90$        ;SAVE (SWR).
917  004656  013777  005072  174316  1$:     MOV     90$,@SWR        ;
918  004664  104402  005052                  TYPE    ,89$            ;
919  004670  104411  005064                  CNVRT   ,88$            ;
920  004674  104402  005074                  TYPE    ,91$            ;
921  004700  105777  174300                  TSTB    @TKCSR          ;WAIT FOR DONE.
922  004704  100375                          BPL     .-4
923  004706  017746  174274                  MOV     @TKDBR,-(SP)    ;
924  004712  042716  000200                  BIC     #BIT7,(SP)      ;
925  004716  122726  000015                  CMPB    #15,(SP)+       ;
926  004722  001450                          BEQ     5$              ;
927  004724  005077  174252                  CLR     @SWR            ;
928  004730  105777  174254          2$:     TSTB    @TPCSR          ;
929  004734  100375                          BPL     .-4
930  004736  016677  177776  174246          MOV     -2(SP),@TPDBR   ;
931  004744  000241                          CLC                     ;
932  004746  006177  174230                  ROL     @SWR            ;
933  004752  006177  174224                  ROL     @SWR            ;
934  004756  006177  174220                  ROL     @SWR            ;
935  004762  103735                          BCS     1$              ;ERROR
936  004764  026627  177776  000060          CMP     -2(SP),#60      ;
937  004772  002731                          BLT     1$              ;
938  004774  026627  177776  000067          CMP     -2(SP),#67      ;
939  005002  003325                          BGT     1$              ;
940  005004  042766  177770  177776          BIC     #^C<7>,-2(SP)   ;
941  005012  056677  177776  174162          BIS     -2(SP),@SWR     ;
942  005020  105777  174160                  TSTB    @TKCSR          ;
943  005024  100375                          BPL     .-4             ;
944  005026  017746  174154                  MOV     @TKDBR,-(SP)
945  005032  042716  000200                  BIC     #BIT7,(SP)
946  005036  122726  000015                  CMPB    #15,(SP)+
947  005042  001332                          BNE     2$              ;
948  005044  104402  005104          5$:     TYPE    ,MCRLF          ;
949  005050  000207                          RTS     PC
950
951  005052  020377  051450  051127  89$:     .ASCIZ  <377>? (SWR)=/?
952  005060  036451  000057
953                                  .EVEN
954  005064  000001                  88$:     1
955  005066     006     000                   .BYTE   6,0
956  005070  005072                          90$
957  005072  000000                  90$:     .WORD   0
958  005074  036457  000057          91$:     .ASCIZ  ?/=/?
959                                  .EVEN
960  005100  020040  000077          MQM:     .ASCIZ  /  ?/
(2)  005104  005015     000          MCRLF:   .ASCIZ   <15><12>
(2)  005107     377  053520  020122  MPFAIL:  .ASCIZ  <377>/PWR FAILED. RESTART AT TEST /
(2)  005145     377  047105  020104  MEPASS:  .ASCIZ  <377>/END PASS CZDVBC0  /
(2)  005171     377  000122          MR:      .ASCIZ  <377>/R/
(2)  005174  050377  047522  051107  MERR2:   .ASCIZ  <377>/PROGRAM INDICATES NO DEVICES PRESENT./
(2)  005243     377  047111  052523  MERR3:   .ASCIZ  <377>/INSUFFICIENT DATA!/
(2)  005267     377  042524  052123  MTSTPC:  .ASCIZ  <377>/TEST PC-/
(2)  005301     377  047514  045503  MLOCK:   .ASCIZ  <377>/LOCK ON SELECTED TEST/
```

```
(2)  005330  051503  035122  000040  MCSRX:  .ASCIZ   /CSR: /
(2)  005336  042526  035103  000040  MVECX:  .ASCIZ  /VEC: /
(2)  005344  040520  051523  051505  MPASSX: .ASCIZ  /PASSES: /
(2)  005355     105  051122  051117  MERRX:  .ASCIZ  /ERRORS: /
(2)  005366  042524  052123  047040  MTSTN:  .ASCIZ  /TEST NO: /
(2)  005400  000052                  MASTEK: .ASCIZ  /*/
(2)  005402  051777  052105  051440  MNEW:   .ASCIZ  <377>/SET SWITCH REG TO DV11'S DESIRED ACTIVE./
(2)  005454  041520  020072     000  MERRPC: .ASCIZ  /PC: /
(2)  005461     377  040515  020120  XHEAD:  .ASCIZ  <377>/MAP OF DV11 STATUS/<377>
(2)                                  .EVEN
(2)  005506  000002                  XSTATQ: 2
961  005510     006     003                  .BYTE   6,3
962  005512  001246                          TEMP1
963  005514     006     002                  .BYTE   6,2
964  005516  001250                          TEMP2
965                                  .EVEN
966
967                                          ;BUFFERS FOR INPUT-OUTPUT
968
969  005520  000000                  INBUF:  0
970          005562                  .=.+40
971  005562  000000                  TEMP:   0
972          005624                  .=.+40
973  005624  000000                  MDATA:  0
974          005666                  .=.+40
```

```
                                        ;
                                        ;ROUTINE USED TO ''CYCLE'' THROUGH UP TO EIGHT DV11'S
                                        ;THIS ROUTINE SETS UP THE CONTROL ADDRESS FOR THE DIAGNOSTIC
                                        ;AND RUNS THE SPECIFIED DV11'S.   THIS ROUTINE *MUST*
                                        ;BE RUN FIRST BEFORE ENTERING THE DIAGNOSTIC FOR THE
                                        ;SETUP NECESSARY.
                                        ;
005666  105737  001300        CYCLE:  TSTB    DVACTV          ;ARE ANY DV11'S TO BE TESTED?
005672  001004                        BNE     1$              ;BR IF OK.
005674  104402  005174                TYPE    ,MERR2          ;NO DV11'S SELECTED!!
005700  000000                        HALT                    ;STOP THE SHOW.
005702  000776                        BR      .-2             ;DISQUALIFY CONT. SW.
005704  133737  001304  001300  1$:   BITB    RUN,DVACTV      ;IS THIS ONE ''ACTIVE''
005712  001020                        BNE     2$              ;BR IF GOOD ONE FOUND.
005714  000241                        CLC                     ;CLEAR PROC. CARRY BIT.
005716  106137  001304                ROLB    RUN             ;UPDATE POINTER
005722  105537  001304                ADCB    RUN             ;CATCH CARRY FROM RUN
005726  062737  000024  001306        ADD     #24,CREAM       ;UPDATE ADDRESS POINTER.
005734  022737  001740  001306        CMP     #DV.END,CREAM
005742  001360                        BNE     1$              ;KEEP GOING; NOT ALL TESTED FOR.
005744  012737  001500  001306        MOV     #DV.MAP,CREAM   ;RESET ADDRESS POINTER.
005752  000754                        BR      1$              ;KEEP LOOKING FOR ACTIVE DV11
005754  000241                2$:     CLC                     ;CLEAR PROC. CARRY.
005756  106137  001304                ROLB    RUN             ;UPDATE POINTER.
005762  105537  001304                ADCB    RUN             ;CATCH CARRY.
005766  013700  001306                MOV     CREAM,R0        ;GET ADDRESS POINTER.
005772  062737  000024  001306        ADD     #24,CREAM       ;UPDATE.
006000  022737  001740  001306        CMP     #DV.END,CREAM
                                                            ;ALL DONE?
006006  001003                        BNE     3$              ;BR IF NO.
006010  012737  001500  001306        MOV     #DV.MAP,CREAM   ;RESTORE POINTER.
006016  012037  001362        3$:     MOV     (R0)+,DVSCR     ;LOAD SYSTEM CTRL. REG
006022  012037  001352                MOV     (R0)+,DVRVEC    ;LOAD VECTOR
006026  012037  001422                MOV     (R0)+,L00.03    ;GET LINE PARAMETERS. 00-03
006032  012037  001432                MOV     (R0)+,SYNC2A    ;
006036  012037  001424                MOV     (R0)+,L04.07    ;                     04-07
006042  012037  001434                MOV     (R0)+,SYNC2B    ;
006046  012037  001426                MOV     (R0)+,L08.11    ;                     08-11
006052  012037  001436                MOV     (R0)+,SYNC2C    ;
006056  012037  001430                MOV     (R0)+,L12.15    ;                     12-15
006062  012037  001440                MOV     (R0)+,SYNC2D    ;
006066  012700  000002                MOV     #2,R0           ;SAVE CORE THIS WAY!
006072  013737  001362  001364        MOV     DVSCR,DVSCRH    ;GET SYS CTRL. REG HIGH BYTE.
006100  005237  001364                INC     DVSCRH          ;GOT IT.
006104  013737  001364  001366        MOV     DVSCRH,DVRIC    ;GET NXT REC. CHAR REG.
006112  005237  001366                INC     DVRIC           ;GOT IT
006116  013737  001366  001370        MOV     DVRIC,DVLCR     ;GET LN. PAR.REG.
006124  060037  001370                ADD     R0,DVLCR        ;GOT IT
006130  013737  001370  001372        MOV     DVLCR,DVSRS     ;GET SEC. REG. SEL. REG.
006136  060037  001372                ADD     R0,DVSRS        ;GOT IT
006142  013737  001372  001374        MOV     DVSRS,DVSRSH    ;GET HIGH BYTE.
006150  005237  001374                INC     DVSRSH          ;GOT IT
006154  013737  001374  001376        MOV     DVSRSH,DVSRA    ;SEC. REG. ACCESS.
006162  005237  001376                INC     DVSRA           ;GOT IT
```

```
1031  006166  013737  001376  001400          MOV     DVSRA,DVSFR     ;SPEC. FUN. REG.
1032  006174  060037  001400                  ADD     R0,DVSFR        ;
1033  006200  013737  001400  001402          MOV     DVSFR,DVNSR     ;NPR STAT. REG.
1034  006206  060037  001402                  ADD     R0,DVNSR        ;
1035  006212  013737  001402  001404          MOV     DVNSR,RESV16    ;RESERVED REG
1036  006220  060037  001404                  ADD     R0,RESV16       ;
1037
1038  006224  013737  001352  001354          MOV     DVRVEC,DVRLVL   ;PTY LVL
1039  006232  060037  001354                  ADD     R0,DVRLVL       ;
1040  006236  013737  001354  001356          MOV     DVRLVL,DVTVEC   ;TX VEC
1041  006244  060037  001356                  ADD     R0,DVTVEC       ;
1042  006250  013737  001356  001360          MOV     DVTVEC,DVTLVL   ;TX LVL
1043  006256  060037  001360                  ADD     R0,DVTLVL
1044
1045  006262  012700  001422                  MOV     #L00.03,R0      ;LOAD STAUS 00-03
1046  006266  012701  001406                  MOV     #MASK.A,R1      ;PREPARE MASK.
1047  006272  012702  001416                  MOV     #CLK.A,R2       ;PREPARE CLOCKS
1048  006276  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1049
1050  006302  012700  001424                  MOV     #L04.07,R0      ;LOAD STAUS 00-03
1051  006306  012701  001410                  MOV     #MASK.B,R1      ;PREPARE MASK.
1052  006312  012702  001417                  MOV     #CLK.B,R2       ;PREPARE CLOCKS
1053  006316  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1054
1055  006322  012700  001426                  MOV     #L08.11,R0      ;LOAD STAUS 00-03
1056  006326  012701  001412                  MOV     #MASK.C,R1      ;PREPARE MASK.
1057  006332  012702  001420                  MOV     #CLK.C,R2       ;PREPARE CLOCKS
1058  006336  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1059
1060  006342  012700  001430                  MOV     #L12.15,R0      ;LOAD STAUS 00-03
1061  006346  012701  001414                  MOV     #MASK.D,R1      ;PREPARE MASK.
1062  006352  012702  001421                  MOV     #CLK.D,R2       ;PREPARE CLOCKS
1063  006356  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1064  006362  032777  000002  172612          BIT     #SW01,@SWR
1065  006370  001445                          BEQ     7$
1066  006372                          4$:
1067  006372  005737  000042                  TST     @#42
1068  006376  001042                          BNE     7$
1069  006400  104402  005104                  TYPE    ,MCRLF
1070  006404  104403                          INSTR
1071  006406  005366                          MTSTN
1072  006410  104405                          PARAM
1073  006412  000001                          1
1074  006414  001000                          1000
1075  006416  001226                          TSTNO
1076  006420     000                  .BYTE   0
1077  006421     001                  .BYTE   1
1078  006422  012700  007260                  MOV     #TST1,R0
1079  006426  022710                  5$:     CMP     (PC)+,(R0)
1080  006430  012737                          MOV     (PC)+,@(PC)+
1081  006432  001015                          BNE     6$
1082  006434  023760  001226  000002          CMP     TSTNO,2(R0)
1083  006442  001011                          BNE     6$
1084  006444  022760  001226  000004          CMP     #TSTNO,4(R0)
1085  006452  001005                          BNE     6$
1086  006454  010037  001214                  MOV     R0;RETURN
```

```
 1087  006460  104402  005104          TYPE    ,MCRLF
 1088  006464  000412                  BR      8$
 1089  006466  005720          6$:     TST     (R0)+
 1090  006470  020027  022760          CMP     R0,#TLAST+10
 1091  006474  001354                  BNE     5$
 1092  006476  104402  005100          TYPE    ,MQM
 1093  006502  000733                  BR      4$
 1094  006504  012737  007260  001214  7$:     MOV     #TST1,RETURN    ;PREPARE RETURN ADDRESS
 1095  006512  000177  172476          8$:     JMP     @RETURN         ;GO START TESTING.
 1096
 1097  006516  011003          FIX.00: MOV     (R0),R3         ;GET PARAMETERS.
 1098  006520  042703  176377          BIC     #^C<1400>,R3    ;CLEAR JUNK.
 1099  006524  005703                  TST     R3              ;TEST FOR EIGHT BITS.
 1100  006526  001005                  BNE     1$              ;BR IF NOT 8 BITS.
 1101  006530  012711  000400          MOV     #400,(R1)               ;SET FOR 8 BITS PER CHAR
 1102  006534  112712  000010          MOVB    #8.,(R2)        ;
 1103  006540  000424                  BR      4$
 1104  006542  022703  000400  1$:     CMP     #400,R3         ;CHECK FOR SEVEN BITS.
 1105  006546  001005                  BNE     2$              ;BR IF NOT 7 BITS.
 1106  006550  112711  000200          MOVB    #200,(R1)       ;
 1107  006554  112712  000007          MOVB    #7,(R2)         ;
 1108  006560  000414                  BR      4$
 1109  006562  022703  001000  2$:     CMP     #1000,R3        ;CHECK FOR SIX BITS.
 1110  006566  001005                  BNE     3$              ;BR IF NOT SIX BITS.
 1111  006570  112711  000300          MOVB    #300,(R1)       ;
 1112  006574  112712  000006          MOVB    #6,(R2)         ;
 1113  006600  000404                  BR      4$
 1114  006602  112711  000340  3$:     MOVB    #340,(R1)       ;IF NONE OF THE ABOVE; MUST BE 5 BITS.
 1115  006606  112712  000005          MOVB    #5,(R2)         ;
 1116  006612  032710  040000  4$:     BIT     #PARBIT,(R0)    ;PARITY ENABLED?
 1117  006616  001401                  BEQ     5$              ;IF =0; THEN NO PARITY.
 1118  006620  105212                  INCB    (R2)            ;PLUS ONE TO THE CLOCK!
 1119  006622  000207          5$:     RTS     PC              ;
 1120
 1121                                  ;*ROUTINE USED TO "AUTO SIZE" THE DV11
 1122                                  ;*CSR AND VECTOR.
 1123                                  ;*NOTE: THE CSR MAY BE ANY WHERE IN THE FLOATING
 1124                                  ;*      ADDRESS RANGE (175000:175400)
 1125                                  ;*      AND THE VECTOR MAY BE ANY WHERE IN THE
 1126                                  ;*      FLOATING VECTOR RANGE (300:770)
 1127                                  ;*
 1128
 1129  006624          AUTO.SIZE:
 1130  006624  000005                  RESET                   ;INSURE A BUS INIT.
 1131  006626  012702  001500  CSRMAP: MOV     #DV.MAP,R2      ;LOAD MAP POINTER.
 1132  006632  005022          1$:     CLR     (R2)+           ;ZERO ENTIRE MAP
 1133  006634  022702  001740          CMP     #DV.END,R2      ;ALL DONE?
 1134  006640  001374                  BNE     1$              ;BR IF NO
 1135  006642  105037  001301          CLRB    DVNUM           ;SET OCTAL NUMBER OF DV11'S TO 0
 1136  006646  012702  001500          MOV     #DV.MAP,R2
 1137  006652  012701  175000          MOV     #175000,R1      ;SET FOR FIRST ADDRESS TO BE TESTED
 1138  006656  012737  007076  000004  MOV     #6$,@#4         ;SET FOR NON-EXISTANT DEVICE TIME OUT
 1139  006664  005711          2$:     TST     (R1)            ;IF DV11 DVSCR S/B 0
 1140  006666  001037                  BNE     3$              ;IF NO DEV ; TRAP TO 4. IF NO BIT 8 THEN NO DV11
 1141  006670  022761  177777  000012  CMP     #177777,12(R1)  ;IF DV11 THEN DVSFR S/B ALL 1'S ON INIT!
 1142  006676  001033                  BNE     3$              ;BR IF NOT DV11
```

```
1143  006700  005761  000016            TST     16(R1)          ;IF DV11 THEN RESV16 S/B ALL 0'S
1144  006704  001030                    BNE     3$              ;BR IF NOT DV11
1145                              ;AT THIS POINT IT IS ASSUMED THAT R1 HOLDS A DV11 CSR ADDRESS.
1146  006706  010122                    MOV     R1,(R2)+        ;STORE CSR IN CORE TABLE.
1147  006710  005722                    TST     (R2)+           ;POP OVER VECTOR STORE AREA
1148  006712  052722  000226            BIS     #226,(R2)+      ;SET LINE CARD 1 STAT AND SYNC
1149  006716  052722  000062            BIS     #62,(R2)+       ;
1150  006722  052722  000226            BIS     #226,(R2)+      ;SET LINE CARD 2 STAT AND SYNC
1151  006726  052722  000062            BIS     #62,(R2)+       ;
1152  006732  052722  000226            BIS     #226,(R2)+      ;SET LINE CARD 3 STAT AND SYNC
1153  006736  052722  000062            BIS     #62,(R2)+       ;
1154  006742  052722  000226            BIS     #226,(R2)+      ;SET LINE CARD 4 STAT AND SYNC
1155  006746  052722  000062            BIS     #62,(R2)+       ;
1156  006752  105237  001301            INCB    DVNUM           ;UPDATE DEVICE COUNTER
1157  006756  122737  000010  001301    CMPB    #10,DVNUM       ;ARE MAX. NO. OF DEV FOUND?
1158  006764  001405                    BEQ     100$            ;YES DON'T LOOK FOR ANY MORE.
1159  006766  062701  000010    3$:     ADD     #10,R1          ;UPDATE CSR POINTER ADDRESS
1160  006772  022701  175400            CMP     #175400,R1
1161  006776  001332                    BNE     2$              ;BR IF MORE ADDRESS TO CHECK.
1162  007000  012722  177777  100$:     MOV     #177777,(R2)+   ;TERMINATER.
1163  007004  105037  001300            CLRB    DVACTV
1164  007010  105737  001301            TSTB    DVNUM           ;WERE ANY DV11'S FOUND AT ALL?
1165  007014  001423                    BEQ     5$              ;ERROR AUTO SIZER FOUND NO DV11'S IN THIS SYS.
1166  007016  113701  001301            MOVB    DVNUM,R1
1167  007022  110137  001303            MOVB    R1,SAVNUM       ;SAVE NUMBER OF DEVICES
1168  007026  000241            4$:     CLC
1169  007030  106137  001300            ROLB    DVACTV          ;GENERATE ACTIVE REGISTER OF DEVICES.
1170  007034  105237  001300            INCB    DVACTV          ;SET THE BIT
1171  007040  005301                    DEC     R1
1172  007042  001371                    BNE     4$              ;BR IF MORE TO GENERATE
1173  007044  012737  000006  000004    MOV     #6,@#4          ;RESTORE TRAP VECTOR
1174  007052  113737  001300  001302    MOVB    DVACTV,SAVACT   ;SAVE ACTIVE REGISTER
1175  007060  000137  007104            JMP     VECMAP          ;GO FIND THE VECTOR NOW.
1176  007064  104402  005174    5$:     TYPE    ,MERR2          ;NOTIFY OPR THAT NO DV11'S FOUND.
1177  007070  005000                    CLR     R0              ;MAKE DATA LIGHTS ZERO
1178  007072  000000                    HALT                    ;STOP THE SHOW
1179  007074  000776                    BR      .-2             ;DISABLE CONT. SW.
1180  007076  012716  006766    6$:     MOV     #3$,(SP)        ;ENTERED BY NON-EXISTANT TIME-OUT.
1181  007102  000002                    RTI                     ;RETURN TO MAINSTREAM
1182
1183  007104  012737  000340  000022  VECMAP: MOV   #340,@#22   ;SET IOT TRAP PRIO TO 7
1184  007112  012737  007234  000020          MOV   #4$,@#20    ;SET IOT TRAP VECTOR
1185  007120  012702  001500                  MOV   #DV.MAP,R2  ;SET SOFTWARE POINTER
1186  007124  012700  000300                  MOV   #300,R0     ;FLOATING VECTORS START HERE.
1187  007130  012701  000302                  MOV   #302,R1     ;PC OF IOT INSTR.
1188  007134  010120                  1$:     MOV   R1,(R0)+    ;START FILLING VECTOR AREA
1189  007136  012721  000004                  MOV   #4,(R1)+    ;WITH .+2; IOT
1190  007142  022021                          CMP   (R0)+,(R1)+ ;ADD 2 TO R0 +R1
1191  007144  020127  001000                  CMP   R1,#1000
1192  007150  101771                          BLOS  1$          ;BR IF MORE TO FILL
1193  007152  113737  001300  001246          MOVB  DVACTV,TEMP1 ;STORE TEMPORALLY
1194  007160  006037  001246          2$:     ROR   TEMP1       ;BRING OUT A BIT
1195  007164  103034                          BCC   5$          ;BR IF ALL DONE
1196  007166  005037  177776                  CLR   PS          ;ZERO CPU PRIO
1197  007172  012772  001300  000000          MOV   #BIT9+BIT7+BIT6,@(R2)
1198  007200  005000                          CLR   R0          ;ATTEMPT TO FORCE AN INTERUPT
```

```
1199  007202  005200                        INC     R0              ;STALL
1200  007204  001376                        BNE     .-2             ;       FOR TIME TO INTERUPT
1201  007206  052762  000300  000002        BIS     #300,2(R2)      ;NO INTERUPT ASSUME 300 AND FIX DV11 LATER
1202  007214  042772  176777  000000  3$:   BIC     #^C<BIT9>,@(R2)
1203  007222  005072  000000                CLR     @(R2)
1204  007226  062702  000024                ADD     #24,R2          ;POP SOFTWARE POINTER
1205  007232  000752                        BR      2$              ;KEEP GOING
1206  007234  051662  000002          4$:   BIS     (SP),2(R2)      ;GET VECTOR ADDRESS
1207  007240  042762  000007  000002        BIC     #7,2(R2)        ;CLEAR JUNK
1208  007246  022626                        CMP     (SP)+,(SP)+     ;POP IOT JUNK OFF STACK
1209  007250  012716  007214                MOV     #3$,(SP)        ;SET FOR RETURN
1210  007254  000002                        RTI
1211  007256  000207                  5$:   RTS     PC              ;ALL DONE WITH ''AUTO SIZING''
1212
```

```
1213
1214                                    ;************************ TEST 1 *******************************
1215                                    ;*TEST THAT ''TRANSMITTER FLAG WAITING''
1216                                    ;*IS TRUE AND THAT 'RECV FLAG WAITING'' IS
1217                                    ;*FALSE  AFTER AN INIT.
1218                                    ;*THIS TEST WILL BE DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
1219                                    ;:*************************************************************************
1220
1221                                    ;  TEST 1
1222                                    ;-------------
1223  007260  012737  000001  001226  TST1:   MOV     #1,TSTNO
1224  007266  012737  007542  001216          MOV     #TST2,NEXT
1225  007274  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
1226  007300  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
1227  007306  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
1228  007310  004737  007376                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
1229  007314  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
1230  007320  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
1231  007326  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
1232  007330  004737  007376                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
1233  007334  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
1234  007340  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
1235  007346  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
1236  007350  004737  007376                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
1237  007354  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
1238  007360  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
1239  007366  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
1240  007370  004737  007376                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
1241  007374  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
1242  007376                          105$:                           ;TEST ENTRANCE.
1243  007376  104413                          RAMCLR                  ;CLEAR ALL DV11 SEC. REGS.
1244  007400  010037  007412                  MOV     R0,65$          ;STORE LINE NO. POINTER.
1245  007404  005001                          CLR     R1              ;ZERO MSCANNER POINTER
1246  007406  004537  023544          1$:     PERFORM ,SETSCAN        ;POSITION SCANNER TO LINE NUMBER.
1247  007412  000001                  65$:    .BLKW 1                 ;INITAL LINE NUMBER HERE.
1248  007414  012703  000004          2$:     MOV     #4,R3           ;SET TO DO 4 LINES AT A TIME
1249  007420  012705  000003          3$:     MOV     #BIT1+BIT0,R5   ;SET EXPECTED RESULTS IN R5
1250  007424  012702  002000                  MOV     #BIT10,R2       ;BR-A 'RX FLAG WAITING'?
1251  007430  010277  171744                  MOV     R2,@DVSFR       ;LOAD DV11 INSTRUCTION
1252  007434  017704  171730                  MOV     @DVLCR,R4       ;READ BR TEST POINTS
1253  007440  020504                          CMP     R5,R4           ;TEST POINTS OK?
1254  007442  001401                          BEQ     4$              ;BR IF YES
1255  007444  104001                          HLT     1               ;EXPECT DVLCR BIT1+BIT0=1
1256  007446  012777  050102  171724  4$:     MOV     #S.C+BIT6+BIT1,@DVSFR
1257  007454  104415                          ROMCLK                  ;S/C ''ADVANCE SCANNER''
1258  007456  005201                          INC     R1              ;UPDATE MSCAN POINTER
1259  007460  010100                          MOV     R1,R0           ;PREPARE TO SET LINE POINTER
1260  007462  000241                          CLC                     ;TO CORRECT POSITION
1261  007464  006000                          ROR     R0              ;
1262  007466  012702  001000                  MOV     #BIT9,R2        ;BR-A ''TX FLAG WAITING''?
1263  007472  010277  171702                  MOV     R2,@DVSFR       ;LOAD DV11 INSTRUCTION
1264  007476  017704  171666                  MOV     @DVLCR,R4       ;READ BR TEST POINT
1265  007502  012705  000002                  MOV     #BIT1,R5        ;SET EXPECTED RESULTS
1266  007506  020504                          CMP     R5,R4           ;TX FLAG WAITING TRUE?
1267  007510  001401                          BEQ     5$              ;BR IF LCR BIT1=1 AND BIT0=0
1268  007512  104001                          HLT     1               ;ERROR.
```

```
1269  007514  012777  050102  171656  5$:     MOV     #S.C+BIT6+BIT1,@DVSFR
1270  007522  104415                          ROMCLK                  ;S/C ''ADVANCE SCANNER''
1271  007524  005201                          INC     R1              ;UPDATE MSCAN POINTER
1272  007526  010100                          MOV     R1,R0           ;UPDATE LINE POINTER
1273  007530  000241                          CLC                     ;
1274  007532  006000                          ROR     R0              ;
1275  007534  005303                          DEC     R3              ;ARE ALL 4 LINES TESTED?
1276  007536  001330                          BNE     3$              ;BR IF NO!
1277  007540  000207                          RTS     PC              ;CHECK NEXT SET OF LINES.
1278
1279
1280                                          ;************************ TEST 2 *********************************
1281                                          ;*TEST THAT 'MATCH DETECT'' IS
1282                                          ;*FALSE AFTER AN INIT.
1283                                          ;*THIS TEST WILL BE DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
1284                                          ;:*************************************************************************
1285
1286                                  ;  TEST 2
1287                                  ;-------------
1288  007542  012737  000002  001226  TST2:   MOV     #2,TSTNO
1289  007550  012737  007744  001216          MOV     #TST3,NEXT
1290  007556  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
1291  007562  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
1292  007570  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
1293  007572  004737  007660                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
1294  007576  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
1295  007602  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
1296  007610  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
1297  007612  004737  007660                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
1298  007616  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
1299  007622  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
1300  007630  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
1301  007632  004737  007660                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
1302  007636  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
1303  007642  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
1304  007650  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
1305  007652  004737  007660                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
1306  007656  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
1307  007660                          105$:                           ;TEST ENTRANCE.
1308  007660  010037  007674                  MOV     R0,65$          ;SET LINE POINTER
1309  007664  104412                          MSTCLR                  ;RESET THE DV11
1310  007666  005001                          CLR     R1              ;ZERO MSCANNER POINTER
1311  007670  004537  023544          1$:     PERFORM ,SETSCAN        ;SET MSCAN TO CORRECT LINE
1312  007674  000001                  65$:    .BLKW 1                 ;INITAL LINE POINTER PLACED HERE.
1313  007676  012703  000004          2$:     MOV     #4,R3           ;SET FOR A FOUR LINE GROUP.
1314  007702  012705  000003          3$:     MOV     #BIT1+BIT0,R5   ;SET EXPECTED RESULTS.
1315  007706  012702  076400          4$:     MOV     #BRB+BIT11+BIT10+BIT8,R2
1316  007712  010277  171462                  MOV     R2,@DVSFR       ;BR-B 'MATCH DET''?
1317  007716  017704  171446                  MOV     @DVLCR,R4       ;READ DVLCR INTO R4
1318  007722  020504                          CMP     R5,R4           ;MATCH DET FALSE??
1319  007724  001401                          BEQ     5$              ;BR IF YES
1320  007726  104001                          HLT     1               ;LCR BIT1=1+BIT0=1 EXPECTED.
1321  007730  004537  023544          5$:     PERFORM ,SETSCAN        ;UPDATE MSCAN POINTER TO NEXT LINE.
1322  007734  000001                          1                       ;1 LINE
1323  007736  005303                          DEC     R3              ;ALL FOUR LINES DONE YET?
1324  007740  001362                          BNE     4$              ;BR IF NO
```

```
 1325  007742  000207                          RTS     PC              ;CHECK NEXT SET OF LINES
 1326
 1327
 1328                                          ;************************ TEST 3 *******************************
 1329                                          ;*TEST THAT MAINT BIT WINDOW IS CLEARED
 1330                                          ;* AFTER AN INIT.
 1331                                          ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
 1332                                          ;:***************************************************************
 1333
 1334
 1335                                  ;  TEST 3
 1336                                  ;-------------
 1337  007744  012737  000003  001226  TST3:   MOV     #3,TSTNO
 1338  007752  012737  010152  001216          MOV     #TST4,NEXT
 1339  007760  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
 1340  007764  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 1341  007772  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 1342  007774  004737  010062                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 1343  010000  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
 1344  010004  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 1345  010012  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
 1346  010014  004737  010062                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
 1347  010020  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
 1348  010024  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 1349  010032  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
 1350  010034  004737  010062                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
 1351  010040  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
 1352  010044  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 1353  010052  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 1354  010054  004737  010062                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 1355  010060  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
 1356  010062                          105$:                           ;TEST ENTRANCE.
 1357  010062  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
 1358  010070  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
 1359  010072  000207                          RTS     PC              ;EXIT TEST
 1360  010074  104412                          MSTCLR                  ;RESET DV11
 1361  010076  005002                          CLR     R2              ;ZERO SFR IMAGE
 1362  010100  017705  171264                  MOV     @DVLCR,R5       ;READ THE DVLCR INTO R5
 1363  010104  042705  000200                  BIC     #BIT7,R5        ;CLEAR MAINT BIT WINDOW EXPECTED
 1364  010110  012703  000004                  MOV     #4,R3           ;SET TO DO 4 LINES.
 1365  010114  010077  171252          1$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
 1366  010120  017704  171244                  MOV     @DVLCR,R4       ;READ DVLCR RESULTS INTO R4
 1367  010124  042705  000060                  BIC     #BIT5+BIT4,R5   ;CLEAR EXTENDED ADDRESS BITS
 1368  010130  042704  000060                  BIC     #BIT5+BIT4,R4   ;""
 1369  010134  020504                          CMP     R5,R4           ;OK?
 1370  010136  001401                          BEQ     2$              ;
 1371  010140  104001                          HLT     1               ;BIT7 INCORRECT
 1372  010142  005200                  2$:     INC     R0              ;UPDATE LINE POINTER
 1373  010144  005303                          DEC     R3              ;ALL LINES DONE?
 1374  010146  001362                          BNE     1$              ;BR IF NO
 1375  010150  000207                          RTS     PC              ;RETURN FOR NEXT SET OF LINES.
 1376
 1377
 1378                                          ;************************ TEST 4 *******************************
 1379                                          ;*TEST THAT THE BIT WINDOW WILL
 1380                                          ;*STAY CLEARED WHEN MAINT INTERNAL
```

```
                                        ;*MODE IS SELECTED BUT COND. STROBE IS
                                        ;*NOT ASSERETED.
                                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
                                        ;;*****************************************************************

                                        ;  TEST 4
                                        ;--------------
010152  012737  000004  001226  TST4:   MOV     #4,TSTNO
010160  012737  010366  001216          MOV     #TST5,NEXT
010166  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
010172  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
010200  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
010202  004737  010270                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
010206  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
010212  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
010220  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
010222  004737  010270                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
010226  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
010232  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
010240  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
010242  004737  010270                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
010246  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
010252  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
010260  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
010262  004737  010270                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
010266  104400          103$:           SCOPE                   ;SCOPE THIS TEST.
010270                  105$:                                   ;TEST ENTRANCE.
010270  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
010276  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
010300  000207                          RTS     PC              ;EXIT TEST
010302  104412                          MSTCLR                  ;RESET DV11
010304  005002                          CLR     R2              ;ZERO SFR IMAGE
010306  012777  004000  171054          MOV     #BIT11,@DVLCR   ;SET INTERNAL MAINT MODE
010314  017705  171050                  MOV     @DVLCR,R5       ;READ THE DVLCR INTO R5
010320  042705  000200                  BIC     #BIT7,R5        ;CLEAR MAINT BIT WINDOW EXPECTED
010324  012703  000004                  MOV     #4,R3           ;SET TO DO 4 LINES.
010330  010077  171036          1$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
010334  017704  171030                  MOV     @DVLCR,R4       ;READ DVLCR RESULTS INTO R4
010340  042705  000060                  BIC     #BIT5+BIT4,R5   ;CLEAR EXTENDED ADDRESS BITS
010344  042704  000060                  BIC     #BIT5+BIT4,R4   ;"
010350  020504                          CMP     R5,R4           ;OK?
010352  001401                          BEQ     2$              ;
010354  104001                          HLT     1               ;BIT7 INCORRECT
010356  005200          2$:             INC     R0              ;UPDATE LINE POINTER
010360  005303                          DEC     R3              ;ALL LINES DONE?
010362  001362                          BNE     1$              ;BR IF NO
010364  000207                          RTS     PC              ;RETURN FOR NEXT SET OF LINES.


                                        ;************************* TEST 5 *******************************
                                        ;*TEST THAT THE BIT WINDOW WILL
                                        ;*SET WHEN MAINT INTERNAL MODE IS SELECTED
                                        ;*AND COND. STOBE IS ASSERTED.
                                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
                                        ;;*****************************************************************
```

```
1437
1438
1439                                        ;  TEST 5
1440                                        ;--------------
1441  010366  012737  000005  001226  TST5:  MOV   #5,TSTNO
1442  010374  012737  010614  001216         MOV   #TST6,NEXT
1443  010402  012700  000000                 MOV   #0.,R0          ;PLACE LINE NUMBER INTO R0
1444  010406  013737  001422  001236         MOV   L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
1445  010414  100402                         BMI   100$            ;BR IF LINE CARD NOT TO BE TESTED
1446  010416  004737  010504                 JSR   PC,105$         ;GO DO THE TEST FOR LINE CARD 1
1447  010422  012700  000004          100$:  MOV   #4.,R0          ;PLACE LINE NUMBER INTO R0
1448  010426  013737  001424  001236         MOV   L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
1449  010434  100402                         BMI   101$            ;BR IF LINE CARD NOT TO BE TESTED
1450  010436  004737  010504                 JSR   PC,105$         ;GO DO THE TEST FOR LINE CARD 2
1451  010442  012700  000010          101$:  MOV   #8.,R0          ;LOAD LINE NUMBER
1452  010446  013737  001426  001236         MOV   L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
1453  010454  100402                         BMI   102$            ;BR IF LINE CARD NOT TO BE TESTED
1454  010456  004737  010504                 JSR   PC,105$         ;DO THE TEST FOR LINE CARD 3
1455  010462  012700  000014          102$:  MOV   #12.,R0         ;LOAD LINE NO.
1456  010466  013737  001430  001236         MOV   L12.15,STAT     ;LOAD LINE CARD STATUS
1457  010474  100402                         BMI   103$            ;BR IF LINE CARD NOT TO BE TESTED
1458  010476  004737  010504                 JSR   PC,105$         ;DO THE TESTS FOR LINE CARD 4
1459  010502  104400                  103$:  SCOPE                 ;SCOPE THIS TEST.
1460  010504                           105$:                        ;TEST ENTRANCE.
1461  010504  032737  004000  001236         BIT   #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
1462  010512  001401                         BEQ   .+4             ;BR IF SYNC LINE CARD.
1463  010514  000207                         RTS   PC              ;EXIT TEST
1464  010516  104412                         MSTCLR                ;RESET DV11
1465  010520  005002                         CLR   R2              ;ZERO SFR IMAGE
1466  010522  012777  004000  170640         MOV   #BIT11,@DVLCR   ;SET INTERNAL MAINT MODE
1467  010530  017705  170634                 MOV   @DVLCR,R5       ;READ THE DVLCR INTO R5
1468  010534  052705  000200                 BIS   #BIT7,R5        ;SET MAINT BIT WINDOW EXP RESULTS
1469  010540  012703  000004                 MOV   #4,R3           ;SET TO DO 4 LINES.
1470  010544  010077  170622           1$:   MOV   R0,@DVSRS       ;LOAD LINE NUMBER
1471  010550  052777  100000  170612         BIS   #BIT15,@DVLCR   ;SET STROBE
1472  010556  004737  023462                 JSR   PC,CKBIT15      ;GO WAIT FOR BIT15 TO =0
1473  010562  017704  170602                 MOV   @DVLCR,R4       ;READ DVLCR RESULTS INTO R4
1474  010566  042705  000060                 BIC   #BIT5+BIT4,R5   ;CLEAR EXTENDED ADDRESS BITS
1475  010572  042704  000060                 BIC   #BIT5+BIT4,R4   ;"
1476  010576  020504                         CMP   R5,R4           ;OK?
1477  010600  001401                         BEQ   2$              ;
1478  010602  104001                         HLT   1               ;BIT7 INCORRECT
1479  010604  005200                   2$:   INC   R0              ;UPDATE LINE POINTER
1480  010606  005303                         DEC   R3              ;ALL LINES DONE?
1481  010610  001355                         BNE   1$              ;BR IF NO
1482  010612  000207                         RTS   PC              ;RETURN FOR NEXT SET OF LINES.
1483
1484
1485                                        ;************************* TEST 6 *******************************
1486                                        ;*TEST THAT THE BIT WINDOW WILL BE CLEARED
1487                                        ;*WHEN MAINT INTERNAL MODE IS SELECTED AND TX DSABLE
1488                                        ;*IS ASSERTED.
1489                                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
1490                                        ;;*************************************************************************
1491
1492
```

```
1493
1494                                        ;  TEST 6
1495                                        ;--------------
1496  010614  012737  000006  001226  TST6:  MOV   #6,TSTNO
1497  010622  012737  011042  001216         MOV   #TST7,NEXT
1498  010630  012700  000000                 MOV   #0.,R0        ;PLACE LINE NUMBER INTO R0
1499  010634  013737  001422  001236         MOV   L00.03,STAT   ;LOAD LINE CARD STATUS INTO STAT
1500  010642  100402                         BMI   100$          ;BR IF LINE CARD NOT TO BE TESTED
1501  010644  004737  010732                 JSR   PC,105$       ;GO DO THE TEST FOR LINE CARD 1
1502  010650  012700  000004          100$:  MOV   #4.,R0        ;PLACE LINE NUMBER INTO R0
1503  010654  013737  001424  001236         MOV   L04.07,STAT   ;LOAD LINE CARD STATUS INTO STAT
1504  010662  100402                         BMI   101$          ;BR IF LINE CARD NOT TO BE TESTED
1505  010664  004737  010732                 JSR   PC,105$       ;GO DO THE TEST FOR LINE CARD 2
1506  010670  012700  000010          101$:  MOV   #8.,R0        ;LOAD LINE NUMBER
1507  010674  013737  001426  001236         MOV   L08.11,STAT   ;LOAD LINE CARD STATUS INTO STAT
1508  010702  100402                         BMI   102$          ;BR IF LINE CARD NOT TO BE TESTED
1509  010704  004737  010732                 JSR   PC,105$       ;DO THE TEST FOR LINE CARD 3
1510  010710  012700  000014          102$:  MOV   #12.,R0       ;LOAD LINE NO.
1511  010714  013737  001430  001236         MOV   L12.15,STAT   ;LOAD LINE CARD STATUS
1512  010722  100402                         BMI   103$          ;BR IF LINE CARD NOT TO BE TESTED
1513  010724  004737  010732                 JSR   PC,105$       ;DO THE TESTS FOR LINE CARD 4
1514  010730  104400                 103$:  SCOPE               ;SCOPE THIS TEST.
1515  010732                         105$:                      ;TEST ENTRANCE.
1516  010732  032737  004000  001236         BIT   #ASYNC,STAT   ;IS THIS A SYNC LINE CARD?
1517  010740  001401                         BEQ   .+4           ;BR IF SYNC LINE CARD.
1518  010742  000207                         RTS   PC            ;EXIT TEST
1519  010744  104412                         MSTCLR              ;RESET DV11
1520  010746  005002                         CLR   R2            ;ZERO SFR IMAGE
1521  010750  012777  005000  170412         MOV   #BIT11+BIT9,@DVLCR      ;SET INTER MAINT MODE FOR SYSTEM TESTING
1522  010756  017705  170406                 MOV   @DVLCR,R5     ;READ THE DVLCR INTO R5
1523  010762  042705  000200                 BIC   #BIT7,R5      ;CLEAR MAINT BIT WINDOW EXPECTED
1524  010766  012703  000004                 MOV   #4,R3         ;SET TO DO 4 LINES.
1525  010772  010077  170374          1$:    MOV   R0,@DVSRS     ;LOAD LINE NUMBER
1526  010776  052777  100000  170364         BIS   #BIT15,@DVLCR ;SET STROBE
1527  011004  004737  023462                 JSR   PC,CKBIT15    ;GO WAIT FOR BIT15 TO =0
1528  011010  017704  170354                 MOV   @DVLCR,R4     ;READ DVLCR RESULTS INTO R4
1529  011014  042705  000060                 BIC   #BIT5+BIT4,R5 ;CLEAR EXTENDED ADDRESS BITS
1530  011020  042704  000060                 BIC   #BIT5+BIT4,R4 ;""
1531  011024  020504                         CMP   R5,R4         ;OK?
1532  011026  001401                         BEQ   2$            ;
1533  011030  104001                         HLT   1             ;BIT7 INCORRECT
1534  011032  005200                 2$:    INC   R0            ;UPDATE LINE POINTER
1535  011034  005303                         DEC   R3            ;ALL LINES DONE?
1536  011036  001355                         BNE   1$            ;BR IF NO
1537  011040  000207                         RTS   PC            ;RETURN FOR NEXT SET OF LINES.
1538
1539                                        ;************************ TEST 7 *********************************
1540                                        ;*TEST THAT 'MAINT DATA' WILL SHOW
1541                                        ;*UP IN 'MAINT BIT WINDOW'.
1542                                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
1543                                        ;:*****************************************************************
1544
1545                                        ;  TEST 7
1546                                        ;--------------
1547  011042  012737  000007  001226  TST7:  MOV   #7,TSTNO
1548  011050  012737  011346  001216         MOV   #TST10,NEXT
```

```
1549  011056  012700  000000            MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
1550  011062  013737  001422  001236    MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
1551  011070  100402                    BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
1552  011072  004737  011160            JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
1553  011076  012700  000004  100$:     MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
1554  011102  013737  001424  001236    MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
1555  011110  100402                    BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
1556  011112  004737  011160            JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
1557  011116  012700  000010  101$:     MOV     #8.,R0          ;LOAD LINE NUMBER
1558  011122  013737  001426  001236    MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
1559  011130  100402                    BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
1560  011132  004737  011160            JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
1561  011136  012700  000014  102$:     MOV     #12.,R0         ;LOAD LINE NO.
1562  011142  013737  001430  001236    MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
1563  011150  100402                    BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
1564  011152  004737  011160            JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
1565  011156  104400          103$:     SCOPE                   ;SCOPE THIS TEST.
1566  011160                  105$:                             ;TEST ENTRANCE.
1567  011160  032737  004000  001236    BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
1568  011166  001401                    BEQ     .+4             ;BR IF SYNC LINE CARD.
1569  011170  000207                    RTS     PC              ;EXIT TEST
1570  011172  104412                    MSTCLR                  ;RESET DV11
1571  011174  005002                    CLR     R2              ;CLEAR DVSFR IMAGE
1572  011176  012703  000004            MOV     #4,R3           ;SET TO DO 4 LINES
1573  011202  010077  170164  1$:       MOV     R0,@DVSRS       ;LOAD LINE NUMBER
1574  011206  004537  023342            PERFORM ,LOAD.MODE      ;LOAD THE MODE
1575  011212  005000                    BIT11+BIT9              ;INT MAIT MODE AND TX DSABLE
1576  011214  017705  170150            MOV     @DVLCR,R5       ;READ LSR
1577  011220  010504                    MOV     R5,R4           ;
1578  011222  042705  000200            BIC     #BIT7,R5        ;CLEAR MAIT BIT WINDOW RESULT
1579  011226  020504                    CMP     R5,R4           ;WAS BIT WINDOW =TO 0
1580  011230  001401                    BEQ     .+4             ;BR IF YES
1581  011232  104001                    HLT     1               ;BIT7 OF LCR S/B=0
1582  011234  012737  000012  001250    MOV     #10.,TEMP2      ;SET FOR 10 BITS
1583  011242  052705  040200    2$:     BIS     #BIT14+BIT7,R5  ;SET MAINT DATA AND MAINT BIT WINDOW
1584  011246  052777  140000  170114    BIS     #BIT15+BIT14,@DVLCR
1585  011254  004737  023462            JSR     PC,CKBIT15      ;STROBE MAINT DATA. WAIT BIT15=0
1586  011260  017704  170104            MOV     @DVLCR,R4       ;READ THE LCR
1587  011264  020504                    CMP     R5,R4           ;BIT14+BIT7=1?
1588  011266  001401                    BEQ     3$              ;YES
1589  011270  104001                    HLT     1               ;MAINT DATA DID NOT SHOW UP IN WINDOW
1590  011272  042705  040200    3$:     BIC     #BIT14+BIT7,R5  ;CLEAR DATA AND WINDOW
1591  011276  042777  040000  170064    BIC     #BIT14,@DVLCR   ;CLEAR MAIT DATA
1592  011304  052777  100000  170056    BIS     #BIT15,@DVLCR   ;SET STROBE ON DV11
1593  011312  004737  023462            JSR     PC,CKBIT15      ;WAIT 15=0
1594  011316  017704  170046            MOV     @DVLCR,R4       ;READ DVLCR
1595  011322  020504                    CMP     R5,R4           ;WINDOW =0?
1596  011324  001401                    BEQ     4$              ;BR IF YES
1597  011326  104001                    HLT     1               ;BIT7 S/B=0
1598  011330  005337  001250    4$:     DEC     TEMP2           ;10 BITS DONE?
1599  011334  001342                    BNE     2$              ;BR IF NO
1600  011336  005200                    INC     R0              ;UPDATE LINE POINTER
1601  011340  005303                    DEC     R3              ;4 LINE GROUP DONE?
1602  011342  001317                    BNE     1$              ;BR IF NO
1603  011344  000207                    RTS     PC              ;RETURN FOR NEXT GROUP
1604
```

```
1605
1606                                    ;************************ TEST 10 *******************************
1607                                    ;*TEST TO XMIT A BINARY COUNT PATTERN
1608                                    ;*THRU THE USE OF THE BIT WINDOW.
1609                                    ;*ONLY ONE LINE AT A TIME WILL BE EXERCISED.
1610                                    ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
1611                                    ;:***************************************************************************
1612
1613                                    ;  TEST 10
1614                                    ;--------------
1615  011346  012737  000010  001226  TST10:  MOV     #10,TSTNO
1616  011354  012737  012242  001216          MOV     #TST11,NEXT
1617  011362  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
1618  011366  113737  001416  001242          MOVB    CLK.A,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
1619  011374  013737  001406  001244          MOV     MASK.A,MASKX    ;PLACE 'MASK'FOR CHARS INTO MASKX
1620  011402  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
1621  011410  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
1622  011412  004737  011544                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
1623  011416  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
1624  011422  113737  001417  001242          MOVB    CLK.B,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
1625  011430  013737  001410  001244          MOV     MASK.B,MASKX    ;GET MASK
1626  011436  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
1627  011444  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
1628  011446  004737  011544                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
1629  011452  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
1630  011456  113737  001420  001242          MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
1631  011464  013737  001412  001244          MOV     MASK.C,MASKX    ;GET MASK
1632  011472  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
1633  011500  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
1634  011502  004737  011544                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
1635  011506  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
1636  011512  113737  001421  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
1637  011520  013737  001414  001244          MOV     MASK.D,MASKX    ;GET MASK
1638  011526  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
1639  011534  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
1640  011536  004737  011544                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
1641  011542  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
1642  011544                          105$:                           ;TEST ENTRANCE.
1643  011544  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
1644  011552  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
1645  011554  000207                          RTS     PC              ;EXIT TEST
1646  011556  010037  011572                  MOV     R0,65$          ;SET LINE NO. POINTER
1647  011562  104412                          MSTCLR                  ;CLEAR DV11
1648  011564  005001                          CLR     R1              ;ZERO MSCANNER POINTER
1649  011566  004537  023544          1$:     PERFORM ,SETSCAN        ;ADJUST SCANNER FOR PROPER LINE
1650  011572  000001                  65$:    .BLKW 1                 ;LINE NUMBER POINTER.
1651  011574  012703  000004          2$:     MOV     #4,R3           ;SET FOR 4 LINES EXERCISED
1652  011600  005005                  3$:     CLR     R5              ;SET DATA POINTER TO 0
1653  011602  012777  050102  167570          MOV     #S.C+BIT6+BIT1,@DVSFR
1654  011610  104415                          ROMCLK                  ;CLOCK SCANNER BY ONE
1655  011612  005201                          INC     R1              ;ADD +1 TO SCANNER POINTER
1656  011614  010077  167552                  MOV     R0,@DVSRS       ;LOAD LINE NUMBER
1657  011620  004537  023342                  PERFORM ,LOAD.MODE      ;LOAD MODE
1658  011624  004000                          BIT11
1659  011626  004537  023634          7$:     PERFORM ,CLR.TMARK      ;CLEAR TMARK BIT.
1660  011632  012777  001000  167540          MOV     #BIT9,@DVSFR    ;DO A BR ''A'' TEST FOR TX FLAG
```

```
1661  011640  005005                        CLR     R5                      ;SET EXPECTED DATA TO 0
1662  011642  032777  000001  167520        BIT     #BIT0,@DVLCR            ;IF FLAG TRUE?
1663  011650  001401                        BEQ     .+4                     ;BR IF YES
1664  011652  104000                        HLT                             ;TX FLAG NO TRUE(LOW(LPR0=0))
1665  011654  005077  167512                CLR     @DVSRS                  ;ZERO LINE TO LINE 0
1666  011660  010577  167512                MOV     R5,@DVSRA               ;LOAD DATA INTO DVSRA
1667  011664  012777  020000  167506        MOV     #BIT13,@DVSFR           ;EXECUTE A 'ROM READ'' INTSTR
1668  011672  104415                        ROMCLK                          ;CLOCK.
1669  011674  012777  030260  167476        MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
1670  011702  104415                        ROMCLK                          ;DO A DATA XFER FROM RAM OUTPUT TO TX BUFFER
1671  011704  104416                        DATACLK                         ;ISSUE A MAINT CLK.
1672  011706  012777  001000  167464        MOV     #BIT9,@DVSFR            ;DO A 'BR A'' TEST FOR TX FLAG
1673  011714  032777  000001  167446        BIT     #BIT0,@DVLCR            ;IS FLAG FALSE?
1674  011722  001001                        BNE     .+4                     ;BR IF YES
1675  011724  104000                        HLT                             ;TX FLAG NOT FALSE(HIGH(LPR0=1))
1676  011726  012737  011734  001220        MOV     #4$,LOCK                ;SET IF SW09=1 GOTO 4$
1677  011734  113702  001242        4$:     MOVB    CLKX,R2                 ;SET REQUIRED SHIFTS
1678  011740  005037  023676                CLR     DATA                    ;CLEAR STUFFER LOCATION
1679  011744  010077  167422                MOV     R0,@DVSRS               ;LOAD LINE NUMBER
1680  011750  104416                5$:     DATACLK                         ;ISSUE MAINT CLK
1681  011752  004537  023252                PERFORM ,TXSHIFT                ;WORK THE TRANSMITTER
1682  011756  005302                        DEC     R2                      ;ALL SHIFTS DONE?
1683  011760  022702  000001                CMP     #1,R2                   ;IS THE BUFFER ALMOST EMPTY?
1684  011764  001030                        BNE     8$                      ;BR IF NO
1685  011766  005077  167400                CLR     @DVSRS                  ;ZERO LINE NUMBER
1686  011772  032777  001000  167202        BIT     #BIT9,@SWR              ;LOCK ON DATA?
1687  012000  001001                        BNE     .+4                     ;BR IF YES!!
1688  012002  005205                        INC     R5                      ;UPDATE DATA.
1689  012004  010577  167366                MOV     R5,@DVSRA               ;LOAD DATA INTO DVSRA
1690  012010  012777  020000  167362        MOV     #BIT13,@DVSFR           ;DO A ROM READ
1691  012016  104415                        ROMCLK                          ;CLK
1692  012020  012777  030260  167352        MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
1693  012026  104415                        ROMCLK                          ;DO A  DATA XFER TO TX BUFF
1694  012030  010077  167336                MOV     R0,@DVSRS               ;RESELECT LINE NUMBER
1695  012034  032777  001000  167140        BIT     #BIT9,@SWR              ;LOCK ON DATA?
1696  012042  001001                        BNE     .+4                     ;BR IF YES!!
1697  012044  005305                        DEC     R5                      ;READJUST DATA CHAR.
1698  012046  005702                8$:     TST     R2                      ;ALL SHIFTS DONE?
1699  012050  001337                        BNE     5$                      ;BR IF NO
1700  012052  022737  000010  001242        CMP     #8.,CLKX                ;IS LINE CARD SET TO 8 BITS?
1701  012060  001420                        BEQ     15$                     ;BR IF YES
1702                                  ;;**********************************************************************
1703  012062  022737  000011  001242        CMP     #9.,CLKX                ;8 BITS WITH PARITY ENABLED?            ;;++C
1704  012070  001414                        BEQ     15$                     ;IF YES, BR
1705                                  ;;**********************************************************************
1706  012072  013737  001242  001246        MOV     CLKX,TEMP1              ;SAVE NUMBER OF SHIFTS DONE.
1707  012100  000241                16$:     CLC                             ;CLEAR CARRY
1708  012102  006037  023676                ROR     DATA                    ;RIGHT JUSTIFY TX RESULTS.
1709  012106  005237  001246                INC     TEMP1                   ;ALL DONE?
1710  012112  022737  000010  001246        CMP     #8.,TEMP1               ;?
1711  012120  001367                        BNE     16$                     ;BR IF NO
1712  012122                         15$:
1713  012122  013704  023676                MOV     DATA,R4                 ;READ IMAGE CHAR FROM TX
1714  012126  043704  001244                BIC     MASKX,R4                ;STRIP PARITY IF IT EXISTS.
1715  012132  020504                        CMP     R5,R4                   ;ARE DATA CHARS THE SAME?
1716  012134  001401                        BEQ     .+4                     ;BR IF GOOD DATA FROM TX
```

```
1717  012136  104003                        HLT     3               ;TX DATA COMPARE ERROR
1718  012140  104401                        SCOP1                   ;LOCK ON DATA?
1719  012142  105205                        INCB    R5              ;UPDATE DATA CHAR.
1720  012144  001403                        BEQ     6$              ;BR IF 8BIT CODE DONE.
1721  012146  133705  001244                BITB    MASKX,R5        ;IF <8BIT SEE IF ALL DONE.
1722  012152  001670                        BEQ     4$              ;BR IF NOT ALL DONE
1723  012154  004537  023622        6$:     PERFORM ,SET.TMARK      ;SET TMARK BIT
1724                                        ;*VERIFY THAT SETTING TMARK BIT PUTS LINE AT MARK.
1725                                        ;*
1726  012160  113702  001242                MOVB    CLKX,R2         ;SET COUNTER
1727  012164  010077  167202                MOV     R0,@DVSRS       ;SET LINE
1728  012170  104416                9$:     DATACLK                 ;CLOCK
1729  012172  005302                        DEC     R2              ;FLUSH LAST CHARACTER.
1730  012174  001375                        BNE     9$              ;CHAR FLUSHED?
1731  012176  012702  000024                MOV     #20.,R2         ;LOOK AT 20. BITS.
1732  012202  104416                10$:    DATACLK                 ;MAINT CLK
1733  012204  032777  000200  167156        BIT     #BIT7,@DVLCR    ;BIT WINDOW
1734  012212  001001                        BNE     11$             ;SET (MARK)
1735  012214  104000                        HLT     0               ;TX BIT WINDOW NOT SET (MARK)
1736  012216  005302                11$:    DEC     R2              ;ALL BITS LOOKED AT?
1737  012220  001370                        BNE     10$             ;BR IF NO
1738  012222  004537  023544                PERFORM ,SETSCAN        ;ADVANCE SCANNER TO NEXT LINE
1739  012226  000001                        1                       ;ONE LINE ADVANCE
1740  012230  005303                        DEC     R3              ;ALL LINES(4) DONE?
1741  012232  001402                        BEQ     12$             ;BR IF YES
1742  012234  000137  011626                JMP     7$              ;IF NO CONTINUE
1743  012240  000207                12$:    RTS     PC              ;GET NEXT GROUP OF 4 LINES.
1744
1745
1746                                        ;************************ TEST 11 ********************************
1747                                        ;*TEST TO CHECK THE IDLE CHARACTER
1748                                        ;*FOR EACH LINE OF THE TRANSMITTER.
1749                                        ;*THIS TEST USES ''SYNCA''.
1750                                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
1751                                        ;:*************************************************************************
1752
1753                                    ;  TEST 11
1754                                    ;--------------
1755  012242  012737  000011  001226    TST11:  MOV     #11,TSTNO
1756  012250  012737  012766  001216            MOV     #TST12,NEXT
1757  012256  012700  000000                    MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
1758  012262  113737  001416  001242            MOVB    CLK.A,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
1759  012270  013737  001406  001244            MOV     MASK.A,MASKX    ;PLACE 'MASK'FOR CHARS INTO MASKX
1760  012276  013737  001422  001236            MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
1761  012304  100402                            BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
1762  012306  004737  012440                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
1763  012312  012700  000004            100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
1764  012316  113737  001417  001242            MOVB    CLK.B,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
1765  012324  013737  001410  001244            MOV     MASK.B,MASKX    ;GET MASK
1766  012332  013737  001424  001236            MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
1767  012340  100402                            BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
1768  012342  004737  012440                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
1769  012346  012700  000010            101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
1770  012352  113737  001420  001242            MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
1771  012360  013737  001412  001244            MOV     MASK.C,MASKX    ;GET MASK
1772  012366  013737  001426  001236            MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
```

```
1773  012374  100402                        BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
1774  012376  004737  012440                JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
1775  012402  012700  000014          102$: MOV     #12.,R0         ;LOAD LINE NO.
1776  012406  113737  001421  001242      MOVB    CLK.D,CLKX      ;GET SHIFTS
1777  012414  013737  001414  001244      MOV     MASK.D,MASKX    ;GET MASK
1778  012422  013737  001430  001236      MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
1779  012430  100402                        BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
1780  012432  004737  012440              JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
1781  012436  104400                  103$: SCOPE                   ;SCOPE THIS TEST.
1782  012440                          105$:                         ;TEST ENTRANCE.
1783  012440  032737  004000  001236      BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
1784  012446  001401                        BEQ     .+4             ;BR IF SYNC LINE CARD.
1785  012450  000207                        RTS     PC              ;EXIT TEST
1786  012452  010037  012466              MOV     R0,65$          ;LOAD LINE NO. POINTER
1787  012456  104412                        MSTCLR                  ;RESET THE DV11
1788  012460  005001                        CLR     R1              ;ZERO MSCANNER POINTER
1789  012462  004537  023544          1$:   PERFORM ,SETSCAN        ;SET MSCANNER TO LINES TESTED
1790  012466  000001                  65$:  .BLKW 1                 ;INITAL LINE VALUE
1791  012470  012703  000004          2$:   MOV     #4,R3           ;SET TO DO 4 LINE GROUP
1792  012474  005005                  3$:   CLR     R5              ;ZERO
1793  012476  012777  050102  166674        MOV     #S.C+BIT6+BIT1,@DVSFR
1794  012504  104415                        ROMCLK                  ;SET/CLEAR ''ADVANCE MSCANNER''
1795  012506  005201                        INC     R1              ;UPDATE MSCANNER POINTER
1796  012510  010077  166656          6$:   MOV     R0,@DVSRS       ;LOAD LINE NUMBER INTO DV11
1797  012514  004537  023634              PERFORM ,CLR.TMARK      ;CLR TMARK BIT.
1798  012520  004537  023342              PERFORM ,LOAD.MODE      ;LOAD THE MODE
1799  012524  004000                        BIT11                   ;INT MAINT MODE
1800  012526  005077  166644              CLR     @DVSRA          ;ZERO DATA FOR XFR
1801  012532  012777  020000  166640        MOV     #BIT13,@DVSFR   ;DO A RAM READ INSTR.
1802  012540  104415                        ROMCLK                  ;
1803  012542  012777  030260  166630        MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
1804  012550  104415                        ROMCLK                  ;DATA XFR  TXBUFFER RAM OUTPUT
1805  012552  104416                        DATACLK                 ;ISSUE MAIT CLOCK PULSE
1806  012554  012737  012606  001220        MOV     #4$,LOCK        ;SET FOR SCOP1
1807  012562  113702  001242              MOVB    CLKX,R2         ;NUMBER OF CLOCK PULSES NEEDED
1808  012566  104416                        DATACLK                 ;MAINT CLOCK PULSE
1809  012570  005302                        DEC     R2              ;ALL CLOCKS DONE?
1810  012572  001375                        BNE     .-4             ;NO , DO MORE
1811  012574  113705  001236              MOVB    STAT,R5         ;GET SYNC (IDLE) CHAR.
1812  012600  012737  000005  001250        MOV     #5,TEMP2        ;SET FOR 5 CHARS
1813  012606  113702  001242          4$:   MOVB    CLKX,R2         ;GET CLOCKS NEEDED
1814  012612  005037  023676              CLR     DATA            ;ZERO STORAGE AREA
1815  012616  010077  166550              MOV     R0,@DVSRS       ;LOAD LINE NUMBER
1816  012622  104416                  5$:   DATACLK                 ;ISSUE MAINT CLK PULSE
1817  012624  004537  023252              PERFORM ,TXSHIFT        ;CLOCK THE TRANSMITTER
1818  012630  005302                        DEC     R2              ;MORE SHIFTS REQUIRED?
1819  012632  001373                        BNE     5$              ;BR IF YES
1820  012634  022737  000010  001242        CMP     #8.,CLKX        ;IS LINE CARD SET TO 8 BITS?
1821  012642  001420                        BEQ     15$             ;BR IF YES
1822                          ;;*************************************************************************
1823  012644  022737  000011  001242        CMP     #9.,CLKX        ;8 BITS WITH PARITY ENABLED?            ;;++C
1824  012652  001414                        BEQ     15$             ;IF YES BR
1825                          ;;*************************************************************************
1826  012654  013737  001242  001246        MOV     CLKX,TEMP1      ;SAVE NUMBER OF SHIFTS DONE.
1827  012662  000241                  16$:  CLC                     ;CLEAR CARRY
1828  012664  006037  023676              ROR     DATA            ;RIGHT JUSTIFY TX RESULTS.
```

```
1829  012670  005237  001246                    INC     TEMP1           ;ALL DONE?
1830  012674  022737  000010  001246            CMP     #8.,TEMP1       ;?
1831  012702  001367                            BNE     16$             ;BR IF NO
1832  012704                                  15$:
1833  012704  013704  023676                    MOV     DATA,R4         ;SAVE DATA SHIFTED OUT OF TX.
1834  012710  143704  001244                    BICB    MASKX,R4        ;CLEAR UNWANTED BITS.
1835  012714  042705  177400                    BIC     #^C<377>,R5     ;CLEAR SIGN EXTEND.
1836  012720  143705  001244                    BICB    MASKX,R5        ;CLEAR UNUSED BITS
1837  012724  042704  177400                    BIC     #^C<377>,R4     ;CLEAR SIGN EXTEND.
1838  012730  020504                            CMP     R5,R4           ;EXPECTED = FOUND ??
1839  012732  001401                            BEQ     .+4             ;BR IF OK
1840  012734  104003                            HLT     3               ;IDLE CHAR NOT WHAT EXPECTED.
1841  012736  005337  001250                    DEC     TEMP2           ;ALL IDLE CHARS DONE?
1842  012742  001321                            BNE     4$              ;BR IF NO
1843  012744  104401                            SCOP1                   ;LOCK (SW09=1)?
1844  012746  004537  023622                    PERFORM ,SET.TMARK      ;SET TMARK BIT
1845  012752  004537  023544                    PERFORM ,SETSCAN        ;UPDATE SCANNER TO NEXT LINE
1846  012756  000001                            1                       ;
1847  012760  005303                            DEC     R3              ;ALL LINES DONE
1848  012762  001252                            BNE     6$              ;BR IF NO
1849  012764  000207                            RTS     PC              ;EXIT FOR NEXT GROUP OF LINES.
1850
1851
1852                                            ;************************ TEST 12 *******************************
1853                                            ;*TEST TO CHECK THE IDLE CHARACTER
1854                                            ;*FOR EACH LINE OF THE TRANSMITTER.
1855                                            ;*THIS TEST USES ''SYNCB''.
1856                                            ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
1857                                            ;:***************************************************************
1858
1859                                          ;  TEST 12
1860                                          ;--------------
1861  012766  012737  000012  001226          TST12:  MOV     #12,TSTNO
1862  012774  012737  013542  001216                  MOV     #TST13,NEXT
1863  013002  012700  000000                          MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
1864  013006  113737  001416  001242                  MOVB    CLK.A,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
1865  013014  013737  001406  001244                  MOV     MASK.A,MASKX    ;PLACE 'MASK'FOR CHARS INTO MASKX
1866  013022  013737  001432  001240                  MOV     SYNC2A,SYNCX    ;
1867  013030  013737  001422  001236                  MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
1868  013036  100402                                  BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
1869  013040  004737  013214                          JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
1870  013044  012700  000004                  100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
1871  013050  113737  001417  001242                  MOVB    CLK.B,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
1872  013056  013737  001410  001244                  MOV     MASK.B,MASKX    ;GET MASK
1873  013064  013737  001434  001240                  MOV     SYNC2B,SYNCX    ;
1874  013072  013737  001424  001236                  MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
1875  013100  100402                                  BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
1876  013102  004737  013214                          JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
1877  013106  012700  000010                  101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
1878  013112  113737  001420  001242                  MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
1879  013120  013737  001412  001244                  MOV     MASK.C,MASKX    ;GET MASK
1880  013126  013737  001436  001240                  MOV     SYNC2C,SYNCX    ;
1881  013134  013737  001426  001236                  MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
1882  013142  100402                                  BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
1883  013144  004737  013214                          JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
1884  013150  012700  000014                  102$:   MOV     #12.,R0         ;LOAD LINE NO.
```

```
 1885  013154  113737  001421  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
 1886  013162  013737  001414  001244          MOV     MASK.D,MASKX    ;GET MASK
 1887  013170  013737  001440  001240          MOV     SYNC2D,SYNCX    ;
 1888  013176  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 1889  013204  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 1890  013206  004737  013214                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 1891  013212  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
 1892  013214                          105$:                           ;TEST ENTRANCE.
 1893  013214  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
 1894  013222  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
 1895  013224  000207                          RTS     PC              ;EXIT TEST
 1896  013226  010037  013242                  MOV     R0,65$          ;LOAD LINE NO. POINTER
 1897  013232  104412                          MSTCLR                  ;RESET THE DV11
 1898  013234  005001                          CLR     R1              ;ZERO MSCANNER POINTER
 1899  013236  004537  023544          1$:     PERFORM ,SETSCAN        ;SET MSCANNER TO LINES TESTED
 1900  013242  000001                  65$:    .BLKW 1                 ;INITAL LINE VALUE
 1901  013244  012703  000004          2$:     MOV     #4,R3           ;SET TO DO 4 LINE GROUP
 1902  013250  005005                  3$:     CLR     R5              ;ZERO
 1903  013252  012777  050102  166120          MOV     #S.C+BIT6+BIT1,@DVSFR
 1904  013260  104415                          ROMCLK                  ;SET/CLEAR ''ADVANCE MSCANNER''
 1905  013262  005201                          INC     R1              ;UPDATE MSCANNER POINTER
 1906  013264  010077  166102          6$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER INTO DV11
 1907  013270  004537  023634                  PERFORM ,CLR.TMARK      ;CLR TMARK BIT.
 1908  013274  004537  023342                  PERFORM ,LOAD.MODE      ;LOAD THE MODE
 1909  013300  006000                          BIT11+BIT10             ;INT MAINT MODE AND SECOND SYNC
 1910  013302  005077  166070                  CLR     @DVSRA          ;ZERO DATA FOR XFR
 1911  013306  012777  020000  166064          MOV     #BIT13,@DVSFR   ;DO A RAM READ INSTR.
 1912  013314  104415                          ROMCLK                  ;
 1913  013316  012777  030260  166054          MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
 1914  013324  104415                          ROMCLK                  ;DATA XFR  TXBUFFER RAM OUTPUT
 1915  013326  104416                          DATACLK                 ;ISSUE MAIT CLOCK PULSE
 1916  013330  012737  013362  001220          MOV     #4$,LOCK        ;SET FOR SCOP1
 1917  013336  113702  001242                  MOVB    CLKX,R2         ;NUMBER OF CLOCK PULSES NEEDED
 1918  013342  104416                          DATACLK                 ;MAINT CLOCK PULSE
 1919  013344  005302                          DEC     R2              ;ALL CLOCKS DONE?
 1920  013346  001375                          BNE     .-4             ;NO , DO MORE
 1921  013350  113705  001240                  MOVB    SYNCX,R5        ;GET SYNC (IDLE CHAR).
 1922  013354  012737  000005  001250          MOV     #5,TEMP2        ;SET FOR 5 CHARS
 1923  013362  113702  001242          4$:     MOVB    CLKX,R2         ;GET CLOCKS NEEDED
 1924  013366  005037  023676                  CLR     DATA            ;ZERO STORAGE AREA
 1925  013372  010077  165774                  MOV     R0,@DVSRS       ;LOAD LINE NUMBER
 1926  013376  104416                  5$:     DATACLK                 ;ISSUE MAINT CLK PULSE
 1927  013400  004537  023252                  PERFORM ,TXSHIFT        ;CLOCK THE TRANSMITTER
 1928  013404  005302                          DEC     R2              ;MORE SHIFTS REQUIRED?
 1929  013406  001373                          BNE     5$              ;BR IF YES
 1930  013410  022737  000010  001242          CMP     #8.,CLKX        ;IS LINE CARD SET TO 8 BITS?
 1931  013416  001420                          BEQ     15$             ;BR IF YES
 1932                          ;;*************************************************************************
 1933  013420  022737  000011  001242          CMP     #9.,CLKX        ;8 BITS WITH PARITY ENABLED?            ;;++C
 1934  013426  001414                          BEQ     15$             ;IF YES BR
 1935                          ;;*************************************************************************
 1936  013430  013737  001242  001246          MOV     CLKX,TEMP1      ;SAVE NUMBER OF SHIFTS DONE.
 1937  013436  000241                  16$:    CLC                     ;CLEAR CARRY
 1938  013440  006037  023676                  ROR     DATA            ;RIGHT JUSTIFY TX RESULTS.
 1939  013444  005237  001246                  INC     TEMP1           ;ALL DONE?
 1940  013450  022737  000010  001246          CMP     #8.,TEMP1       ;?
```

```
1941  013456  001367                          BNE     16$             ;BR IF NO
1942  013460                          15$:
1943  013460  013704  023676                  MOV     DATA,R4         ;SAVE DATA SHIFTED OUT OF TX.
1944  013464  143704  001244                  BICB    MASKX,R4        ;CLEAR UNWANTED BITS.
1945  013470  042705  177400                  BIC     #^C<377>,R5     ;CLEAR SIGN EXTEND.
1946  013474  143705  001244                  BICB    MASKX,R5        ;CLEAR UNUSED BITS
1947  013500  042704  177400                  BIC     #^C<377>,R4     ;CLEAR SIGN EXTEND.
1948  013504  020504                          CMP     R5,R4           ;EXPECTED = FOUND ??
1949  013506  001401                          BEQ     .+4             ;BR IF OK
1950  013510  104003                          HLT     3               ;IDLE CHAR NOT WHAT EXPECTED.
1951  013512  005337  001250                  DEC     TEMP2           ;ALL IDLE CHARS DONE?
1952  013516  001321                          BNE     4$              ;BR IF NO
1953  013520  104401                          SCOP1                   ;LOCK (SW09=1)?
1954  013522  004537  023622                  PERFORM ,SET.TMARK      ;SET TMARK BIT
1955  013526  004537  023544                  PERFORM ,SETSCAN        ;UPDATE SCANNER TO NEXT LINE
1956  013532  000001                          1                       ;
1957  013534  005303                          DEC     R3              ;ALL LINES DONE
1958  013536  001252                          BNE     6$              ;BR IF NO
1959  013540  000207                          RTS     PC              ;EXIT FOR NEXT GROUP OF LINES.
1960
1961
1962                                  ;********************** TEST 13 ******************************
1963                                  ;*THIS TEST CHECKS "RECEIVE CHAR SILO" TO BE
1964                                  ;*ALL ZERO'S WHEN "DATA ENABLE" IS NOT SET.
1965                                  ;*EXPECTED DATA SHOULD BE LINE NUMBER ONLY
1966                                  ;*DATA 0'S AND ERROR FLAGS 0.
1967                                  ;*THIS TEST WILL BE DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
1968                                  ;:************************************************************
1969
1970                          ;  TEST 13
1971                          ;--------------
1972  013542  012737  000013  001226  TST13:  MOV     #13,TSTNO
1973  013550  012737  014050  001216          MOV     #TST14,NEXT
1974  013556  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
1975  013562  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
1976  013570  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
1977  013572  004737  013660                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
1978  013576  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
1979  013602  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
1980  013610  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
1981  013612  004737  013660                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
1982  013616  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
1983  013622  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
1984  013630  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
1985  013632  004737  013660                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
1986  013636  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
1987  013642  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
1988  013650  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
1989  013652  004737  013660                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
1990  013656  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
1991  013660                          105$:                           ;TEST ENTRANCE.
1992  013660  010037  013700                  MOV     R0,65$          ;STORE LINE NO. POINTER
1993  013664  012703  000004                  MOV     #4,R3           ;SET FOR 4 LINE GROUP
1994  013670  104412                  1$:     MSTCLR                  ;RESET DV11
1995  013672  005001                          CLR     R1              ;ZERO MSCANNER POINTER
1996  013674  004537  023544                  PERFORM ,SETSCAN        ;ADJUST SCANNER
```

```
 1997  013700  000001                  65$:    .BLKW 1                 ;TO CORRECT LINE NO.
 1998  013702  010005                          MOV     R0,R5           ;PLACE LINE NUMBER INTO R5
 1999  013704  000305                          SWAB    R5              ;PLACE LINE NO. IN HIGH BYTE
 2000  013706  105005                          CLRB    R5              ;CLEAR LOW BYTE OF EXPECTED
 2001  013710                          3$:
 2002  013710  012777  050021  165462          MOV     #S.C+BIT4+BIT0,@DVSFR
 2003  013716  104415                          ROMCLK                  ;SET/CLEAR SILO IN
 2004  013720  005002                          CLR     R2              ;
 2005  013722  012777  001400  165450          MOV     #BIT9+BIT8,@DVSFR
 2006  013730  032777  000001  165432  4$:     BIT     #BIT0,@DVLCR    ;'RECV CHAR WAITING TRUE''
 2007  013736  001403                          BEQ     5$              ;BR IF YES
 2008  013740  005202                          INC     R2              ;DELAY IF NOT READY
 2009  013742  001372                          BNE     4$              ;END OF DELAY?
 2010  013744  104000                          HLT     0               ;'RECV CHAR WAITING'' NOT TRUE
 2011  013746  012777  030306  165424  5$:     MOV     #XFR+BIT7+BIT6+BIT2+BIT1,@DVSFR
 2012  013754  017702  165420                  MOV     @DVSFR,R2       ;XFR RICR SILO OUT
 2013  013760  104415                          ROMCLK                  ;DATA/XFER  RICR_SILO OUT
 2014  013762  017704  165400                  MOV     @DVRIC,R4       ;READ RIC
 2015  013766  020504                          CMP     R5,R4           ;EXPECTED OK?
 2016  013770  001401                          BEQ     .+4             ;
 2017  013772  104001                          HLT     1               ;
 2018  013774  062705  000400                  ADD     #400,R5         ;UPDATE LINE NO. (POINTER)
 2019  014000  005002                          CLR     R2              ;SFR IMAGE
 2020  014002  012777  050020  165370          MOV     #S.C+BIT4,@DVSFR
 2021  014010  104415                          ROMCLK                  ;S/C ''SET SILO OUT''
 2022  014012  012777  001400  165360          MOV     #BIT9+BIT8,@DVSFR
 2023  014020  032777  000001  165342  6$:     BIT     #BIT0,@DVLCR    ;'RECV CHAR WAITING'
 2024  014026  001003                          BNE     7$              ;FALSE?
 2025  014030  005202                          INC     R2              ;DELAY WAITING....
 2026  014032  001372                          BNE     6$              ;DELAY DONE?
 2027  014034  104000                          HLT     0               ;
 2028  014036  005237  013700          7$:     INC     65$             ;UPDATE MSCANNER POINTER(LINE)
 2029  014042  005303                          DEC     R3              ;GROUP OF 4 LINES DONE.
 2030  014044  001311                          BNE     1$              ;BR IF YES
 2031  014046  000207                          RTS     PC              ;EXIT FOR NEXT GROUP OF LINES
 2032
 2033
 2034                                          ;************************ TEST 14 *******************************
 2035                                          ;*THIS TEST CHECKS 'RECEIVER CHAR SILO''
 2036                                          ;*WHEN 'DATA ENABLE IS SET'' EXPECTED DATA S/B
 2037                                          ;*ALL 1'S FOR RX DATA, LINE NUMBER CORRECT,
 2038                                          ;*AND ERROR FLAGS =0.
 2039                                          ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
 2040                                          ;:***************************************************************
 2041
 2042                                  ;  TEST 14
 2043                                  ;--------------
 2044  014050  012737  000014  001226  TST14:  MOV     #14,TSTNO
 2045  014056  012737  014402  001216          MOV     #TST15,NEXT
 2046  014064  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
 2047  014070  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2048  014076  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 2049  014100  004737  014166                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 2050  014104  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
 2051  014110  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2052  014116  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
```

```
2053  014120  004737  014166                        JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2054  014124  012700  000010              101$:     MOV     #8.,R0          ;LOAD LINE NUMBER
2055  014130  013737  001426  001236              MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2056  014136  100402                              BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2057  014140  004737  014166                        JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2058  014144  012700  000014              102$:     MOV     #12.,R0         ;LOAD LINE NO.
2059  014150  013737  001430  001236              MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2060  014156  100402                              BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2061  014160  004737  014166                        JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2062  014164  104400                      103$:     SCOPE                   ;SCOPE THIS TEST.
2063  014166                              105$:                             ;TEST ENTRANCE.
2064  014166  032737  004000  001236              BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
2065  014174  001401                              BEQ     .+4             ;BR IF SYNC LINE CARD.
2066  014176  000207                              RTS     PC              ;EXIT TEST
2067  014200  010037  014220                      MOV     R0,65$          ;STORE LINE NO. POINTER
2068  014204  012703  000004                      MOV     #4,R3           ;SET FOR 4 LINE GROUP
2069  014210  104412                      1$:     MSTCLR                  ;RESET DV11
2070  014212  005001                              CLR     R1              ;ZERO MSCANNER POINTER
2071  014214  004537  023544                      PERFORM ,SETSCAN        ;ADJUST SCANNER
2072  014220  000001                      65$:    .BLKW 1                 ;TO CORRECT LINE NO.
2073  014222  010005                              MOV     R0,R5           ;PLACE LINE NUMBER INTO R5
2074  014224  000305                              SWAB    R5              ;PLACE LINE NO. IN HIGH BYTE
2075  014226  052705  000377                      BIS     #377,R5         ;SET LOW BYTE TO ALL 1'S
2076  014232                              3$:
2077  014232  012777  050023  165140              MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
2078  014240  104415                              ROMCLK                  ;S/C ''SET RECV DATA ENABLE''
2079  014242  012777  050021  165130              MOV     #S.C+BIT4+BIT0,@DVSFR
2080  014250  104415                              ROMCLK                  ;SET/CLEAR SILO IN
2081  014252  005002                              CLR     R2              ;
2082  014254  012777  001400  165116              MOV     #BIT9+BIT8,@DVSFR
2083  014262  032777  000001  165100      4$:     BIT     #BIT0,@DVLCR    ;''RECV CHAR WAITING TRUE''
2084  014270  001403                              BEQ     5$              ;BR IF YES
2085  014272  005202                              INC     R2              ;DELAY IF NOT READY
2086  014274  001372                              BNE     4$              ;END OF DELAY?
2087  014276  104000                              HLT     0               ;''RECV CHAR WAITING'' NOT TRUE
2088  014300  012777  030306  165072      5$:     MOV     #XFR+BIT7+BIT6+BIT2+BIT1,@DVSFR
2089  014306  017702  165066                      MOV     @DVSFR,R2       ;XFR RICR_SILO OUT
2090  014312  104415                              ROMCLK                  ;DATA/XFER  RICR_SILO OUT
2091  014314  017704  165046                      MOV     @DVRIC,R4       ;READ RIC
2092  014320  020504                              CMP     R5,R4           ;EXPECTED OK?
2093  014322  001401                              BEQ     .+4             ;
2094  014324  104001                              HLT     1               ;
2095  014326  062705  000400                      ADD     #400,R5         ;UPDATE LINE NO. (POINTER)
2096  014332  005002                              CLR     R2              ;SFR IMAGE
2097  014334  012777  050020  165036              MOV     #S.C+BIT4,@DVSFR
2098  014342  104415                              ROMCLK                  ;S/C ''SET SILO OUT''
2099  014344  012777  001400  165026              MOV     #BIT9+BIT8,@DVSFR
2100  014352  032777  000001  165010      6$:     BIT     #BIT0,@DVLCR    ;''RECV CHAR WAITING''
2101  014360  001003                              BNE     7$              ;FALSE?
2102  014362  005202                              INC     R2              ;DELAY WAITING....
2103  014364  001372                              BNE     6$              ;DELAY DONE?
2104  014366  104000                              HLT     0               ;
2105  014370  005237  014220              7$:     INC     65$             ;UPDATE MSCANNER POINTER(LINE)
2106  014374  005303                              DEC     R3              ;GROUP OF 4 LINES DONE.
2107  014376  001304                              BNE     1$              ;BR IF YES
2108  014400  000207                              RTS     PC              ;EXIT FOR NEXT GROUP OF LINES
```

```
2109
2110
2111                                    ;************************ TEST 15 ****************************
2112                                    ;*TEST THAT EACH RECEIVER WILL SET
2113                                    ;*'MATCH DETECT' WHEN THE FIRST SYNC
2114                                    ;*CHARACTER IS PUMPED INTO IT.
2115                                    ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
2116                                    ;;*****************************************************************
2117
2118                                    ;  TEST 15
2119                                    ;--------------
2120  014402  012737  000015  001226  TST15:  MOV     #15,TSTNO
2121  014410  012737  014700  001216          MOV     #TST16,NEXT
2122  014416  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2123  014422  113737  001416  001242          MOVB    CLK.A,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
2124  014430  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2125  014436  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2126  014440  004737  014550                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2127  014444  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2128  014450  113737  001417  001242          MOVB    CLK.B,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
2129  014456  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2130  014464  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2131  014466  004737  014550                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2132  014472  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2133  014476  113737  001420  001242          MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
2134  014504  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2135  014512  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2136  014514  004737  014550                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2137  014520  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
2138  014524  113737  001421  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
2139  014532  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2140  014540  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2141  014542  004737  014550                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2142  014546  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
2143  014550                          105$:                           ;TEST ENTRANCE.
2144  014550  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
2145  014556  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
2146  014560  000207                          RTS     PC              ;EXIT TEST
2147  014562  012703  000004                  MOV     #4,R3
2148  014566  010037  014602                  MOV     R0,65$          ;SET LINE NO. POINTER
2149  014572  104412                  1$:     MSTCLR                  ;RESET DV11
2150  014574  005001                          CLR     R1              ;ZERO MSCANNER POINTER
2151  014576  004537  023544                  PERFORM ,SETSCAN
2152  014602  000001                  65$:    .BLKW 1                 ;SET MSCANNER TO CORRECT LINE.
2153  014604  010077  164562          3$:     MOV     R0,@DVSRS       ;LOAD LINE NO.
2154  014610  004537  023342                  PERFORM ,LOAD.MODE      ;LOAD THE MODE
2155  014614  025000                          BIT13+BIT11+BIT9        ;RECV ENABLE,INT MAINT,TX DSABLE
2156  014616  113737  001236  023676          MOVB    STAT,DATA       ;GET ''SYNC'' CHAR.
2157  014624  104416                          DATACLK                 ;PRIME DV11
2158  014626  004537  023402                  PERFORM ,RXSHIFT        ;SHIFT DATA INTO RECEIVER
2159  014632  001242                          CLKX                    ;NO. OF SHIFTS GIVEN
2160  014634  012777  076400  164536          MOV     #BRB+BIT11+BIT10+BIT8,@DVSFR
2161  014642  017704  164522                  MOV     @DVLCR,R4       ;BRB ''MATCH DET''
2162  014646  010405                          MOV     R4,R5           ;
2163  014650  052705  000001                  BIS     #BIT0,R5        ;
2164  014654  042705  000002                  BIC     #BIT1,R5        ;
```

```
2165  014660  020504                        CMP     R5,R4           ;MATCH DET TRUE??
2166  014662  001401                        BEQ     4$              ;BR IF YES
2167  014664  104001                        HLT     1               ;
2168  014666  005237  014602        4$:     INC     65$             ;UPDATE TO NEXT LINE.
2169  014672  005303                        DEC     R3              ;4 LINE GROUP DONE?
2170  014674  001336                        BNE     1$              ;BR IF NO
2171  014676  000207                        RTS     PC              ;OBTAIN NEXT 4 LINE GROUP
2172
2173
2174                                        ;************************ TEST 16 *******************************
2175                                        ;*TEST TO VERIFY THAT IF THE DV11 RECEIVER
2176                                        ;*IS SET FOR ONE SYNC CHAR;
2177                                        ;*''MATCH DET'' *AND* ''CHAR FLAG'' ARE
2178                                        ;*SET AFTER ONE SYNC IS PUSHED INTO THE RECEIVER
2179                                        ;*              HOWEVER...
2180                                        ;*IF THE DV11 RECEIVER IS SET FOR
2181                                        ;*TWO SYNC CHARS....
2182                                        ;*VERIFY THAT ''MATCH DET'' SETS ON THE FIRST SYNC
2183                                        ;*AND VERIFY THAT ''MATCH DET'' *AND* ''CHAR FLAG''
2184                                        ;*ARE SET ON THE SECOND SYNC.
2185                                        ;*THIS TEST USES ''SYNC A''.
2186                                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
2187                                        ;:*************************************************************
2188
2189                                ;  TEST 16
2190                                ;--------------
2191  014700  012737  000016  001226  TST16:  MOV     #16,TSTNO
2192  014706  012737  015334  001216          MOV     #TST17,NEXT
2193  014714  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2194  014720  113737  001416  001242          MOVB    CLK.A,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
2195  014726  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2196  014734  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2197  014736  004737  015046                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2198  014742  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2199  014746  113737  001417  001242          MOVB    CLK.B,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
2200  014754  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2201  014762  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2202  014764  004737  015046                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2203  014770  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2204  014774  113737  001420  001242          MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
2205  015002  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2206  015010  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2207  015012  004737  015046                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2208  015016  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
2209  015022  113737  001421  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
2210  015030  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2211  015036  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2212  015040  004737  015046                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2213  015044  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
2214  015046                          105$:                           ;TEST ENTRANCE.
2215  015046  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
2216  015054  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
2217  015056  000207                          RTS     PC              ;EXIT TEST
2218  015060  012703  000004                  MOV     #4,R3           ;SET FOR 4 LINES
2219  015064  010037  015100                  MOV     R0,65$          ;PLACE LINE NO. POINTER
2220  015070  104412                  1$:     MSTCLR                  ;INIT DV11
```

```
2221  015072  005001                        CLR     R1                      ;ZERO MSCANNER POINTER
2222  015074  004537  023544                PERFORM ,SETSCAN                ;SET SCANNER TO LINE DESIRED
2223  015100  000001                65$:    .BLKW 1                         ;INITIAL LINE NUMBER.
2224  015102  010077  164264        3$:     MOV     R0,@DVSRS               ;LOAD LINE NUMBER
2225  015106  004537  023342                PERFORM ,LOAD.MODE              ;LOAD
2226  015112  025000                        BIT13+BIT11+BIT9                ;MODE AND RX ENABLEAND TX DSABLE
2227  015114  113737  001236  023676        MOVB    STAT,DATA               ;PLACE SYNC CHAR IN DATA
2228  015122  104416                        DATACLK                         ;INIT DATA CLOCK.
2229  015124  004537  023402                PERFORM ,RXSHIFT                ;SHIFT DATA INTO RX
2230  015130  001242                        CLKX                            ;NUMBER OF SHIFTS NEEDED
2231  015132  012777  076400  164240        MOV     #BRB+BIT11+BIT10+BIT8,@DVSFR
2232                                                                        ;SET BR 'B'' AND MATCH DET.
2233  015140  017704  164224                MOV     @DVLCR,R4               ;SAVE LPR IN R4
2234  015144  010405                        MOV     R4,R5                   ;SET FOR COMPARE
2235  015146  052705  000001                BIS     #BIT0,R5                ;BR ''A'' FALSE
2236  015152  042705  000002                BIC     #BIT1,R5                ;BR ''B'' TRUE
2237  015156  020504                        CMP     R5,R4
2238  015160  001401                        BEQ     .+4                     ;BR IF LPR OK.
2239  015162  104001                        HLT     1                       ;EXPECT B TRUE; A FALSE
2240  015164  012777  002000  164206        MOV     #BIT10,@DVSFR           ;SET BR ''A'' AND RX CHAR FLAG.
2241  015172  017704  164172                MOV     @DVLCR,R4               ;SAVE LPR IN R4
2242  015176  010405                        MOV     R4,R5                   ;SET FOR COMPARE
2243  015200  032737  010000  001236        BIT     #TWOSYN,STAT            ;SET FOR ONE SYNC OR TWO?
2244  015206  001036                        BNE     4$                      ;BR IF SET FOR ONE SYNC
2245  015210  052705  000003                BIS     #BIT1+BIT0,R5
2246  015214  020504                        CMP     R5,R4
2247  015216  001401                        BEQ     .+4
2248  015220  104001                        HLT     1
2249  015222  113737  001236  023676        MOVB    STAT,DATA
2250  015230  004537  023402                PERFORM ,RXSHIFT
2251  015234  001242                        CLKX
2252  015236  012777  076400  164134        MOV     #BRB+BIT11+BIT10+BIT8,@DVSFR
2253                                                                        ;SET BR 'B'' AND MATCH DET.
2254  015244  017704  164120                MOV     @DVLCR,R4               ;SAVE LPR IN R4
2255  015250  010405                        MOV     R4,R5                   ;SET FOR COMPARE
2256  015252  052705  000001                BIS     #BIT0,R5                ;BR ''A'' FALSE
2257  015256  042705  000002                BIC     #BIT1,R5                ;BR ''B'' TRUE
2258  015262  020504                        CMP     R5,R4
2259  015264  001401                        BEQ     .+4                     ;BR IF LPR OK.
2260  015266  104001                        HLT     1                       ;EXPECT B TRUE; A FALSE
2261  015270  012777  002000  164102        MOV     #BIT10,@DVSFR           ;SET BR ''A'' AND RX CHAR FLAG.
2262  015276  017704  164066                MOV     @DVLCR,R4               ;SAVE LPR IN R4
2263  015302  010405                        MOV     R4,R5                   ;SET FOR COMPARE
2264  015304  052705  000002        4$:     BIS     #BIT1,R5
2265  015310  042705  000001                BIC     #BIT0,R5
2266  015314  020504                        CMP     R5,R4
2267  015316  001401                        BEQ     .+4
2268  015320  104001                        HLT     1
2269  015322  005237  015100                INC     65$                     ;UPDATE LINE NUMBER
2270  015326  005303                        DEC     R3
2271  015330  001257                        BNE     1$
2272  015332  000207                        RTS     PC
2273
2274
2275                                    ;************************* TEST 17 ********************************
2276                                    ;*TEST TO VERIFY THAT IF THE DV11 RECEIVER
```

```
                                        ;*IS SET FOR ONE SYNC CHAR;
                                        ;*'MATCH DET'' *AND* ''CHAR FLAG'' ARE
                                        ;*SET AFTER ONE SYNC IS PUSHED INTO THE RECEIVER
                                        ;*              HOWEVER...
                                        ;*IF THE DV11 RECEIVER IS SET FOR
                                        ;*TWO SYNC CHARS....
                                        ;*VERIFY THAT 'MATCH DET'' SETS ON THE FIRST SYNC
                                        ;*AND VERIFY THAT 'MATCH DET'' *AND* ''CHAR FLAG''
                                        ;*ARE SET ON THE SECOND SYNC.
                                        ;*THIS TEST USES ''SYNC B''.
                                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
                                        ;:*****************************************************************

                                ;  TEST 17
                                ;-------------
015334  012737  000017  001226  TST17:  MOV     #17,TSTNO
015342  012737  016020  001216         MOV     #TST20,NEXT
015350  012700  000000                 MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
015354  113737  001416  001242         MOVB    CLK.A,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
015362  013737  001432  001240         MOV     SYNC2A,SYNCX    ;
015370  013737  001422  001236         MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
015376  100402                         BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
015400  004737  015532                 JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
015404  012700  000004         100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
015410  113737  001417  001242         MOVB    CLK.B,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
015416  013737  001434  001240         MOV     SYNC2B,SYNCX    ;
015424  013737  001424  001236         MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
015432  100402                         BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
015434  004737  015532                 JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
015440  012700  000010         101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
015444  113737  001420  001242         MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
015452  013737  001436  001240         MOV     SYNC2C,SYNCX    ;
015460  013737  001426  001236         MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
015466  100402                         BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
015470  004737  015532                 JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
015474  012700  000014         102$:   MOV     #12.,R0         ;LOAD LINE NO.
015500  113737  001421  001242         MOVB    CLK.D,CLKX      ;GET SHIFTS
015506  013737  001440  001240         MOV     SYNC2D,SYNCX    ;
015514  013737  001430  001236         MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
015522  100402                         BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
015524  004737  015532                 JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
015530  104400                 103$:   SCOPE                   ;SCOPE THIS TEST.
015532                         105$:                           ;TEST ENTRANCE.
015532  032737  004000  001236         BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
015540  001401                         BEQ     .+4             ;BR IF SYNC LINE CARD.
015542  000207                         RTS     PC              ;EXIT TEST
015544  012703  000004                 MOV     #4,R3           ;SET FOR 4 LINES
015550  010037  015564                 MOV     R0,65$          ;PLACE LINE NO. POINTER
015554  104412                 1$:     MSTCLR                  ;INIT DV11
015556  005001                         CLR     R1              ;ZERO MSCANNER POINTER
015560  004537  023544                 PERFORM ,SETSCAN         ;SET SCANNER TO LINE DESIRED
015564  000001                 65$:    .BLKW 1                 ;INITIAL LINE NUMBER.
015566  010077  163600         3$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
015572  004537  023342                 PERFORM ,LOAD.MODE       ;LOAD
015576  027000                         BIT13+BIT11+BIT10+BIT9  ;MODE, RX ENABL, TX DSABL, SYNC2
015600  013737  001240  023676         MOV     SYNCX,DATA      ;PLACE SYNC 2 IN  DATA
```

```
2333  015606  104416                          DATACLK                 ;INIT DATA CLOCK.
2334  015610  004537  023402                  PERFORM ,RXSHIFT        ;SHIFT DATA INTO RX
2335  015614  001242                          CLKX                    ;NUMBER OF SHIFTS NEEDED
2336  015616  012777  076400  163554          MOV     #BRB+BIT11+BIT10+BIT8,@DVSFR
2337                                                                  ;SET BR 'B' AND MATCH DET.
2338  015624  017704  163540                  MOV     @DVLCR,R4       ;SAVE LPR IN R4
2339  015630  010405                          MOV     R4,R5           ;SET FOR COMPARE
2340  015632  052705  000001                  BIS     #BIT0,R5        ;BR 'A' FALSE
2341  015636  042705  000002                  BIC     #BIT1,R5        ;BR 'B' TRUE
2342  015642  020504                          CMP     R5,R4
2343  015644  001401                          BEQ     .+4             ;BR IF LPR OK.
2344  015646  104001                          HLT     1               ;EXPECT B TRUE; A FALSE
2345  015650  012777  002000  163522          MOV     #BIT10,@DVSFR   ;SET BR 'A' AND RX CHAR FLAG.
2346  015656  017704  163506                  MOV     @DVLCR,R4       ;SAVE LPR IN R4
2347  015662  010405                          MOV     R4,R5           ;SET FOR COMPARE
2348  015664  032737  010000  001236          BIT     #TWOSYN,STAT    ;SET FOR ONE SYNC OR TWO?
2349  015672  001036                          BNE     4$              ;BR IF SET FOR ONE SYNC
2350  015674  052705  000003                  BIS     #BIT1+BIT0,R5
2351  015700  020504                          CMP     R5,R4
2352  015702  001401                          BEQ     .+4
2353  015704  104001                          HLT     1
2354  015706  013737  001240  023676          MOV     SYNCX,DATA
2355  015714  004537  023402                  PERFORM ,RXSHIFT
2356  015720  001242                          CLKX
2357  015722  012777  076400  163450          MOV     #BRB+BIT11+BIT10+BIT8,@DVSFR
2358                                                                  ;SET BR 'B' AND MATCH DET.
2359  015730  017704  163434                  MOV     @DVLCR,R4       ;SAVE LPR IN R4
2360  015734  010405                          MOV     R4,R5           ;SET FOR COMPARE
2361  015736  052705  000001                  BIS     #BIT0,R5        ;BR 'A' FALSE
2362  015742  042705  000002                  BIC     #BIT1,R5        ;BR 'B' TRUE
2363  015746  020504                          CMP     R5,R4
2364  015750  001401                          BEQ     .+4             ;BR IF LPR OK.
2365  015752  104001                          HLT     1               ;EXPECT B TRUE; A FALSE
2366  015754  012777  002000  163416          MOV     #BIT10,@DVSFR   ;SET BR 'A' AND RX CHAR FLAG.
2367  015762  017704  163402                  MOV     @DVLCR,R4       ;SAVE LPR IN R4
2368  015766  010405                          MOV     R4,R5           ;SET FOR COMPARE
2369  015770  052705  000002          4$:     BIS     #BIT1,R5
2370  015774  042705  000001                  BIC     #BIT0,R5
2371  016000  020504                          CMP     R5,R4
2372  016002  001401                          BEQ     .+4
2373  016004  104001                          HLT     1
2374  016006  005237  015564                  INC     65$             ;UPDATE LINE NUMBER
2375  016012  005303                          DEC     R3
2376  016014  001257                          BNE     1$
2377  016016  000207                          RTS     PC
2378
2379
2380                                  ;********************** TEST 20 ********************************
2381                                  ;*TEST TO FORCE RECEIVER OVERRUN.
2382                                  ;*THIS TEST WILL PUSH INTO THE RECEIVER
2383                                  ;*TWO FULL CHARS (SYNCS) AND ONE MORE CHAR MINUS
2384                                  ;*ONE BIT. THE PROGRAM WILL VERIFY NO OVERRUN EXISTS
2385                                  ;*THEN THE LAST BITS WILL BE PUSHED IN VERIFING
2386                                  ;*THAT THE OVERRUN WAS GENERATED.
2387                                  ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
2388                                  ;:*************************************************************************
```

```
2389
2390                                    ;  TEST 20
2391                                    ;--------------
2392  016020  012737  000020  001226    TST20:  MOV     #20,TSTNO
2393  016026  012737  016652  001216            MOV     #TST21,NEXT
2394  016034  012700  000000                    MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2395  016040  113737  001416  001242            MOVB    CLK.A,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
2396  016046  013737  001406  001244            MOV     MASK.A,MASKX    ;PLACE 'MASK'FOR CHARS INTO MASKX
2397  016054  013737  001422  001236            MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2398  016062  100402                            BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2399  016064  004737  016216                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2400  016070  012700  000004            100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2401  016074  113737  001417  001242            MOVB    CLK.B,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
2402  016102  013737  001410  001244            MOV     MASK.B,MASKX    ;GET MASK
2403  016110  013737  001424  001236            MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2404  016116  100402                            BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2405  016120  004737  016216                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2406  016124  012700  000010            101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2407  016130  113737  001420  001242            MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
2408  016136  013737  001412  001244            MOV     MASK.C,MASKX    ;GET MASK
2409  016144  013737  001426  001236            MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2410  016152  100402                            BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2411  016154  004737  016216                    JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2412  016160  012700  000014            102$:   MOV     #12.,R0         ;LOAD LINE NO.
2413  016164  113737  001421  001242            MOVB    CLK.D,CLKX      ;GET SHIFTS
2414  016172  013737  001414  001244            MOV     MASK.D,MASKX    ;GET MASK
2415  016200  013737  001430  001236            MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2416  016206  100402                            BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2417  016210  004737  016216                    JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2418  016214  104400                    103$:   SCOPE                   ;SCOPE THIS TEST.
2419  016216                            105$:                           ;TEST ENTRANCE.
2420  016216  032737  004000  001236            BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
2421  016224  001401                            BEQ     .+4             ;BR IF SYNC LINE CARD.
2422  016226  000207                            RTS     PC              ;EXIT TEST
2423  016230  012703  000004                    MOV     #4,R3           ;SET FOR 4 LINE GROUP
2424  016234  010037  016250                    MOV     R0,65$          ;SET LINE POINTER
2425  016240  104412                    1$:     MSTCLR                  ;RESET DV11
2426  016242  005001                            CLR     R1              ;ZERO MSCANNER POINTER
2427  016244  004537  023544                    PERFORM ,SETSCAN         ;ADJUST MSCANNER
2428  016250  000001                    65$:    .BLKW 1                 ;LINE POINTER
2429  016252  010077  163114            3$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
2430  016256  012777  125000  163104            MOV     #BIT15+BIT13+BIT11+BIT9,@DVLCR
2431  016264  004737  023462                    JSR     PC,CKBIT15
2432  016270  113737  001236  023676            MOVB    STAT,DATA       ;GET SYNC CHAR
2433  016276  104416                            DATACLK                 ;INIT DV11 BY ONE CLOCK
2434  016300  113737  001242  016650            MOVB    CLKX,10$        ;GET NUMBER OF SHIFTS PER CHAR.
2435  016306  004537  023402                    PERFORM ,RXSHIFT        ;CLOCK RX
2436  016312  016650                            10$                     ;NUMBER OF SHIFTS
2437  016314  113737  001236  023676            MOVB    STAT,DATA       ;GET ANOTHER SYNC
2438  016322  004537  023402                    PERFORM ,RXSHIFT        ;SHIFT RX
2439  016326  016650                            10$                     ;NUMBER OF SHIFTS
2440  016330  113737  001236  023676            MOVB    STAT,DATA       ;SYNC CHAR
2441  016336  162737  000001  016650            SUB     #1,10$          ;SET NUMBER OF SHIFTS -1
2442  016344  004537  023402                    PERFORM ,RXSHIFT        ;SHIFT RX
2443  016350  016650                            10$                     ;SHIFTS
2444  016352  012777  050023  163020            MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
```

```
2445  016360  104415                          ROMCLK                          ;S/C "SET RECV DATA ENABLE"
2446  016362  012777  050021  163010          MOV     #S.C+BIT4+BIT0,@DVSFR
2447  016370  104415                          ROMCLK                          ;SET/CLEAR SILO IN
2448  016372  012777  001400  163000          MOV     #BIT9+BIT8,@DVSFR
2449  016400  032777  000001  162762  4$:     BIT     #BIT0,@DVLCR            ;RCV CHAR WAITING??
2450  016406  001374                          BNE     4$                      ;BR IF YES
2451  016410  012702  030306                  MOV     #XFR+BIT7+BIT6+BIT2+BIT1,R2
2452  016414  010277  162760                  MOV     R2,@DVSFR               ;XFR RIC_SILO OUT
2453  016420  104415                          ROMCLK                          ;DATA/XFER  RICR_SILO OUT
2454  016422  017704  162740                  MOV     @DVRIC,R4               ;READ DVRIC REG
2455  016426  010405                          MOV     R4,R5                   ;
2456  016430  042705  020000                  BIC     #BIT13,R5               ;
2457  016434  020504                          CMP     R5,R4                   ;OVERRUN??
2458  016436  001401                          BEQ     .+4                     ;BR IF NO
2459  016440  104001                          HLT     1                       ;OVERRUN OCCURED TO SOON.
2460  016442  004537  023532                  PERFORM ,SILO.OUT               ;SILO OUT
2461  016446  113737  001236  023676          MOVB    STAT,DATA
2462  016454  113704  001242                  MOVB    CLKX,R4
2463  016460  005304                          DEC     R4
2464  016462  000241                  66$:    CLC
2465  016464  106037  023676                  RORB    DATA
2466  016470  105304                          DECB    R4
2467  016472  001373                          BNE     66$
2468  016474  012737  000001  016650          MOV     #1,10$
2469  016502  004537  023402                  PERFORM ,RXSHIFT
2470  016506  016650                          10$
2471  016510  012777  050021  162662          MOV     #S.C+BIT4+BIT0,@DVSFR
2472  016516  104415                          ROMCLK                          ;SET/CLEAR SILO IN
2473  016520  012777  001400  162652          MOV     #BIT9+BIT8,@DVSFR
2474  016526  032777  000001  162634  5$:     BIT     #BIT0,@DVLCR            ;RECV CHAR WAITING
2475  016534  001374                          BNE     5$                      ;
2476  016536  010005                          MOV     R0,R5                   ;GET LINE NUMBER
2477  016540  000305                          SWAB    R5                      ;PUT LINE NUMBER INTO HIGH BYTE
2478  016542  153705  001236                  BISB    STAT,R5                 ;PLACE SYNC INTO EXPECTED
2479                                  ;;*****************************************************************
2480  016546  032737  020000  001236          BIT     #BIT13,STAT             ;IS PARITY EVEN ?                ;;++C
2481  016554  001402                          BEQ     7$                      ;IF NO BR. DO NOT MASK PARITY BIT
2482                                                                          ;WHEN PARITY EVEN BECAUSE IT MASKS
2483                                                                          ;THE SYNC CHARACTER MAKING RECEIVED
2484                                                                          ;SYNC ODD, CAUSING DATA COMPARE
2485                                                                          ; ERRORS.
2486                                  ;;*****************************************************************
2487  016556  143705  001244                  BICB    MASKX,R5                ;CLEAR UNUSED BITS.
2488  016562  052705  020000          7$:     BIS     #BIT13,R5               ;SET OVERRUN
2489  016566  012702  030306                  MOV     #XFR+BIT7+BIT6+BIT2+BIT1,R2
2490  016572  010277  162602                  MOV     R2,@DVSFR
2491  016576  104415                          ROMCLK                          ;DATA/XFER  RICR_SILO OUT
2492  016600  017704  162562                  MOV     @DVRIC,R4               ;READ DVRIC
2493  016604  032737  040000  001236          BIT     #PARBIT,STAT            ;PARITY?
2494  016612  001402                          BEQ     6$                      ;BR IF NO
2495  016614  042704  010000                  BIC     #BIT12,R4               ;CLEAR PARITY ERROR IF IT EXISTS
2496  016620  020504                  6$:     CMP     R5,R4                   ;OVERRUN SET?
2497  016622  001401                          BEQ     .+4                     ;BR IF YES
2498  016624  104001                          HLT     1                       ;LINE,CHAR,AND OVERRUN EXPECTED.
2499  016626  004537  023532                  PERFORM ,SILO.OUT               ;SILO OUT
2500  016632  005237  016250                  INC     65$                     ;UPDATE LINE POINTER
```

```
2501  016636  005303                    DEC     R3              ;4 LINE GROUP DONE?
2502  016640  001402                    BEQ     11$             ;BR IF YES
2503  016642  000137  016240            JMP     1$              ;IN NOT CONTINUE
2504  016646  000207              11$:  RTS     PC              ;RETURN FOR NEXT 4 LINE GROUP
2505  016650  000001              10$:  .BLKW 1
2506
2507
2508                                    ;************************* TEST 21 *********************************
2509                                    ;*TEST OF RECEIVER DATA .
2510                                    ;*THIS TEST RUNS A BINARY COUNT PATTERN THROUGH
2511                                    ;*THE RECEIVER OF EACH LINE
2512                                    ;*THROUGH THE USE OF MAINT. DATA BIT.
2513                                    ;*THE TX IS NEVER ENABLED.
2514                                    ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
2515                                    ;:*****************************************************************
2516
2517                              ;  TEST 21
2518                              ;--------------
2519  016652  012737  000021  001226  TST21:  MOV     #21,TSTNO
2520  016660  012737  017424  001216          MOV     #TST22,NEXT
2521  016666  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2522  016672  113737  C01416  001242          MOVB    CLK.A,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
2523  016700  013737  001406  001244          MOV     MASK.A,MASKX    ;PLACE 'MASK'FOR CHARS INTO MASKX
2524  016706  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2525  016714  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2526  016716  004737  017050                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2527  016722  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2528  016726  113737  001417  001242          MOVB    CLK.B,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
2529  016734  013737  001410  001244          MOV     MASK.B,MASKX    ;GET MASK
2530  016742  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2531  016750  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2532  016752  004737  017050                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2533  016756  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2534  016762  113737  001420  001242          MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
2535  016770  013737  001412  001244          MOV     MASK.C,MASKX    ;GET MASK
2536  016776  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2537  017004  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2538  017006  004737  017050                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2539  017012  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
2540  017016  113737  001421  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
2541  017024  013737  001414  001244          MOV     MASK.D,MASKX    ;GET MASK
2542  017032  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2543  017040  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2544  017042  004737  017050                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2545  017046  104400              103$:   SCOPE                   ;SCOPE THIS TEST.
2546  017050                      105$:                           ;TEST ENTRANCE.
2547  017050  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
2548  017056  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
2549  017060  000207                          RTS     PC              ;EXIT TEST
2550  017062  012703  000004                  MOV     #4,R3           ;SET FOR 4 LINE GROUP.
2551  017066  010037  017102                  MOV     R0,65$          ;PLACE LINE POINTER
2552  017072  104412              1$:     MSTCLR                  ;CLEAR THE DV11
2553  017074  005001                          CLR     R1              ;ZERO MSCANNER POINTER
2554  017076  004537  023544                  PERFORM ,SETSCAN         ;SET SCANNER
2555  017102  000001              65$:    .BLKW 1                 ;POSITION MSCAN TO LINE NO.
2556  017104  010077  162262      3$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
```

```
2557  017110  012777  125000  162252          MOV     #BIT15+BIT13+BIT11+BIT9,@DVLCR
2558  017116  004737  023462                  JSR     PC,CKBIT15      ;GO WAIT FOR BIT15 TO=0
2559  017122  113737  001236  023676          MOVB    STAT,DATA       ;LOAD SYNC CHAR
2560  017130  104416                          DATACLK                 ;GIVE AN INITIAL CLOCK
2561  017132  004537  023402                  PERFORM ,RXSHIFT        ;STROBE CHAR INTO RX.
2562  017136  001242                          CLKX                    ;PICK UP NO. OF CLOCKS.
2563  017140  032737  010000  001236          BIT     #TWOSYN,STAT    ;TWO SYNCS REQUIRED??
2564  017146  001006                          BNE     4$              ;BR IF ONLY ONE SYNC..
2565  017150  113737  001236  023676          MOVB    STAT,DATA       ;GIVE ANOTHER SYNC TO THE RX
2566  017156  004537  023402                  PERFORM ,RXSHIFT        ;STROBE IT IN
2567  017162  001242                          CLKX                    ;SHIFTS REQUIRED
2568  017164  010005                  4$:     MOV     R0,R5           ;LOAD LINE NUMBER INTO 'EXPECTED''
2569  017166  000305                          SWAB    R5              ;PLACE IT INTO HIGH BYTE
2570  017170  105005                          CLRB    R5              ;ZERO LOW BYTE
2571  017172  012737  017246  001220          MOV     #5$,LOCK        ;SET IF SW09=1; GOTO 5$
2572  017200  012777  050023  162172          MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
2573  017206  104415                          ROMCLK                  ;CLOCK 'DATA ENABLE''
2574  017210  004537  023510                  PERFORM ,SILO.IN        ;READ RX BUFFER INTO SILO
2575  017214  005002                          CLR     R2              ;SET FOR DELAY
2576  017216  012777  001400  162154          MOV     #BIT9+BIT8,@DVSFR
2577  017224  032777  000001  162136  10$:    BIT     #BIT0,@DVLCR    ;IS 'RX CHAR WAITING'' TRUE?
2578  017232  001403                          BEQ     9$              ;BR IF TRUE..
2579  017234  005202                          INC     R2              ;DELAY.....
2580  017236  001372                          BNE     10$             ;BR IF DELAY NOTDONE
2581  017240  104000                          HLT     0               ;RX CHAR WAITING NOT TRUE!
2582  017242  004537  023532          9$:     PERFORM ,SILO.OUT       ;REMOVE CHAR FROM SILO
2583  017246  010537  023676          5$:     MOV     R5,DATA         ;PLACE CHAR INTO SOFTWARE LOC.
2584  017252  105037  023677                  CLRB    DATA+1          ;ZERO LINE NUMBER.
2585  017256  004537  023402                  PERFORM ,RXSHIFT        ;PLACE CHAR INTO RX BUFFER.
2586  017262  001242                          CLKX                    ;CLOCKS.
2587  017264  012777  050023  162106          MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
2588  017272  104415                          ROMCLK                  ;SET RX DATA ENABLE
2589  017274  004537  023510                  PERFORM ,SILO.IN        ;READ FROM RX BUFFER INTO SILO
2590  017300  005002                          CLR     R2              ;SET DELAY
2591  017302  012777  001400  162070          MOV     #BIT9+BIT8,@DVSFR
2592  017310  032777  000001  162052  6$:     BIT     #BIT0,@DVLCR    ;WAIT FOR RX CHAR WAITING
2593  017316  001403                          BEQ     7$              ;BR IF TRUE
2594  017320  005202                          INC     R2              ;UPDATE DELAY
2595  017322  001372                          BNE     6$              ;GOBACK
2596  017324  104000                          HLT     0               ;RX CHAR WAITING NOT TRUE
2597  017326  012702  030306          7$:     MOV     #XFR+BIT7+BIT6+BIT2+BIT1,R2
2598  017332  010277  162042                  MOV     R2,@DVSFR       ;DO DATA XFER FROM SILO TO DVRIC
2599  017336  104415                          ROMCLK                  ;CLOCK
2600  017340  017704  162022                  MOV     @DVRIC,R4       ;LOAD DVRIC TO 'FOUND'' LOC.
2601  017344  032737  040000  001236          BIT     #PARBIT,STAT    ;PARITY ON??
2602  017352  001402                          BEQ     16$             ;BR IF PARITY NOT ON.
2603  017354  042704  010000                  BIC     #BIT12,R4       ;CLEAR PARITY ERROR (DON'T WORRY ABOUT PARITY NOW!)
2604  017360                          16$:
2605  017360  020504                          CMP     R5,R4           ;RX DATA AND LINE NUMBER OK??
2606  017362  001401                          BEQ     .+4             ;BR IF EXPECTED =FOUND.
2607  017364  104002                          HLT     2               ;RX DATA ERROR
2608  017366  004537  023532                  PERFORM ,SILO.OUT       ;REMOVE RX DATA FROM SILO
2609  017372  104401                          SCOP1                   ;SW09=1?
2610  017374  105205                          INCB    R5              ;UPDATE DATA
2611  017376  001403                          BEQ     8$              ;BR IF ALL DATA DONE
2612  017400  133705  001244                  BITB    MASKX,R5        ;IF <8BITS CHECK END OF DATA.
```

```
 2613  017404  001720                    BEQ     5$              ;BR IF MORE TO GO
 2614  017406  005237  017102    8$:     INC     65$             ;UPDATE TO NEXT LINE.
 2615  017412  005303                    DEC     R3              ;ALL 4 LINES DONE?
 2616  017414  001402                    BEQ     22$
 2617  017416  000137  017072            JMP     1$              ;BR IF NOT ALL DONE
 2618  017422  000207            22$:    RTS     PC              ;SCOPE THIS TEST
 2619
 2620
 2621
 2622
 2623                                    ;************************ TEST 22 *********************************
 2624                                    ;*TEST OF RECEIVER PARITY LOGIC.
 2625                                    ;*THIS TEST RUNS PREDETERMINED DATA PATTERNS
 2626                                    ;*THROUGH THE RECEIVER OF EACH LINE, BY
 2627                                    ;*MEANS OF THE MAINTENACE DATA BIT. IF ODD
 2628                                    ;*PARITY IS SELECTED, AN EVEN DATA PATTERN
 2629                                    ;*IS GENERATED THROUGH THE RECEIVER WITH
 2630                                    ;*THE PARITY BIT CLEAR. THIS SHOULD CAUSE A
 2631                                    ;*RECEIVER PARITY ERROR. IF NOT, THEN WE CAN
 2632                                    ;*ASSUME THE PARITY CHECKING LOGIC IN THE
 2633                                    ;*RECEIVER IS DEFECTIVE. DATA IS STILL
 2634                                    ;*CHECKED TO INSURE INTEGERITY. EVEN PARITY
 2635                                    ;*WILL LIKEWISE BE TESTED BY GENERATING
 2636                                    ;*AN ODD DATA PATTERN. ALL CHARACTER LENGTHS
 2637                                    ;*MAY BE TESTED. THE TX IS NEVER ENABLED.
 2638                                    ;*THIS TEST WILL DE DONE FOR SYNC LINE CARDS ONLY.
 2639                                    ;:*************************************************************
 2640
 2641                            ;  TEST 22
 2642                            ;--------------
 2643  017424  012737  000022  001226  TST22:  MOV     #22,TSTNO
 2644  017432  012737  020336  001216          MOV     #TST23,NEXT
 2645  017440  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
 2646  017444  113737  001416  001242          MOVB    CLK.A,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
 2647  017452  013737  001406  001244          MOV     MASK.A,MASKX    ;PLACE 'MASK'FOR CHARS INTO MASKX
 2648  017460  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2649  017466  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 2650  017470  004737  017622                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 2651  017474  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
 2652  017500  113737  001417  001242          MOVB    CLK.B,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
 2653  017506  013737  001410  001244          MOV     MASK.B,MASKX    ;GET MASK
 2654  017514  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2655  017522  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
 2656  017524  004737  017622                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
 2657  017530  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
 2658  017534  113737  001420  001242          MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
 2659  017542  013737  001412  001244          MOV     MASK.C,MASKX    ;GET MASK
 2660  017550  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2661  017556  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
 2662  017560  004737  017622                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
 2663  017564  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
 2664  017570  113737  001421  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
 2665  017576  013737  001414  001244          MOV     MASK.D,MASKX    ;GET MASK
 2666  017604  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 2667  017612  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 2668  017614  004737  017622                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
```

```
2669  017620  104400                    103$:   SCOPE                   ;SCOPE THIS TEST.
2670  017622                            105$:                           ;TEST ENTRANCE.
2671                                    ;;*****************************************************************
2672  017622  032737  040000  001236            BIT     #PARBIT,STAT    ;IS PARITY ENABLED?          ;;++C
2673  017630  001404                            BEQ     23$             ;IF NO BR
2674                                    ;;*****************************************************************
2675  017632  032737  004000  001236            BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
2676  017640  001401                            BEQ     .+4             ;BR IF SYNC LINE CARD.
2677  017642                            23$:
2678  017642  000207                            RTS     PC              ;EXIT TEST
2679  017644  012703  000004                    MOV     #4,R3           ;SET FOR 4 LINE GROUP.
2680  017650  010037  017664                    MOV     R0,65$          ;PLACE LINE POINTER
2681  017654  104412                    1$:     MSTCLR                  ;CLEAR THE DV11
2682  017656  005001                            CLR     R1              ;ZERO MSCANNER POINTER
2683  017660  004537  023544                    PERFORM ,SETSCAN         ;SET SCANNER
2684  017664  000001                    65$:    .BLKW 1                 ;POSITION MSCAN TO LINE NO.
2685  017666  010077  161500            3$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
2686  017672  012777  125000  161470            MOV     #BIT15+BIT13+BIT11+BIT9,@DVLCR
2687  017700  004737  023462                    JSR     PC,CKBIT15      ;GO WAIT FOR BIT15 TO=0
2688  017704  113737  001236  023676            MOVB    STAT,DATA       ;LOAD SYNC CHAR
2689  017712  104416                            DATACLK                 ;GIVE AN INITIAL CLOCK
2690  017714  004537  023402                    PERFORM ,RXSHIFT         ;STROBE CHAR INTO RX.
2691  017720  001242                            CLKX                    ;PICK UP NO. OF CLOCKS.
2692  017722  032737  010000  001236            BIT     #TWOSYN,STAT    ;TWO SYNCS REQUIRED??
2693  017730  001006                            BNE     4$              ;BR IF ONLY ONE SYNC..
2694  017732  113737  001236  023676            MOVB    STAT,DATA       ;GIVE ANOTHER SYNC TO THE RX
2695  017740  004537  023402                    PERFORM ,RXSHIFT         ;STROBE IT IN
2696  017744  001242                            CLKX                    ;SHIFTS REQUIRED
2697  017746  010005                    4$:     MOV     R0,R5           ;LOAD LINE NUMBER INTO 'EXPECTED''
2698  017750  000305                            SWAB    R5              ;PLACE IT INTO HIGH BYTE
2699  017752  105005                            CLRB    R5              ;ZERO LOW BYTE
2700  017754  012737  020056  001220            MOV     #5$,LOCK        ;SET IF SW09=1; GOTO 5$
2701  017762  012777  050023  161410            MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
2702  017770  104415                            ROMCLK                  ;CLOCK 'DATA ENABLE''
2703  017772  004537  023510                    PERFORM ,SILO.IN         ;READ RX BUFFER INTO SILO
2704  017776  005002                            CLR     R2              ;SET FOR DELAY
2705  020000  012777  001400  161372            MOV     #BIT9+BIT8,@DVSFR
2706  020006  032777  000001  161354    10$:    BIT     #BIT0,@DVLCR    ;IS 'RX CHAR WAITING'' TRUE?
2707  020014  001403                            BEQ     9$              ;BR IF TRUE..
2708  020016  005202                            INC     R2              ;DELAY.....
2709  020020  001372                            BNE     10$             ;BR IF DELAY NOTDONE
2710  020022  104000                            HLT     0               ;RX CHAR WAITING NOT TRUE!
2711                                    ;;*****************************************************************
2712  020024  032737  020000  001236    9$:     BIT     #BIT13,STAT     ;IS PARITY EVEN SET?          ;;++C
2713  020032  001404                            BEQ     20$             ;IF NO,BR
2714  020034  012737  020262  001254            MOV     #ODDCH,TEMP4    ;SINCE PARITY IS EVEN, LOAD ODD CHARACTER
2715                                            ;TABLE ADR IN R5 WITH PARITY BIT CLEAR
2716  020042  000403                            BR      21$             ;GO LOAD DATA
2717  020044  012737  020310  001254    20$:    MOV     #EVENCH,TEMP4   ;SINCE PARITY IS ODD, LOAD EVEN CHARACTER
2718                                            ;TABLE WITH PARITY BIT CLEAR
2719  020052  004537  023532            21$:    PERFORM ,SILO.OUT        ;REMOVE CHAR FROM SILO
2720  020056  017737  161172  023676    5$:     MOV     @TEMP4,DATA     ;LOAD DATA
2721  020064  010537  001256                    MOV     R5,TEMP5        ;SAVE LINE #
2722  020070  117737  161160  001256            MOVB    @TEMP4,TEMP5    ;SAVE DATA
2723  020076  013705  001256                    MOV     TEMP5,R5        ;UPDATE EXPECTED DATA AND LINE #
2724                                    ;;*****************************************************************
```

```
2725  020102  004537  023402          PERFORM ,RXSHIFT         ;PLACE CHAR INTO RX BUFFER.
2726  020106  001242                  CLKX                     ;CLOCKS.
2727  020110  012777  050023  161262  MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
2728  020116  104415                  ROMCLK                   ;SET RX DATA ENABLE
2729  020120  004537  023510          PERFORM ,SILO.IN         ;READ FROM RX BUFFER INTO SILO
2730  020124  005002                  CLR     R2               ;SET DELAY
2731  020126  012777  001400  161244  MOV     #BIT9+BIT8,@DVSFR
2732  020134  032777  000001  161226  6$:     BIT     #BIT0,@DVLCR     ;WAIT FOR RX CHAR WAITING
2733  020142  001403                  BEQ     7$               ;BR IF TRUE
2734  020144  005202                  INC     R2               ;UPDATE DELAY
2735  020146  001372                  BNE     6$               ;GOBACK
2736  020150  104000                  HLT     0                ;RX CHAR WAITING NOT TRUE
2737  020152  012702  030306          7$:     MOV     #XFR+BIT7+BIT6+BIT2+BIT1,R2
2738  020156  010277  161216          MOV     R2,@DVSFR        ;DO DATA XFER FROM SILO TO DVRIC
2739  020162  104415                  ROMCLK                   ;CLOCK
2740  020164  017704  161176          MOV     @DVRIC,R4        ;LOAD DVRIC TO "FOUND" LOC.
2741                                  ;;***************************************************************
2742  020170  032704  010000          BIT     #BIT12,R4        ;IS PARITY ERROR SET?                    ;;++C
2743  020174  001002                  BNE     17$              ;IF YES BR.
2744  020176  104004                  HLT     4                ;RECEIVER PARITY ERROR NOT DETECTED
2745  020200  000402                  BR      16$
2746  020202  042704  010000          17$:    BIC     #BIT12,R4        ;CLEAR PARITY ERROR INDICATOR
2747                                  ;;***************************************************************
2748  020206                          16$:
2749  020206  020504                  CMP     R5,R4            ;RX DATA AND LINE NUMBER OK??
2750  020210  001401                  BEQ     .+4              ;BR IF EXPECTED =FOUND.
2751  020212  104002                  HLT     2                ;RX DATA ERROR
2752  020214  004537  023532          PERFORM ,SILO.OUT        ;REMOVE RX DATA FROM SILO
2753  020220  104401                  SCOP1                    ;SW09=1?
2754                                  ;;***************************************************************
2755  020222  062737  000002  001254          ADD     #2,TEMP4         ;UPDATE POINTER TO DATA          ;;++C
2756  020230  005777  161020                  TST     @TEMP4           ;END OF DATA?
2757                                  ;;***************************************************************
2758  020234  001403                  BEQ     8$               ;BR IF ALL DATA DONE
2759  020236  133705  001244          BITB    MASKX,R5         ;IF <8BITS CHECK END OF DATA.
2760  020242  001705                  BEQ     5$               ;BR IF MORE TO GO
2761  020244  005237  017664          8$:     INC     65$              ;UPDATE TO NEXT LINE.
2762  020250  005303                  DEC     R3               ;ALL 4 LINES DONE?
2763  020252  001402                  BEQ     22$
2764  020254  000137  017654          JMP     1$               ;BR IF NOT ALL DONE
2765  020260  000207                  22$:    RTS     PC               ;SCOPE THIS TEST
2766
2767
2768  020262  000001  000002  000004  ODDCH:  .WORD   1,2,4,10,13,16,20,31,32,34,0
2769  020270  000010  000013  000016
2770  020276  000020  000031  000032
2771  020304  000034  000000
2772  020310  000003  000006  000011  EVENCH: .WORD   3,6,11,12,14,21,22,24,30,33,0
2773  020316  000012  000014  000021
2774  020324  000022  000024  000030
2775  020332  000033  000000
2776
2777
2778                                  ;************************* TEST 23 *********************************
2779                                  ;*TEST OF RECEIVER DATA .
2780                                  ;*THIS TEST RUNS A SET PATTERN THROUGH
```

```
2781                                    ;*THE RECEIVER OF EACH LINE
2782                                    ;*THROUGH THE USE OF THE TRANSMITTER.
2783                                    ;*THIS TEST EXERCISES ALL LINES IN GROUPS OF 4.
2784                                    ;*NOTE: SHOULD A DATA COMPARE ERROR OCCUR; THE PROGRAM
2785                                    ;*      REPORTS THE ERROR AS A RECEIVER DATA ERROR BASED
2786                                    ;*      ON THE TRANSMITTER HAS PREVIOUSLY BEEN CHECKED AND ASSUMED GOOD.
2787                                    ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
2788                                    ;:**********************************************************************
2789
2790                                    ;  TEST 23
2791                                    ;--------------
2792  020336  012737  000023  001226    TST23:  MOV     #23,TSTNO
2793  020344  012737  021342  001216            MOV     #TST24,NEXT
2794  020352  012700  000000                    MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2795  020356  113737  001416  001242            MOVB    CLK.A,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
2796  020364  013737  001406  001244            MOV     MASK.A,MASKX    ;PLACE 'MASK'FOR CHARS INTO MASKX
2797  020372  013737  001422  001236            MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2798  020400  100402                            BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2799  020402  004737  020534                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2800  020406  012700  000004    100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2801  020412  113737  001417  001242            MOVB    CLK.B,CLKX      ;PLACE ''SHIFTS/PER/CHAR'' IN CLKX
2802  020420  013737  001410  001244            MOV     MASK.B,MASKX    ;GET MASK
2803  020426  013737  001424  001236            MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2804  020434  100402                            BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2805  020436  004737  020534                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2806  020442  012700  000010    101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2807  020446  113737  001420  001242            MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
2808  020454  013737  001412  001244            MOV     MASK.C,MASKX    ;GET MASK
2809  020462  013737  001426  001236            MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2810  020470  100402                            BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2811  020472  004737  020534                    JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2812  020476  012700  000014    102$:   MOV     #12.,R0         ;LOAD LINE NO.
2813  020502  113737  001421  001242            MOVB    CLK.D,CLKX      ;GET SHIFTS
2814  020510  013737  001414  001244            MOV     MASK.D,MASKX    ;GET MASK
2815  020516  013737  001430  001236            MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2816  020524  100402                            BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2817  020526  004737  020534                    JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2818  020532  104400            103$:   SCOPE                   ;SCOPE THIS TEST.
2819  020534                    105$:                           ;TEST ENTRANCE.
2820  020534  032737  004000  001236            BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
2821  020542  001401                            BEQ     .+4             ;BR IF SYNC LINE CARD.
2822  020544  000207                            RTS     PC              ;EXIT TEST
2823  020546  010037  020644                    MOV     R0,65$          ;PLACE LINE NO.
2824  020552  005037  001250                    CLR     TEMP2
2825  020556  113704  001244                    MOVB    MASKX,R4
2826  020562  005037  001252                    CLR     TEMP3
2827  020566  110437  001252                    MOVB    R4,TEMP3
2828  020572  000241                            CLC
2829  020574  006104                            ROL     R4
2830  020576  050437  001252                    BIS     R4,TEMP3
2831  020602  000241                            CLC
2832  020604  006104                            ROL     R4
2833  020606  050437  001252                    BIS     R4,TEMP3
2834  020612  013737  001236  023646            MOV     STAT,SYNC
2835  020620  113737  001236  023647            MOVB    STAT,SYNC+1
2836  020626  012737  000004  001246            MOV     #4,TEMP1        ;SET FOR 4 LINES
```

```
2837  020634  104412                  1$:     MSTCLR                  ;RESET DV11
2838  020636  005001                          CLR     R1              ;ZERO MSCANNER POINTER
2839  020640  004537  023544                  PERFORM ,SETSCAN         ;ADJUST SCANNER FOR PROPER LINE
2840  020644  000001                  65$:    .BLKW 1                 ;
2841  020646                          3$:
2842                                                                  ;SET SOURCE SELECT
2843  020646  010077  160520          7$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
2844  020652  004537  023634                  PERFORM ,CLR.TMARK       ;CLEAR TMARK BIT.
2845  020656  004537  023342                  PERFORM ,LOAD.MODE       ;LOAD
2846  020662  024000                          BIT13+BIT11             ;MODE AND RX ENABLE
2847  020664  032737  010000  001236          BIT     #TWOSYN,STAT
2848  020672  001003                          BNE     9$
2849  020674  012703  023646                  MOV     #SYNC,R3
2850  020700  000402                          BR      10$
2851  020702  012703  023647          9$:     MOV     #SYNC+1,R3
2852  020706  111337  001250          10$:    MOVB    (R3),TEMP2
2853  020712  043737  001252  001250        BIC     TEMP3,TEMP2
2854  020720  005077  160446                  CLR     @DVSRS          ;ZERO LINE TO LINE 0
2855  020724  013777  001250  160444        MOV     TEMP2,@DVSRA    ;LOAD DATA INTO DVSRA
2856  020732  012777  020000  160440        MOV     #BIT13,@DVSFR   ;EXECUTE A 'ROM READ'' INTSTR
2857  020740  104415                          ROMCLK                  ;CLOCK.
2858  020742  012777  030260  160430        MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
2859  020750  104415                          ROMCLK                  ;DO A DATA XFER FROM RAM OUTPUT TO TX BUFFER
2860  020752  104416                          DATACLK                 ;ISSUE A MAINT CLK.
2861  020754  012737  020766  001220        MOV     #4$,LOCK        ;SET IF SW09=1 GOTO 4$
2862  020762  010005                          MOV     R0,R5
2863  020764  000305                          SWAB    R5
2864  020766  113702  001242          4$:     MOVB    CLKX,R2         ;SET REQUIRED SHIFTS
2865  020772  010077  160374                  MOV     R0,@DVSRS       ;LOAD LINE NUMBER
2866  020776  111337  001250                  MOVB    (R3),TEMP2
2867  021002  043737  001252  001250        BIC     TEMP3,TEMP2
2868  021010  105005                          CLRB    R5
2869  021012  053705  001250                  BIS     TEMP2,R5
2870  021016  104416                  5$:     DATACLK                 ;ISSUE MAINT CLK
2871  021020  005302                          DEC     R2              ;ALL SHIFTS DONE?
2872  021022  022702  000001                  CMP     #1,R2           ;IS THE BUFFER ALMOST EMPTY?
2873  021026  001033                          BNE     8$              ;BR IF NO
2874  021030  005077  160336                  CLR     @DVSRS          ;ZERO LINE NUMBER
2875  021034  032777  001000  160140        BIT     #BIT9,@SWR      ;LOCK ON DATA?
2876  021042  001001                          BNE     .+4             ;BR IF YES!!
2877  021044  005203                          INC     R3              ;UPDATE DATA POINTER.
2878  021046  111337  001250                  MOVB    (R3),TEMP2      ;STORE DATA
2879  021052  013777  001250  160316        MOV     TEMP2,@DVSRA    ;LOAD DATA INTO DVSRA
2880  021060  012777  020000  160312        MOV     #BIT13,@DVSFR   ;DO A ROM READ
2881  021066  104415                          ROMCLK                  ;CLK
2882  021070  012777  030260  160302        MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
2883  021076  104415                          ROMCLK                  ;DO A  DATA XFER TO TX BUFF
2884  021100  010077  160266                  MOV     R0,@DVSRS       ;RESELECT LINE NUMBER
2885  021104  032777  001000  160070        BIT     #BIT9,@SWR      ;LOCK ON DATA?
2886  021112  001001                          BNE     .+4             ;BR IF YES!!
2887  021114  005303                          DEC     R3              ;READJUST DATA CHAR POINTER.
2888  021116  005702                  8$:     TST     R2              ;ALL SHIFTS DONE?
2889  021120  001336                          BNE     5$              ;BR IF NO
2890  021122  022703  023646                  CMP     #SYNC,R3
2891  021126  001473                          BEQ     50$
2892  021130  022703  023647                  CMP     #SYNC+1,R3
```

```
2893  021134  001470                        BEQ     50$
2894  021136  012777  050023  160234        MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
2895  021144  104415                        ROMCLK                      ;SET RX DATA ENABLE
2896  021146  004537  023510                PERFORM ,SILO.IN            ;READ FROM RX BUFFER INTO SILO
2897  021152  005002                        CLR     R2                  ;SET DELAY
2898  021154  012777  001400  160216        MOV     #BIT9+BIT8,@DVSFR
2899  021162  032777  000001  160200  26$:  BIT     #BIT0,@DVLCR        ;WAIT FOR RX CHAR WAITING
2900  021170  001403                        BEQ     27$                 ;BR IF TRUE
2901  021172  005202                        INC     R2                  ;UPDATE DELAY
2902  021174  001372                        BNE     26$                 ;GOBACK
2903  021176  104000                        HLT     0                   ;RX CHAR WAITING NOT TRUE
2904  021200  012702  030306        27$:    MOV     #XFR+BIT7+BIT6+BIT2+BIT1,R2
2905  021204  010277  160170                MOV     R2,@DVSFR           ;DO DATA XFFR FROM SILO TO DVRIC
2906  021210  104415                        ROMCLK                      ;CLOCK
2907  021212  017704  160150                MOV     @DVRIC,R4           ;LOAD DVRIC TO 'FOUND'' LOC.
2908  021216  032737  040000  001236        BIT     #PARBIT,STAT        ;PARITY ON??
2909  021224  001410                        BEQ     36$                 ;BR IF PARITY DISABLED
2910                          ;;*************************************************************
2911  021226  032704  010000                BIT     #BIT12,R4           ;IS THERE A PARITY ERROR?                ;;++C
2912  021232  001403                        BEQ     12$                 ;IF NO BR
2913  021234  104005                        HLT     5                   ;IF YES, THEN IT IS A A BAD TRANSMITTER
2914                          ;;*************************************************************
2915  021236  042704  010000                BIC     #BIT12,R4           ;CLEAR PARITY ERROR (DON'T WORRY ABOUT PARITY NOW!)
2916                          ;;*************************************************************
2917  021242  143704  001244        12$:    BICB    MASKX,R4            ;ELSE CLEAR PARITY BIT                   ;;++C
2918                          ;;*************************************************************
2919  021246  020504        36$:    CMP     R5,R4                       ;RX DATA AND LINE NUMBER OK??
2920  021250  001401                        BEQ     .+4                 ;BR IF EXPECTED =FOUND.
2921  021252  104002                        HLT     2                   ;RX DATA ERROR
2922  021254  004537  023532                PERFORM ,SILO.OUT           ;REMOVE RX DATA FROM SILO
2923  021260  104401                        SCOP1                       ;LOCK ON DATA?
2924  021262  005203        11$:    INC     R3
2925  021264  020327  023674                CMP     R3,#ENDPAT
2926  021270  001236                        BNE     4$
2927  021272  004537  023622        6$:     PERFORM ,SET.TMARK          ;SET TMARK BIT.
2928  021276  005237  020644                INC     65$                 ;UPDATE LINE NO.
2929  021302  005337  001246                DEC     TEMP1               ;ALL LINES(4) DONE?
2930  021306  001402                        BEQ     46$
2931  021310  000137  020634                JMP     1$
2932  021314  000207        46$:    RTS     PC                                  ;SCOPE THESE 4 LINES!
2933  021316  012777  050023  160054  50$:  MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
2934  021324  104415                        ROMCLK
2935  021326  012777  050022  160044        MOV     #S.C+BIT4+BIT1,@DVSFR
2936  021334  104415                        ROMCLK
2937  021336  000137  021262                JMP     11$                                                  !
2938
2939
2940                          ;************************* TEST 24 *********************************
2941                          ;*TEST OF RECEIVER 'RE-SYNC''
2942                          ;*THIS TEST WILL SEND (BY BIT WINDOW) TWO SYNC CHARS AND
2943                          ;*THEN VERIFY THAT RX CHAR FLAG IS TRUE.
2944                          ;*THEN A 'RE-SYNC'' WILL BE ISSUED AND
2945                          ;*TWO NON-SYNC CHARS WIIL BE SENT INTO THE RX
2946                          ;*VERIFYING THAT THERE IS NO RX CHAR FLAG.
2947                          ;*NEXT TWO SYNC CHARS ARE AGAIN MOVED INTO THE RX
2948                          ;*VERIFING CHAR FLAG AND THE THE RX SOULD INDEED
```

```
                        ;*          RE SYNC!
                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
                        ;:*****************************************************************

                        ;  TEST 24
                        ;--------------
021342 012737 000024 001226  TST24:  MOV   #24,TSTNO
021350 012737 022046 001216          MOV   #TST25,NEXT
021356 012700 000000                 MOV   #0.,R0          ;PLACE LINE NUMBER INTO R0
021362 113737 001416 001242          MOVB  CLK.A,CLKX      ;PLACE 'SHIFTS/PER/CHAR'' IN CLKX
021370 013737 001422 001236          MOV   L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
021376 100402                        BMI   100$            ;BR IF LINE CARD NOT TO BE TESTED
021400 004737 021510                 JSR   PC,105$         ;GO DO THE TEST FOR LINE CARD 1
021404 012700 000004          100$:  MOV   #4.,R0          ;PLACE LINE NUMBER INTO R0
021410 113737 001417 001242          MOVB  CLK.B,CLKX      ;PLACE 'SHIFTS/PER/CHAR'' IN CLKX
021416 013737 001424 001236          MOV   L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
021424 100402                        BMI   101$            ;BR IF LINE CARD NOT TO BE TESTED
021426 004737 021510                 JSR   PC,105$         ;GO DO THE TEST FOR LINE CARD 2
021432 012700 000010          101$:  MOV   #8.,R0          ;LOAD LINE NUMBER
021436 113737 001420 001242          MOVB  CLK.C,CLKX      ;GET SHIFTS PER CHAR
021444 013737 001426 001236          MOV   L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
021452 100402                        BMI   102$            ;BR IF LINE CARD NOT TO BE TESTED
021454 004737 021510                 JSR   PC,105$         ;DO THE TEST FOR LINE CARD 3
021460 012700 000014          102$:  MOV   #12.,R0         ;LOAD LINE NO.
021464 113737 001421 001242          MOVB  CLK.D,CLKX      ;GET SHIFTS
021472 013737 001430 001236          MOV   L12.15,STAT     ;LOAD LINE CARD STATUS
021500 100402                        BMI   103$            ;BR IF LINE CARD NOT TO BE TESTED
021502 004737 021510                 JSR   PC,105$         ;DO THE TESTS FOR LINE CARD 4
021506 104400                 103$:  SCOPE                 ;SCOPE THIS TEST.
021510                        105$:                        ;TEST ENTRANCE.
021510 032737 004000 001236          BIT   #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
021516 001401                        BEQ   .+4             ;BR IF SYNC LINE CARD.
021520 000207                        RTS   PC              ;EXIT TEST
021522 012703 000004                 MOV   #4,R3           ;SET FOR 4 LINE GROUP
021526 010037 021542                 MOV   R0,68$          ;SAVE LINE NO
021532 104412                 1$:    MSTCLR                ;RESET
021534 005001                        CLR   R1              ;ZERO MSCANNER POINTER
021536 004537 023544                 PERFORM ,SETSCAN       ;SET SCANNER
021542 000001                 68$:   .BLKW 1               ;TO RIGHT LINE
021544 012737 021552 001220          MOV   #3$,LOCK        ;SET IF SW09=1
021552 010077 157614          3$:    MOV   R0,@DVSRS       ;LOAD LINE
021556 004537 023342                 PERFORM ,LOAD.MODE     ;LOAD
021562 025000                        BIT13+BIT11+BIT9      ;MODE
021564 012702 000002                 MOV   #2,R2           ;SET COUNT
021570 104416                        DATACLK               ;INIT DV11 SAT/SAR
021572 013737 001236 023676  4$:     MOV   STAT,DATA       ;GET SYNC
021600 004537 023402                 PERFORM ,RXSHIFT       ;SHIFT INTO RX
021604 001242                        CLKX                  ;CLOCKS
021606 005302                        DEC   R2              ;TWO CHARS YET
021610 001370                        BNE   4$              ;
021612 012702 002000                 MOV   #BIT10,R2       ;BRA TEST
021616 010277 157556                 MOV   R2,@DVSFR       ;
021622 017704 157542                 MOV   @DVLCR,R4       ;
021626 010405                        MOV   R4,R5           ;
021630 042705 000001                 BIC   #BIT0,R5        ;
021634 020504                        CMP   R5,R4           ;BRANCH TEST POINT BAD
```

```
3005  021636  001401                          BEQ     64$             ;
3006  021640  104001                          HLT     1               ;
3007  021642  012777  050106  157530  64$:    MOV     #S.C+BIT6+BIT2+BIT1,@DVSFR
3008  021650  104415                          ROMCLK                  ;S/C 'RESYNC PULSE''
3009  021652  010277  157522                  MOV     R2,@DVSFR       ;
3010  021656  017704  157506                  MOV     @DVLCR,R4       ;
3011  021662  010405                          MOV     R4,R5           ;
3012  021664  052705  000001                  BIS     #BIT0,R5        ;
3013  021670  020504                          CMP     R5,R4           ;
3014  021672  001401                          BEQ     65$             ;
3015  021674  104001                          HLT     1               ;RESYNC FAILED.
3016  021676  012702  000002          65$:    MOV     #2,R2           ;
3017  021702  013737  001236  023676  5$:     MOV     STAT,DATA       ;GET SYNC
3018  021710  005437  023676                  NEG     DATA            ;MAKE IT A NON-SYNC
3019  021714  004537  023402                  PERFORM ,RXSHIFT        ;SHIFT
3020  021720  001242                          CLKX                    ;INTO RX
3021  021722  005302                          DEC     R2              ;TWO DONE?
3022  021724  001366                          BNE     5$              ;
3023  021726  012702  002000                  MOV     #BIT10,R2       ;
3024  021732  010277  157442                  MOV     R2,@DVSFR       ;
3025  021736  017704  157426                  MOV     @DVLCR,R4       ;
3026  021742  010405                          MOV     R4,R5           ;
3027  021744  052705  000001                  BIS     #BIT0,R5        ;
3028  021750  020504                          CMP     R5,R4           ;
3029  021752  001401                          BEQ     66$             ;
3030  021754  104001                          HLT     1               ;
3031  021756  012702  000002          66$:    MOV     #2,R2           ;
3032  021762  013737  001236  023676  6$:     MOV     STAT,DATA       ;
3033  021770  004537  023402                  PERFORM ,RXSHIFT        ;
3034  021774  001242                          CLKX                    ;
3035  021776  005302                          DEC     R2              ;
3036  022000  001370                          BNE     6$              ;
3037  022002  012702  002000                  MOV     #BIT10,R2       ;
3038  022006  010277  157366                  MOV     R2,@DVSFR       ;
3039  022012  017704  157352                  MOV     @DVLCR,R4       ;
3040  022016  010405                          MOV     R4,R5   ;       ;
3041  022020  042705  000001                  BIC     #BIT0,R5        ;
3042  022024  020504                          CMP     R5,R4           ;
3043  022026  001401                          BEQ     67$             ;
3044  022030  104001                          HLT     1               ;
3045  022032  104401                  67$:    SCOP1                   ;
3046  022034  005237  021542                  INC     68$             ;
3047  022040  005303                          DEC     R3              ;
3048  022042  001233                          BNE     1$              ;
3049  022044  000207                          RTS     PC              ;EXIT
3050
3051
3052                                  ;************************** TEST 25 ******************************
3053                                  ;*TEST TO VERIFY THAT SETTING RECEIVER ENABLE
3054                                  ;*WILL SET RX FLAG AND MATCH DETECT.
3055                                  ;*TEST WILL ALSO VERIFY THAT CLEARING RECEIVER
3056                                  ;*ENABLE WILL CLEAR RX FLAG AND MATCH DETECT.
3057                                  ;*THIS TEST WILL BE DONE FOR ASYNC LINE CARDS ONLY.
3058
3059                                  ;:*************************************************************************
3060
```

```
3061                                    ;  TEST 25
3062                                    ;---------------
3063  022046  012737  000025  001226    TST25:  MOV     #25,TSTNO
3064  022054  012737  022436  001216            MOV     #TST26,NEXT
3065  022062  012700  000000                    MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
3066  022066  013737  001422  001236            MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
3067  022074  100402                            BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
3068  022076  004737  022164                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
3069  022102  012700  000004            100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
3070  022106  013737  001424  001236            MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
3071  022114  100402                            BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
3072  022116  004737  022164                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
3073  022122  012700  000010            101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
3074  022126  013737  001426  001236            MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
3075  022134  100402                            BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
3076  022136  004737  022164                    JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
3077  022142  012700  000014            102$:   MOV     #12.,R0         ;LOAD LINE NO.
3078  022146  013737  001430  001236            MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
3079  022154  100402                            BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
3080  022156  004737  022164                    JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
3081  022162  104400                    103$:   SCOPE                   ;SCOPE THIS TEST.
3082  022164                            105$:                           ;TEST ENTRANCE.
3083  022164  032737  004000  001236            BIT     #ASYNC,STAT     ;IS THIS AN ASYNC LINE CAR?
3084  022172  001001                            BNE     .+4             ;BR IF ASYNC.
3085  022174  000207                            RTS     PC              ;EXIT TEST
3086  022176  012703  000004                    MOV     #4,R3           ;SET TO TEST 4 LINES.
3087  022202  104412                    1$:     MSTCLR                  ;INIT DV11
3088  022204  005001                            CLR     R1              ;INIT SCANNER POINTER.
3089  022206  012777  000010  157146            MOV     #BIT3,@DVSCR    ;SET SOURCE ENABLE
3090  022214  010037  022224                    MOV     R0,65$          ;PREPARE MASTER SCANNER.
3091  022220  004537  023544                    PERFORM ,SETSCAN        ;SET SCANNER
3092  022224  000001                    65$:    .BLKW   1               ;POSITION OF SCANNER.
3093  022226  010077  157140                    MOV     R0,@DVSRS       ;LOAD LINE NO.
3094  022232  004537  023342                    PERFORM ,LOAD.MODE      ;SET RX ENABLE.
3095  022236  020000                            BIT13                   ;
3096  022240  012702  076400                    MOV     #BRB+BIT11+BIT10+BIT8,R2
3097  022244  010277  157130                    MOV     R2,@DVSFR       ;BRB MATCH DETECT.
3098  022250  017704  157114                    MOV     @DVLCR,R4       ;READ BR POINTS.
3099  022254  010405                            MOV     R4,R5           ;
3100  022256  052705  000001                    BIS     #BIT0,R5        ;BR A FALSE.
3101  022262  042705  000002                    BIC     #BIT1,R5        ;BR B TRUE.
3102  022266  020504                            CMP     R5,R4           ;MATCH DETECT TRUE?
3103  022270  001401                            BEQ     2$              ;BR IF YES
3104  022272  104001                            HLT     1               ;RX FLAG NOT TRUE.
3105  022274  012702  002000            2$:     MOV     #BIT10,R2       ;BRA RX FLAG.
3106  022300  010277  157074                    MOV     R2,@DVSFR       ;LOAD INSTRUCTION.
3107  022304  017704  157060                    MOV     @DVLCR,R4       ;READ BR POINTS.
3108  022310  010405                            MOV     R4,R5           ;
3109  022312  052705  000002                    BIS     #BIT1,R5        ;BR B FALSE
3110  022316  042705  000001                    BIC     #BIT0,R5        ;BR A TRUE.
3111  022322  020504                            CMP     R5,R4           ;RX FLAG TRUE?
3112  022324  001401                            BEQ     3$              ;BR IF YES
3113  022326  104001                            HLT     1               ;RX FLAG NOT TRUE.
3114  022330  004537  023342            3$:     PERFORM ,LOAD.MODE      ;CLEAR RX ENABLE.
3115  022334  000000                            0                       ;
3116  022336  012702  076400                    MOV     #BRB+BIT11+BIT10+BIT8,R2
```

```
3117  022342  010277  157032          MOV     R2,@DVSFR       ;BRB MATCH DETECT.
3118  022346  017704  157016          MOV     @DVLCR,R4       ;READ BR POINTS.
3119  022352  010405                  MOV     R4,R5           ;
3120  022354  052705  000001          BIS     #BIT0,R5        ;BR A FALSE.
3121  022360  052705  000002          BIS     #BIT1,R5        ;BR B FALSE.
3122  022364  020504                  CMP     R5,R4           ;MATCH DETECT FALSE?
3123  022366  001401                  BEQ     4$              ;BR IF YES
3124  022370  104001                  HLT     1               ;RX FLAG NOT FALSE.
3125  022372  012702  002000  4$:     MOV     #BIT10,R2       ;BRA RX FLAG.
3126  022376  010277  156776          MOV     R2,@DVSFR       ;LOAD INSTRUCTION.
3127  022402  017704  156762          MOV     @DVLCR,R4       ;READ BR POINTS.
3128  022406  010405                  MOV     R4,R5           ;
3129  022410  052705  000002          BIS     #BIT1,R5        ;BR B FALSE
3130  022414  052705  000001          BIS     #BIT0,R5        ;BR A FALSE.
3131  022420  020504                  CMP     R5,R4           ;RX FLAG FALSE?
3132  022422  001401                  BEQ     5$              ;BR IF YES
3133  022424  104001                  HLT     1               ;RX FLAG NOT FALSE.
3134  022426  005200  5$:     INC     R0              ;UPDATE LINE NO.
3135  022430  005303                  DEC     R3              ;4 LINES DONE?
3136  022432  001263                  BNE     1$              ;BR IF NO.
3137  022434  000207                  RTS     PC              ;EXIT TEST.
3138
3139
3140                          ;************************ TEST 26 *******************************
3141                          ;*TEST TO SET RECEIVER ENABLE.
3142                          ;*SET 'RX DATA ENABLE'.
3143                          ;*CLR 'RX DATA ENABLE'.
3144                          ;*AND EXPECT BOTH RX FLAG AND MATCH DETECT TO BE FALSE.
3145                          ;*THIS TEST WILL BE DONE FOR ASYNC LINE CARDS ONLY.
3146                          ;;*************************************************************
3147
3148                  ;  TEST 26
3149                  ;--------------
3150  022436  012737  000026  001226  TST26:  MOV     #26,TSTNO
3151  022444  012737  022750  001216          MOV     #TST27,NEXT
3152  022452  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
3153  022456  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
3154  022464  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
3155  022466  004737  022554                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
3156  022472  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
3157  022476  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
3158  022504  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
3159  022506  004737  022554                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
3160  022512  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
3161  022516  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
3162  022524  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
3163  022526  004737  022554                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
3164  022532  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
3165  022536  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
3166  022544  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
3167  022546  004737  022554                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
3168  022552  104400          103$:   SCOPE                   ;SCOPE THIS TEST.
3169  022554                  105$:                           ;TEST ENTRANCE.
3170  022554  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS AN ASYNC LINE CAR?
3171  022562  001001                          BNE     .+4             ;BR IF ASYNC.
3172  022564  000207                          RTS     PC              ;EXIT TEST
```

```
3173  022566  012703  000004              MOV     #4,R3           ;SET TO TEST 4 LINES.
3174  022572  104412              1$:     MSTCLR                  ;INIT DV11
3175  022574  005001                      CLR     R1              ;INIT SCANNER POINTER.
3176  022576  012777  000010  156556      MOV     #BIT3,@DVSCR    ;SET SOURCE ENABLE
3177  022604  010037  022614              MOV     R0,65$          ;PREPARE MASTER SCANNER.
3178  022610  004537  023544              PERFORM ,SETSCAN        ;SET SCANNER
3179  022614  000001              65$:    .BLKW   1               ;POSITION OF SCANNER.
3180  022616  010077  156550              MOV     R0,@DVSRS       ;LOAD LINE NO.
3181  022622  004537  023342              PERFORM ,LOAD.MODE      ;SET RX ENABLE.
3182  022626  020000                      BIT13                   ;
3183  022630  012777  050023  156542      MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
3184  022636  104415                      ROMCLK                  ;SET RX DATA ENABLE.
3185  022640  012777  050022  156532      MOV     #S.C+BIT4+BIT1,@DVSFR
3186  022646  104415                      ROMCLK                  ;CLEAR RX DATA ENABLE.
3187  022650  012702  076400              MOV     #BRB+BIT11+BIT10+BIT8,R2
3188  022654  010277  156520              MOV     R2,@DVSFR       ;BRB MATCH DETECT.
3189  022660  017704  156504              MOV     @DVLCR,R4       ;READ BR POINTS.
3190  022664  010405                      MOV     R4,R5           ;
3191  022666  052705  000001              BIS     #BIT0,R5        ;BR A FALSE.
3192  022672  052705  000002              BIS     #BIT1,R5        ;BR B FALSE.
3193  022676  020504                      CMP     R5,R4           ;MATCH DETECT FALSE?
3194  022700  001401                      BEQ     4$              ;BR IF YES
3195  022702  104001                      HLT     1               ;RX FLAG NOT FALSE.
3196  022704  012702  002000      4$:     MOV     #BIT10,R2       ;BRA RX FLAG.
3197  022710  010277  156464              MOV     R2,@DVSFR       ;LOAD INSTRUCTION.
3198  022714  017704  156450              MOV     @DVLCR,R4       ;READ BR POINTS.
3199  022720  010405                      MOV     R4,R5           ;
3200  022722  052705  000002              BIS     #BIT1,R5        ;BR B FALSE
3201  022726  052705  000001              BIS     #BIT0,R5        ;BR A FALSE.
3202  022732  020504                      CMP     R5,R4           ;RX FLAG FALSE?
3203  022734  001401                      BEQ     5$              ;BR IF YES
3204  022736  104001                      HLT     1               ;RX FLAG NOT FALSE.
3205  022740  005200              5$:     INC     R0              ;UPDATE LINE NO.
3206  022742  005303                      DEC     R3              ;4 LINES DONE?
3207  022744  001312                      BNE     1$              ;BR IF NO.
3208  022746  000207                      RTS     PC              ;EXIT TEST.
3209
3210
3211                                      ;************************* TEST 27 *******************************
3212                                      ;*TEST TO SET RECEIVER ENABLE.
3213                                      ;*ISSUE A RESYNC  SIGNAL.
3214                                      ;*AND EXPECT BOTH RX FLAG AND MATCH DETECT TO BE FALSE.
3215                                      ;*THIS TEST WILL B    E FOR ASYNC LINE CARDS ONLY.
3216                                      ;:*****************    ***********************************************
3217
3218                              ;  TEST 27
3219                              ;--------------
3220  022750  012737  000027  001226  TST27:  MOV     #27,TSTNO
3221  022756  012737  002436  001216          MOV     #.EOP,NEXT
3222  022764  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
3223  022770  013737  001422  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
3224  022776  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
3225  023000  004737  023066                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
3226  023004  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
3227  023010  013737  001424  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
3228  023016  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
```

```
3229  023020  004737  023066                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
3230  023024  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
3231  023030  013737  001426  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
3232  023036  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
3233  023040  004737  023066                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
3234  023044  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
3235  023050  013737  001430  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
3236  023056  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
3237  023060  004737  023066                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
3238  023064  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
3239  023066                          105$:                           ;TEST ENTRANCE.
3240  023066  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS AN ASYNC LINE CAR?
3241  023074  001001                          BNE     .+4             ;BR IF ASYNC.
3242  023076  000207                          RTS     PC              ;EXIT TEST
3243  023100  012703  000004                  MOV     #4,R3           ;SET TO TEST 4 LINES.
3244  023104  104412                  1$:     MSTCLR                  ;INIT DV11
3245  023106  005001                          CLR     R1              ;INIT SCANNER POINTER.
3246  023110  012777  000010  156244          MOV     #BIT3,@DVSCR    ;SET SOURCE ENABLE
3247  023116  010037  023126                  MOV     R0,65$          ;PREPARE MASTER SCANNER.
3248  023122  004537  023544                  PERFORM ,SETSCAN         ;SET SCANNER
3249  023126  000001                  65$:    .BLKW   1               ;POSITION OF SCANNER.
3250  023130  010077  156236                  MOV     R0,@DVSRS       ;LOAD LINE NO.
3251  023134  004537  023342                  PERFORM ,LOAD.MODE       ;SET RX ENABLE.
3252  023140  020000                          BIT13                   ;
3253  023142  012777  050106  156230          MOV     #S.C+BIT6+BIT2+BIT1,@DVSFR
3254  023150  104415                          ROMCLK                  ;ISSUE RESYNC.
3255  023152  012702  076400                  MOV     #BRB+BIT11+BIT10+BIT8,R2
3256  023156  010277  156216                  MOV     R2,@DVSFR       ;BRB MATCH DETECT.
3257  023162  017704  156202                  MOV     @DVLCR,R4       ;READ BR POINTS.
3258  023166  010405                          MOV     R4,R5           ;
3259  023170  052705  000001                  BIS     #BIT0,R5        ;BR A FALSE.
3260  023174  052705  000002                  BIS     #BIT1,R5        ;BR B FALSE.
3261  023200  020504                          CMP     R5,R4           ;MATCH DETECT FALSE?
3262  023202  001401                          BEQ     4$              ;BR IF YES
3263  023204  104001                          HLT     1               ;RX FLAG NOT FALSE.
3264  023206  012702  002000          4$:     MOV     #BIT10,R2       ;BRA RX FLAG.
3265  023212  010277  156162                  MOV     R2,@DVSFR       ;LOAD INSTRUCTION.
3266  023216  017704  156146                  MOV     @DVLCR,R4       ;READ BR POINTS.
3267  023222  010405                          MOV     R4,R5           ;
3268  023224  052705  000002                  BIS     #BIT1,R5        ;BR B FALSE
3269  023230  052705  000001                  BIS     #BIT0,R5        ;BR A FALSE.
3270  023234  020504                          CMP     R5,R4           ;RX FLAG FALSE?
3271  023236  001401                          BEQ     5$              ;BR IF YES
3272  023240  104001                          HLT     1               ;RX FLAG NOT FALSE.
3273  023242  005200                  5$:     INC     R0              ;UPDATE LINE NO.
3274  023244  005303                          DEC     R3              ;4 LINES DONE?
3275  023246  001316                          BNE     1$              ;BR IF NO.
3276  023250  000207                          RTS     PC              ;EXIT TEST.
3277
3278  023252                          TXSHIFT:
3279  023252  010046                          MOV     R0,-(SP)
3280  023254  017700  156110                  MOV     @DVLCR,R0
3281                                  ;;**************************************************************************
3282  023260  022737  000010  001242          CMP     #8.,CLKX        ;SEE IF 8 BIT CHAR                      ;;++C
3283                                                                  ;OR 7 BIT W/PARITY ENABLED
3284  023266  001404                          BEQ     1$              ;IF YES,BR
```

```
3285  023270  022737  000011  001242          CMP     #9.,CLKX        ;SEE IF I BIT CHAR W/PARTIY ENABLED
3286  023276  001404                          BEQ     2$              ;IF YES BR
3287                                  ;;*************************************************************************
3288  023300  106100                  1$:     ROLB    R0
3289  023302  106037  023676                  RORB    DATA
3290                                  ;;*************************************************************************
3291  023306  000413                          BR      4$              ;CHAR IS 8 BIT WITHOUT PARITY ENABLED          ;;++C
3292                                                                  ;OR LESS THAN 8 BITS IN LENGTH WITH
3293                                                                  ;OR WITHOUT PARITY ENABLED
3294  023310  042737  001000  023676  2$:     BIC     #BIT9,DATA      ;SINCE PARTIY IS ENABLED, ROOM
3295                                                                  ;MUST BE PROVIDED FOR INSERTION
3296                                                                  ;OF PARITY BIT ON THE END OF 8
3297                                                                  ;BIT CHARACTER
3298  023316  106100                          ROLB    R0              ;LOAD CARRY BIT WITH DATA
3299  023320  103004                          BCC     3$              ;IF CARRY IS CLEAR OR NO DATA BR
3300  023322  000241                          CLC                     ;IF CARRY SET , CLEAR IT SO DATA
3301                                                                  ;WON'T BE SHIFTED INTO HIGH BYTE
3302                                                                  ;(BIT15) OF DATA LOCATION WHEN
3303                                                                  ;SHIFTINF OF DATA TAKES PLACE.
3304  023324  052737  001000  023676          BIS     #BIT9,DATA      ;ELSE START BEGINNING OF DATA
3305                                                                  ;IN BIT POSITION 9.
3306  023332  006037  023676          3$:     ROR     DATA            ;SHIFT DATA
3307                                  ;;*************************************************************************
3308  023336  012600                  4$:     MOV     (SP)+,R0
3309  023340  000205                          EXIT
3310  023342                          LOAD.MODE:
3311  023342  012577  156022                  MOV     (R5)+,@DVLCR
3312  023346  052777  100000  156014          BIS     #BIT15,@DVLCR
3313  023354  010046                          MOV     R0,-(SP)
3314  023356  005000                          CLR     R0
3315  023360  005777  156004          1$:     TST     @DVLCR
3316  023364  100004                          BPL     2$
3317  023366  104414                          DELAY
3318  023370  005200                          INC     R0
3319  023372  001372                          BNE     1$
3320  023374  104000                          HLT     0                       ;BIT 15 FAILED TO CLEAR
3321  023376  012600                  2$:     MOV     (SP)+,R0
3322  023400  000205                          EXIT
3323  023402                          RXSHIFT:
3324  023402  010046                          MOV     R0,-(SP)
3325  023404  010246                          MOV     R2,-(SP)
3326  023406  113502                          MOVB    @(R5)+,R2
3327  023410  042777  040000  155752  1$:     BIC     #BIT14,@DVLCR
3328  023416  005000                          CLR     R0
3329  023420  000241                          CLC
3330  023422  006037  023676                  ROR     DATA
3331  023426  006000                          ROR     R0
3332  023430  006000                          ROR     R0
3333  023432  052700  100000                  BIS     #BIT15,R0
3334  023436  050077  155726                  BIS     R0,@DVLCR
3335  023442  004737  023462                  JSR     PC,CKBIT15
3336  023446  104416                          DATACLK
3337  023450  105302                          DECB    R2
3338  023452  001356                          BNE     1$
3339  023454  012602                          MOV     (SP)+,R2
3340  023456  012600                          MOV     (SP)+,R0
```

```
3341  023460  000205                          EXIT
3342
3343  023462                          CKBIT15:
3344  023462  010046                          MOV     R0,-(SP)
3345  023464  005000                          CLR     R0
3346  023466  005777  155676          64$:    TST     @DVLCR
3347  023472  100004                          BPL     65$
3348  023474  104414                          DELAY
3349  023476  005200                          INC     R0
3350  023500  001372                          BNE     64$
3351  023502  104000                          HLT     0                ;BIT 15 FAILED TO CLEAR
3352  023504  012600                  65$:    MOV     (SP)+,R0
3353  023506  000207                          RTS     PC
3354  023510                          SILO.IN:
3355  023510  012777  050021  155662          MOV     #BIT14+BIT12+BIT4+BIT0,@DVSFR
3356  023516  104415                          ROMCLK
3357  023520  012777  050022  155652          MOV     #BIT14+BIT12+BIT4+BIT1,@DVSFR
3358  023526  104415                          ROMCLK
3359  023530  000205                          EXIT
3360
3361  023532                          SILO.OUT:
3362  023532  012777  050020  155640          MOV     #BIT14+BIT12+BIT4,@DVSFR
3363  023540  104415                          ROMCLK
3364  023542  000205                          EXIT
3365
3366
3367  023544                          SETSCAN:
3368  023544  010346                          MOV     R3,-(SP)
3369  023546  052777  000010  155606          BIS     #BIT3,@DVSCR
3370  023554  012503                          MOV     (R5)+,R3
3371  023556  001414                          BEQ     2$
3372  023560  012777  050102  155612  1$:     MOV     #BIT14+BIT12+BIT6+BIT1,@DVSFR
3373  023566  104415                          ROMCLK
3374  023570  005201                          INC     R1
3375  023572  012777  050102  155600          MOV     #BIT14+BIT12+BIT6+BIT1,@DVSFR
3376  023600  104415                          ROMCLK
3377  023602  005201                          INC     R1
3378  023604  005303                          DEC     R3
3379  023606  001364                          BNE     1$
3380  023610  012603                  2$:     MOV     (SP)+,R3
3381  023612  010100                          MOV     R1,R0
3382  023614  000241                          CLC
3383  023616  006000                          ROR     R0
3384  023620  000205                          EXIT
3385  023622                          SET.TMARK:
3386  023622  012777  050105  155550          MOV     #BIT14+BIT12+BIT6+BIT2+BIT0,@DVSFR
3387  023630  104415                          ROMCLK                   ;SET/CLEAR ''SET TMARK''
3388  023632  000205                          EXIT
3389  023634                          CLR.TMARK:
3390  023634  012777  050101  155536          MOV     #BIT14+BIT12+BIT6+BIT0,@DVSFR
3391  023642  104415                          ROMCLK           ;SET/CLEAR ''CLEAR TMARK''
3392  023644  000205                          EXIT
3393
3394  023646  000001                  SYNC:   .BLKW 1
3395  023650     000          DATPAT: .BYTE   ^B<00000000>     ;ALL ZERO'S
3396  023651     377                  .BYTE   ^B<11111111>     ;ALL ONE'S
```

```
 3397  023652      125                        .BYTE   ^B<01010101>    ;ALTERNATE ONE'S
 3398  023653      252                        .BYTE   ^B<10101010>    ;ALTERNATE ZERO'S
 3399  023654      001                        .BYTE   ^B<00000001>    ;F
 3400  023655      002                        .BYTE   ^B<00000010>    ; L
 3401  023656      004                        .BYTE   ^B<00000100>    ;  O
 3402  023657      010                        .BYTE   ^B<00001000>    ;   A
 3403  023660      020                        .BYTE   ^B<00010000>    ;    T
 3404  023661      040                        .BYTE   ^B<00100000>    ;     I
 3405  023662      100                        .BYTE   ^B<01000000>    ;      N
 3406  023663      200                        .BYTE   ^B<10000000>    ;       G  ONE!
 3407  023664      177                        .BYTE   ^B<01111111>    ;F
 3408  023665      277                        .BYTE   ^B<10111111>    ; L
 3409  023666      337                        .BYTE   ^B<11011111>    ;  O
 3410  023667      357                        .BYTE   ^B<11101111>    ;   A
 3411  023670      367                        .BYTE   ^B<11110111>    ;    T
 3412  023671      373                        .BYTE   ^B<11111011>    ;     I
 3413  023672      375                        .BYTE   ^B<11111101>    ;      N
 3414  023673      376                        .BYTE   ^B<11111110>    ;       G  ZERO!
 3415  023674                             ENDPAT:
 3416  023674  000000                     NPRLOC: 0
 3417  023676  000000                     DATA:   0
 3418  023700  046377  047111  020105     EM1:    .ASCIZ  <377>/LINE CARD STATIC TEST/
       023727     377  042522  042503     EM2:    .ASCIZ  <377>/RECEIVER DATA COMAPRISON ERROR/
       023767     377  051124  047101     EM3:    .ASCIZ  <377>/TRANSMITTER DATA COMPARISON ERROR/
       024032  051377  041505  044505     EM4:    .ASCIZ  <377>/RECEIVER PARITY ERROR NOT DETECTED/
       024076  052377  040522  051516     EM5:    .ASCIZ  <377>/TRANSMITTER PARITY ERROR/
       024130  046777  052123  041523     DH1:    .ASCIZ  <377>/MSTSCAN  DVSFR  EXPECTED  FOUND  LINE(8)/
                                          .EVEN
               000000                     SKIP=000000
 3419  024202  000005                     DT6:    5
 3420  024204     006      003                    .BYTE   6,3
 3421  024206  001262                             SAVR1
 3422  024210     006      001                    .BYTE   6,1
 3423  024212  001264                             SAVR2
 3424  024214     006      004                    .BYTE   6,4
 3425  024216  001272                             SAVR5
 3426  024220     006      001                    .BYTE   6,1
 3427  024222  001270                             SAVR4
 3428  024224     002      001                    .BYTE   2,1
 3429  024226  001260                             SAVR0
 3430
 3431  024230                             .ERRTAB:
 3432  024230  000000                             0
 3433  024232  000000                             0
 3434  024234  000000                             0
 3435  024236  023700                             EM1
 3436  024240  024130                             DH1     ;HALT 1
 3437  024242  024202                             DT6
 3438  024244  023727                             EM2
 3439  024246  024130                             DH1     ;HALT 2
 3440  024250  024202                             DT6
 3441  024252  023767                             EM3
 3442  024254  024130                             DH1     ;HALT 3
 3443  024256  024202                             DT6
 3444  024260  024037                             EM4
 3445  024262  024130                             DH1     ;HALT 4
```

```
3446  024264  024202                DT6
3447  024266  024076                EM5
3448  024270  024130                DH1     ;HALT 5
3449  024272  024202                DT6
3450  024274                    CORMAX:
3451          000001            .END
```

```
ADRCNT= 003443   619* 655* 664#
ALU   = 010000    73#
ASYNC = 004000    81# 1357 1409 1461 1515 1567 1643 1783 1893 2064 2144 2215 2320
                2420 2547 2675 2820 2979 3083 3170 3240
AUTO.S  006624  1129#
BCC   = 060000    78#
BINWRD  003746   705* 706 743#
BIT0  = 000001    71# 886 1249 1314 1662 1673 2002 2006 2023 2077 2079 2083 2100
                2163 2235 2245 2256 2265 2340 2350 2361 2370 2444 2446 2449 2471
                2474 2572 2577 2587 2592 2701 2706 2727 2732 2894 2899 2933 3003
                3012 3027 3041 3100 3110 3120 3130 3183 3191 3201 3259 3269 3355
                3386 3390
BIT1  = 000002    70# 886 897 1249 1256 1265 1269 1314 1653 1793 1903 2011 2077
                2088 2164 2236 2245 2257 2264 2341 2350 2362 2369 2444 2451 2489
                2572 2587 2597 2701 2727 2737 2894 2904 2933 2935 3007 3101 3109
                3121 3129 3183 3185 3192 3200 3253 3260 3268 3357 3372 3375
BIT10 = 002000    61# 886 1250 1315 1909 2160 2231 2240 2252 2261 2331 2336 2345
                2357 2366 2999 3023 3037 3096 3105 3116 3125 3187 3196 3255 3264
BIT11 = 004000    60# 886 1315 1414 1466 1520 1575 1658 1799 1909 2155 2160 2226
                2231 2252 2331 2336 2357 2430 2557 2686 2846 2991 3096 3116 3187
                3255
BIT12 = 010000    59# 73 75 77 79 2495 2603 2742 2746 2911 2915 3355 3357
                3362 3372 3375 3386 3390
BIT13 = 020000    58# 74 75 78 79 1667 1690 1801 1911 2155 2226 2331 2430
                2456 2480 2488 2557 2686 2712 2846 2856 2880 2991 3095 3182 3252
BIT14 = 040000    57# 76 77 78 79 526 1583 1584 1590 1591 3327 3355 3357
                3362 3372 3375 3386 3390
BIT15 = 100000    56# 1471 1525 1584 1592 2430 2557 2686 3312 3333
BIT2  = 000004    69# 453 886 2011 2088 2451 2489 2597 2737 2904 3007 3253 3386
BIT3  = 000010    68# 886 3089 3176 3246 3369
BIT4  = 000020    67# 1367 1368 1420 1421 1474 1475 1528 1529 1669 1692 1803 1913
                2002 2020 2077 2079 2097 2444 2446 2471 2572 2587 2701 2727 2858
                2882 2894 2933 2935 3183 3185 3355 3357 3362
BIT5  = 000040    66# 1367 1368 1420 1421 1474 1475 1528 1529 1669 1692 1803 1913
                2858 2882
BIT6  = 000100    65# 1197 1256 1269 1653 1793 1903 2011 2088 2451 2489 2597 2737
                2904 3007 3253 3372 3375 3386 3390
BIT7  = 000200    64# 520 767 924 945 1197 1363 1416 1468 1522 1578 1583 1590
                1669 1692 1733 1803 1913 2011 2088 2451 2489 2597 2737 2858 2882
                2904
BIT8  = 000400    63# 886 903 1315 2005 2022 2082 2099 2160 2231 2252 2336 2357
                2448 2473 2576 2591 2705 2731 2898 3096 3116 3187 3255
BIT9  = 001000    62# 886 1197 1202 1262 1520 1575 1660 1672 1686 1695 2005 2022
                2082 2099 2155 2226 2331 2430 2448 2473 2557 2576 2591 2686 2705
                2731 2875 2885 2898 2991 3294 3304
BRB   = 070000    79# 1315 2160 2231 2252 2336 2357 3096 3116 3187 3255
BRW     003014   459 548#
BRX     003016   460 549#
CHRCNT  003744   703* 707 723* 741# 742
CKBIT1  023462  1472 1526 1585 1593 2431 2558 2687 3335 3343#
CLKX    001242   150# 1618* 1624* 1630* 1636* 1677 1700 1703 1706 1726 1758* 1764* 1770*
                1776* 1807 1813 1820 1823 1826 1864* 1871* 1878* 1885* 1917 1923 1930
                1933 1936 2123* 2128* 2133* 2138* 2159 2194* 2199* 2204* 2209* 2230 2251
                2295* 2301* 2307* 2313* 2335 2356 2395* 2401* 2407* 2413* 2434 2462 2522*
                2528* 2534* 2540* 2562 2567 2586 2646* 2652* 2658* 2664* 2691 2696 2726
                2795* 2801* 2807* 2813* 2864 2958* 2963* 2968* 2973* 2996 3020 3034 3282
```

```
                    3285
CLK.A   001416       250#  1047  1618  1758  1864  2123  2194  2295  2395  2522  2646  2795  2958
CLK.B   001417       251#  1052  1624  1764  1871  2128  2199  2301  2401  2528  2652  2801  2963
CLK.C   001420       252#  1057  1630  1770  1878  2133  2204  2307  2407  2534  2658  2807  2968
CLK.D   001421       253#  1062  1636  1776  1885  2138  2209  2313  2413  2540  2664  2813  2973
CLR.TM  023634      1659   1797  1907  2844  3389#
CNVRT = 104411       209#   478   480   482   484   804   806   862   919
CONVRT= 104410       207#   418   820
CORMAX  024274      3450#  3451
CREAM   001306       171#   387*  994*  995   997*  1002  1003* 1004  1007*
CSRMAP  006626       412   1131#
CYCLE   005666       462    498   499   984#
DATA    023676      1678*  1708* 1713  1814* 1828* 1833  1924* 1938* 1943  2156* 2227* 2249* 2332*
                    2354*  2432* 2437* 2440* 2461* 2465* 2559* 2565* 2583* 2584* 2688* 2694* 2720*
                    2994*  3017* 3018* 3032* 3289* 3294* 3304* 3306* 3330* 3417#
DATABP  004276       793*   796   818   821#
DATACL= 104416       219#  1671  1680  1728  1732  1805  1808  1816  1915  1918  1926  2157  2228
                    2333   2433  2560  2689  2860  2870  2993  3336
DATAHD  004264       792*   814   817#
DATPAT  023650      3395#
DELAY = 104414       215#  3317  3348
DEVADR  003440       617*   652   662#
DH1     024130      3418#  3436  3439  3442  3445  3448
DT6     024202      3419#  3437  3440  3443  3446  3449
DVACTV  001300       165#   434*  435   984   989  1163* 1169* 1170* 1174  1193
DVCR00  001500       283#
DVCR01  001524       294#
DVCR02  001550       305#
DVCR03  001574       316#
DVCR04  001620       327#
DVCR05  001644       338#
DVCR06  001670       349#
DVCR07  001714       360#
DVLCR   001370       233#   903*  904   1023* 1024* 1025  1252  1264  1317  1362  1366  1414* 1415
                    1419   1466* 1467  1471* 1473  1520* 1521  1525* 1527  1576  1584* 1586  1591*
                    1592*  1594  1662  1673  1733  2006  2023  2083  2100  2161  2233  2241  2254
                    2262   2338  2346  2359  2367  2430* 2449  2474  2557* 2577  2592  2686* 2706
                    2732   2899  3001  3010  3025  3039  3098  3107  3118  3127  3189  3198  3257
                    3266   3280  3311* 3312* 3315  3327* 3334* 3346
DVNSR   001402       238#  1033* 1034* 1035
DVNUM   001301       166#   383   488  1135* 1156* 1157  1164  1166
DVRIC   001366       232#  1021* 1022* 1023  2014  2091  2454  2492  2600  2740  2907
DVRLVL  001354       227#  1038* 1039* 1040
DVRVEC  001352       226#   505  1009* 1038
DVSCR   001362       230#   502   880* 893*  897* 1008* 1019  3089* 3176* 3246* 3369*
DVSCRH  001364       231#  1019* 1020* 1021
DVSFR   001400       237#  1031* 1032* 1033  1251* 1256* 1263* 1269* 1316* 1653* 1660* 1667* 1669*
                    1672*  1690* 1692* 1793* 1801* 1803* 1903* 1911* 1913* 2002* 2005* 2011* 2012
                    2020*  2022* 2077* 2079* 2082* 2088* 2089  2097* 2099* 2160* 2231* 2240* 2252*
                    2261*  2336* 2345* 2357* 2366* 2444* 2446* 2448* 2452* 2471* 2473* 2490* 2572*
                    2576*  2587* 2591* 2598* 2701* 2705* 2727* 2731* 2738* 2856* 2858* 2880* 2882*
                    2894*  2898* 2905* 2933* 2935* 3000* 3007* 3009* 3024* 3038* 3097* 3106* 3117*
                    3126*  3183* 3185* 3188* 3197* 3253* 3256* 3265* 3355* 3357* 3362* 3372* 3375*
                    3386*  3390*
DVSRA   001376       236#   884  1029* 1030* 1031  1666* 1689* 1800* 1910* 2855* 2879*
DVSRS   001372       234#   883  1025* 1026* 1027  1365* 1418* 1470* 1524* 1573* 1656* 1665* 1679*
```

```
                    1685*  1694*  1727*  1796*  1815*  1906*  1925*  2153*  2224*  2329*  2429*  2556*  2685*
                    2843*  2854*  2865*  2874*  2884*  2989*  3093*  3180*  3250*
DVSRSH  001374       235#  1027*  1028*  1029
DVTLVL  001360       229#  1042*  1043*
DVTR00  001502       284#
DVTR01  001526       295#
DVTR02  001552       306#
DVTR03  001576       317#
DVTR04  001622       328#
DVTR05  001646       339#
DVTR06  001672       350#
DVTR07  001716       361#
DVTVEC  001356       228#  1040*  1041*  1042
DV.END  001740       371#   995   1004   1133
DV.MAP  001500       171    282#   387    414    997   1007   1131   1136   1185
DV00.A  001504       285#
DV00.B  001510       287#
DV00.C  001514       289#
DV00.D  001520       291#
DV01.A  001530       296#
DV01.B  001534       298#
DV01.C  001540       300#
DV01.D  001544       302#
DV02.A  001554       307#
DV02.B  001560       309#
DV02.C  001564       311#
DV02.D  001570       313#
DV03.A  001600       318#
DV03.B  001604       320#
DV03.C  001610       322#
DV03.D  001614       324#
DV04.A  001624       329#
DV04.B  001630       331#
DV04.C  001634       333#
DV04.D  001640       335#
DV05.A  001650       340#
DV05.B  001654       342#
DV05.C  001660       344#
DV05.D  001664       346#
DV06.A  001674       351#
DV06.B  001700       353#
DV06.C  001704       355#
DV06.D  001710       357#
DV07.A  001720       362#
DV07.B  001724       364#
DV07.C  001730       366#
DV07.D  001734       368#
EM1     023700      3418#  3435
EM2     023727      3418#  3438
EM3     023767      3418#  3441
EM4     024032      3418#  3444
EM5     024076      3418#  3447
ENDPAT  023674      2925   3415#
ERRCNT  001232       142#   389*   511    829*
ERRFLG  001311       177#   385*   473*   540*   781*   794    808*   863*
ERRMSG  004252       791*   809    812#
```

```
ERTABO  004366          806     838#
EVENCH  020310         2717    2772#
EXIT  = 000205           81#    3309    3322    3341    3359    3364    3384    3388    3392
EXITER  004322          824     829#
FIX.00  006516         1048    1053    1058    1063    1097#
HALTS   004302          777     823#
HILIM   003436          616*    643     661#
ICOUNT  001222          138#    538     543*
INBUF   005520          586     622     969#
INIFLG  001310          176#    394     409*
INSTER= 104404          199#    637
INSTR = 104403          197#   1070
INSTR2  003236          593     605#
LIGHT   000174          110#    404
LIGHTS  001200          121#    404*    475*
LIMITS  003364          632     643#
LOAD.M  023342         1574    1657    1798    1908    2154    2225    2330    2845    2990    3094    3114    3181    3251
                       3310#
LOBITS  003442          618*    647     663#    664
LOCK    001220          137#    542*    556     558     800    1676*   1806*   1916*   2571*   2700*   2861*   2988*
LOGICA  002560          107     492#
LOKFLG  001312          178#
LOLIM   003434          615*    645     660#
LPCNT   001224          139#    537*    538     541*
LSTERR  001234          143#    390*    472*    524*    778     780*    864*
LO0.03  001422          255#   1010*   1045    1226    1291    1340    1392    1444    1498    1550    1620    1760    1867
                       1975    2047    2124    2195    2297    2397    2524    2648    2797    2959    3066    3153    3223
LO4.07  001424          256#   1012*   1050    1230    1295    1344    1396    1448    1502    1554    1626    1766    1874
                       1979    2051    2129    2200    2303    2403    2530    2654    2803    2964    3070    3157    3227
LO8.11  001426          257#   1014*   1055    1234    1299    1348    1400    1452    1506    1558    1632    1772    1881
                       1983    2055    2134    2205    2309    2409    2536    2660    2809    2969    3074    3161    3231
L12.15  001430          258#   1016*   1060    1238    1303    1352    1404    1456    1510    1562    1638    1778    1888
                       1987    2059    2139    2210    2315    2415    2542    2666    2815    2974    3078    3165    3235
MASKX   001244          151#   1619*   1625*   1631*   1637*   1714    1721    1759*   1765*   1771*   1777*   1834    1836
                       1865*   1872*   1879*   1886*   1944    1946    2396*   2402*   2408*   2414*   2487    2523*   2529*
                       2535*   2541*   2612    2647*   2653*   2659*   2665*   2759    2796*   2802*   2808*   2814*   2825
                       2917
MASK.A  001406          245#   1046    1619    1759    1865    2396    2523    2647    2796
MASK.B  001410          246#   1051    1625    1765    1872    2402    2529    2653    2802
MASK.C  001412          247#   1056    1631    1771    1879    2408    2535    2659    2808
MASK.D  001414          248#   1061    1637    1777    1886    2414    2541    2665    2814
MASTEK  005400          802     960#
MCRLF   005104          571     694     798     799     807     948     960#    1069    1087
MCSRX   005330          477     960#
MDATA   005624          721     731     973#
MEPASS  005145          476     960#
MERRPC  005454          805     960#
MERRX   005355          483     960#
MERR2   005174          960#    986    1176
MERR3   005243          431     960#
MLOCK   005301          455     960#
MNEW    005402          426     960#
MPASSX  005344          481     960#
MPFAIL  005107          861     960#
MQM     005100          601     960#    1092
MR      005171          463     960#
```

```
MRESET= 004000          81#   880   893
MSTCLR= 104412         211#   865  1309  1360  1412  1464  1518  1570  1647  1787  1897  1994  2069
                      2149  2220  2325  2425  2552  2681  2837  2984  3087  3174  3244
MTITLE  001000         119#   408
MTSTN   005366         803   960#  1071
MTSTPC  005267         960#
MVECX   005336         479   960#
NEXT    001216         136#   544   834  1224* 1289* 1338* 1390* 1442* 1496* 1548* 1616* 1756* 1862*
                      1973* 2045* 2121* 2192* 2293* 2393* 2520* 2644* 2793* 2956* 3064* 3151* 3221*
NOLIST= ****** U         1
NPR   = 040000          76#
NPRLOC  023674         3416#
ODDCH   020262         2714  2768#
PARAM = 104405          201#  1072
PARAM1  003304          621#   638
PARBIT= 040000           81#  1116  2493  2601  2672  2908
PARERR  003360          624   626   628   637#  644   646   648
PASCNT  001230          141#   384*  474*  475   508
PERFOR= 004537           81#  1246  1311  1321  1574  1649  1657  1659  1681  1723  1738  1789  1797
                      1798  1817  1844  1845  1899  1907  1908  1927  1954  1955  1996  2071  2151
                      2154  2158  2222  2225  2229  2250  2327  2330  2334  2355  2427  2435  2438
                      2442  2460  2469  2499  2554  2561  2566  2574  2582  2585  2589  2608  2683
                      2690  2695  2703  2719  2725  2729  2752  2839  2844  2845  2896  2922  2927
                      2986  2990  2995  3019  3033  3091  3094  3114  3178  3181  3248  3251
PFTAB   004470          862   868#
POPR0 = 012600           55#   828
POP1SP= 005726           55#
POP2SP= 022626           55#   546
PS    = 177776           53#   380*  449*  1196*
PUSHR0= 010046           55#   825
PUSH1S= 005746           55#
PUSH2S= 024646           55#
QV.FLG  001313         179#   386*  487*  535
RAM   = 020000          74#
RAMCLR= 104413         213#   866  1243
RESREG  004300         819   822#
RESTAR  004414         849   855#
RESTRT  002572         486   490   498#
RESV16  001404         239#  1035* 1036*
RESO5 = 104407         205#   822
RETURN  001214         135#   392*  462*  464   498*  544*  547   834*  836   867  1086* 1094* 1095
ROMCLK= 104415         217#  1257  1270  1654  1668  1670  1691  1693  1794  1802  1804  1904  1912
                      1914  2003  2013  2021  2078  2080  2090  2098  2445  2447  2453  2472  2491
                      2573  2588  2599  2702  2728  2739  2857  2859  2881  2883  2895  2906  2934
                      2936  3008  3184  3186  3254  3356  3358  3363  3373  3376  3387  3391
RUN     001304         169#   388*  989   992*  993*  1000* 1001*
RXSHIF  023402        2158  2229  2250  2334  2355  2435  2438  2442  2469  2561  2566  2585  2690
                      2695  2725  2995  3019  3033  3323#
SAVACT  001302         167#   429  1174*
SAVNUM  001303         168#   383*  485*  488*  1167*
SAVPC   001276         164#   669*  840
SAVR0   001260         157#   678*  683  3429
SAVR1   001262         158#   677*  684  3421
SAVR2   001264         159#   676*  685  3423
SAVR3   001266         160#   675*  686
SAVR4   001270         161#   674*  687  3427
```

```
SAVR5   001272          162#    673*    688     3425
SAVSP   001274          163#
SAV05 = 104406          203#    782
SCOPE = 104400          191#   1241    1306    1355    1407    1459    1513    1565    1641    1781    1891    1990    2062
                       2142    2213    2318    2418    2545    2669    2818    2977    3081    3168    3238
SCOP1 = 104401          193#   1718    1843    1953    2609    2753    2923    3045
SERV.G  004640          523     770     914#    915
SETSCA  023544         1246    1311    1321    1649    1738    1789    1845    1899    1955    1996    2071    2151    2222
                       2327    2427    2554    2683    2839    2986    3091    3178    3248    3367#
SET.TM  023622         1723    1844    1954    2927    3385#
SILO.I  023510         2574    2589    2703    2729    2896    3354#
SILO.O  023532         2460    2499    2582    2608    2719    2752    2922    3361#
SKIP  = 000000         3418#
SPACNT= 003745          704*    725     728*    742#
SSWR    000176          112#    405
STACK = 001200           54#    381     450     835     857
STAT    001236          148#   1226*   1230*   1234*   1238*   1291*   1295*   1299*   1303*   1340*   1344*   1348*   1352*
                       1357    1392*   1396*   1400*   1404*   1409    1444*   1448*   1452*   1456*   1461    1498*   1502*
                       1506*   1510*   1515    1550*   1554*   1558*   1562*   1567    1620*   1626*   1632*   1638*   1643
                       1760*   1766*   1772*   1778*   1783    1811    1867*   1874*   1881*   1888*   1893    1975*   1979*
                       1983*   1987*   2047*   2051*   2055*   2059*   2064    2124*   2129*   2134*   2139*   2144    2156
                       2195*   2200*   2205*   2210*   2215    2227    2243    2249    2297*   2303*   2309*   2315*   2320
                       2348    2397*   2403*   2409*   2415*   2420    2432    2437    2440    2461    2478    2480    2493
                       2524*   2530*   2536*   2542*   2547    2559    2563    2565    2601    2648*   2654*   2660*   2666*
                       2672    2675    2688    2692    2694    2712    2797*   2803*   2809*   2815*   2820    2834    2835
                       2847    2908    2959*   2964*   2969*   2974*   2979    2994    3017    3032    3066*   3070*   3074*
                       3078*   3083    3153*   3157*   3161*   3165*   3170    3223*   3227*   3231*   3235*   3240
SV05    003452          673#
SWR     001202          123#    400     405*    410     424     429     434     453     517     526     533     554     567
                        764     771     776     823     830     832     916     917*    927*    932*    933*    934*    941*
                       1064    1686    1695    2875    2885
SW00  = 000001           34#    424
SW01  = 000002           33#   1064
SW02  = 000004           32#
SW03  = 000010           31#
SW04  = 000020           30#
SW05  = 000040           29#
SW06  = 000100           28#
SW07  = 000200           27#
SW08  = 000400           26#    830
SW09  = 001000           25#    554
SW10  = 002000           24#    832
SW11  = 004000           23#    533
SW12  = 010000           22#    567     771
SW13  = 020000           21#    776
SW14  = 040000           20#
SW15  = 100000           19#
SYNA00  001506          286#
SYNA01  001532          297#
SYNA02  001556          308#
SYNA03  001602          319#
SYNA04  001626          330#
SYNA05  001652          341#
SYNA06  001676          352#
SYNA07  001722          363#
SYNB00  001512          288#
```

```
SYNB01  001536          299#
SYNB02  001562          310#
SYNB03  001606          321#
SYNB04  001632          332#
SYNB05  001656          343#
SYNB06  001702          354#
SYNB07  001726          365#
SYNC    023646         2834*  2835*  2849   2851   2890   2892   3394#
SYNCX   001240          149#  1866*  1873*  1880*  1887*  1921   2296*  2302*  2308*  2314*  2332   2354
SYNC00  001516          290#
SYNC01  001542          301#
SYNC02  001566          312#
SYNC03  001612          323#
SYNC04  001636          334#
SYNC05  001662          345#
SYNC06  001706          356#
SYNC07  001732          367#
SYNC2A  001432          260#  1011*  1866   2296
SYNC2B  001434          261#  1013*  1873   2302
SYNC2C  001436          262#  1015*  1880   2308
SYNC2D  001440          263#  1017*  1887   2314
SYND00  001522          292#
SYND01  001546          303#
SYND02  001572          314#
SYND03  001616          325#
SYND04  001642          336#
SYND05  001666          347#
SYND06  001712          358#
SYND07  001736          369#
S.C   = 050000           77#  1256   1269   1653   1793   1903   2002   2020   2077   2079   2097   2444   2446
                        2471  2572   2587   2701   2727   2894   2933   2935   3007   3183   3185   3253
TEMP    005562          708   858*   859*   971#
TEMP1   001246          152#  414*   415    420*   962    1193*  1194*  1706*  1709*  1710   1826*  1829*  1830
                       1936*  1939*  1940   2836*  2929*
TEMP2   001250          153#  415*   416    964    1582*  1598*  1812*  1841*  1922*  1951*  2824*  2852*  2853*
                       2855   2866*  2867*  2869   2878*  2879
TEMP3   001252          154#  2826*  2827*  2830*  2833*  2853   2867
TEMP4   001254          155#  2714*  2717*  2720   2722   2755*  2756
TEMP5   001256          156#  2721*  2722*  2723
TKCSR   001204          127#  529    588    914    921    942
TKDBR   001206          128#  519    531    590    596    766    923    944
TLAST = 022750         1090  3418#
TPCSR   001210          129#  572    594    773    928
TPDBR   001212          130#  574*   596*   775*   930*
TRPOK   003762          754#
TSTNO   001226          140#  391*   843    870    1075   1082   1084   1223*  1288*  1337*  1389*  1441*  1495*
                       1547*  1615*  1755*  1861*  1972*  2044*  2120*  2191*  2292*  2392*  2519*  2643*  2792*
                       2955*  3063*  3150*  3220*
TST1    007260         1078   1094   1223#
TST10   011346         1548   1615#
TST11   012242         1616   1755#
TST12   012766         1756   1861#
TST13   013542         1862   1972#
TST14   014050         1973   2044#
TST15   014402         2045   2120#
TST16   014700         2121   2191#
```

```
TST17   015334            2192    2292#
TST2    007542            1224    1288#
TST20   016020            2293    2392#
TST21   016652            2393    2519#
TST22   017424            2520    2643#
TST23   020336            2644    2792#
TST24   021342            2793    2955#
TST25   022046            2956    3063#
TST26   022436            3064    3150#
TST27   022750            3151    3220#   3418
TST3    007744            1289    1337#
TST30 = ****** U          3221
TST4    010152            1338    1389#
TST5    010366            1390    1441#
TST6    010614            1442    1495#
TST7    011042            1496    1547#
TTST    002702             456*    457*    459*    460*    527#
TWOSYN= 010000              81#    2243    2348    2563    2692    2847
TXSHIF  023252            1681    1817    1927    3278#
TYPDAT  004266             797     815     818#
TYPE  = 104402             195#    408     413     426     431     455     463     476     477     479     481     483     571
                           584     601     694     731     798     799     802     803     805     807     811     816     861
                           918     920     948     986    1069    1087    1092    1176
TYPMSG  004166             795     798#
VECMAP  007104            1175    1183#
WRDCNT  003742             702*    732*    740#
WRKO.F  004254             810     813#
XBX     004060             772     774     776#
XCSR    002604             478     500#
XERR    002626             484     509#
XFR   = 030000              75#    1669    1692    1803    1913    2011    2088    2451    2489    2597    2737    2858    2882
                           2904
XHEAD   005461             413     960#
XPASS   002620             482     506#
XSTATQ  005506             419     960#
XTSTN   004374             804     841#
XVEC    002612             480     503#
$CRAP = 177777              1#    1214#   1219#   1280#   1284#   1328#   1332#   1378#   1384#   1431#   1436#   1485#   1490#
                          1539#   1543#   1606#   1611#   1746#   1751#   1852#   1857#   1962#   1968#   2034#   2040#   2111#
                          2116#   2174#   2187#   2275#   2288#   2380#   2388#   2508#   2515#   2623#   2639#   2778#   2788#
                          2940#   2951#   3052#   3059#   3140#   3146#   3211#   3216#
$E    = 000031              1#    1224    1225#   1289    1290#   1338    1339#   1390    1391#   1442    1443#   1496    1497#
                          1548    1549#   1616    1617#   1756    1757#   1862    1863#   1973    1974#   2045    2046#   2121
                          2122#   2192    2193#   2293    2294#   2393    2394#   2520    2521#   2644    2645#   2793    2794#
                          2956    2957#   3064    3065#   3151    3152#   3221    3222#
$N    = 000027              1#    1214    1221    1225#   1280    1286    1290#   1328    1335    1339#   1378    1387    1391#
                          1431    1439    1443#   1485    1493    1497#   1539    1545    1549#   1606    1613    1617#   1746
                          1753    1757#   1852    1859    1863#   1962    1970    1974#   2034    2042    2046#   2111    2118
                          2122#   2174    2189    2193#   2275    2290    2294#   2380    2390    2394#   2508    2517    2521#
                          2623    2641    2645#   2778    2790    2794#   2940    2953    2957#   3052    3061    3065#   3140
                          3148    3152#   321*    3218    3222#   3418#
$Y    = 000017              1#    182#    191     193#    195#    197#    199#    201#    203#    205#    207#    209#    211#
                          213#    215#    217#    219#    221#
.     = 024274             92#     93      96#    103#    104#    105#    106#    109#    111#    114#    118#    120#    165#
                          166#    167#    168#    169#    170#    281#    283#    284#    285#    286#    287#    288#    289#
                          290#    291#    292#    294#    295#    296#    297#    298#    299#    300#    301#    302#    303#
```

		305#	306#	307#	308#	309#	310#	311#	312#	313#	314#	316#	317#	318#
		319#	320#	321#	322#	323#	324#	325#	327#	328#	329#	330#	331#	332#
		333#	334#	335#	336#	338#	339#	340#	341#	342#	343#	344#	345#	346#
		347#	349#	350#	351#	352#	353#	354#	355#	356#	357#	358#	360#	361#
		362#	363#	364#	365#	366#	367#	368#	369#	433	522	769	851	860
		874	912#	922	929	943	970#	972#	974#	988	1179	1200	1247#	1312#
		1358	1410	1462	1516	1568	1580	1644	1650#	1663	1674	1687	1696	1716
		1784	1790#	1810	1839	1894	1900#	1920	1949	1997#	2016	2065	2072#	2093
		2145	2152#	2216	2223#	2238	2247	2259	2267	2321	2328#	2343	2352	2364
		2372	2421	2428#	2458	2497	2505#	2548	2555#	2606	2676	2684#	2750	2821
		2840#	2876	2886	2920	2980	2987#	3084	3092#	3171	3179#	3241	3249#	3394#
.BEGIN	002332	449#												
.CNVRI	003542	210	695#											
.CONVR	003536	208	694#											
.DATAC	004576	220	900#											
.DELAY	004476	216	871#											
.EOP	002436	471#	3221											
.ERRTA	024230	790	3431#											
.HLT	004002	99	763#											
.INSTE	003224	200	601#											
.INSTR	003120	198	580#											
.INST1	003140	584#	604											
.MSG	003142	582*	585#											
.MSTCL	004556	212	892#											
.PARAM	003244	202	612#											
.PFAIL	004402	97	382	848#	856									
.RAMCL	004516	214	879#											
.RESO5	003504	206	683#											
.ROMCL	004566	218	896#											
.SAV05	003444	204	669#											
.SCOPE	002634	192	516#											
.SCOP1	003020	194	554#											
.START	001742	115	380#	392										
.TRPSR	003750	101	751#											
.TRPTA	001314	190#	756											
.TYPE	003044	196	564#											

```
DVEND       1#    465
DVFRNT      1#
HLT        55#    909   1255   1268   1320   1371   1424   1478   1532   1581   1589   1597   1664   1675   1717
         1735    1840   1950   2010   2017   2027   2087   2094   2104   2167   2239   2248   2260   2268   2344
         2353    2365   2373   2459   2498   2581   2596   2607   2710   2736   2744   2751   2903   2913   2921
         3006    3015   3030   3044   3104   3113   3124   3133   3195   3204   3263   3272   3320   3351
$BUFFE      1#    966
$CK15       1#   3335
$CK150      1#   3343
$CLR.T      1#   3389
$CYCLE      1#    975
$EOP        1#    465
$FINI       1#   3418
$GETFL      1#
$GETPA      1#   1070
$HEADE      1#
$LC01       1#   1213
$LC02       1#   1327   1377   1430   1484
$LC02A      1#   1538
$LC03       1#   1605
$LC03A      1#
$LC04       1#   1745   1851
$LC05       1#   1961   2033
$LC06       1#   2110
$LC06A      1#   2173   2274
$LC06B      1#   2379
$LC07       1#   2507   2622
$LC07A      1#   2777
$LC10       1#   2939
$LC11       1#   3051
$LC12       1#   3139
$LC13       1#   3210
$MSG        1#    960
$PFAIL      1#    844
$RAMCL      1#    871
$RXSHI      1#   3323
$SCOPE      1#    512
$SETLI      1#   1221   1286   1335   1387   1439   1493   1545   1613   1753   1859   1970   2042   2118   2189
         2290    2390   2517   2641   2790   2953   3061   3148   3218
$SETSC      1#   3366
$SETSY      1#
$SET.T      1#   3385
$SILOI      1#   3354
$SIMBC      1#
$TRPDE      1#    191    193    195    197    199    201    203    205    207    209    211    213    215    217
          219
$TSTN       1#   1221   1286   1335   1387   1439   1493   1545   1613   1753   1859   1970   2042   2118   2189
         2290    2390   2517   2641   2790   2953   3061   3148   3218
$TXSHI      1#   3278
$VARIA      1#    117
$XZ         1#   1214   1219   1280   1284   1328   1332   1378   1384   1431   1436   1485   1490   1539   1543
         1606    1611   1746   1751   1852   1857   1962   1968   2034   2040   2111   2116   2174   2187   2275
         2288    2380   2388   2508   2515   2623   2639   2778   2788   2940   2951   3052   3059   3140   3146
         3211    3216
```

```
. ABS.  024274     000


 ERRORS DETECTED:  0

 CZDVBC,CZDVBC/SOL/CRF/DOC=CZDVBC.MAC,CZDVBC.P11
 RUN-TIME: 31 45 3 SECONDS
 RUN-TIME RATIO: 279/81=3.4
 CORE USED:  27K  (53 PAGES)

 DOCUMENT PAGES:  77
```