DV11

DV11 MODEM CNTRL

CZDVEC0

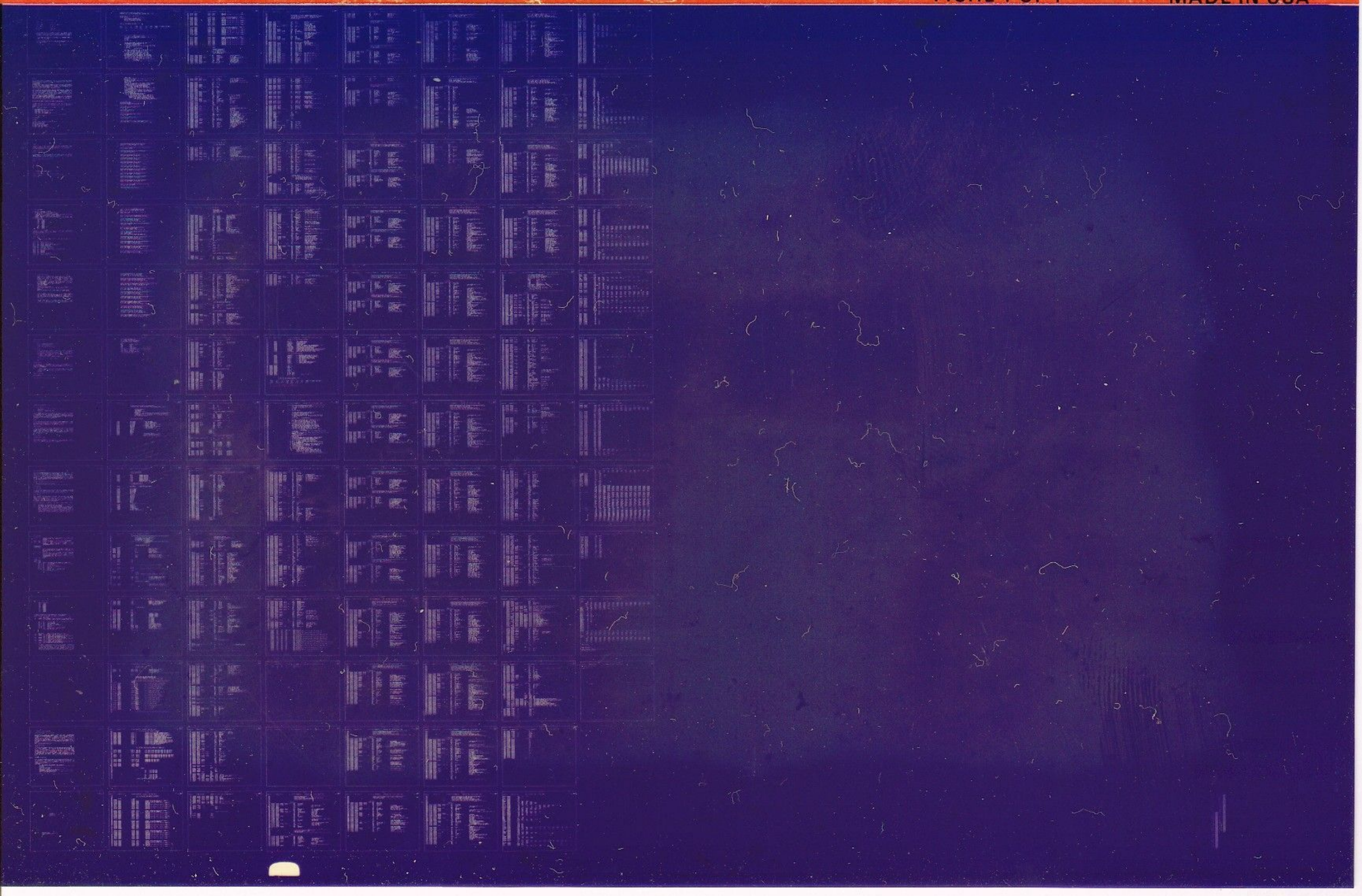
AH-8745C-MC

FICHE 1 OF 1

SEP 1979

digital

MADE IN USA



IDENTIFICATION

PRODUCT CODE: AC-8744C-MC

PRODUCT NAME: CZDVECO DV11 MODEM CNTRL

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).

CZDVE is used to verify the cables used for modem hook up. Modem bits are tested and interupts are also checked. All signals are tested and the turn around is either through the single line tester(h325) or 16 line turn around(h861). All signals that are looped around by the test connecter are checked. Modem control signals AND DV11 transmitter and receiver data is checked. Any combination of lines may be selected and these inturn will be tested individually.
Part 2 -THE MANUAL PARAMETER INPUT(TRIAL)- IS USED TO GET THE PARAMETERS INTO THE STATUS TABLE FOR REFERENCE BY THE DIAGNOSTIC IF "AUTO SIZING" does not work or is not desired. Starting address is at 210 and the execution of the program is self explanatory. (answer the questions).

Note:czdvec has been enhanced to be able to run with all the character lengths (5,6,7 and 8), with parity option enabled(odd/even).

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. CZDVF [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

SEQ 0003

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 real 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:	Set- single H325 turn around Clr- multi H325 turn around
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 DAiA: 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' until 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 CZDVEC0 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) 1276/0000000000010001 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.

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.**note** when character length is less than 8 bits, be sure to setup correct sync character in switches of sync recognition logic. ie: if 226 for 8 bit character ,

then it should be 026 for 7 bit character.

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 175200 is reached. If a 'SLAVE SYNC RESPONSE' was issued by the DV11 (or any other device) (no nxm trap) ; 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 chars.
5) SYNC ''A''=226 AND SYNC ''B''=062

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

DOCUMENT
****
CZDVEC LST
****

```
2       AC-8744C-MC/<377>/CZDVECO DV11 MODEM CNTRL
        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)


        TABLE OF LOOP AROUND FUNCTIONS  (H325)
        -----------------------------

        RING     CO      CTS      SECRX   SECTX    RTS      TRDY    LENAB  *** SIGNALS FOR ASYNC LC.
        RING     CO      CTS       DSR     NS      RTS      TRDY    LENAB  *** SIGNALS FOR SYNC LC
        BIT07   BIT06   BIT05     BIT04   BIT03   BIT02    BIT01    BIT00
                                    ^---------^
                  ^---------^------------------------^
          ^-------------------------------------------------^


1267    ******************************************************************
                THIS ''TEST 1'' IS NOT ACTUALLY A TEST.
                IT IS USED TO GET USERS INPUTS FOR WHICH LINE(S) ARE TO BE
                EXERCISED. THE PROGRAM WILL TYPE OUT:
                (A) H325
                (B) H861
                TYPE ''A'' ''OR ''B''

                THE H325 TURN AROUND IS USED FOR THE SINGLE LINE
                TURN AROUND AT THE DISTRIBUTION PANEL OR
                AT THE END OF THE MODEM CABLE.
                THE H861 TURN AROUND IS USED FOR THE 16 LINE TURN AROUND.
                IF THE H325 WAS SELECTED (A) THE FOLLOWING WILL BE TYPED
                IF SW06=0:
                SELECT LINE(S):  XXXXXXXXXXXXXXXX

                THE FIRST ''X'' REPRESENTS LINE 15 AND EACH ''X'' IS THE
                NEXT LOWER LINE TILL THE LAST ''X'' IS LINE 0. TYPE
                A ''1'' OR A ''0'' UNDER THE APPROIATE ''X''(LINE)
                TO EITHER SELECT(1) OR NOT TEST(0) EACH LINE.
                AFTER ALL 1'S AND 0'S ARE TYPED; TYPE A <CR>.
                THE PROGRAM WILL TYPE OUT IN OCTAL THE LINES YOU
                HAVE SELECTED; AND THE PROGRAM WILL BEGIN RUNNING
                THE HIGHEST SELECTED LINE THROUGH *ALL* TESTS THEN
                UPDATING TO THE NEXT LOWEST LINE TILL ALL SELECTED
                LINES ARE DONE. THEN THE PROGRAM WILL TYPE AN
                ''END'' CHAR.  PLEASE READ  THE SECTION ON PASS COMPLETE
                          IN DOCUMENT.
                IF THE H325 IS SELECTED AND SW06=1 THE FOLLOWING WILL BE TYPED:
```

```
                SINGLE LINE:
                THE USER MUST THEN TYPE IN A SINGLE LINE HE DESIRES (00-17) -OCIAL-
                END PASS IS THE SAME.
                REGUARDLESS OF WHICH CONNECTOR WAS SELECTED; THE
                THE LAST QUESTION IS:
                MODEM VECTOR:
                (THIS WILL BE ASKED ONLY AT THE INIATL START OF PROGRAM
                OR WHEN A DIFFERENT DV11 IN THE SYSTEM IS UNDER TEST)
                TYPE IN THE VECTOR OF THE MODEM CONTROL(300:774).
                THE CSR(MC.CSR) IS ASSUMED TO BE =DVSCR+20.
                NOTE:   IF CABLE TESTS ARE TO BE DONE ON OTHER
                DV11'S IN SYSTEM; SELECT THEM BY USING SW00 AS DESCRIBED
                IN THE DOCUMENTATION.
                UNLESS LOCATION 42 IS NON-ZERO IN WHICH CASE THE PROGRAM
                ASSUMES ITS UNDER ACT-11 MONITOR.  THE PROGRAM WILL
                CYCLE THROUGH ALL DV11S AND MODEM CONTROL *HOWEVER*
                THE RESTRICTIONS ARE:
                ***ALL*** MODEM VECTORS MUST BE AT 300
                ***ALL*** TURN AROUNDS MUST BE H861.
                ''LONG END PASS'' WILL BE GIVEN AT END OF LARGE END TO
                        INDICATE DEVICES TESTED. PASSES TYPED IN THIS
                        MESSAGE DO NOT INDICATE PASSES BUT RATHER THE
                        NUMBER OF FULL PASSES THROUGH MULTIPLE DEVICES.
                                !LARGE END AND TYPE OUT MAY BE INHIBITED BY SW12!
        ********************************************************************************

1466    ************************ TEST 2 ********************************
        INITIALIZATION CHECK
        VERIFY THAT CONTROL STATUS REGISTER AND LINE STATUS
        REGISTER WERE CLEARED BY INITIALIZE

1503    ************************ TEST 3 ********************************
        VERIFY THAT ''INTERUPT ENABLE'' CAN BE
        SET AND CLEARED.

1530    ************************ TEST 4 ********************************
        VERIFY THAT ''DONE'' CAN BE
        SET AND CLEARED.

1557    ************************ TEST 5 ********************************
        VERIFY THAT ''MAINTENANCE MODE'' CAN BE
        SET AND CLEARED.

1584    ************************ TEST 6 ********************************
        VERIFY THAT ''SCAN ENABLE'' CAN BE
        SET AND CLEARED.

1610    ************************ TEST 7 ********************************
        VERIFY THAT ''BUSY'' IS SET WHEN ''SCAN ENABLE'' IS SET
        VERIFY THAT ''BUSY'' IS CLEARED WHEN ''SCAN ENABLE'' IS CLEARED
```

```
1638    ********************** TEST 10 ********************************
        VERIFY THAT SETTING 'DONE'' DOES NOT CAUSE AN
        INTERRUPT IF ''INTERRUPT ENABLE'' IS CLEARED.

1659    ********************** TEST 11 ********************************
        VERIFY THAT NO INTERRUPT OCCURS WITH ''INTERRUPT ENABLE''
        SET AND 'DONE'' CLEARED.

1680    ********************** TEST 12 ********************************
        VERIFY THAT SETTING 'DONE'' CAUSES AN INTERRUPT
        WITH ''INTERRUPT ENABLE'' SET

1703    ********************** TEST 13 ********************************
        VERIFY THAT NO INTERRUPT OCCURS WITH
        ''INTERRUPT ENABLE'' SET AND 'DONE'' SET AT PRIORITY 7.

1724    ********************** TEST 14 ********************************
        VERIFY THAT NO INTERRUPT OCCURS WITH
        ''INTERRUPT ENABLE'' SET AND 'DONE'' SET AT PRIORITY 6.

1745    ********************** TEST 15 ********************************
        VERIFY THAT NO INTERRUPT OCCURS WITH
        ''INTERRUPT ENABLE'' SET AND 'DONE'' SET AT PRIORITY 5.

1766    ********************** TEST 16 ********************************
        VERIFY THAT NO INTERRUPT OCCURS WITH
        ''INTERRUPT ENABLE'' SET AND 'DONE'' SET AT PRIORITY 4.

1787    ********************** TEST 17 ********************************
        VERIFY THAT AN INTERRUPT OCCURS WITH ''INTERRUPT
        ENABLE'' SET AND 'DONE'' SET AT PRIORITY 0.

1808    ********************** TEST 20 ********************************
        VERIFY THAT AN INTERRUPT OCCURS WITH ''INTERRUPT
        ENABLE'' SET AND 'DONE'' SET AT PRIORITY 1.

1829    ********************** TEST 21 ********************************
        VERIFY THAT AN INTERRUPT OCCURS WITH ''INTERRUPT
        ENABLE'' SET AND 'DONE'' SET AT PRIORITY 2.

1850    ********************** TEST 22 ********************************
        VERIFY THAT AN INTERRUPT OCCURS WITH ''INTERRUPT
        ENABLE'' SET AND 'DONE'' SET AT PRIORITY 3.

1870    ********************** TEST 23 ********************************
        VERIFY THAT ALL LINE NUMBERS CAN BE WRITTEN INTO AND
        READ BACK FROM LINE COUNTER

1896    ********************** TEST 24 ********************************
        USING ''STEP'' MODE, VERIFY THAT THE
        LINE COUNTER CAN BE STEPPED THRU ALL STATES.
```

```
1923    ********** TEST 25 **********
        WRITE 1'S INTO ALL SCANNER MEMORY LOCATIONS.
        VERIFY THAT ALL LOCATIONS HAVE BEEN WRITTEN
        TO 1'S.
        VERIFY THAT ''CLEAR SCAN'' CLEARS ALL SCANNER
        MEMORY LOCATIONS.

1975    ********** TEST 26 **********
        WRITE 1'S INTO SELECTED SCANNER MEMORY LOCATION.
        VERIFY THAT ONLY SELECTED LOCATION WAS WRITTEN INTO.

2018    ********** TEST 27 **********
        WITH ALL SCANNER MEMORY LOCATIONS SET TO 1'S,
        WRITE 0'S INTO SELECTED LOCATION
        VERIFY THAT ONLY SELECTED LOCATION WAS CLEARED.

2062    ********** TEST 30 **********
        VERIFY THAT ''CLEAR MULTIPLXER'' CLEARS ALL MULTIPLEXER
        FUNCTION FLIP-FLOPS

2103    ********** TEST 31 **********
        WRITE 1'S INTO ALL SCANNER MEMORY LOCATIONS
        SET 'LINE ENABLE FOR ALL LINES
        VERIFY THAT AN INTERRUPT OCCURS FOR EACH LINE

2157    ********** TEST 32 **********
        WRITE 1'S INTO ALL MULTIPLEXER FUNCTION FLIP-FLOPS
        CLEAR SCANNER MEMORY
        VERIFY THAT AN INTERRUPT OCCURS FOR EACH LINE
        THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.

2235    ********** TEST 33 **********
        VERIFY THAT LINE ENABLE FUNCTION FLIP-FLOP CAN
        BE SET AND CLEARED FOR SELECTED LINE
        THIS TEST IS DONE IF THE H325 TURN AROUND IS USED

2282    ********** TEST 34 **********
        VERIFY THAT TERMINAL READY FUNCTION FLIP-FLOP CAN
        BE SET AND CLEARED FOR SELECTED LINE
        THIS TEST IS DONE IF THE H325 TURN AROUND IS USED

2329    ********** TEST 35 **********
        VERIFY THAT REQUEST TO SEND FUNCTION FLIP-FLOP CAN
        BE SET AND CLEARED FOR SELECTED LINE
        THIS TEST IS DONE IF THE H325 TURN AROUND IS USED

2376    ********** TEST 36 **********
        VERIFY THAT NEW SYNC (SECTX IF ASYNC LC) FUNCTION FLIP-FLOP CAN
        BE SET AND CLEARED FOR SELECTED LINE
        THIS TEST IS DONE IF THE H325 TURN AROUND IS USED
```

```
2424    ********** TEST 37 **********
        VERIFY THAT RING IS SET IF 'LINE ENABLE'
        AND TERMINAL ARE SET FOR SELECTED LINE.
        THIS TEST IS DONE IF THE H325 TURN AROUND IS USED

2471    ********** TEST 40 **********
        VERIFY THAT CLEAR TO SEND AND CARRIER ARE SET IF 'LINE ENABLE'
        AND REQUEST TO SEND ARE SET FOR SELECTED LINE.
        THIS TEST IS DONE IF THE H325 TURN AROUND IS USED

2518    ********** TEST 41 **********
        VERIFY THAT DATA SET READY(SECRX IF ASYNC LC) IS SET IF 'LINE ENABLE'
        AND NEW SYNC (SECTX IF ASYNC LC) ARE SET FOR SELECTED LINE.
        THIS TEST IS DONE IF THE H325 TURN AROUND IS USED

2564    ********** TEST 42 **********
        VERIFY THAT LINE ENABLE FUNCTION FLIP-FLOP CAN
        BE SET AND CLEARED FOR SELECTED LINE
        THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.

2618    ********** TEST 43 **********
        VERIFY THAT TERMINAL READY FUNCTION FLIP-FLOP CAN
        BE SET AND CLEARED FOR SELECTED LINE
        THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.

2672    ********** TEST 44 **********
        VERIFY THAT REQUEST TO SEND FUNCTION FLIP-FLOP CAN
        BE SET AND CLEARED FOR SELECTED LINE
        THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.

2726    ********** TEST 45 **********
        VERIFY THAT SECONDARY TRANSMIT FUNCTION FLIP-FLOP CAN
        BE SET AND CLEARED FOR SELECTED LINE
        THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.

2781    ********** TEST 46 **********
        VERIFY THAT CLEAR TO SEND AND CARRIER ARE SET IF 'LINE ENABLE'
        AND TERMINAL ARE SET FOR SELECTED LINE.
        THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.

2835    ********** TEST 47 **********
        VERIFY THAT RING IS SET IF 'LINE ENABLE'
        AND REQUEST TO SEND ARE SET FOR SELECTED LINE.
        THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.

2889    ********** TEST 50 **********
        VERIFY THAT SECONDARY RECEIVE IS SET IF 'LINE ENABLE'
        AND SECONDARY TRANSMIT ARE SET FOR SELECTED LINE.
        THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.
```

```
2942    ************************ TEST 51 ********************************
        DV11 SINGLE LINE CABLE TEST.
        TEST TO RUN A 5 BIT BLOCK (000-037)
        OF DATA FROM THE DV11 TRANSMITTER INTO THE
        DV11 RECEIVER THROUGH THE CABLE.
        SETUP:
        MODE:              EXTERNAL LOOP BACK
        TXBA:              SYNC
        TXWC:              -42(8)-BIT15
        RXBA               RXBA
        RXWC:              -40(8)-BIT15
        LINE PROTOCOL TXDDCMP,RXDDCMP,LRC8,STRIP SYNC,IDLE MARK
        LINE STATE         EXPECT BCC,TX GO
        LINE PROGRESS      SEND BCC
        NOTE: FOR TEST OF ASYNC LINE CARD;
         "SYNC 'A'" MUST BE SET TO ALL ZEROS
          IN SOFTWARE STATUS MAP.
```

```
                        ;*AC-8744C-MC/<377>/CZDVEC0 DV11 MODEM CNTRL
                        ;*COPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
                        ;*-------------------------------------------------------------

                                ;STARTING PROCEDURE
                                ;LOAD PROGRAM
                                ;LOAD ADDRESS 000200
                                ;PRESS START
                                ;PROGRAM WILL TYPE ''AC-8744C-MC/<377>/CZDVEC0 DV11 MODEM CNTRL''
                                ;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-8744C-MC/<377>/CZDVEC0 DV11 MODEM CNTRL/<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  DVPVEC: 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
```

```
 390  002024  005037  001234                 CLR     LSTERR                  ;CLEAR LAST ERROR POINTER
 391  002030  012737  000001  001226         MOV     #1,TSTNO                ;SET UP FOR TEST 1
 392  002036  012737  001742  001214         MOV     #.START,RETURN          ;SET UP FOR POWER FAIL BEFORE
 393                                                                          ;TESTING STARTS
 394  002044  105737  001310                 TSTB    INIFLG                  ;HAS INITIALIZATION BEEN PERFORMED
 395  002050  001063                         BNE     1$                      ;BR IF YES
 396  002052  013746  000004                 MOV     4,-(SP)                 ;
 397  002056  013746  000006                 MOV     6,-(SP)                 ;
 398  002062  005037  000006                 CLR     6                       ;
 399  002066  012737  002104  000004         MOV     #80$,4                  ;
 400  002074  005777  177102                 TST     @SWR                    ;
 401  002100  000240                         NOP                             ;
 402  002102  000407                         BR      81$                     ;
 403  002104  022626                 80$:    CMP     (SP)+,(SP)+             ;
 404  002106  012737  000174  001200         MOV     #LIGHT,LIGHTS           ;
 405  002114  012737  000176  001202         MOV     #SSWR,SWR               ;
 406  002122  012637  000006         81$:    MOV     (SP)+,6                 ;
 407  002126  012637  000004                 MOV     (SP)+,4                 ;
 408  002132  104402  001000                 TYPE    ,MTITLE                 ;TYPE TITLE MESSAGE
 409  002136  105137  001310                 COMB    INIFLG                  ;IF NOT SET FLAG AND DO
 410  002142  105777  177034                 TSTB    @SWR                    ;BIT7=1??
 411  002146  100402                         BMI     16$                     ;BR IF NO AUTO SIZE
 412  002150  004737  006626                 JSR     PC,CSRMAP               ;GO DO THE AUTO SIZE
 413  002154  104402  005461         16$:    TYPE    ,XHEAD          ;TYPE HEADER
 414  002160  012737  001500  001246         MOV     #DV.MAP,TEMP1           ;SET POINTER
 415  002166  017737  177054  001250  5$:     MOV     @TEMP1,TEMP2            ;SET DATA
 416  002174  022737  177777  001250         CMP     #177777,TEMP2           ;ALL DONE?
 417  002202  001406                         BEQ     1$                      ;BR IF YES
 418  002204  104410                         CONVRT                          ;
 419  002206  005506                         XSTATQ                          ;
 420  002210  062737  000002  001246         ADD     #2,TEMP1                ;UPDATE POINTER
 421  002216  000763                         BR      5$
 422  002220  005737  000042          1$:    TST     @#42                    ;IS PROGRAM RUNNING UNDER MONITOR
 423  002224  001030                         BNE     3$                      ;BR IF YES
 424  002226  032777  000001  176746         BIT     #SW00,@SWR              ;SELECT SPECIFIC DEVICES??
 425  002234  001424                         BEQ     3$                      ;BR IF NO.
 426  002236  104402  005402                 TYPE    ,MNEW                   ;TYPE THE MESSAGE.
 427  002242  005000                         CLR     R0                      ;ZERO DATA LIGHTS
 428  002244  000000                         HALT                            ;WAIT FOR USER TO TELL WHAT DEVICES TO RUN
 429  002246  127737  176730  001302         CMPB    @SWR,SAVACT             ;IS THE NUMBER VALID?
 430  002254  101404                         BLOS    2$                      ;BR IF NUMBER IS OK.
 431  002256  104402  005243                 TYPE    ,MERR3                  ;TELL USER OF INVALID NUMBER.
 432  002262  000000                         HALT                            ;STOP EVERY THING.
 433  002264  000776                         BR      .-2                     ;RESTART THE PROGRAM AGAIN.
 434  002266  117737  176710  001300  2$:     MOVB    @SWR,DVACTV             ;GET NEW DEVICE PATTERN
 435  002274  113700  001300                 MOVB    DVACTV,R0               ;SHOW THE USER WHAT HE SELECTED.
 436  002300  042700  177400                 BIC     #^C<377>,R0             ;USE ONLY LOW BYTE.
 437  002304  000000                         HALT                            ;CONTINUE DYNAMIC SWITCHES.
 438  002306  012700  000300          3$:    MOV     #300,R0                 ;PREPARE TO CLEAR THE FLOATING
 439  002312  012701  000302                 MOV     #302,R1                 ;VECTOR AREA. 300-776
 440  002316  010120                  4$:    MOV     R1,(R0)+                ;START PUTTING 'PC+2 - HALT''
 441  002320  005021                         CLR     (R1)+                   ;IN VECTOR AREA.
 442  002322  022021                         CMP     (R0)+,(R1)+             ;POP POINTERS
 443  002324  022700  001000                 CMP     #1000,R0                ;ALL DONE??
 444  002330  001372                         BNE     4$                      ;BR IF NO.
 445
```

```
                                        ;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$:     CLRB    ERRFLG          ;PREPARE FOR NEW TEST
                                        CLR     LPCNT           ;START ICOUNTER AT 0
                                        CLR     LOCK
                                        MOV     #5.,ICOUNT      ;RESET ITERATIONS
                                        MOV     NEXT,RETURN     ;GET NEXT TEST
                                3$:     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
```

```
003350  006305                  ASL     R5
003352  006305                  ASL     R5
003354  006305                  ASL     R5
003356  000760                  BR      1$
003360  104404          PARERR: INSTER
003362  000750                  BR      PARAM1

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

003364  020537  003436  LIMITS: CMP     R5,HILIM
003370  101373                  BHI     PARERR
003372  020537  003434          CMP     R5,LOLIM
003376  103770                  BLO     PARERR
003400  133705  003442          BITB    LOBITS,R5
003404  001365                  BNE     PARERR

                                ;STORE NUMBER AT SPECIFIED ADDRESS

003406  013704  003440          MOV     DEVADR,R4
003412  010524          1$:     MOV     R5,(R4)+
003414  062705  000002          ADD     #2,R5
003420  105337  003443          DECB    ADRCNT
003424  001372                  BNE     1$
003426  012604                  MOV     (SP)+,R4
003430  012605                  MOV     (SP)+,R5
003432  000002                  RTI
003434  000000          LOLIM:  0
003436  000000          HILIM:  0
003440  000000          DEVADR: 0
003442  000000          LOBITS: 0
        003443          ADRCNT=LOBITS+1

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

003444  016637  000004  001276  .SAV05: MOV     4(SP),SAVPC     ;SAVE R7 (PC)

                                ;SAVE R0-R5

003452  010537  001272          SV05:   MOV     R5,SAVR5        ;SAVE R5
003456  010437  001270                  MOV     R4,SAVR4        ;SAVE R4
003462  010337  001266                  MOV     R3,SAVR3        ;SAVE R3
003466  010237  001264                  MOV     R2,SAVR2        ;SAVE R2
003472  010137  001262                  MOV     R1,SAVR1        ;SAVE R1
003476  010037  001260                  MOV     R0,SAVR0        ;SAVE R0
003502  000002                  RTI                             ;LEAVE.

                                ;RESTORE R0-R5

003504  013700  001260          .RES05: MOV     SAVR0,R0        ;RESTORE R0
003510  013701  001262                  MOV     SAVR1,R1        ;RESTORE R1
003514  013702  001264                  MOV     SAVR2,R2        ;RESTORE R2
003520  013703  001266                  MOV     SAVR3,R3        ;RESTORE R3
003524  013704  001270                  MOV     SAVR4,R4        ;RESTORE R4
003530  013705  001272                  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  025364          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
```

```
000801 004202 001402                    BEQ     1$
000802 004204 104402 005400             TYPE    ,MASTEK
000803 004210 104402 005366     1$:     TYPE    ,MTSTN
000804 004214 104411 004374             CNVRT   ,XTSTN          ;SHOW IT
000805 004220 104402 005454             TYPE    ,MERRPC         ;TYPE PC.
000806 004224 104411 004366             CNVRT   ,ERTABO         ;SHOW IT
000807 004230 104402 005104             TYPE    ,MCRLF          ;GIVE A CR/LF
000808 004234 112737 177777 001311      MOVB    #-1,ERRFLG      ;NO MORE HEADER UNLESS NO DATA TABLE.
000809 004242 005737 004252             TST     ERRMSG          ;IS THERE AN ERROR MESSAGE?
000810 004246 001402                    BEQ     WRKO.FM         ;BR IF NO.
000811 004250 104402                    TYPE                    ;TYPE
000812 004252 000000     ERRMSG: 0      ;    ERROR MESSAGE
000813 004254            WRKO.FM:       ;
000814 004254 005737 004264             TST     DATAHD          ;DATA HEADER?
000815 004260 001402                    BEQ     TYPDAT          ;BR IF NO
000816 004262 104402                    TYPE                    ;TYPE
000817 004264 000000     DATAHD: 0      ;    DATA HEADER
000818 004266 005737 004276     TYPDAT: TST     DATABP          ;DATA TABLE?
000819 004272 001402                    BEQ     RESREG          ;BR IF NO.
000820 004274 104410                    CONVRT                  ;SHOW
000821 004276 000000     DATABP: 0      ;    DATA TABLE
000822 004300 104407     RESREG: RES05  ;RESTORE PROC REGISTERS
000823 004302 005777 174674     HALTS:  TST     @SWR            ;HALT ON ERROR?
000824 004306 100005                    BPL     EXITER          ;BR IF NO HALT ON ERROR
000825 004310 010046                    PUSHR0                  ;SAVE R0
000826 004312 016600 000002             MOV     2(SP),R0        ;SHOW ERROR PC IN DATA LIGHTS
000827 004316 000000                    HALT                    ;HALT
000828 004320 012600                    POPR0                   ;GET R0
000829 004322 005237 001232     EXITER: INC     ERRCNT          ;UPDATE ERROR COUNT
000830 004326 032777 000400 174646      BIT     #SW08,@SWR      ;GOTO TOP OF TEST?
000831 004334 001007                    BNE     1$              ;BR IF YES
000832 004336 032777 002000 174636      BIT     #SW10,@SWR      ;GOTO NEXT TEST?
000833 004344 001407                    BEQ     2$              ;BR IF NO
000834 004346 013737 001216 001214      MOV     NEXT,RETURN     ;SET FOR NEXT TEST
000835 004354 012706 001200     1$:     MOV     #STACK,SP       ;RESET SP
000836 004360 000177 174630             JMP     @RETURN         ;GOTO SPECIFIED TEST
000837 004364 000002     2$:     RTI                     ;RETURN
000838 004366 000001     ERTABO: 1
000839 004370     006     002            .BYTE   6,2
000840 004372 001276                    SAVPC
000841 004374 000001     XTSTN:  1
000842 004376     003     002            .BYTE   3,2
000843 004400 001226                    TSTNO
000844                                  ;ENTER HERE ON POWER FAILURE
000845                                  ;--------------------------
000846
000847
000848 004402             .PFAIL:
000849 004402 012737 004414 000024      MOV     #RESTART,24             ;SET UP FOR POWER UP TRAP
000850 004410 000000                    HALT                            ;HALT ON POWER DOWN NORMAL
000851 004412 000777                    BR      .
000852
000853                                  ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
000854
000855 004414             RESTAR:
000856 004414 012737 004402 000024      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 CZDVEC0  /
   (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
```

```
 975
 976                                        ;
 977                                        ;ROUTINE USED TO ''CYCLE'' THROUGH UP TO EIGHT DV11'S
 978                                        ;THIS ROUTINE SETS UP THE CONTROL ADDRESS FOR THE DIAGNOSTIC
 979                                        ;AND RUNS THE SPECIFIED DV11'S.   THIS ROUTINE *MUST*
 980                                        ;BE RUN FIRST BEFORE ENTERING THE DIAGNOSTIC FOR THE
 981                                        ;SETUP NECESSARY.
 982                                        ;
 983
 984  005666  105737  001300        CYCLE:  TSTB    DVACTV          ;ARE ANY DV11'S TO BE TESTED?
 985  005672  001004                        BNE     1$              ;BR IF OK.
 986  005674  104402  005174                TYPE    ,MERR2          ;NO DV11'S SELECTED!!
 987  005700  000000                        HALT                    ;STOP THE SHOW.
 988  005702  000776                        BR      .-2             ;DISQUALIFY CONT. SW.
 989  005704  133737  001304  001300  1$:   BITB    RUN,DVACTV      ;IS THIS ONE ''ACTIVE''
 990  005712  001020                        BNE     2$              ;BR IF GOOD ONE FOUND.
 991  005714  000241                        CLC                     ;CLEAR PROC. CARRY BIT.
 992  005716  106137  001304                ROLB    RUN             ;UPDATE POINTER
 993  005722  105537  001304                ADCB    RUN             ;CATCH CARRY FROM RUN
 994  005726  062737  000024  001306        ADD     #24,CREAM       ;UPDATE ADDRESS POINTER.
 995  005734  022737  001740  001306        CMP     #DV.END,CREAM
 996  005742  001360                        BNE     1$              ;KEEP GOING; NOT ALL TESTED FOR.
 997  005744  012737  001500  001306        MOV     #DV.MAP,CREAM   ;RESET ADDRESS POINTER.
 998  005752  000754                        BR      1$              ;KEEP LOOKING FOR ACTIVE DV11
 999  005754  000241                2$:     CLC                     ;CLEAR PROC. CARRY.
1000  005756  106137  001304                ROLB    RUN             ;UPDATE POINTER.
1001  005762  105537  001304                ADCB    RUN             ;CATCH CARRY.
1002  005766  013700  001306                MOV     CREAM,R0        ;GET ADDRESS POINTER.
1003  005772  062737  000024  001306        ADD     #24,CREAM       ;UPDATE.
1004  006000  022737  001740  001306        CMP     #DV.END,CREAM
1005                                                                ;ALL DONE?
1006  006006  001003                        BNE     3$              ;BR IF NO.
1007  006010  012737  001500  001306        MOV     #DV.MAP,CREAM   ;RESTORE POINTER.
1008  006016  012037  001362        3$:     MOV     (R0)+,DVSCR     ;LOAD SYSTEM CTRL. REG
1009  006022  012037  001352                MOV     (R0)+,DVRVEC    ;LOAD VECTOR
1010  006026  012037  001422                MOV     (R0)+,L00.03    ;GET LINE PARAMETERS. 00-03
1011  006032  012037  001432                MOV     (R0)+,SYNC2A    ;
1012  006036  012037  001424                MOV     (R0)+,L04.07    ;                     04-07
1013  006042  012037  001434                MOV     (R0)+,SYNC2B    ;
1014  006046  012037  001426                MOV     (R0)+,L08.11    ;                     08-11
1015  006052  012037  001436                MOV     (R0)+,SYNC2C    ;
1016  006056  012037  001430                MOV     (R0)+,L12.15    ;                     12-15
1017  006062  012037  001440                MOV     (R0)+,SYNC2D    ;
1018  006066  012700  000002                MOV     #2,R0           ;SAVE CORE THIS WAY.
1019  006072  013737  001362  001364        MOV     DVSCR,DVSCRH    ;GET SYS CTRL. REG HIGH BYTE.
1020  006100  005237  001364                INC     DVSCRH          ;GOT IT.
1021  006104  013737  001364  001366        MOV     DVSCRH,DVRIC    ;GET NXT REC. CHAR REG.
1022  006112  005237  001366                INC     DVRIC           ;GOT IT
1023  006116  013737  001366  001370        MOV     DVRIC,DVLCR     ;GET LN. PAR.REG.
1024  006124  060037  001370                ADD     R0,DVLCR        ;GOT IT
1025  006130  013737  001370  001372        MOV     DVLCR,DVSRS     ;GET SEC. REG. SEL. REG.
1026  006136  060037  001372                ADD     R0,DVSRS        ;GOT IT
1027  006142  013737  001372  001374        MOV     DVSRS,DVSRSH    ;GET HIGH BYTE.
1028  006150  005237  001374                INC     DVSRSH          ;GOT IT
1029  006154  013737  001374  001376        MOV     DVSRSH,DVSRA    ;SEC. REG. ACCESS.
1030  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  007310                  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$:     TST     (R0)+
1090  006470  020027  020460          CMP     R0,#TLAST+10
1091  006474  001354                  BNE     5$
1092  006476  104402  005100          TYPE    ,MQM
1093  006502  000733                  BR      4$
1094  006504  012737  007310  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
```

```
                                ;CONTROL STATUS REGISTER BIT FUNCTIONS

            000020              BUSY=20         ;LINE SCANNER RUNNING
            000040              SCNENA=40       ;LINE SCANNER ENABLE
            000100              INTENA=100      ;INTERRUPT ENABLE
            000200              DONE=200        ;SCANNER DONE
            000400              STEP=400        ;CAUSES LINE COUNTER TO BE INCREMENTED BY 1 COUNT
            001000              MAINT=1000      ;FORCES 1S TO INPUT OF SCRATCH PAD MEMORY
            002000              CLRMUX=2000     ;CLEAR MULTIPLEXER FUNCTION FLIPFLOPS
            004000              CLRSCN=4000     ;CLEARS SCANNER SCRATCHPAD MEMORY
            010000              SECRXF=10000    ;SECONDARY RECEIVE TRANSITION WAS DETECTED BY SCANNER
            020000              CSF=20000       ;CLEAR TO SEND TRANSITION WAS DETECTED BY SCANNER
            040000              COF=40000       ;CARRIER TRANSITION WAS DETECTED BY SCANNER
            100000              RINGF=100000    ;RING SIGNAL WAS DETECTED BY SCANNER

                                ;LINE REGISTER BIT FUNCTIONS

            000001              LINENA=BIT0     ;=1, RECOGNIZE TRANSITIONS ON THIS LINE
            000010              SECTX=10        ;=1, SEND SECONDARY TRANSMIT TO MODEM
            000020              SECRX=20        ;=1, SECONDARY RECEIVE TURNED ON BY MODEM
            000002              TRMRDY=BIT1     ;=1, SEND TERMINAL READY TO MODEM
            000004              RS=BIT2         ;=1, SEND REQUEST TO SEND TO MODEM
            000010              NS=BIT3         ;=1,  NEW SYNC LEAD.
            000020              DSR=BIT4        ;=1, DATA SET READY.
            000040              CS=BIT5         ;=1, CLEAR TO SEND TURNED ON BY MODEM
            000100              CO=BIT6         ;=1, CARRIER TURNED ON BY MODEM
            000200              RING=BIT7       ;=1, RING TURNED ON BY MODEM

007260      000000              TURFLG: 0
007262      000000              LINE:   0
007264      000000              POINTER:        0
007266      000000              CHAR:   0
007270      000000              COUNT:  0
007272      000000              SELECT: 0
007274      000000              EXERCISE:       0
007276      000000              TOTAL:  0
007300      000001              MC.CSR: .BLKW 1
007302      000001              MC.LSR: .BLKW 1
007304      000300              MC.VEC: 300             ;DEFAULT VECTOR!.
007306      000001              MC.LVL: .BLKW 1


                        ;*TABLE OF LOOP AROUND FUNCTIONS  (H325)
                        ;*-------------------------------
;*
;*RING      CO       CTS      SECRX   SECTX     RTS     TRDY    LENAB  *** SIGNALS FOR ASYNC LC.
;*RING      CO       CTS       DSR     NS       RTS     TRDY    LENAB  *** SIGNALS FOR SYNC LC
;*BIT07   BIT06    BIT05    BIT04    BIT03    BIT02   BIT01   BIT00
;*                            ^        ^
;*          ^        ^        ---------        ^
;*   ^      ---------------------------        ^
;*    ---------------------------------------------------^
;*
;*
;*
```

```
1267                    ;*********************************************************************
1268                    ;*      THIS ''TEST 1'' IS NOT ACTUALLY A TEST.
1269                    ;*      IT IS USED TO GET USERS INPUTS FOR WHICH LINE(S) ARE TO BE
1270                    ;*      EXERCISED. THE PROGRAM WILL TYPE OUT:
1271                    ;*      (A) H325
1272                    ;*      (B) H861
1273                    ;*      TYPE ''A'' 'OR 'B''
1274                    ;*
1275                    ;*      THE H325 TURN AROUND IS USED FOR THE SINGLE LINE
1276                    ;*      TURN AROUND AT THE DISTRIBUTION PANEL OR
1277                    ;*      AT THE END OF THE MODEM CABLE.
1278                    ;*      THE H861 TURN AROUND IS USED FOR THE 16 LINE TURN AROUND.
1279                    ;*      IF THE H325 WAS SELECTED (A) THE FOLLOWING WILL BE TYPED
1280                    ;*      IF SW06=0:
1281                    ;*      SELECT LINE(S):  XXXXXXXXXXXXXXXX
1282                    ;*
1283                    ;*      THE FIRST ''X'' REPRESENTS LINE 15 AND EACH ''X'' IS THE
1284                    ;*      NEXT LOWER LINE TILL THE LAST ''X'' IS LINE 0. TYPE
1285                    ;*      A ''1'' OR A ''0'' UNDER THE APPROIATE ''X''(LINE)
1286                    ;*      TO EITHER SELECT(1) OR NOT TEST(0) EACH LINE.
1287                    ;*      AFTER ALL 1'S AND 0'S ARE TYPED; TYPE A <CR>.
1288                    ;*      THE PROGRAM WILL TYPE OUT IN OCTAL THE LINES YOU
1289                    ;*      HAVE SELECTED; AND THE PROGRAM WILL BEGIN RUNNING
1290                    ;*      THE HIGHEST SELECTED LINE THROUGH *ALL* TESTS THEN
1291                    ;*      UPDATING TO THE NEXT LOWEST LINE TILL ALL SELECTED
1292                    ;*      LINES ARE DONE. THEN THE PROGRAM WILL TYPE AN
1293                    ;*      ''END'' CHAR.  PLEASE READ  THE SECTION ON PASS COMPLETE
1294                    ;*              IN DOCUMENT.
1295                    ;*      IF THE H325 IS SELECTED AND SW06=1 THE FOLLOWING WILL BE TYPED:
1296                    ;*      SINGLE LINE:
1297                    ;*      THE USER MUST THEN TYPE IN A SINGLE LINE HE DESIRES (00-17) -OCTAL-
1298                    ;*      END PASS IS THE SAME.
1299                    ;*      REGUARDLESS OF WHICH CONNECTOR WAS SELECTED; THE
1300                    ;*      THE LAST QUESTION IS:
1301                    ;*      MODEM VECTOR:
1302                    ;*      (THIS WILL BE ASKED ONLY AT THE INIATL START OF PROGRAM
1303                    ;*      OR WHEN A DIFFERENT DV11 IN THE SYSTEM IS UNDER TEST)
1304                    ;*      TYPE IN THE VECTOR OF THE MODEM CONTROL(300:774).
1305                    ;*      THE CSR(MC.CSR) IS ASSUMED TO BE =DVSCR+20.
1306                    ;*      NOTE:   IF CABLE TESTS ARE TO BE DONE ON OTHER
1307                    ;*      DV11'S IN SYSTEM; SELECT THEM BY USING SW00 AS DESCRIBED
1308                    ;*      IN THE DOCUMENTATION.
1309                    ;*      UNLESS LOCATION 42 IS NON-ZERO IN WHICH CASE THE PROGRAM
1310                    ;*      ASSUMES ITS UNDER ACT-11 MONITOR.  THE PROGRAM WILL
1311                    ;*      CYCLE THROUGH ALL DV11S AND MODEM CONTROL *HOWEVER*
1312                    ;*      THE RESTRICTIONS ARE:
1313                    ;*      ***ALL*** MODEM VECTORS MUST BE AT 300
1314                    ;*      ***ALL*** TURN AROUNDS MUST BE H861.
1315                    ;*      ''LONG END PASS'' WILL BE GIVEN AT END OF LARGE END TO
1316                    ;*              INDICATE DEVICES TESTED. PASSES TYPED IN THIS
1317                    ;*              MESSAGE DO NOT INDICATE PASSES BUT RATHER THE
1318                    ;*              NUMBER OF FULL PASSES THROUGH MULTIPLE DEVICES.
1319                    ;*                      !LARGE END AND TYPE OUT MAY BE INHIBITED BY SW12.
1320                    ;*********************************************************************
```

```
1321
1322                                    ;  TEST 1
1323                                    ;-------------
1323  007310  012737  000001  001226  TST1:   MOV     #1,TSTNO
1324  007316  012737  010770  001216          MOV     #TST2,NEXT
1325  007324  005037  177776                  CLR     PS                  ;CLEAR CPU STATUS
1326  007330  013737  001362  007300          MOV     DVSCR,MC.CSR        ;GET MODEM CSR
1327  007336  062737  000020  007300          ADD     #20,MC.CSR          ;IT HAS TO BE 20(8) MORE THAN DVSCR.
1328  007344  013737  007300  007302          MOV     MC.CSR,MC.LSR       ;GET MODEM LSR
1329  007352  062737  000002  007302          ADD     #2,MC.LSR           ;MUST BE 2 MORE THAN CSR
1330  007360  012737  010276  000060          MOV     #KBISR,@#60         ;SET KEYBOARD INTERUPT VEC
1331  007366  012737  000340  000062          MOV     #340,@#62           ;SET LEV TO 7
1332  007374  012777  000100  171602          MOV     #100,@TKCSR         ;SET INTERUPT ENABLE
1333  007402  012737  000340  177776          MOV     #340,PS             ;LOCK OUT TTY
1334  007410  005737  000042                  TST     @#42
1335  007414  001020                          BNE     44$
1336  007416  104402  022672          1$:     TYPE    ,MTURN
1337  007422  004737  022760                  JSR     PC,TKRDY
1338  007426  122737  000101  001272          CMPB    #101,SAVR5
1339  007434  001004                          BNE     70$
1340  007436  012737  000377  007260          MOV     #377,TURFLG
1341  007444  000412                          BR      71$
1342  007446  122737  000102  001272  70$:    CMPB    #102,SAVR5
1343  007454  001360                          BNE     1$
1344  007456  005037  007260          44$:    CLR     TURFLG
1345  007462  012737  000001  007272          MOV     #1,SELECT
1346  007470  000523                          BR      68$
1347  007472  032777  000100  171502  71$:    BIT     #SW06,@SWR
1348  007500  001421                          BEQ     72$
1349          007502                  MAR18=.
1350  007502  104403  022234                  INSTR   ,MSING
1351  007506  104405                          PARAM
1352  007510  000000                          00
1353  007512  000017                          17
1354  007514  007262                          LINE
1355  007516     000     001                  .BYTE   0,1
1356  007520  012737  000001  007272          MOV     #1,SELECT
1357  007526  005337  007262          74$:    DEC     LINE
1358  007532  100502                          BMI     68$
1359  007534  000241                          CLC
1360  007536  006137  007272                  ROL     SELECT
1361  007542  000771                          BR      74$
1362  007544  104402  022121          72$:    TYPE    ,MSEL               ;ASK FOR LINES
1363  007550  013737  007272  001252          MOV     SELECT,TEMP3        ;GET PREVIOUS LINE SELECT
1364  007556  005037  007272                  CLR     SELECT              ;MAKE IT 0
1365  007562  105777  171416          2$:     TSTB    @TKCSR              ;READY?
1366  007566  100375                          BPL     2$                  ;BR IF NO
1367  007570  017700  171412                  MOV     @TKDBR,R0           ;READ CHAR
1368  007574  010077  171412                  MOV     R0,@TPDBR           ;ECHO CHAR
1369  007600  042700  177600                  BIC     #^C<177>,R0         ;STRIP ALL BUT DATA
1370  007604  022700  000123                  CMP     #123,R0             ;WAS IT ''S(AME)''
1371  007610  001004                          BNE     .+12                ;BR IF NO
1372  007612  013737  001252  007272          MOV     TEMP3,SELECT        ;RESTORE PREVIOUS LINES SELECTED
1373  007620  000415                          BR      4$                  ;GO ON
1374  007622  022700  000015                  CMP     #15,R0              ;WAS IT ''<CR>''
1375  007626  001412                          BEQ     4$                  ;BR IF YES
1376  007630  022700  000060                  CMP     #60,R0              ;WAS IT ''0''
```

```
1377  007634  001403                          BEQ     3$              ;BR IF YES
1378  007636  022700  000061                  CMP     #61,R0          ;WAS IT "1"
1379  007642  001265                          BNE     1$              ;BR IF NO. RETYPE MSG
1380  007644  006000                  3$:     ROR     R0              ;SHIFT THE BITS
1381  007646  006137  007272                  ROL     SELECT          ;BRING CARRY INTO SELECT
1382  007652  000743                          BR      2$              ;CONT.
1383  007654  005737  007272          4$:     TST     SELECT          ;ARE ANY LINES SELECTED?
1384  007660  001656                          BEQ     1$              ;BR IF NO. AND TYPE MSG
1385  007662  005037  001266                  CLR     SAVR3           ;SET TYPE OUT
1386  007666  013705  007272                  MOV     SELECT,R5       ;SAVE
1387  007672  104402  022202                  TYPE    ,MLINE          ;ALERT USER TO WHAT
1388  007676  005037  177776          65$:    CLR     PS              ;HE SELECTED
1389  007702  006005                  5$:     ROR     R5              ;
1390  007704  103002                          BCC     6$              ;
1391  007706  104411  023044                  CNVRT   ,XXLIN          ;
1392  007712  005237  001266          6$:     INC     SAVR3           ;
1393  007716  022737  000020  001266          CMP     #16.,SAVR3      ;
1394  007724  001366                          BNE     5$              ;
1395  007726  104402  022231                  TYPE    ,M.CRLF         ;
1396  007732  022700  000123                  CMP     #123,R0         ;
1397  007736  001427                          BEQ     69$             ;
1398  007740  005737  000042          68$:    TST     @#42
1399  007744  001016                          BNE     98$
1400  007746  022737                          CMP     (PC)+,@(PC)+
1401  007750  000000                  80$:    .WORD   0
1402  007752  001362                          DVSCR
1403  007754  001412                          BEQ     98$
1404  007756  104403  022737                  INSTR   ,MVECZ          ;
1405  007762  104405                          PARAM
1406  007764  000300                          300
1407  007766  000774                          774
1408  007770  007304                          MC.VEC
1409  007772      003     001                 .BYTE   3,1
1410  007774  013737  001362  007750          MOV     DVSCR,80$
1411  010002  013737  007304  007306  98$:    MOV     MC.VEC,MC.LVL   ;GET PRIORITY LEVEL
1412  010010  062737  000002  007306          ADD     #2,MC.LVL       ;UP IT.
1413  010016  012737  010330  007264  69$.    MOV     #TABLE,POINTER  ;
1414  010024  117737  177234  007270          MOVB    @POINTER,COUNT  ;
1415  010032  005237  007264                  INC     POINTER         ;
1416  010036  117737  177222  007266          MOVB    @POINTER,CHAR   ;
1417  010044  005237  007264                  INC     POINTER         ;
1418  010050  013737  007272  007274          MOV     SELECT,EXERCISE ;
1419  010056  012737  000020  007262          MOV     #20,LINE        ;
1420  010064  005337  007262          TESTER: DEC     LINE            ;
1421  010070  006337  007274                  ASL     EXERCISE        ;
1422  010074  103451                          BCS     2$              ;
1423  010076  001372                          BNE     TESTER          ;
1424  010100  112737  000377  001313          MOVB    #377,QV.FLG     ;
1425  010106  104402  007266                  TYPE    ,CHAR           ;
1426  010112  005337  007270                  DEC     COUNT           ;
1427  010116  001031                          BNE     3$              ;
1428  010120  117737  177140  007270          MOVB    @POINTER,COUNT  ;
1429  010126  001016                          BNE     4$              ;
1430  010130  005737  000042                  TST     42
1431  010134  001405                          BEQ     .+14
1432  010136  012737  002436  001214          MOV     #.EOP,RETURN
```

```
1433  010144  000177  171044                    JMP     @RETURN
1434  010150  012737  010330  007264            MOV     #TABLE,POINTER  ;
1435  010156  117737  177102  007270            MOVB    @POINTER,COUNT  ;
1436  010164  005237  007264            4$:     INC     POINTER         ;
1437  010170  117737  177070  007266            MOVB    @POINTER,CHAR
1438  010176  005237  007264                    INC     POINTER         ;
1439  010202  013737  007272  007274    3$:     MOV     SELECT,EXERCISE ;
1440  010210  012737  000020  007262            MOV     #20,LINE        ;
1441  010216  000722                            BR      TESTER          ;
1442  010220  012737  010770  001214    2$:     MOV     #TST2,RETURN    ;
1443  010226  013737  001214  001216            MOV     RETURN,NEXT     ;
1444  010234  005046                            CLR     -(SP)           ;SET FOR FAKE INTR
1445  010236  012746  010272                    MOV     #5$,-(SP)       ;SET FAKE PC OF INTR
1446  010242  032777  004000  170734            BIT     #BIT11,@TKCSR   ;TTY ACTIVE?
1447  010250  001374                            BNE     .-6             ;YES WAIT TILL DONE.
1448  010252  017746  170730                    MOV     @TKDBR,-(SP)
1449  010256  042716  000200                    BIC     #BIT7,(SP)      ;CLEAR PARITY
1450  010262  122726  000001                    CMPB    #1,(SP)+        ;WAS ^A (CHANGE LINES) HIT?
1451  010266  001403                            BEQ     KBISR           ;BR IF YES
1452  010270  022626                            CMP     (SP)+,(SP)+     ;BR TO KBISR NOT TAKEN
1453                                                                    ;POP FAKE INTR OFF STACK
1454  010272  000177  170716            5$:     JMP     @RETURN         ;
1455
1456  010276  010046                    KBISR:  MOV     R0,-(SP)        ;
1457  010300  017700  170702                    MOV     @TKDBR,R0       ;SAVE CHAR IN R0
1458  010304  042700  177600                    BIC     #^C<177>,R0     ;CLEAR ALL BUT DATA
1459  010310  022700  000001                    CMP     #1,R0           ;WAS IT <^A> (CNTRL A)?
1460  010314  001003                            BNE     1$              ;BR IF NO
1461  010316  012766  007502  000002            MOV     #MAR18,2(SP)    ;SET RETURN
1462  010324  012600                    1$:     MOV     (SP)+,R0        ;RESTORE R0
1463  010326  000002                            RTI                     ;CONT.
1464
1465  010330     001     015     002    TABLE:  .BYTE 1,15,2,12
      010334     010     040     012            .BYTE 8.,40,10.,105,4,40,2,116,6,40,2,116,4,40,8.,104
      010354     001     015     001            .BYTE 1,15,1,12
      010360     010     040     012            .BYTE 8.,40,10.,105,4,40,2,116,6,40,2,116,4,40,8.,104
      010400     001     015     001            .BYTE 1,15,1,12
      010404     010     040     002            .BYTE 8.,40,2,105,12.,40,2,116,6,40,2,116,4,40,2,104,6,40,2,104
      010430     001     015     001            .BYTE 1,15,1,12
      010434     010     040     002            .BYTE 8.,40,2,105,12.,40,2,116,6,40,2,116,4,40,2,104,6,40,2,104
      010460     001     015     001            .BYTE 1,15,1,12
      010464     010     040     002            .BYTE 8.,40,2,105,12.,40,4,116,4,40,2,116,4,40,2,104,6,40,2,104
      010510     001     015     001            .BYTE 1,15,1,12
      010514     010     040     002            .BYTE 8.,40,2,105,12.,40,4,116,4,40,2,116,4,40,2,104,6,40,2,104
      010540     001     015     001            .BYTE 1,15,1,12
      010544     010     040     010            .BYTE 8.,40,8.,105,6,40,2,116,2,40,2,116,2,40,2,116,4,40,2,104,6,40,2,104
      010574     001     015     001            .BYTE 1,15,1,12
      010600     010     040     010            .BYTE 8.,40,8.,105,6,40,2,116,2,40,2,116,2,40,2,116,4,40,2,104,6,40,2,104
      010630     001     015     001            .BYTE 1,15,1,12
      010634     010     040     002            .BYTE 8.,40,2,105,12.,40,2,116,4,40,4,116,4,40,2,104,6,40,2,104
      010660     001     015     001            .BYTE 1,15,1,12
      010664     010     040     002            .BYTE 8.,40,2,105,12.,40,2,116,4,40,4,116,4,40,2,104,6,40,2,104
      010710     001     015     001            .BYTE 1,15,1,12
      010714     010     040     012            .BYTE 8.,40,10.,105,4,40,2,116,6,40,2,116,4,40,8.,104
      010734     001     015     001            .BYTE 1,15,1,12
      010740     010     040     012            .BYTE 8.,40,10.,105,4,40,2,116,6,40,2,116,4,40,8.,104
```

```
010760  001  015  001          .BYTE 1,15,1,12
010764  000  000  000          .BYTE 0,0,0
        010770                  .EVEN
```

```
                        ;************************ TEST 2 *******************************
                        ;*INITIALIZATION CHECK
                        ;*VERIFY THAT CONTROL STATUS REGISTER AND LINE STATUS
                        ;*REGISTER WERE CLEARED BY INITIALIZE
                        ;;***************************************************************

                        ;  TEST 2
                        ;--------------
001000 010770 012737 000002 001226 TST2:   MOV     #2,TSTNO
001006 010776 012737 011126 001216         MOV     #TST3,NEXT
001014 011004 105777 170200                TSTB    @TPCSR          ;WAIT FOR TTY READY
001020 011010 100375                       BPL     .-4             ;BR IF NOT READY
001022 011012 000005                       RESET                   ;INIT
001024 011014 005005                       CLR     R5
001026 011016 052777 000100 170160         BIS     #100,@TKCSR     ;SET TTY INTERUPT ENABLE
001034 011024 012737 011114 000004         MOV     #1$,@#4         ;SET FOR NON-EX DEVICE.
001042 011032 012702 000010                MOV     #8.,R2          ;SET COUNTER
001046 011036 027777 170142 170140 65$:    CMP     @TKCSR,@TKCSR   ;WASTE TIME
001054 011044 027777 170134 170132         CMP     @TKCSR,@TKCSR   ;WASTE TIME
001062 011052 005302                       DEC     R2              ;DELAY DONE?
001064 011054 001370                       BNE     65$             ;BR IF NO
001066 011056 005200                       INC     R0              ;FLASH LIGHTS
001070 011060 013703 007300                MOV     MC.CSR,R3       ;SET MC.CSR POINTER
001074 011064 011304                       MOV     (R3),R4         ;READ REGISTER
001076 011066 001401                       BEQ     .+4
001100 011070 104002                       HLT     2               ;CONTROL STATUS NOT CLEARED, ERROR
001102 011072 013703 007302                MOV     MC.LSR,R3       ;SET POINTER
001106 011076 011304                       MOV     (R3),R4         ;READ MC.LSR
001110 011100 001401                       BEQ     .+4
001112 011102 104002                       HLT     2               ;LINE STATUS NOT CLEARED, ERROR
001114 011104 012737 000006 000004         MOV     #6,@#4          ;RESET TRAP CATCHER
001122 011112 104400                       SCOPE                   ;CHECK FOR LOOP
001124 011114 104005                1$:    HLT     5               ;SHOULD NOT TRAP.
001126 011116 012706 001200                MOV     #STACK,SP       ;
001132 011122 000177 170066                JMP     @RETURN

                        ;************************ TEST 3 *******************************
                        ;*VERIFY THAT ''INTERUPT ENABLE'' CAN BE
                        ;*SET AND CLEARED.
                        ;;***************************************************************

                        ;  TEST 3
                        ;--------------
001136 011126 012737 000003 001226 TST3:   MOV     #3,TSTNO
001144 011134 012737 011220 001216         MOV     #TST4,NEXT
001152 011142 013703 007300                MOV     MC.CSR,R3       ;SET POINTER TO MC.CSR
001156 011146 012713 000100                MOV     #INTENA,(R3)    ;LOAD FUNCTION
001162 011152 011304                       MOV     (R3),R4         ;READ RESULTS
001164 011154 042704 177677                BIC     #^C<INTENA>,R4  ;MASK OFF ALL OTHER BITS.
001170 011160 012705 000100                MOV     #INTENA,R5              ;MAKE R5=GOOD
001174 011164 020504                       CMP     R5,R4           ;RESULTS OK?
001176 011166 001401                       BEQ     .+4             ;BR IF YES
001200 011170 104002                       HLT     2               ;ERROR. R5=GOOD,R4=BAD,R3=REGISTER
001202 011172 042705 000100                BIC     #INTENA,R5
001206 011176 042713 000100                BIC     #INTENA,(R3)    ;CLEAR BIT
```

```
1522  011202  011304                  MOV     (R3),R4                 ;READ REGISTER
1523  011204  042704  177677          BIC     #^C<INTENA>,R4          ;MASK OFF ALL OTHER BITS.
1524  011210  020504                  CMP     R5,R4                   ;REGISTER OK?
1525  011212  001401                  BEQ     .+4                     ;BR IF YES
1526  011214  104002                  HLT     2                       ;BIT FAILED TO CLEAR
1527  011216  104400                  SCOPE                           ;SCOPE TEST.
1528
1529
1530                                  ;************************* TEST 4 ******************************
1531                                  ;*VERIFY THAT 'DONE' CAN BE
1532                                  ;*SET AND CLEARED.
1533                                  ;:**************************************************************
1534
1535                          ;  TEST 4
1536                          ;--------------
1537  011220  012737  000004  001226  TST4:   MOV     #4,TSTNO
1538  011226  012737  011312  001216          MOV     #TST5,NEXT
1539  011234  013703  007300                  MOV     MC.CSR,R3               ;SET POINTER TO MC.CSR
1540  011240  012713  000200                  MOV     #DONE,(R3)              ;LOAD FUNCTION
1541  011244  011304                          MOV     (R3),R4                 ;READ RESULTS
1542  011246  042704  177577                  BIC     #^C<DONE>,R4            ;MASK OFF ALL OTHER BITS.
1543  011252  012705  000200                  MOV     #DONE,R5                        ;MAKE R5-GOOD
1544  011256  020504                          CMP     R5,R4                   ;RESULTS OK?
1545  011260  001401                          BEQ     .+4                     ;BR IF YES
1546  011262  104002                          HLT     2                       ;ERROR. R5=GOOD,R4 BAD,R3 REGISTER
1547  011264  042705  000200                  BIC     #DONE,R5
1548  011270  042713  000200                  BIC     #DONE,(R3)              ;CLEAR BIT
1549  011274  011304                          MOV     (R3),R4                 ;READ REGISTER
1550  011276  042704  177577                  BIC     #^C<DONE>,R4            ;MASK OFF ALL OTHER BITS.
1551  011302  020504                          CMP     R5,R4                   ;REGISTER OK?
1552  011304  001401                          BEQ     .+4                     ;BR IF YES
1553  011306  104002                          HLT     2                       ;BIT FAILED TO CLEAR
1554  011310  104400                          SCOPE                           ;SCOPE TEST.
1555
1556
1557                                  ;************************* TEST 5 ******************************
1558                                  ;*VERIFY THAT 'MAINTENANCE MODE' CAN BE
1559                                  ;*SET AND CLEARED.
1560                                  ;:**************************************************************
1561
1562                          ;  TEST 5
1563                          ;--------------
1564  011312  012737  000005  001226  TST5:   MOV     #5,TSTNO
1565  011320  012737  011404  001216          MOV     #TST6,NEXT
1566  011326  013703  007300                  MOV     MC.CSR,R3               ;SET POINTER TO MC.CSR
1567  011332  012713  001000                  MOV     #MAINT,(R3)             ;LOAD FUNCTION
1568  011336  011304                          MOV     (R3),R4                 ;READ RESULTS
1569  011340  042704  176777                  BIC     #^C<MAINT>,R4           ;MASK OFF ALL OTHER BITS.
1570  011344  012705  001000                  MOV     #MAINT,R5                       ;MAKE R5=GOOD
1571  011350  020504                          CMP     R5,R4                   ;RESULTS OK?
1572  011352  001401                          BEQ     .+4                     ;BR IF YES
1573  011354  104002                          HLT     2                       ;ERROR. R5=GOOD,R4=BAD,R3=REGISTER
1574  011356  042705  001000                  BIC     #MAINT,R5
1575  011362  042713  001000                  BIC     #MAINT,(R3)             ;CLEAR BIT
1576  011366  011304                          MOV     (R3),R4                 ;READ REGISTER
1577  011370  042704  176777                  BIC     #^C<MAINT>,R4           ;MASK OFF ALL OTHER BITS.
```

```
                                        CMP     R5,R4                   ;REGISTER OK?
                                        BEQ     .+4                     ;BR IF YES
                                        HLT     2                       ;BIT FAILED TO CLEAR
                                        SCOPE                           ;SCOPE TEST.


                                        ;************************ TEST 6 *******************************
                                        ;*VERIFY THAT ''SCAN ENABLE'' CAN BE
                                        ;*SET AND CLEARED.
                                        ;;***************************************************************

                                    ;  TEST 6
                                    ;--------------
011404  012737  000006  001226  TST6:   MOV     #6,TSTNO
011412  012737  011476  001216          MOV     #TST7,NEXT
011420  013703  007300                  MOV     MC.CSR,R3               ;SET POINTER TO MC.CSR
011424  012713  000040                  MOV     #SCNENA,(R3)            ;LOAD FUNCTION
011430  011304                          MOV     (R3),R4                 ;READ RESULTS
011432  042704  177737                  BIC     #^C<SCNENA>,R4          ;MASK OFF ALL OTHER BITS.
011436  012705  000040                  MOV     #SCNENA,R5                      ;MAKE R5-GOOD
011442  020504                          CMP     R5,R4                   ;RESULTS OK?
011444  001401                          BEQ     .+4                     ;BR IF YES
011446  104002                          HLT     2                       ;ERROR. R5=GOOD,R4=BAD,R3=REGISTER
011450  042705  000040                  BIC     #SCNENA,R5
011454  042713  000040                  BIC     #SCNENA,(R3)            ;CLEAR BIT
011460  011304                          MOV     (R3),R4                 ;READ REGISTER
011462  042704  177737                  BIC     #^C<SCNENA>,R4          ;MASK OFF ALL OTHER BITS.
011466  020504                          CMP     R5,R4                   ;REGISTER OK?
011470  001401                          BEQ     .+4                     ;BR IF YES
011472  104002                          HLT     2                       ;BIT FAILED TO CLEAR
011474  104400                          SCOPE                           ;SCOPE TEST.
```

```
                                        ;************************ TEST 7 *******************************
                                        ;*VERIFY THAT 'BUSY'' IS SET WHEN ''SCAN ENABLE'' IS SET
                                        ;*VERIFY THAT 'BUSY'' IS CLEARED WHEN ''SCAN ENABLE'' IS CLEARED
                                        ;;*************************************************************************

                                ;  TEST 7
                                ;-------------
011476  012737  000007  001226  TST7:   MOV     #7,TSTNO
011504  012737  011600  001216          MOV     #TST10,NEXT
011512  013703  007300                  MOV     MC.CSR,R3               ;SET REGISTER POINTER
011516  012713  000040                  MOV     #SCNENA,(R3)            ;SET SCAN ENABLE
011522  011304                          MOV     (R3),R4                 ;READ REGISTER
011524  010405                          MOV     R4,R5                   ;GET IMAGE
011526  052705  000020                  BIS     #BUSY,R5                ;SET BUSY BIT IN GOOD.
011532  020504                          CMP     R5,R4                   ;REGISTER OK?
011534  001401                          BEQ     .+4
011536  104002                          HLT     2                       ;BUSY NOT SET, ERROR
011540  042713  000040                  BIC     #SCNENA,(R3)            ;CLEAR SCAN ENABLE
011544  023737  000000  000000          CMP     0,0                     ;GIVE BUSY A CHANCE TO CLEAR
011552  023737  000000  000000          CMP     0,0                     ;WHEN ON A HOT ROD MACHINE (11/70)!
011560  011304                          MOV     (R3),R4                 ;READ MC.CSR
011562  010405                          MOV     R4,R5                   ;GET IMAGE
011564  042705  000020                  BIC     #BUSY,R5                ;CLEAR BUSY IN GOOD.
011570  020504                          CMP     R5,R4                   ;BUSY CLEARED?
011572  001401                          BEQ     .+4
011574  104002                          HLT     2                       ;BUSY NOT CLEARED, ERROR
011576  104400                          SCOPE                           ;CHECK FOR LOOP, ITERATIONS

                                        ;************************ TEST 10 ******************************
                                        ;*VERIFY THAT SETTING 'DONE'' DOES NOT CAUSE AN
                                        ;*INTERRUPT IF ''INTERRUPT ENABLE'' IS CLEARED.
                                        ;;*************************************************************************

                                ;  TEST 10
                                ;--------------
011600  012737  000010  001226  TST10:  MOV     #10,TSTNO
011606  012737  011666  001216          MOV     #TST11,NEXT
011614  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
011622  005077  175452                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
011626  012777  011660  175450          MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
011634  012777  000340  175444          MOV     #340,@MC.LVL            ;SET UP INTERRUPT PRIORITY
011642  052777  000200  175430          BIS     #DONE,@MC.CSR           ;SET DONE
011650  005037  177776                  CLR     PS                      ;ALLOW INTERRUPTS
011654  000240                          NOP                             ;DELAY FOR INTERRUPT
011656  000402                          BR      2$                      ;NO INTERRUPT, CONTINUE
011660  022626                  1$:     POP2SP                          ;RESTORE STACK, INTERRUPT
011662  104003                          HLT     3                       ;OCCURED, ERROR
011664  104400                  2$:     SCOPE                           ;CHECK FOR LOOP, ITERATIONS
```

```
                                        ;************************ TEST 11 *******************************
                                        ;*VERIFY THAT NO INTERRUPT OCCURS WITH ''INTERRUPT ENABLE''
                                        ;*SET AND 'DONE'' CLEARED.
                                        ;;*************************************************************************

                                        ;  TEST 11
                                        ;--------------
011666  012737  000011  001226  TST11:  MOV     #11,TSTNO
011674  012737  011754  001216          MOV     #TST12,NEXT
011702  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
011710  005077  175364                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
011714  012777  011746  175362          MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
011722  012777  000340  175356          MOV     #340,@MC.LVL            ;SET UP INTERRUPT SERVICE LEVEL
011730  052777  000100  175342          BIS     #INTENA,@MC.CSR         ;SET INTERRUPT ENABLE
011736  005037  177776                  CLR     PS                      ;ALLOW INTERRUPTS
011742  000240                          NOP                             ;DELAY FOR INTERRUPTS
011744  000402                          BR      2$                      ;NO INTERRUPT, CONTINUE
011746  022626                  1$:     POP2SP                          ;RESTORE STACK
011750  104003                          HLT     3                       ;INTERRUPT OCCURED, ERROR
011752  104400                  2$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP

                                        ;************************ TEST 12 *******************************
                                        ;*VERIFY THAT SETTING 'DONE'' CAUSES AN INTERRUPT
                                        ;*WITH ''INTERRUPT ENABLE'' SET
                                        ;;*************************************************************************

                                        ;  TEST 12
                                        ;--------------
011754  012737  000012  001226  TST12:  MOV     #12,TSTNO
011762  012737  012050  001216          MOV     #TST13,NEXT
011770  012737  000340  177776          MOV     #340,PS                 ;LOCK OUT INTERRUPTS
011776  005077  175276                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
012002  012777  012044  175274          MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
012010  012777  000100  175262          MOV     #INTENA,@MC.CSR         ;SET ''INTERRUPT ENABLE''
012016  012777  000340  175262          MOV     #340,@MC.LVL            ;SET ''INTERRUPT LEVEL''
012024  005037  177776                  CLR     PS                      ;ALLOW INTERRUPTS
012030  052777  000200  175242          BIS     #DONE,@MC.CSR           ;SET 'DONE''
012036  000240                          NOP                             ;DELAY FOR INTERRUPT
012040  104004                          HLT     4                       ;INTERRUPT OCCURED, ERROR
012042  000401                          BR      2$                      ;CONTINUE
012044  022626                  1$:     POP2SP                          ;INTERRUPT OCCURED, RESTOR STACK
012046  104400                  2$:     SCOPE                           ;CHECK FOR ITERATION, LOOP
```

```
1702
1703                                    ;************************ TEST 13 *******************************
1704                                    ;*VERIFY THAT NO INTERRUPT OCCURS WITH
1705                                    ;*''INTERRUPT ENABLE'' SET AND 'DONE'' SET AT PRIORITY 7.
1706                                    ;:***************************************************************
1707
1708                                    ;  TEST 13
1709                                    ;--------------
1710  012050  012737  000013  001226  TST13:  MOV     #13,TSTNO
1711  012056  012737  012140  001216          MOV     #TST14,NEXT
1712  012064  005077  175210                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
1713  012070  012737  000340  177776          MOV     #340,PS                 ;TO LEVEL 7.
1714  012076  012777  012132  175200          MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
1715  012104  012777  000340  175174          MOV     #340,@MC.LVL            ;SET UP INTERRUPT SERVICE LEVEL
1716  012112  012777  000100  175160          MOV     #INTENA,@MC.CSR         ;SET INTERRUPT ENABLE
1717  012120  052777  000200  175152          BIS     #DONE,@MC.CSR           ;GENERATE INTERRUPT
1718  012126  000240                          NOP                             ;DELAY FOR INTERRUPT
1719  012130  000402                          BR      2$                      ;NO INTERRUPT, CONTINUE
1720  012132  022626                  1$:     POP2SP                          ;RESTORE STACK
1721  012134  104003                          HLT     3                       ;INTERRUPT OCCURED, ERROR
1722  012136  104400                  2$:     SCOPE                           ;CHECK FOR ITERATION, LOOP
1723
1724                                    ;************************ TEST 14 *******************************
1725                                    ;*VERIFY THAT NO INTERRUPT OCCURS WITH
1726                                    ;*''INTERRUPT ENABLE'' SET AND 'DONE'' SET AT PRIORITY 6.
1727                                    ;:***************************************************************
1728
1729                                    ;  TEST 14
1730                                    ;--------------
1731  012140  012737  000014  001226  TST14:  MOV     #14,TSTNO
1732  012146  012737  012230  001216          MOV     #TST15,NEXT
1733  012154  005077  175120                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
1734  012160  012737  000300  177776          MOV     #300,PS                 ;TO LEVEL 6.
1735  012166  012777  012222  175110          MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
1736  012174  012777  000300  175104          MOV     #300,@MC.LVL            ;SET UP INTERRUPT SERVICE LEVEL
1737  012202  012777  000100  175070          MOV     #INTENA,@MC.CSR         ;SET INTERRUPT ENABLE
1738  012210  052777  000200  175062          BIS     #DONE,@MC.CSR           ;GENERATE INTERRUPT
1739  012216  000240                          NOP                             ;DELAY FOR INTERRUPT
1740  012220  000402                          BR      2$                      ;NO INTERRUPT, CONTINUE
1741  012222  022626                  1$:     POP2SP                          ;RESTORE STACK
1742  012224  104003                          HLT     3                       ;INTERRUPT OCCURED, ERROR
1743  012226  104400                  2$:     SCOPE                           ;CHECK FOR ITERATION, LOOP
```

```
1744
1745                                    ;************************ TEST 15 *******************************
1746                                    ;*VERIFY THAT NO INTERRUPT OCCURS WITH
1747                                    ;*''INTERRUPT ENABLE'' SET AND 'DONE'' SET AT PRIORITY 5.
1748                                    ;;*************************************************************************
1749
1750                                    ;  TEST 15
1751                                    ;--------------
1752  012230  012737  000015  001226  TST15:  MOV     #15,TSTNO
1753  012236  012737  012320  001216          MOV     #TST16,NEXT
1754  012244  005077  175030                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
1755  012250  012737  000240  177776          MOV     #240,PS                 ;TO LEVEL 5.
1756  012256  012777  012312  175020          MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
1757  012264  012777  000240  175014          MOV     #240,@MC.LVL            ;SET UP INTERRUPT SERVICE LEVEL
1758  012272  012777  000100  175000          MOV     #INTENA,@MC.CSR         ;SET INTERRUPT ENABLE
1759  012300  052777  000200  174772          BIS     #DONE,@MC.CSR           ;GENERATE INTERRUPT
1760  012306  000240                          NOP                             ;DELAY FOR INTERRUPT
1761  012310  000402                          BR      2$                      ;NO INTERRUPT, CONTINUE
1762  012312  022626                  1$:     POP2SP                          ;RESTORE STACK
1763  012314  104003                          HLT     3                       ;INTERRUPT OCCURED, ERROR
1764  012316  104400                  2$:     SCOPE                           ;CHECK FOR ITERATION, LOOP
1765
1766                                    ;************************ TEST 16 *******************************
1767                                    ;*VERIFY THAT NO INTERRUPT OCCURS WITH
1768                                    ;*''INTERRUPT ENABLE'' SET AND 'DONE'' SET AT PRIORITY 4.
1769                                    ;;*************************************************************************
1770
1771                                    ;  TEST 16
1772                                    ;--------------
1773  012320  012737  000016  001226  TST16:  MOV     #16,TSTNO
1774  012326  012737  012410  001216          MOV     #TST17,NEXT
1775  012334  005077  174740                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
1776  012340  012737  000200  177776          MOV     #200,PS                 ;TO LEVEL 4.
1777  012346  012777  012402  174730          MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
1778  012354  012777  000200  174724          MOV     #200,@MC.LVL            ;SET UP INTERRUPT SERVICE LEVEL
1779  012362  012777  000100  174710          MOV     #INTENA,@MC.CSR         ;SET INTERRUPT ENABLE
1780  012370  052777  000200  174702          BIS     #DONE,@MC.CSR           ;GENERATE INTERRUPT
1781  012376  000240                          NOP                             ;DELAY FOR INTERRUPT
1782  012400  000402                          BR      2$                      ;NO INTERRUPT, CONTINUE
1783  012402  022626                  1$:     POP2SP                          ;RESTORE STACK
1784  012404  104003                          HLT     3                       ;INTERRUPT OCCURED, ERROR
1785  012406  104400                  2$:     SCOPE                           ;CHECK FOR ITERATION, LOOP
```

```
                                        ;************************ TEST 17 *******************************
                                        ;*VERIFY THAT AN INTERRUPT OCCURS WITH ''INTERRUPT
                                        ;*ENABLE'' SET AND 'DONE'' SET AT PRIORITY 0.
                                        ;;*****************************************************************

                                ;  TEST 17
                                ;--------------
012410  012737  000017  001226 TST17:  MOV     #17,TSTNO
012416  012737  012476  001216         MOV     #TST20,NEXT
012424  005077  174650                 CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
012430  012777  012472  174646         MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
012436  005077  174644                 CLR     @MC.LVL                 ;SET UP INTERRUPT SERVICE PRIORITY
012442  012737  000000  177776         MOV     #0,PS                   ;SET PROCESSOR PRIORITY TO LEVEL 0.
012450  012777  000100  174622         MOV     #INTENA,@MC.CSR         ;SET INTERRUPT ENABLE
012456  052777  000200  174614         BIS     #DONE,@MC.CSR           ;GENERATE INTERRUPT
012464  000240                         NOP                             ;WAIT FOR INTERRUPT
012466  104004                         HLT     4                       ;NO INTERRUPT, ERROR.
012470  000401                         BR      2$                      ;CONTINUE
012472  022626                 1$:     POP2SP                          ;INTERRUPT OCCURED, RESTORE STACK
012474  104400                 2$:     SCOPE                           ;CHECK FOR INTERATIONS, LOOP.

                                        ;************************ TEST 20 *******************************
                                        ;*VERIFY THAT AN INTERRUPT OCCURS WITH ''INTERRU
                                        ;*ENABLE'' SET AND 'DONE'' SET AT PRIORITY 1.
                                        ;;*****************************************************************

                                ;  TEST 20
                                ;--------------
012476  012737  000020  001226 TST20:  MOV     #20,TSTNO
012504  012737  012564  001216         MOV     #TST21,NEXT
012512  005077  174562                 CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
012516  012777  012560  174560         MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
012524  005077  174556                 CLR     @MC.LVL                 ;SET UP INTERRUPT SERVICE PRIORITY
012530  012737  000040  177776         MOV     #40,PS                  ;SET PROCESSOR PRIORITY TO LEVEL 1.
012536  012777  000100  174534         MOV     #INTENA,@MC.CSR         ;SET INTERRUPT ENABLE
012544  052777  000200  174526         BIS     #DONE,@MC.CSR           ;GENERATE INTERRUPT
012552  000240                         NOP                             ;WAIT FOR INTERRUPT
012554  104004                         HLT     4                       ;NO INTERRUPT, ERROR.
012556  000401                         BR      2$                      ;CONTINUE
012560  022626                 1$:     POP2SP                          ;INTERRUPT OCCURED, RESTORE STACK
012562  104400                 2$:     SCOPE                           ;CHECK FOR INTERATIONS, LOOP.
```

```
                                        ;************************ TEST 21 *********************************
                                        ;*VERIFY THAT AN INTERRUPT OCCURS WITH ''INTERRUPT
                                        ;*ENABLE'' SET AND 'DONE'' SET AT PRIORITY 2.
                                        ;;*************************************************************************

                                ;  TEST 21
                                ;-------------
012564  012737  000021  001226  TST21:  MOV     #21,TSTNO
012572  012737  012652  001216          MOV     #TST22,NEXT
012600  005077  174474                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
012604  012777  012646  174472          MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
012612  005077  174470                  CLR     @MC.LVL                 ;SET UP INTERRUPT SERVICE PRIORITY
012616  012737  000100  177776          MOV     #100,PS                 ;SET PROCESSOR PRIORITY TO LEVEL 2.
012624  012777  000100  174446          MOV     #INTENA,@MC.CSR         ;SET INTERRUPT ENABLE
012632  052777  000200  174440          BIS     #DONE,@MC.CSR           ;GENERATE INTERRUPT
012640  000240                          NOP                             ;WAIT FOR INTERRUPT
012642  104004                          HLT     4                       ;NO INTERRUPT, ERROR.
012644  000401                          BR      2$                      ;CONTINUE
012646  022626                  1$:     POP2SP                          ;INTERRUPT OCCURED, RESTORE STACK
012650  104400                  2$:     SCOPE                           ;CHECK FOR INTERATIONS, LOOP.

                                        ;************************ TEST 22 *********************************
                                        ;*VERIFY THAT AN INTERRUPT OCCURS WITH ''INTERRUPT
                                        ;*ENABLE'' SET AND 'DONE'' SET AT PRIORITY 3.
                                        ;;*************************************************************************

                                ;  TEST 22
                                ;-------------
012652  012737  000022  001226  TST22:  MOV     #22,TSTNO
012660  012737  012740  001216          MOV     #TST23,NEXT
012666  005077  174406                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
012672  012777  012734  174404          MOV     #1$,@MC.VEC             ;SET UP INTERRUPT SERVICE ADDRESS
012700  005077  174402                  CLR     @MC.LVL                 ;SET UP INTERRUPT SERVICE PRIORITY
012704  012737  000140  177776          MOV     #140,PS                 ;SET PROCESSOR PRIORITY TO LEVEL 3.
012712  012777  000100  174360          MOV     #INTENA,@MC.CSR         ;SET INTERRUPT ENABLE
012720  052777  000200  174352          BIS     #DONE,@MC.CSR           ;GENERATE INTERRUPT
012726  000240                          NOP                             ;WAIT FOR INTERRUPT
012730  104004                          HLT     4                       ;NO INTERRUPT, ERROR.
012732  000401                          BR      2$                      ;CONTINUE
012734  022626                  1$:     POP2SP                          ;INTERRUPT OCCURED, RESTORE STACK
012736  104400                  2$:     SCOPE                           ;CHECK FOR INTERATIONS, LOOP.
```

```
1870                                    ;************************ TEST 23 *******************************
1871                                    ;*VERIFY THAT ALL LINE NUMBERS CAN BE WRITTEN INTO AND
1872                                    ;*READ BACK FROM LINE COUNTER
1873                                    ;:***************************************************************
1874
1875                                    ;  TEST 23
1876                                    ;--------------
1877  012740  012737  000023  001226    TST23:  MOV     #23,TSTNO
1878  012746  012737  013026  001216            MOV     #TST24,NEXT
1879  012754  012737  013002  001220            MOV     #1$,LOCK
1880  012762  013703  007300                    MOV     MC.CSR,R3               ;SET POINTER
1881  012766  005013                            CLR     (R3)                    ;CLEAR CONTROL STATUS REGISTER
1882  012770  005037  177776                    CLR     PS                      ;ENABLE INTERRUPTS
1883  012774  005005                            CLR     R5                      ;CLEAR EXPECTED LINE NUMBER
1884  012776  012700  000020                    MOV     #16.,R0                 ;SET UP TO TEST 16 LINE NUMBERS
1885  013002  010513                    1$:     MOV     R5,(R3)                 ;SET LINE NUMBER
1886  013004  011304                            MOV     (R3),R4                 ;READ BACK LINE NUMBER
1887  013006  020504                            CMP     R5,R4                   ;ARE EXPECTED AND RECEIVED
1888  013010  001401                            BEQ     2$                      ;LINE NUBMERS THE SAME
1889  013012  104002                            HLT     2                       ;LINE NUMBERS DIFFERENT, ERROR
1890  013014  104401                    2$:     SCOP1                           ;CHECK FOR DATA FREEZE
1891  013016  005205                            INC     R5                      ;UPDATE LINE COUNT
1892  013020  005300                            DEC     R0                      ;UPDATE LINE NUMBER
1893  013022  001367                            BNE     1$                      ;CONTINUE
1894  013024  104400                            SCOPE                           ;CHECK FOR ITERATION, LOOP
1895
1896                                    ;************************ TEST 24 *******************************
1897                                    ;*USING "STEP" MODE, VERIFY THAT THE
1898                                    ;*LINE COUNTER CAN BE STEPPED THRU ALL STATES.
1899                                    ;:***************************************************************
1900
1901                                    ;  TEST 24
1902                                    ;--------------
1903  013026  012737  000024  001226    TST24:  MOV     #24,TSTNO
1904  013034  012737  013124  001216            MOV     #TST25,NEXT
1905  013042  012737  013054  001220            MOV     #1$,LOCK
1906  013050  013703  007300                    MOV     MC.CSR,R3               ;SET POINTER
1907  013054  005037  177776            1$:     CLR     PS                      ;ENABLE INTERRUPTS
1908  013060  005013                            CLR     (R3)                    ;CLEAR CONTROL STATUS REGISTER
1909  013062  005005                            CLR     R5                      ;CLEAR EXPECTED LINE COUNT
1910  013064  012700  000020                    MOV     #16.,R0                 ;SET UP TO TEST 16 VALUES
1911  013070  012713  000017                    MOV     #17,(R3)                ;FIRST VALUE =0
1912  013074  052713  000400            2$:     BIS     #STEP,(R3)              ;STEP LINE COUNTER
1913  013100  104414                            DELAY
1914  013102  011304                            MOV     (R3),R4                 ;READ LINE COUNTER
1915  013104  020504                            CMP     R5,R4                   ;COMPARE EXPECTED AND
1916  013106  001401                            BEQ     3$                      ;RECEIVED LINE NUMBERS
1917  013110  104002                            HLT     2                       ;LINE COUNTER ERROR
1918  013112  104401                    3$:     SCOP1                           ;CHECK FOR DATA FREEZE
1919  013114  005205                            INC     R5                      ;UPDATE EXPECTED LINE NUMBER
1920  013116  005300                            DEC     R0
1921  013120  001365                            BNE     2$
1922  013122  104400                            SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                        ;************************ TEST 25 *******************************
                        ;*WRITE 1'S INTO ALL SCANNER MEMORY LOCATIONS.
                        ;*VERIFY THAT ALL LOCATIONS HAVE BEEN WRITTEN
                        ;*TO 1'S.
                        ;*VERIFY THAT ''CLEAR SCAN'' CLEARS ALL SCANNER
                        ;*MEMORY LOCATIONS.
                        ;:************************************************************************

                        ;  TEST 25
                        ;-------------
013124  012737  000025  001226  TST25:  MOV     #25,TSTNO
013132  012737  013326  001216          MOV     #TST26,NEXT
013140  012737  013152  001220          MOV     #1$,LOCK
013146  013703  007300                  MOV     MC.CSR,R3               ;SET POINTER
013152  012713  002000          1$:     MOV     #CLRMUX,(R3)            ;CLEAR CONTROL STATUS REGISTER
013156  005037  177776                  CLR     PS                      ;ENABLE INTERRUPTS
013162  012700  000020                  MOV     #16.,R0                 ;SET UP TO TEST 16 LOCATIONS
013166  052713  001017                  BIS     #MAINT+17,(R3)          ;SET MAINTEANCE MODE
013172  052713  000400          2$:     BIS     #STEP,(R3)              ;SET LINE COUNTER THRU ALL
013176  005300                          DEC     R0                      ;STATES, WRITING 1'S INTO
013200  001374                          BNE     2$                      ;ALL MEMORY WORDS
013202  012700  000020                  MOV     #16.,R0                 ;SET UP TO TEST 16 WORDS
013206  012705  070000                  MOV     #70000,R5               ;SET UP EXPECTED STATUS REGISTER
013212  012713  000017                  MOV     #17,(R3)                ;START WITH LINE 0
013216  052713  000400          3$:     BIS     #STEP,(R3)              ;ACCESS SCANNER MEMORY
013222  104414                          DELAY
013224  011304                          MOV     (R3),R4                 ;READ DATA
013226  020504                          CMP     R5,R4                   ;COMPARE EXPECTED AND RECEIVED
013230  001401                          BEQ     4$                      ;DATA
013232  104002                          HLT     2                       ;CONTROL STATUS OR MEMORY ERROR
013234  104401                  4$:     SCOP1                           ;CHECK FOR DATA FREEZE
013236  005205                          INC     R5                      ;UPDATE EXPECTED STATUS
013240  005300                          DEC     R0                      ;UPDATE LINE COUNT
013242  001365                          BNE     3$                      ;CONTINUE
013244  012737  013252  001220          MOV     #5$,LOCK                ;SET RETURN
013252  012713  004000          5$:     MOV     #CLRSCN,(R3)            ;SET ''CLEAR SCAN''
013256  032713  000020                  BIT     #BUSY,(R3)              ;WAIT FOR ''CLEAR CYCLES''
013262  001375                          BNE     .-4
013264  012700  000020                  MOV     #16.,R0                 ;SET UP TO TEST 16 MEMORY
013270  005005                          CLR     R5                      ;LOCATIONS
013272  012713  000017                  MOV     #17,(R3)                ;FIRST TO BE TESTED=0
013276  052713  000400          6$:     BIS     #STEP,(R3)              ;ACCESS SEANNER MEMORY
013302  104414                          DELAY
013304  011304                          MOV     (R3),R4                 ;READ DATA
013306  020504                          CMP     R5,R4                   ;COMPARE EXPECTED AND RECEIVED
013310  001402                          BEQ     7$                      ;DATA
013312  104002                          HLT     2                       ;CONTROL  STATUS OF MEMORY ERROR
013314  104401                          SCOP1                           ;CHECK FOR DATA FREEZE
013316  005205                  7$:     INC     R5                      ;UPDATE EXPECTED DATA
013320  005300                          DEC     R0                      ;UPDATE LINE COUNT
013322  001365                          BNE     6$                      ;CONTTINUE
013324  104400                          SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                        ;************************ TEST 26 *********************************
                        ;*WRITE 1'S INTO SELECTED SCANNER MEMORY LOCATION.
                        ;*VERIFY THAT ONLY SELECTED LOCATION WAS WRITTEN INTO.
                        ;:*************************************************************************

                        ;  TEST 26
                        ;-------------
013326  012737  000026  001226  TST26:  MOV     #26,TSTNO
013334  012737  013510  001216         MOV     #TST27,NEXT
013342  012737  013354  001220         MOV     #1$,LOCK
013350  013703  007300                 MOV     MC.CSR,R3               ;SET POINTER
013354  005013                  1$:    CLR     (R3)                    ;CLEAR CONTROL STATUS REGISTER
013356  005037  177776                 CLR     PS                      ;ENABLE INTERRUPTS
013362  012700  000020                 MOV     #16.,R0                 ;SET UP TO TEST 16 ADDRESSES
013366  012702  000017                 MOV     #17,R2                  ;FIRST ADDRESS TO BE TESTED=0
013372  012713  004000          2$:    MOV     #CLRSCN,(R3)            ;CLEAR ACANNER MEMORY
013376  032713  000020                 BIT     #BUSY,(R3)              ;WAIT FOR CLEAR CYCLE
013402  001375                       BNE     .-4
013404  012713  001000                 MOV     #MAINT,(R3)             ;SET 'MAINTENANCE MODE''
013410  050213                       BIS     R2,(R3)                 ;SET LINE COUNTER TO TEST ADDRESS-1
013412  052713  000400                 BIS     #STEP,(R3)              ;WRITE 1'S INTO TEST ADDRESS
013416  042713  001000                 BIC     #MAINT,(R3)             ;CLEAR 'MAINTENANCE MODE''
013422  012737  000020  001252         MOV     #16.,TEMP3              ;SET UP TO TEST ALL 16
013430  012713  000017                 MOV     #17,(R3)                ;SCANNER MEMORY LOCATIONS
013434  005202                       INC     R2
013436  005001                       CLR     R1
013440  052713  000400          3$:    BIS     #STEP,(R3)              ;ACCESS SCANNER MEMORY
013444  104414                       DELAY
013446  111304                       MOVB    (R3),R4                 ;READ CONPENTS OF MEMORY
013450  010105                       MOV     R1,R5                   ;SET UP EXPECTED CONTENTS
013452  120402                       CMPB    R4,R2                   ;OF SCANNER MEMORY
013454  001002                       BNE     4$
013456  052705  070000                 BIS     #70000,R5
013462  020405                  4$:    CMP     R4,R5                   ;COMPARE EXPECTED AND RECEIVED
013464  001402                       BEQ     5$                      ;VALUES
013466  104002                       HLT     2                       ;SCANNER MEMORY ERROR
013470  104401                       SCOP1                           ;CHECK FOR DATA FREEZE
013472  005201                  5$:    INC     R1
013474  005337  001252               DEC     TEMP3                   ;TEST NEXT SCANNEB LOCATION
013500  001357                       BNE     3$
013502  005300                       DEC     R0                      ;UPDATE LINE COUNT
013504  001332                       BNE     2$
013506  104400                       SCOPE                           ;CHECK FOR ITERATION, LOOP
```

```
                                        ;************************ TEST 27 *******************************
                                        ;*WITH ALL SCANNER MEMORY LOCATIONS SET TO 1'S,
                                        ;*WRITE 0'S INTO SELECTED LOCATION
                                        ;*VERIFY THAT ONLY SELECTED LOCATION WAS CLEARED.
                                        ;:*************************************************************************

                                        ;  TEST 27
                                        ;--------------
013510  012737  000027  001226  TST27:  MOV     #27,TSTNO
013516  012737  013674  001216         MOV     #TST30,NEXT
013524  012737  013554  001220         MOV     #2$,LOCK
013532  013703  007300                 MOV     MC.CSR,R3               ;SET POINTER
013536  005013                  1$:    CLR     (R3)                    ;CLEAR CONTROL STATUS REGISTER
013540  005037  177776                 CLR     PS                      ;ENABLE INTERRUPTS
013544  012700  000020                 MOV     #16.,R0                 ;SET UP TO TEST 16 ADDRESSES
013550  012702  000017                 MOV     #17,R2                  ;FIRST ADDRESS TO BE TESTED=0
013554  012737  000020  001252  2$:    MOV     #16.,TEMP3              ;WRITE 1'S INTO ALL SCANNER
013562  012713  001017                 MOV     #MAINT+17,(R3)  ;MEMORY LOCATIONS
013566  052713  000400           3$:    BIS     #STEP,(R3)
013572  005337  001252                 DEC     TEMP3
013576  001373                       BNE     3$
013600  010213                       MOV     R2,(R3)                 ;SET LINE COUNTER TO TEST ADDRESS-1
013602  052713  000400                 BIS     #STEP,(R3)              ;WRITE 0'S INTO TEST ADDRESS
013606  012737  000020  001252         MOV     #16.,TEMP3              ;SET UP TO TEST ALL 16
013614  012713  000017                 MOV     #17,(R3)                ;SCANNER MEMORY LOCATIONS
013620  005202                       INC     R2
013622  005001                       CLR     R1
013624  052713  000400           4$:    BIS     #STEP,(R3)              ;ACESS SCANNER MEMORY
013630  104414                       DELAY
013632  111304                       MOVB    (R3),R4                 ;READ CONTENTS OF MEMORY
013634  010105                       MOV     R1,R5                   ;SET UP EXPECTED CONTENTS
013636  120402                       CMPB    R4,R2                   ;OF SCANNER MEIORY
013640  001002                       BNE     5$
013642  052705  070000                 BIS     #70000,R5
013646  020405           5$:    CMP     R4,R5                   ;COMPARE EXPECTED AND
013650  001402                       BEQ     6$                      ;RECEIVED VALUES
013652  104002                       HLT     2                       ;SCANNER MEMORY ERROR
013654  104401                       SCOP1                           ;CHECK FOR DATA FREEZE
013656  005201           6$:    INC     R1
013660  005337  001252               DEC     TEMP3                   ;TEST NEXT SCANNER LOCATION
013664  001357                       BNE     4$
013666  005300                       DEC     R0                      ;UPDATE ADDRESS COUNT
013670  001331                       BNE     2$
013672  104400                       SCOPE                           ;CHECK FOR ITERATION, LOOP
```

```
2062                                    ;************************ TEST 30 ********************************
2063                                    ;*VERIFY THAT "CLEAR MULTIPLXER" CLEARS ALL MULTIPLEXER
2064                                    ;*FUNCTION FLIP-FLOPS
2065                                    ;::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2066
2067                                    ;  TEST 30
2068                                    ;--------------
2069  013674  012737  000030  001226  TST30:  MOV     #30,TSTNO
2070  013702  012737  014052  001216           MOV     #TST31,NEXT
2071  013710  012737  013762  001220           MOV     #3$,LOCK
2072  013716  013703  007300                    MOV     MC.CSR,R3               ;SET POINTER
2073  013722  005013                  1$:      CLR     (R3)                    ;CLEAR CONTROL REGISTER
2074  013724  005037  177776                    CLR     PS                      ;ENABLE INTERRUPTS
2075  013730  012700  000020                    MOV     #16.,R0                 ;SET UP TO TEST 16 LINES
2076  013734  012777  000017  173340  2$:      MOV     #17,@MC.LSR             ;WRITE 1S INTO  ALL MULTIPLEXER
2077  013742  052713  000400                    BIS     #STEP,(R3)              ;FUNCTION FLIPFLOPS
2078  013746  005300                           DEC     R0
2079  013750  001371                           BNE     2$
2080  013752  005037  001252                    CLR     TEMP3                   ;SET UP FOR 16 LINES
2081  013756  012700  000020                    MOV     #16.,R0
2082  013762  012713  002000          3$:      MOV     #CLRMUX,(R3)            ;CLEAR MULTIPLEXER
2083  013766  013713  001252          4$:      MOV     TEMP3,(R3)                      ;SELECT LINE
2084  013772  017704  173304                    MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
2085  013776  005005                           CLR     R5                      ;EXPECT 0S
2086  014000  005704                           TST     R4                      ;WAS LINE STATUS REGISTER CLEARED
2087  014002  001402                           BEQ     5$
2088  014004  104002                           HLT     2                       ;LINE STATUS ERROR
2089  014006  104401                           SCOP1                           ;CHECK FOR LOOP ON SAME DATA
2090  014010  005205                  5$:      INC     R5                      ;EXPECT LINE ENABLE
2091  014012  052777  000001  173262           BIS     #LINENA,@MC.LSR         ;SET LINE ENABLE ON SELECTED LINE
2092  014020  017704  173256                    MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
2093  014024  020504                           CMP     R5,R4                   ;IS ANYTHING BUT LINE ENABLE SET
2094  014026  001402                           BEQ     6$
2095  014030  104002                           HLT     2                       ;LINE STATUS ERROR
2096  014032  104401                           SCOP1                           ;CHECK FOR LOOP ON SAME DATA
2097  014034  005237  001252          6$:      INC     TEMP3                   ;UPDATE LINE NUMBER
2098  014040  005077  173236                    CLR     @MC.LSR                 ;CLEAR CURRENT LINE
2099  014044  005300                           DEC     R0                      ;CONTINUE IF ALL LINES NOT
2100  014046  001347                           BNE     4$                      ;TESTED
2101  014050  104400                           SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                                        ;************************ TEST 31 ******************************
                                        ;*WRITE 1'S INTO ALL SCANNER MEMORY LOCATIONS
                                        ;*SET 'LINE ENABLE FOR ALL LINES
                                        ;*VERIFY THAT AN INTERRUPT OCCURS FOR EACH LINE
                                        ;:**************************************************************

                                ;  TEST 31
                                ;--------------
014052  012737  000031  001226  TST31:  MOV     #31,TSTNO
014060  012737  014304  001216          MOV     #TST32,NEXT
014066  012737  014100  001220          MOV     #1$,LOCK
014074  013703  007300                  MOV     MC.CSR,R3               ;SET POINTER
014100  012713  002000          1$:     MOV     #CLRMUX,(R3)            ;CLEAR ALL MULTIPLEXER FLIPFLOPS
014104  005013                          CLR     (R3)                    ;CLEAR CONTROL REGISTER
014106  005037  177776                  CLR     PS                      ;ENABLE INTERRUPTS
014112  012700  000020                  MOV     #16.,R0                 ;SET UP TO WRITE 1'S INTO
014116  012713  001017                  MOV     #MAINT+17,(R3)          ;ALL SCANNER MEMORY LOCATION
014122  052713  000400          2$:     BIS     #STEP,(R3)              ;WRITE A LOCATION
014126  012777  000001  173146          MOV     #LINENA,@MC.LSR         ;LET 'LINE ENABLE''
014134  005300                          DEC     R0
014136  001371                          BNE     2$
014140  012705  070340                  MOV     #70340,R5               ;EXPECT 'DONE''+''SCNENA''+''COF''+''CSF''+'SECRXF''
014144  012777  014254  173132          MOV     #4$,@MC.VEC             ;SET UP LOCAL INTERRUPT SERVICE
014152  013777  177776  173126          MOV     PS,@MC.LVL              ;SERVICE AT LEVEL 7
014160  012700  000020                  MOV     #16.,R0
014164  012713  000117                  MOV     #INTENA+17,(R3)         ;SET INTERRUPT ENABLE
014170  012737  000340  177776  3$:     MOV     #340,PS                 ;LOCK OUT INTERRUPTS
014176  052713  000040                  BIS     #SCNENA,(R3)            ;START SCANNER
014202  005037  177776                  CLR     PS                      ;ENABLE INTERRUPTS
014206  005037  001270                  CLR     SAVR4
014212  105713                          TSTB    (R3)                    ;WAIT FOR DONE
014214  100410                          BMI     .+22
014216  104414                          DELAY
014220  000240                          NOP
014222  000240                          NOP
014224  062737  000001  001270          ADD     #1,SAVR4
014232  001367                          BNE     .-20
014234  104006                          HLT     6
014236  012737  000340  177776          MOV     #340,PS                 ;INTERRUPT DID NOT OCCUR
014244  011304                          MOV     (R3),R4                 ;ERROR
014246  104004                          HLT     4                       ;CONTROL STATUS ERROR
014250  104401                          SCOP1                           ;CHECK FOR LOOP ON SAME DATA
014252  000406                          BR      5$
014254  022626                  4$:     POP2SP                          ;INTERRUPT OCCURED, REPOSITION STACK
014256  011304                          MOV     (R3),R4                 ;READ CONTROL STATUS
014260  020504                          CMP     R5,R4                   ;ARE EXPECTED AND RECEIVED
014262  001402                          BEQ     5$                      ;REGISTERS THE SAME
014264  104002                          HLT     2                       ;NO, LINE STATUS ERROR
014266  104401                          SCOP1                           ;CHECK FOR LOOP WITH CURRENT DATA
014270  042713  000240          5$:     BIC     #SCNENA+DONE,(R3)       ;CLEAR SCAN ENABLE AND DONE
014274  005205                          INC     R5                      ;UPDATE EXPECTED RESULT
014276  005300                          DEC     R0                      ;CONTINUE IF NOT DONE
014300  001333                          BNE     3$
014302  104400                          SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                                        ;************************ TEST 32 *******************************
                                        ;*WRITE 1'S INTO ALL MULTIPLEXER FUNCTION FLIP-FLOPS
                                        ;*CLEAR SCANNER MEMORY
                                        ;*VERIFY THAT AN INTERRUPT OCCURS FOR EACH LINE
                                        ;*THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.
                                        ;:***************************************************************

                                ;  TEST 32
                                ;-------------
014304  012737  000032  001226  TST32:  MOV     #32,TSTNO
014312  012737  014650  001216          MOV     #TST33,NEXT
014320  012737  014436  001220          MOV     #1$,LOCK
014326  005000                          CLR     R0
014330  005737  001422                  TST     L00.03
014334  100402                          BMI     68$
014336  062700  000004                  ADD     #4,R0
014342  005737  001424          68$:    TST     L04.07
014346  100402                          BMI     69$
014350  062700  000004                  ADD     #4,R0
014354  005737  001426          69$:    TST     L08.11
014360  100402                          BMI     70$
014362  062700  000004                  ADD     #4,R0
014366  005737  001430          70$:    TST     L12.15
014372  100402                          BMI     71$
014374  062700  000004                  ADD     #4,R0
014400  005700                  71$:    TST     R0
014402  001001                          BNE     .+4
014404  000000                          HALT                    ;TEST CAN NOT RUN WITH NO LINE CARDS.!
014406  010037  007276                  MOV     R0,TOTAL
014412  005737  007260                  TST     TURFLG
014416  001405                          BEQ     65$
014420  013737  001216  001214          MOV     NEXT,RETURN
014426  000177  164562                  JMP     @RETURN
014432  013703  007300          65$:    MOV     MC.CSR,R3       ;SET POINTER
014436  012700  000020          1$:     MOV     #16.,R0         ;WRITE 1S INTO ALL
014442  012713  002000                  MOV     #CLRMUX,(R3)    ;CLEAR MULTIPLEXER
014446  005013                          CLR     (R3)            ;MULTIPLEXER FUNCTION
014450  005037  177776                  CLR     PS              ;ENABLE TELETYPE INTERRUPTS
014454  012777  000017  172620  2$:     MOV     #17,@MC.LSR     ;FLIPFLOPS
014462  052713  000400                  BIS     #STEP,(R3)
014466  005300                          DEC     R0
014470  001371                          BNE     2$
014472  012713  004000                  MOV     #CLRSCN,(R3)    ;CLEAR SCANNER MEMORY
014476  032713  000020                  BIT     #BUSY,(R3)      ;WAIT FOR CLEAR CYCLE TO COMPLETE
014502  001375                          BNE     .-4
014504  013700  007276                  MOV     TOTAL,R0
014510  012705  170340                  MOV     #170340,R5      ;FIRST EXPECTED RESULT
014514  012777  014620  172562          MOV     #4$,@MC.VEC     ;SET UP LOCAL INTERRUPT RETURN
014522  013777  177776  172556          MOV     PS,@MC.LVL
014530  012713  000117                  MOV     #INTENA+17,(R3) ;SET INTERRUPT ENABLE
014534  012737  000340  177776  3$:     MOV     #340,PS         ;LOCK OUT INTERRUPTS
014542  052713  000040                  BIS     #SCNENA,(R3)    ;START SCANNER
014546  005037  177776                  CLR     PS              ;ENABLE INTERRUPTS
014552  005037  001270                  CLR     SAVR4
014556  105713                          TSTB    (R3)            ;WAIT FOR DONE
014560  100410                          BMI     .+22
```

```
2213  014562  104414                        DELAY
2214  014564  000240                        NOP
2215  014566  000240                        NOP
2216  014570  062737  000001  001270        ADD     #1,SAVR4
2217  014576  001367                        BNE     .-20
2218  014600  104006                        HLT     6
2219  014602  012737  000340  177776        MOV     #340,PS                 ;LOCK OUT INTERRUPTS
2220  014610  011304                        MOV     (R3),R4                 ;READ CONTROL STATUS
2221  014612  104004                        HLT     4                       ;INTERRUPT DID NOT OCCUR
2222  014614  104401                        SCOP1                           ;CHECK FOR LOOP ON CURRENT DATA
2223  014616  000406                        BR      5$                      ;CONTINUE
2224  014620  022626                4$:     POP2SP                          ;INTERRUPT OCCURED, RESTORE STACK
2225  014622  011304                        MOV     (R3),R4                 ;READ CONTROL STATUS REGISTER
2226  014624  020504                        CMP     R5,R4                   ;COMPARE TO EXPECTED RESULT
2227  014626  001402                        BEQ     5$
2228  014630  104002                        HLT     2                       ;CONTROL STATUS ERROR
2229  014632  104401                        SCOP1                           ;CHECK FOR LOOP ON CURRENT DATA
2230  014634  042713  000240        5$:     BIC     #SCNENA+DONE,(R3)       ;CLEAR SCAN ENABLE AND DONE
2231  014640  005205                        INC     R5                      ;UPDATE EXPECTED RESULT
2232  014642  005300                        DEC     R0                      ;CONTINUE IF ALL
2233  014644  001333                        BNE     3$                      ;LINES NOT TESTED
2234  014646  104400                        SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
2235                                    ;************************ TEST 33 *******************************
2236                                    ;*VERIFY THAT LINE ENABLE FUNCTION FLIP-FLOP CAN
2237                                    ;*BE SET AND CLEARED FOR SELECTED LINE
2238                                    ;*THIS TEST IS DONE IF THE H325 TURN AROUND IS USED
2239                                    ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
2240                                    ;:*************************************************************************
2241
2242                                    ;  TEST 33
2243                                    ;--------------
2244  014650  012737  000033  001226  TST33:  MOV     #33,TSTNO
2245  014656  012737  015050  001216          MOV     #TST34,NEXT
2246  014664  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
2247  014670  001005                          BNE     1$                      ;BR IF H325
2248  014672  013737  001216  001214          MOV     NEXT,RETURN
2249  014700  000177  164310                  JMP     @RETURN
2250  014704  005077  172370          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL STATUS REGISTER
2251  014710  005037  177776                  CLR     PS                      ;ZERO PSW.
2252  014714  013701  007262                  MOV     LINE,R1                 ;SET LINE IMAGE
2253  014720  012777  002000  172352  2$:     MOV     #CLRMUX,@MC.CSR         ;CLEAR MUX
2254  014726  012702  000020                  MOV     #16.,R2                 ;SET FOR 16 LINES
2255  014732  010177  172342                  MOV     R1,@MC.CSR              ;SELECT LINE TO BE TESTED
2256  014736  012777  000001  172336          MOV     #LINENA,@MC.LSR         ;SET LINE ENABLE FUNCTION FLIP-FLOP
2257  014744  005077  172330                  CLR     @MC.CSR                 ;ZERO CSR
2258  014750  005005                  3$:     CLR     R5                      ;SET EXPECTED
2259  014752  017704  172324                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
2260  014756  117703  172316                  MOVB    @MC.CSR,R3              ;READ CONTROL STATUS REGISTER
2261  014762  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
2262  014766  020103                          CMP     R1,R3                   ;IF LINE NUMBER=SELECTED LINE NUMBER,
2263  014770  001002                          BNE     4$                      ;EXCEPT LINE ENABLE FUNCTION FLIP FLOP
2264  014772  012705  000001                  MOV     #LINENA,R5              ;SET "GOOD"
2265                                                                          ;TO BE SET
2266  014776  020504                  4$:     CMP     R5,R4                   ;COMPARE EXPECTED AND RECEIVED
2267  015000  001401                          BEQ     5$                      ;RESULTS
2268  015002  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
2269  015004  052777  000400  172266  5$:     BIS     #STEP,@MC.CSR           ;EXAMINE NEXT LINE
2270  015012  005302                          DEC     R2                      ;ALL LINES DONE?
2271  015014  001355                          BNE     3$                      ;BR IF NO
2272  015016  005005                          CLR     R5                      ;CLEAR "GOOD"
2273  015020  010177  172254          6$:     MOV     R1,@MC.CSR              ;LOAD LINE
2274  015024  010103                          MOV     R1,R3                   ;SET LINE COUNTER TO SELECTED LINE
2275  015026  005077  172250                  CLR     @MC.LSR                 ;CLEAR LINE ENABLE FLIP FLOP
2276  015032  104414                          DELAY                           ;DELAY FOR CABLE
2277  015034  017704  172242                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
2278  015040  005704                          TST     R4                      ;WAS LINE ENABLE FUNCTION FLIP FLOP
2279  015042  001401                          BEQ     .+4                     ;CLEARED
2280  015044  104001                          HLT     1                       ;R5-EXPECTED R4=FOUND
2281  015046  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                                        ;************************ TEST 34 *******************************
                                        ;*VERIFY THAT TERMINAL READY FUNCTION FLIP-FLOP CAN
                                        ;*BE SET AND CLEARED FOR SELECTED LINE
                                        ;*THIS TEST IS DONE IF THE H325 TURN AROUND IS USED
                                        ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
                                        ;:***************************************************************************

                                        ;  TEST 34
                                        ;--------------
015050  012737  000034  001226  TST34:  MOV     #34,TSTNO
015056  012737  015250  001216          MOV     #TST35,NEXT
015064  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
015070  001005                          BNE     1$                      ;BR IF H325
015072  013737  001216  001214          MOV     NEXT,RETURN
015100  000177  164110                  JMP     @RETURN
015104  005077  172170          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL STATUS REGISTER
015110  005037  177776                  CLR     PS                      ;ZERO PSW.
015114  013701  007262                  MOV     LINE,R1                 ;SET LINE IMAGE
015120  012777  002000  172152  2$:     MOV     #CLRMUX,@MC.CSR         ;CLEAR MUX
015126  012702  000020                  MOV     #16.,R2                 ;SET FOR 16 LINES
015132  010177  172142                  MOV     R1,@MC.CSR              ;SELECT LINE TO BE TESTED
015136  012777  000002  172136          MOV     #TRMRDY,@MC.LSR         ;SET TERMINAL READY FUNCTION FLIP-FLOP
015144  005077  172130                  CLR     @MC.CSR                 ;ZERO CSR
015150  005005                  3$:     CLR     R5                      ;SET EXPECTED
015152  017704  172124                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
015156  117703  172116                  MOVB    @MC.CSR,R3              ;READ CONTROL STATUS REGISTER
015162  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
015166  020103                          CMP     R1,R3                   ;IF LINE NUMBER=SELECTED LINE NUMBER,
015170  001002                          BNE     4$                      ;EXCEPT TERMINAL READY FUNCTION FLIP FLOP
015172  012705  000002                  MOV     #TRMRDY,R5              ;SET ''GOOD''
                                                                        ;TO BE SET
015176  020504                  4$:     CMP     R5,R4                   ;COMPARE EXPECTED AND RECEIVED
015200  001401                          BEQ     5$                      ;RESULTS
015202  104001                          HLT     1                       ;R5=EXPECTED R4-FOUND
015204  052777  000400  172066  5$:     BIS     #STEP,@MC.CSR           ;EXAMINE NEXT LINE
015212  005302                          DEC     R2                      ;ALL LINES DONE?
015214  001355                          BNE     3$                      ;BR IF NO
015216  005005                          CLR     R5                      ;CLEAR ''GOOD''
015220  010177  172054          6$:     MOV     R1,@MC.CSR              ;LOAD LINE
015224  010103                          MOV     R1,R3                   ;SET LINE COUNTER TO SELECTED LINE
015226  005077  172050                  CLR     @MC.LSR                 ;CLEAR TERMINAL READY FLIP FLOP
015232  104414                          DELAY                           ;DELAY FOR CABLE
015234  017704  172042                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
015240  005704                          TST     R4                      ;WAS TERMINAL READY FUNCTION FLIP FLOP
015242  001401                          BEQ     .+4                     ;CLEARED
015244  104001                          HLT     1                       ;R5=EXPECTFD R4 FOUND
015246  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
2329                                    ;************************ TEST 35 *******************************
2330                                    ;*VERIFY THAT REQUEST TO SEND FUNCTION FLIP-FLOP CAN
2331                                    ;*BE SET AND CLEARED FOR SELECTED LINE
2332                                    ;*THIS TEST IS DONE IF THE H325 TURN AROUND IS USED
2333                                    ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
2334                                    ;:***************************************************************************
2335
2336                                    ;  TEST 35
2337                                    ;--------------
2338  015250  012737  000035  001226  TST35:  MOV     #35,TSTNO
2339  015256  012737  015450  001216          MOV     #TST36,NEXT
2340  015264  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
2341  015270  001005                          BNE     1$                      ;BR IF H325
2342  015272  013737  001216  001214          MOV     NEXT,RETURN
2343  015300  000177  163710                  JMP     @RETURN
2344  015304  005077  171770          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL STATUS REGISTER
2345  015310  005037  177776                  CLR     PS                      ;ZERO PSW.
2346  015314  013701  007262                  MOV     LINE,R1                 ;SET LINE IMAGE
2347  015320  012777  002000  171752  2$:     MOV     #CLRMUX,@MC.CSR         ;CLEAR MUX
2348  015326  012702  000020                  MOV     #16.,R2                 ;SET FOR 16 LINES
2349  015332  010177  171742                  MOV     R1,@MC.CSR              ;SELECT LINE TO BE TESTED
2350  015336  012777  000004  171736          MOV     #RS,@MC.LSR             ;SET REQUEST TO SEND FUNCTION FLIP-FLOP
2351  015344  005077  171730                  CLR     @MC.CSR                 ;ZERO CSR
2352  015350  005005                  3$:     CLR     R5                      ;SET EXPECTED
2353  015352  017704  171724                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
2354  015356  117703  171716                  MOVB    @MC.CSR,R3              ;READ CONTROL STATUS REGISTER
2355  015362  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
2356  015366  020103                          CMP     R1,R3                   ;IF LINE NUMBER=SELECTED LINE NUMBER,
2357  015370  001002                          BNE     4$                      ;EXCEPT REQUEST TO SEND FUNCTION FLIP FLOP
2358  015372  012705  000004                  MOV     #RS,R5          ;SET ''GOOD''
2359                                                                          ;TO BE SET
2360  015376  020504                  4$:     CMP     R5,R4                   ;COMPARE EXPECTED AND RECEIVED
2361  015400  001401                          BEQ     5$                      ;RESULTS
2362  015402  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
2363  015404  052777  000400  171666  5$:     BIS     #STEP,@MC.CSR           ;EXAMINE NEXT LINE
2364  015412  005302                          DEC     R2                      ;ALL LINES DONE?
2365  015414  001355                          BNE     3$                      ;BR IF NO
2366  015416  005005                          CLR     R5                      ;CLEAR ''GOOD''
2367  015420  010177  171654          6$:     MOV     R1,@MC.CSR              ;LOAD LINE
2368  015424  010103                          MOV     R1,R3                   ;SET LINE COUNTER TO SELECTED LINE
2369  015426  005077  171650                  CLR     @MC.LSR                 ;CLEAR REQUEST TO SEND FLIP FLOP
2370  015432  104414                          DELAY                           ;DELAY FOR CABLE
2371  015434  017704  171642                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
2372  015440  005704                          TST     R4                      ;WAS REQUEST TO SEND FUNCTION FLIP FLOP
2373  015442  001401                          BEQ     .+4                     ;CLEARED
2374  015444  104001                          HLT     1                       ;R5=EXPECTED R4 FOUND
2375  015446  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                                        ;************************ TEST 36 *******************************
                                        ;*VERIFY THAT NEW SYNC (SECTX IF ASYNC LC) FUNCTION FLIP-FLOP CAN
                                        ;*BE SET AND CLEARED FOR SELECTED LINE
                                        ;*THIS TEST IS DONE IF THE H325 TURN AROUND IS USED
                                        ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
                                        ;*************************************************************

                                        ;  TEST 36
                                        ;--------------
015450  012737  000036  001226  TST36:  MOV     #36,TSTNO
015456  012737  015650  001216          MOV     #TST37,NEXT
015464  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
015470  001005                          BNE     1$                      ;BR IF H325
015472  013737  001216  001214          MOV     NEXT,RETURN
015500  000177  163510                  JMP     @RETURN
015504  005077  171570          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL STATUS REGISTER
015510  005037  177776                  CLR     PS                      ;ZERO PSW.
015514  013701  007262                  MOV     LINE,R1                 ;SET LINE IMAGE
015520  012777  002000  171552  2$:     MOV     #CLRMUX,@MC.CSR         ;CLEAR MUX
015526  012702  000020                  MOV     #16.,R2                 ;SET FOR 16 LINES
015532  010177  171542                  MOV     R1,@MC.CSR              ;SELECT LINE TO BE TESTED
015536  012777  000010  171536          MOV     #NS,@MC.LSR             ;SET NEW SYNC (SECTX IF ASYNC LC) FUNCTION FLIP-
015544  005077  171530                  CLR     @MC.CSR                 ;ZERO CSR
015550  005005                  3$:     CLR     R5                      ;SET EXPECTED
015552  017704  171524                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
015556  117703  171516                  MOVB    @MC.CSR,R3              ;READ CONTROL STATUS REGISTER
015562  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
015566  020103                          CMP     R1,R3                   ;IF LINE NUMBER-SELECTED LINE NUMBER,
015570  001002                          BNE     4$                      ;EXCEPT NEW SYNC (SECTX IF ASYNC LC) FUNCTION FL
015572  012705  000010                  MOV     #NS,R5          ;SET ''GOOD''
                                                                        ;TO BE SET
015576  020504                  4$:     CMP     R5,R4                   ;COMPARE EXPECTED AND RECEIVED
015600  001401                          BEQ     5$                      ;RESULTS
015602  104001                          HLT     1                       ;R5=EXPECTED R4-FOUND
015604  052777  000400  171466  5$:     BIS     #STEP,@MC.CSR           ;EXAMINE NEXT LINE
015612  005302                          DEC     R2                      ;ALL LINES DONE?
015614  001355                          BNE     3$                      ;BR IF NO
015616  005005                          CLR     R5                      ;CLEAR ''GOOD''
015620  010177  171454          6$:     MOV     R1,@MC.CSR              ;LOAD LINE
015624  010103                          MOV     R1,R3                   ;SET LINE COUNTER TO SELECTED LINE
015626  005077  171450                  CLR     @MC.LSR                 ;CLEAR NEW SYNC (SECTX IF ASYNC LC) FLIP FLOP
015632  104414                          DELAY                           ;DELAY FOR CABLE
015634  017704  171442                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
015640  005704                          TST     R4                      ;WAS NEW SYNC (SECTX IF ASYNC LC) FUNCTION FLIP
015642  001401                          BEQ     .+4                     ;CLEARED
015644  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
015646  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                                        ;************************ TEST 37 *******************************
                                        ;*VERIFY THAT RING IS SET IF 'LINE ENABLE''
                                        ;*AND TERMINAL ARE SET FOR SELECTED LINE.
                                        ;*THIS TEST IS DONE IF THE H325 TURN AROUND IS USED
                                        ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
                                        ;;*************************************************************

                                        ;  TEST 37
                                        ;--------------
015650  012737  000037  001226  TST37:  MOV     #37,TSTNO
015656  012737  016046  001216          MOV     #TST40,NEXT
015664  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
015670  001005                          BNE     1$                      ;BR IF H325
015672  013737  001216  001214          MOV     NEXT,RETURN
015700  000177  163310                  JMP     @RETURN
015704  005077  171370          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
015710  005037  177776                  CLR     PS                      ;ZERO PSW
015714  013701  007262                  MOV     LINE,R1                 ;LINE NUMBER
015720  012702  000020          2$:     MOV     #16.,R2                 ;16 LINES
015724  010177  171350                  MOV     R1,@MC.CSR              ;SELECT A LINE
015730  012777  000003  171344          MOV     #LINENA+TRMRDY,@MC.LSR  ;SET LINE ENABLE +TRMRDY
015736  005077  171336                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
015742  005005                  3$:     CLR     R5                      ;CLEAR EXPECTED RESULT
015744  017704  171332                  MOV     @MC.LSR,R4              ;READ LINE STATUS
015750  117703  171324                  MOVB    @MC.CSR,R3              ;READ LINE NUMBER
015754  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
015760  020103                          CMP     R1,R3                   ;IF RECEIVED LINE=SELECTED LINE
015762  001002                          BNE     4$                      ;EXPECT LINE ENABLE AND
015764  012705  000203                  MOV     #LINENA+TRMRDY+RING,R5
                                                                        ;RING IS SET
015770  020405                  4$:     CMP     R4,R5                   ;COMPARE EXPECTED AND
015772  001401                          BEQ     5$                      ;RECEIVED RESULTS
015774  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
015776  052777  000400  171274  5$:     BIS     #STEP,@MC.CSR           ;UPDATE LINE COUNTER
016004  005302                          DEC     R2                      ;CONTINUE IF ALL CHECKS
016006  001355                          BNE     3$                      ;ARE NOT DONE FOR THIS LINE
016010  012705  000001                  MOV     #LINENA,R5              ;EXPECT LINE ENABLE
016014  010103                  6$:     MOV     R1,R3                   ;ON SELECTED LINE
016016  010177  171256                  MOV     R1,@MC.CSR              ;SELECT LINE
016022  042777  000002  171252          BIC     #TRMRDY,@MC.LSR         ;CLEAR TERMINAL
016030  104414                          DELAY                           ;DELAY FOR CABLE
016032  017704  171244                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
016036  020504                          CMP     R5,R4                   ;ONLY LINE ENABLE SHOULD BE
016040  001401                          BEQ     .+4                     ;SET ON THIS LINE
016042  104001                          HLT     1                       ;R5=EXPECTED R4 FOUND
016044  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                        ;************************ TEST 40 *********************************
                        ;*VERIFY THAT CLEAR TO SEND AND CARRIER ARE SET IF 'LINE ENABLE''
                        ;*AND REQUEST TO SEND ARE SET FOR SELECTED LINE.
                        ;*THIS TEST IS DONE IF THE H325 TURN AROUND IS USED
                        ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
                        ;:*****************************************************************

                        ;  TEST 40
                        ;--------------
016046  012737  000040  001226  TST40:  MOV     #40,TSTNO
016054  012737  016244  001216          MOV     #TST41,NEXT
016062  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
016066  001005                          BNE     1$                      ;BR IF H325
016070  013737  001216  001214          MOV     NEXT,RETURN
016076  000177  163112                  JMP     @RETURN
016102  005077  171172          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
016106  005037  177776                  CLR     PS                      ;ZERO PSW
016112  013701  007262                  MOV     LINE,R1                 ;LINE NUMBER
016116  012702  000020          2$:     MOV     #16.,R2                 ;16 LINES
016122  010177  171152                  MOV     R1,@MC.CSR              ;SELECT A LINE
016126  012777  000005  171146          MOV     #LINENA+RS,@MC.LSR      ;SET LINE ENABLE +RS
016134  005077  171140                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
016140  005005                  3$:     CLR     R5                      ;CLEAR EXPECTED RESULT
016142  017704  171134                  MOV     @MC.LSR,R4              ;READ LINE STATUS
016146  117703  171126                  MOVB    @MC.CSR,R3              ;READ LINE NUMBER
016152  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
016156  020103                          CMP     R1,R3                   ;IF RECEIVED LINE=SELECTED LINE
016160  001002                          BNE     4$                      ;EXPECT LINE ENABLE AND
016162  012705  000145                  MOV     #LINENA+RS+CO+CS,R5
                                                                        ;CLEAR TO SEND AND CARRIER ARE SET
016166  020405                  4$:     CMP     R4,R5                   ;COMPARE EXPECTED AND
016170  001401                          BEQ     5$                      ;RECEIVED RESULTS
016172  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
016174  052777  000400  171076  5$:     BIS     #STEP,@MC.CSR           ;UPDATE LINE COUNTER
016202  005302                          DEC     R2                      ;CONTINUE IF ALL CHECKS
016204  001355                          BNE     3$                      ;ARE NOT DONE FOR THIS LINE
016206  012705  000001                  MOV     #LINENA,R5              ;EXPECT LINE ENABLE
016212  010103                  6$:     MOV     R1,R3                   ;ON SELECTED LINE
016214  010177  171060                  MOV     R1,@MC.CSR              ;SELECT LINE
016220  042777  000004  171054          BIC     #RS,@MC.LSR             ;CLEAR REQUEST TO SEND
016226  104414                          DELAY                           ;DELAY FOR CABLE
016230  017704  171046                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
016234  020504                          CMP     R5,R4                   ;ONLY LINE ENABLE SHOULD BE
016236  001401                          BEQ     .+4                     ;SET ON THIS LINE
016240  104001                          HLT     1                       ;R5=EXPECTED R4 FOUND
016242  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                                        ;************************ TEST 41 *******************************
                                        ;*VERIFY THAT DATA SET READY(SECRX IF ASYNC LC) IS SET IF 'LINE ENABLE''
                                        ;*AND NEW SYNC (SECTX IF ASYNC LC) ARE SET FOR SELECTED LINE.
                                        ;*THIS TEST IS DONE IF THE H325 TURN AROUND IS USED
                                        ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
                                        ;:***************************************************************

                                        ;  TEST 41
                                        ;--------------
016244  012737  000041  001226  TST41:  MOV     #41,TSTNO
016252  012737  016442  001216          MOV     #TST42,NEXT
016260  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
016264  001005                          BNE     1$                      ;BR IF H325
016266  013737  001214  001214          MOV     NEXT,RETURN
016274  000177  1[illegible]            JMP     @RETURN
016300  005077  1[illegible]    1$:     CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
016304  005037  [illegible]76           CLR     PS                      ;ZERO PSW
016310  013701  00[illegible]262        MOV     LINE,R1                 ;LINE NUMBER
016314  012702  000020          2$:     MOV     #16.,R2                 ;16 LINES
016320  010177  170754                  MOV     R1,@MC.CSR              ;SELECT A LINE
016324  012777  000011  170750          MOV     #LINENA+NS,@MC.LSR      ;SET LINE ENABLE +NS
016332  005077  170742                  CLR     @MC.CSR                 ;CLEAR CONTROL RFGISTER
016336  005005                  3$:     CLR     R5                      ;CLEAR EXPECTED RESULT
016340  017704  170736                  MOV     @MC.LSR,R4              ;READ LINE STATUS
016344  117703  170730                  MOVB    @MC.CSR,R3              ;READ LINE NUMBER
016350  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
016354  020103                          CMP     R1,R3                   ;IF RECEIVED LINE=SELECTED LINE
016356  001002                          BNE     4$                      ;EXPECT LINE ENABLE AND
016360  012705  000031                  MOV     #LINENA+NS+DSR,R5
                                                                        ;DATA SET READY(SECRX IF ASYNC LC) IS SET
016364  020405          4$:     CMP     R4,R5                           ;COMPARE EXPECTED AND
016366  001401                  BEQ     5$                              ;RECEIVED RESULTS
016370  104001                  HLT     1                               ;R5=EXPECTED R4=FOUND
016372  052777  000400  170700  5$:     BIS     #STEP,@MC.CSR           ;UPDATE LINE COUNTER
016400  005302                  DEC     R2                              ;CONTINUE IF ALL CHECKS
016402  001355                  BNE     3$                              ;ARE NOT DONE FOR THIS LINE
016404  012705  000001                  MOV     #LINENA,R5              ;EXPECT LINE ENABLE
016410  010103          6$:     MOV     R1,R3                           ;ON SELECTED LINE
016412  010177  170662                  MOV     R1,@MC.CSR              ;SELECT LINE
016416  042777  000010  170656          BIC     #NS,@MC.LSR             ;CLEAR NEW SYNC (SECTX IF ASYNC LC)
016424  104414                          DELAY                           ;DELAY FOR CABLE
016426  017704  170650                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
016432  020504                          CMP     R5,R4                   ;ONLY LINE ENABLE SHOULD BE
016434  001401                          BEQ     .+4                     ;SET ON THIS LINE
016436  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
016440  104400          7$:     SCOPE                                   ;CHECK FOR ITERATIONS, LOOP
```

```
                                        ;************************ TEST 42 *******************************
                                        ;*VERIFY THAT LINE ENABLE FUNCTION FLIP-FLOP CAN
                                        ;*BE SET AND CLEARED FOR SELECTED LINE
                                        ;*THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.
                                        ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
                                        ;:***************************************************************************

                                ;  TEST 42
                                ;--------------
016442  012737  000042  001226  TST42:  MOV     #42,TSTNO
016450  012737  016666  001216          MOV     #TST43,NEXT
016456  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
016462  001405                          BEQ     1$                      ;BR IF H861
016464  013737  001216  001214          MOV     NEXT,RETURN
016472  000177  162516                  JMP     @RETURN
016476  005077  170576          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL STATUS REGISTER
016502  005037  177776                  CLR     PS                      ;ZERO PSW.
016506  013700  007276                  MOV     TOTAL,R0                ;SET THE TOTAL NUMBER OF LINES TO BE TESTED IN R
016512  005001                          CLR     R1
016514  012737  016522  001220          MOV     #2$,LOCK
016522  012777  002000  170550  2$:     MOV     #CLRMUX,@MC.CSR         ;CLEAR MUX
016530  012702  000020                  MOV     #16.,R2                 ;SET FOR 16 LINES
016534  010177  170540                  MOV     R1,@MC.CSR              ;SELECT LINE TO BE TESTED
016540  010137  007262                  MOV     R1,LINE                 ;SET IMAGE
016544  012777  000001  170530          MOV     #LINENA,@MC.LSR         ;SET LINE ENABLE FUNCTION FLIP-FLOP
016552  005077  170522                  CLR     @MC.CSR                 ;ZERO CSR
016556  005005                  3$:     CLR     R5                      ;SET EXPECTED
016560  017704  170516                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
016564  117703  170510                  MOVB    @MC.CSR,R3              ;READ CONTROL STATUS REGISTER
016570  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
016574  020103                          CMP     R1,R3                   ;IF LINE NUMBER=SELECTED LINE NUMBER,
016576  001002                          BNE     4$                      ;EXCEPT LINE ENABLE FUNCTION FLIP FLOP
016600  012705  000001                  MOV     #LINENA,R5              ;SET ''GOOD''
                                                                        ;TO BE SET
016604  020504                  4$:     CMP     R5,R4                   ;COMPARE EXPECTED AND RECEIVED
016606  001401                          BEQ     5$                      ;RESULTS
016610  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
016612  052777  000400  170460  5$:     BIS     #STEP,@MC.CSR           ;EXAMINE NEXT LINE
016620  005302                          DEC     R2                      ;ALL LINES DONE?
016622  001355                          BNE     3$                      ;BR IF NO
016624  005005                          CLR     R5                      ;CLEAR ''GOOD''
016626  010177  170446          6$:     MOV     R1,@MC.CSR              ;LOAD LINE
016632  010103                          MOV     R1,R3                   ;SET LINE COUNTER TO SELECTED LINE
016634  005077  170442                  CLR     @MC.LSR                 ;CLEAR LINE ENABLE FLIP FLOP
016640  104414                          DELAY                           ;DELAY FOR CABLE
016642  017704  170434                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
016646  005704                          TST     R4                      ;WAS LINE ENABLE FUNCTION FLIP FLOP
016650  001401                          BEQ     .+4                     ;CLEARED
016652  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
016654  104401                          SCOP1
016656  005201                          INC     R1
016660  005300                          DEC     R0
016662  001317                          BNE     2$
016664  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                        ;************************** TEST 43 ********************************
                        ;*VERIFY THAT TERMINAL READY FUNCTION FLIP-FLOP CAN
                        ;*BE SET AND CLEARED FOR SELECTED LINE
                        ;*THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.
                        ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
                        ;:*****************************************************************

                        ;  TEST 43
                        ;--------------
016666  012737  000043  001226  TST43:  MOV     #43,TSTNO
016674  012737  017112  001216          MOV     #TST44,NEXT
016702  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
016706  001405                          BEQ     1$                      ;BR IF H861
016710  013737  001216  001214          MOV     NEXT,RETURN
016716  000177  162272                  JMP     @RETURN
016722  005077  170352          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL STATUS REGISTER
016726  005037  177776                  CLR     PS                      ;ZERO PSW.
016732  013700  007276                  MOV     TOTAL,R0                ;SET THE TOTAL NUMBER OF LINES TO BE TESTED IN R
016736  005001                          CLR     R1
016740  012737  016746  001220          MOV     #2$,LOCK
016746  012777  002000  170324  2$:     MOV     #CLRMUX,@MC.CSR         ;CLEAR MUX
016754  012702  000020                  MOV     #16.,R2                 ;SET FOR 16 LINES
016760  010177  170314                  MOV     R1,@MC.CSR              ;SELECT LINE TO BE TESTED
016764  010137  007262                  MOV     R1,LINE                 ;SET IMAGE
016770  012777  000002  170304          MOV     #TRMRDY,@MC.LSR         ;SET TERMINAL READY FUNCTION FLIP-FLOP
016776  005077  170276                  CLR     @MC.CSR                 ;ZERO CSR
017002  005005                  3$:     CLR     R5                      ;SET EXPECTED
017004  017704  170272                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
017010  117703  170264                  MOVB    @MC.CSR,R3              ;READ CONTROL STATUS REGISTER
017014  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
017020  020103                          CMP     R1,R3                   ;IF LINE NUMBER=SELECTED LINE NUMBER,
017022  001002                          BNE     4$                      ;EXCEPT TERMINAL READY FUNCTION FLIP FLOP
017024  012705  000002                  MOV     #TRMRDY,R5              ;SET 'GOOD'
                                                                        ;TO BE SET
017030  020504                  4$:     CMP     R5,R4                   ;COMPARE EXPECTED AND RECEIVED
017032  001401                          BEQ     5$                      ;RESULTS
017034  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
017036  052777  000400  170234  5$:     BIS     #STEP,@MC.CSR           ;EXAMINE NEXT LINE
017044  005302                          DEC     R2                      ;ALL LINES DONE?
017046  001355                          BNE     3$                      ;BR IF NO
017050  005005                          CLR     R5                      ;CLEAR 'GOOD'
017052  010177  170222          6$:     MOV     R1,@MC.CSR              ;LOAD LINE
017056  010103                          MOV     R1,R3                   ;SET LINE COUNTER TO SELECTED LINE
017060  005077  170216                  CLR     @MC.LSR                 ;CLEAR TERMINAL READY FLIP FLOP
017064  104414                          DELAY                           ;DELAY FOR CABLE
017066  017704  170210                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
017072  005704                          TST     R4                      ;WAS TERMINAL READY FUNCTION FLIP FLOP
017074  001401                          BEQ     .+4                     ;CLEARED
017076  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
017100  104401                          SCOP1
017102  005201                          INC     R1
017104  005300                          DEC     R0
017106  001317                          BNE     2$
017110  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                        ;************************ TEST 44 *******************************
                        ;*VERIFY THAT REQUEST TO SEND FUNCTION FLIP-FLOP CAN
                        ;*BE SET AND CLEARED FOR SELECTED LINE
                        ;*THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.
                        ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
                        ;:*************************************************************

                        ;  TEST 44
                        ;--------------
017112 012737 000044 001226 TST44:  MOV   #44,TSTNO
017120 012737 017336 001216         MOV   #TST45,NEXT
017126 005737 007260                TST   TURFLG              ;TURN AROUND H861 OR H325?
017132 001405                       BEQ   1$                  ;BR IF H861
017134 013737 001216 001214         MOV   NEXT,RETURN
017142 000177 162046                JMP   @RETURN
017146 005077 170126          1$:   CLR   @MC.CSR             ;CLEAR CONTROL STATUS REGISTER
017152 005037 177776                CLR   PS                  ;ZERO PSW.
017156 013700 007276                MOV   TOTAL,R0            ;SET THE TOTAL NUMBER OF LINES TO BE TESTED IN R
017162 005001                       CLR   R1
017164 012737 017172 001220         MOV   #2$,LOCK
017172 012777 002000 170100   2$:   MOV   #CLRMUX,@MC.CSR      ;CLEAR MUX
017200 012702 000020                MOV   #16.,R2             ;SET FOR 16 LINES
017204 010177 170070                MOV   R1,@MC.CSR          ;SELECT LINE TO BE TESTED
017210 010137 007262                MOV   R1,LINE             ;SET IMAGE
017214 012777 000004 170060         MOV   #RS,@MC.LSR         ;SET REQUEST TO SEND FUNCTION FLIP-FLOP
017222 005077 170052                CLR   @MC.CSR             ;ZERO CSR
017226 005005                 3$:   CLR   R5                  ;SET EXPECTED
017230 017704 170046                MOV   @MC.LSR,R4          ;READ LINE STATUS REGISTER
017234 117703 170040                MOVB  @MC.CSR,R3          ;READ CONTROL STATUS REGISTER
017240 042703 177760                BIC   #^C<17>,R3          ;CLEAR UNWANTED BITS
017244 020103                       CMP   R1,R3               ;IF LINE NUMBER=SELECTED LINE NUMBER,
017246 001002                       BNE   4$                  ;EXCEPT REQUEST TO SEND FUNCTION FLIP FLOP
017250 012705 000004                MOV   #RS,R5        ;SET ''GOOD''
                                                              ;TO BE SET
017254 020504                 4$:   CMP   R5,R4               ;COMPARE EXPECTED AND RECEIVED
017256 001401                       BEQ   5$                  ;RESULTS
017260 104001                       HLT   1                   ;R5=EXPECTED R4=FOUND
017262 052777 000400 170010   5$:   BIS   #STEP,@MC.CSR      ;EXAMINE NEXT LINE
017270 005302                       DEC   R2                  ;ALL LINES DONE?
017272 001355                       BNE   3$                  ;BR IF NO
017274 005005                       CLR   R5                  ;CLEAR ''GOOD''
017276 010177 167776          6$:   MOV   R1,@MC.CSR          ;LOAD LINE
017302 010103                       MOV   R1,R3               ;SET LINE COUNTER TO SELECTED LINE
017304 005077 167772                CLR   @MC.LSR             ;CLEAR REQUEST TO SEND FLIP FLOP
017310 104414                       DELAY                     ;DELAY FOR CABLE
017312 017704 167764                MOV   @MC.LSR,R4          ;READ LINE STATUS REGISTER
017316 005704                       TST   R4                  ;WAS REQUEST TO SEND FUNCTION FLIP FLOP
017320 001401                       BEQ   .+4                 ;CLEARED
017322 104001                       HLT   1                   ;R5=EXPECTED R4=FOUND
017324 104401                       SCOP1
017326 005201                       INC   R1
017330 005300                       DEC   R0
017332 001317                       BNE   2$
017334 104400                 7$:   SCOPE                     ;CHECK FOR ITERATIONS, LOOP
```

```
                                        ;************************ TEST 45 ******************************
                                        ;*VERIFY THAT SECONDARY TRANSMIT FUNCTION FLIP-FLOP CAN
                                        ;*BE SET AND CLEARED FOR SELECTED LINE
                                        ;*THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.
                                        ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
                                        ;*************************************************************

                                        ;  TEST 45
                                        ;-------------
017336  012737  000045  001226  TST45:  MOV     #45,TSTNO
017344  012737  017562  001216          MOV     #TST46,NEXT
017352  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
017356  001405                          BEQ     1$                      ;BR IF H861
017360  013737  001216  001214          MOV     NEXT,RETURN
017366  000177  161622                  JMP     @RETURN
017372  005077  167702          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL STATUS REGISTER
017376  005037  177776                  CLR     PS                      ;ZERO PSW.
017402  013700  007276                  MOV     TOTAL,R0                ;SET THE TOTAL NUMBER OF LINES TO BE TESTED IN R
017406  005001                          CLR     R1
017410  012737  017416  001220          MOV     #2$,LOCK
017416  012777  002000  167654  2$:     MOV     #CLRMUX,@MC.CSR         ;CLEAR MUX
017424  012702  000020                  MOV     #16.,R2                 ;SET FOR 16 LINES
017430  010177  167644                  MOV     R1,@MC.CSR              ;SELECT LINE TO BE TESTED
017434  010137  007262                  MOV     R1,LINE                 ;SET IMAGE
017440  012777  000010  167634          MOV     #SECTX,@MC.LSR          ;SET SECONDARY TRANSMIT FUNCTION FLIP-FLOP
017446  005077  167626                  CLR     @MC.CSR                 ;ZERO CSR
017452  005005                  3$:     CLR     R5                      ;SET EXPECTED
017454  017704  167622                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
017460  117703  167614                  MOVB    @MC.CSR,R3              ;READ CONTROL STATUS REGISTER
017464  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
017470  020103                          CMP     R1,R3                   ;IF LINE NUMBER=SELECTED LINE NUMBER,
017472  001002                          BNE     4$                      ;EXCEPT SECONDARY TRANSMIT FUNCTION FLIP FLOP
017474  012705  000010                  MOV     #SECTX,R5               ;SET "GOOD"
                                                                        ;TO BE SET
017500  020504                  4$:     CMP     R5,R4                   ;COMPARE EXPECTED AND RECEIVED
017502  001401                          BEQ     5$                      ;RESULTS
017504  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
017506  052777  000400  167564  5$:     BIS     #STEP,@MC.CSR           ;EXAMINE NEXT LINE
017514  005302                          DEC     R2                      ;ALL LINES DONE?
017516  001355                          BNE     3$                      ;BR IF NO
017520  005005                          CLR     R5                      ;CLEAR "GOOD"
017522  010177  167552          6$:     MOV     R1,@MC.CSR              ;LOAD LINE
017526  010103                          MOV     R1,R3                   ;SET LINE COUNTER TO SELECTED LINE
017530  005077  167546                  CLR     @MC.LSR                 ;CLEAR SECONDARY TRANSMIT FLIP FLOP
017534  104414                          DELAY                           ;DELAY FOR CABLE
017536  017704  167540                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
017542  005704                          TST     R4                      ;WAS SECONDARY TRANSMIT FUNCTION FLIP FLOP
017544  001401                          BEQ     .+4                     ;CLEARED
017546  104001                          HLT     1                       ;R5-EXPECTED R4=FOUND
017550  104401                          SCOP1
017552  005201                          INC     R1
017554  005300                          DEC     R0
017556  001317                          BNE     2$
017560  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
2780
2781                            ;************************ TEST 46 *******************************
2782                            ;*VERIFY THAT CLEAR TO SEND AND CARRIER ARE SET IF 'LINE ENABLE''
2783                            ;*AND TERMINAL ARE SET FOR SELECTED LINE.
2784                            ;*THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.
2785                            ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
2786                            ;:***************************************************************
2787
2788                            ;  TEST 46
2789                            ;--------------
2790  017562  012737  000046  001226  TST46:  MOV     #46,TSTNO
2791  017570  012737  020004  001216          MOV     #TST47,NEXT
2792  017576  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
2793  017602  001405                          BEQ     1$                      ;BR IF H861
2794  017604  013737  001216  001214          MOV     NEXT,RETURN
2795  017612  000177  161376                  JMP     @RETURN
2796  017616  005077  167456          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
2797  017622  005037  177776                  CLR     PS                      ;ZERO PSW
2798  017626  013700  007276                  MOV     TOTAL,R0                ;SET THE TOTAL NUMBER OF LINES TO BE TESTED IN R
2799  017632  005001                          CLR     R1
2800  017634  012737  017642  001220          MOV     #2$,LOCK
2801  017642  012702  000020          2$:     MOV     #16.,R2                 ;16 LINES
2802  017646  010177  167426                  MOV     R1,@MC.CSR              ;SELECT A LINE
2803  017652  012777  000003  167422          MOV     #LINENA+TRMRDY,@MC.LSR  ;SET LINE ENABLE +TRMRDY
2804  017660  005077  167414                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
2805  017664  005005                  3$:     CLR     R5                      ;CLEAR EXPECTED RESULT
2806  017666  017704  167410                  MOV     @MC.LSR,R4              ;READ LINE STATUS
2807  017672  117703  167402                  MOVB    @MC.CSR,R3              ;READ LINE NUMBER
2808  017676  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
2809  017702  020103                          CMP     R1,R3                   ;IF RECEIVED LINE=SELECTED LINE
2810  017704  001002                          BNE     4$                      ;EXPECT LINE ENABLE AND
2811  017706  012705  000143                  MOV     #LINENA+TRMRDY+CO+CS,R5
2812                                                                          ;CLEAR TO SEND AND CARRIER ARE SET
2813  017712  020405                  4$:     CMP     R4,R5                   ;COMPARE EXPECTED AND
2814  017714  001401                          BEQ     5$                      ;RECEIVED RESULTS
2815  017716  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
2816  017720  052777  000400  167352  5$:     BIS     #STEP,@MC.CSR           ;UPDATE LINE COUNTER
2817  017726  005302                          DEC     R2                      ;CONTINUE IF ALL CHECKS
2818  017730  001355                          BNE     3$                      ;ARE NOT DONE FOR THIS LINE
2819  017732  012705  000001                  MOV     #LINENA,R5              ;EXPECT LINE ENABLE
2820  017736  010103                  6$:     MOV     R1,R3                   ;ON SELECTED LINE
2821  017740  010177  167334                  MOV     R1,@MC.CSR              ;SELECT LINE
2822  017744  042777  000002  167330          BIC     #TRMRDY,@MC.LSR         ;CLEAR TERMINAL
2823  017752  104414                          DELAY                           ;DELAY FOR CABLE
2824  017754  017704  167322                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
2825  017760  020504                          CMP     R5,R4                   ;ONLY LINE ENABLE SHOULD BE
2826  017762  001401                          BEQ     .+4                     ;SET ON THIS LINE
2827  017764  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
2828  017766  104401                          SCOP1
2829  017770  005201                          INC     R1
2830  017772  005077  167304                  CLR     @MC.LSR
2831  017776  005300                          DEC     R0
2832  020000  001320                          BNE     2$
2833  020002  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
2834
2835                                    ;************************ TEST 47 *******************************
2836                                    ;*VERIFY THAT RING IS SET IF 'LINE ENABLE'
2837                                    ;*AND REQUEST TO SEND ARE SET FOR SELECTED LINE.
2838                                    ;*THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.
2839                                    ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
2840                                    ;*************************************************************
2841
2842                                    ;  TEST 47
2843                                    ;--------------
2844  020004  012737  000047  001226  TST47:  MOV     #47,TSTNO
2845  020012  012737  020226  001216          MOV     #TST50,NEXT
2846  020020  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
2847  020024  001405                          BEQ     1$                      ;BR IF H861
2848  020026  013737  001216  001214          MOV     NEXT,RETURN
2849  020034  000177  161154                  JMP     @RETURN
2850  020040  005077  167234          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
2851  020044  005037  177776                  CLR     PS                      ;ZERO PSW
2852  020050  013700  007276                  MOV     TOTAL,R0                ;SET THE TOTAL NUMBER OF LINES TO BE TESTED IN R
2853  020054  005001                          CLR     R1
2854  020056  012737  020064  001220          MOV     #2$,LOCK
2855  020064  012702  000020          2$:     MOV     #16.,R2                 ;16 LINES
2856  020070  010177  167204                  MOV     R1,@MC.CSR              ;SELECT A LINE
2857  020074  012777  000005  167200          MOV     #LINENA+RS,@MC.LSR      ;SET LINE ENABLE +RS
2858  020102  005077  167172                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
2859  020106  005005                  3$:     CLR     R5                      ;CLEAR EXPECTED RESULT
2860  020110  017704  167166                  MOV     @MC.LSR,R4              ;READ LINE STATUS
2861  020114  117703  167160                  MOVB    @MC.CSR,R3              ;READ LINE NUMBER
2862  020120  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
2863  020124  020103                          CMP     R1,R3                   ;IF RECEIVED LINE=SELECTED LINE
2864  020126  001002                          BNE     4$                      ;EXPECT LINE ENABLE AND
2865  020130  012705  000205                  MOV     #LINENA+RS+RING,R5
2866                                                                          ;RING IS SET
2867  020134  020405                  4$:     CMP     R4,R5                   ;COMPARE EXPECTED AND
2868  020136  001401                          BEQ     5$                      ;RECEIVED RESULTS
2869  020140  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
2870  020142  052777  000400  167130  5$:     BIS     #STEP,@MC.CSR           ;UPDATE LINE COUNTER
2871  020150  005302                          DEC     R2                      ;CONTINUE IF ALL CHECKS
2872  020152  001355                          BNE                             ;ARE NOT DONE FOR THIS LINE
2873  020154  012705  000001                  MOV     #LINENA,R5              ;EXPECT LINE ENABLE
2874  020160  010103                  6$:     MOV     R1,R3                   ;ON SELECTED LINE
2875  020162  010177  167112                  MOV     R1,@MC.CSR              ;SELECT LINE
2876  020166  042777  000004  167106          BIC     #RS,@MC.LSR             ;CLEAR REQUEST TO SEND
2877  020174  104414                          DELAY                           ;DELAY FOR CABLE
2878  020176  017704  167100                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
2879  020202  020504                          CMP     R5,R4                   ;ONLY LINE ENABLE SHOULD BE
2880  020204  001401                          BEQ     .+4                     ;SET ON THIS LINE
2881  020206  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
2882  020210  104401                          SCOP1
2883  020212  005201                          INC     R1
2884  020214  005077  167062                  CLR     @MC.LSR
2885  020220  005300                          DEC     R0
2886  020222  001320                          BNE     2$
2887  020224  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                                        ;************************* TEST 50 *********************************
                                        ;*VERIFY THAT SECONDARY RECEIVE IS SET IF 'LINE ENABLE''
                                        ;*AND SECONDARY TRANSMIT ARE SET FOR SELECTED LINE.
                                        ;*THIS TEST IS DONE IF THE H861 TURN AROUND IS USED.
                                        ; MODEM CONTROL LINES *MUST* BE CONTIGUOUS FROM LINE 00.
                                        ;:*****************************************************************

                                        ;  TEST 50
                                        ;--------------
020226  012737  000050  001226  TST50:  MOV     #50,TSTNO
020234  012737  020450  001216          MOV     #TST51,NEXT
020242  005737  007260                  TST     TURFLG                  ;TURN AROUND H861 OR H325?
020246  001405                          BEQ     1$                      ;BR IF H861
020250  013737  001216  001214          MOV     NEXT,RETURN
020256  000177  160732                  JMP     @RETURN
020262  005077  167012          1$:     CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
020266  005037  177776                  CLR     PS                      ;ZERO PSW
020272  013700  007276                  MOV     TOTAL,R0                ;SET THE TOTAL NUMBER OF LINES TO BE TESTED IN R
020276  005001                          CLR     R1
020300  012737  020306  001220          MOV     #2$,LOCK
020306  012702  000020          2$:     MOV     #16.,R2                 ;16 LINES
020312  010177  166762                  MOV     R1,@MC.CSR              ;SELECT A LINE
020316  012777  000011  166756          MOV     #LINENA+SECTX,@MC.LSR   ;SET LINE ENABLE +SECTX
020324  005077  166750                  CLR     @MC.CSR                 ;CLEAR CONTROL REGISTER
020330  005005                  3$:     CLR     R5                      ;CLEAR EXPECTED RESULT
020332  017704  166744                  MOV     @MC.LSR,R4              ;READ LINE STATUS
020336  117703  166736                  MOVB    @MC.CSR,R3              ;READ LINE NUMBER
020342  042703  177760                  BIC     #^C<17>,R3              ;CLEAR UNWANTED BITS
020346  020103                          CMP     R1,R3                   ;IF RECEIVED LINE-SELECTED LINE
020350  001002                          BNE     4$                      ;EXPECT LINE ENABLE AND
020352  012705  000031                  MOV     #LINENA+SECTX+SECRX,R5
020356                                                                  ;SECONDARY RECEIVE IS SET
020356  020405                  4$:     CMP     R4,R5                   ;COMPARE EXPECTED AND
020360  001401                          BEQ     5$                      ;RECEIVED RESULTS
020362  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
020364  052777  000400  166706  5$:     BIS     #STEP,@MC.CSR           ;UPDATE LINE COUNTER
020372  005302                          DEC     R2                      ;CONTINUE IF ALL CHECKS
020374  001355                          BNE     3$                      ;ARE NOT DONE FOR THIS LINE
020376  012705  000001                  MOV     #LINENA,R5              ;EXPECT LINE ENABLE
020402  010103                  6$:     MOV     R1,R3                   ;ON SELECTED LINE
020404  010177  166670                  MOV     R1,@MC.CSR              ;SELECT LINE
020410  042777  000010  166664          BIC     #SECTX,@MC.LSR          ;CLEAR SECONDARY TRANSMIT
020416  104414                          DELAY                           ;DELAY FOR CABLE
020420  017704  166656                  MOV     @MC.LSR,R4              ;READ LINE STATUS REGISTER
020424  020504                          CMP     R5,R4                   ;ONLY LINE ENABLE SHOULD BE
020426  001401                          BEQ     .+4                     ;SET ON THIS LINE
020430  104001                          HLT     1                       ;R5=EXPECTED R4=FOUND
020432  104401                          SCOP1
020434  005201                          INC     R1
020436  005077  166640                  CLR     @MC.LSR
020442  005300                          DEC     R0
020444  001320                          BNE     2$
020446  104400                  7$:     SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                                        ;************************ TEST 51 *********************************
                                        ;*DV11 SINGLE LINE CABLE TEST.
                                        ;*TEST TO RUN A 5 BIT BLOCK (000-037)
                                        ;*OF DATA FROM THE DV11 TRANSMITTER INTO THE
                                        ;*DV11 RECEIVER THROUGH THE CABLE.
                                        ;*SETUP:
                                        ;*MODE:          EXTERNAL LOOP BACK
                                        ;*TXBA:          SYNC
                                        ;*TXWC:          -42(8)-BIT15
                                        ;*RXBA           RXBA
                                        ;*RXWC:          -40(8)-BIT15
                                        ;*LINE PROTOCOL TXDDCMP,RXDDCMP,LRC8,STRIP SYNC,IDLE MARK
                                        ;*LINE STATE    EXPECT BCC,TX GO
                                        ;*LINE PROGRESS SEND BCC
                                        ;*NOTE: FOR TEST OF ASYNC LINE CARD;
                                        ;* ''SYNC 'A''' MUST BE SET TO ALL ZEROS
                                        ;* IN SOFTWARE STATUS MAP.
                                        ;*
                                        ;:*****************************************************************

                                        ;  TEST 51
                                        ;-------------
020450  012737  000051  001226  TST51:  MOV     #51,TSTNO
020456  012737  010064  001216          MOV     #TESTER,NEXT
020464  005737  007260                  TST     TURFLG
020470  001005                          BNE     88$
020472  013737  001216  001214          MOV     NEXT,RETURN
020500  000177  160510                  JMP     @RETURN
020504  104413                  88$:    RAMCLR                  ;CLEAR DV11
020506  032737  000010  007262          BIT #BIT3,LINE          ;DETERMINE LINE NO.
020514  001422                          BEQ     91$             ;
020516  032737  000004  007262          BIT     #BIT2,LINE      ;
020524  001412                          BEQ     89$             ;
020526  013737  001414  001244          MOV     MASK.D,MASKX    ;MASK PRRITY BIT IF SET         ;;++C
020534  113737  001430  023204          MOVB    L12.15,SYNC     ;SET SYNC FOR 12-15
020542  000430                          BR      100$            ;
020544  013737  001412  001244          MOV     MASK.C,MASKX    ;MASK PARITY BIT IF SET         ;;++C
020552  113737  001426  023204  89$:    MOVB    L08.11,SYNC     ;SET SYNC FOR 08-11
020560  000421                          BR      100$            ;
020562  032737  000004  007262  91$:    BIT     #BIT2,LINE      ;
020570  001412                          BEQ     90$             ;
020572  013737  001410  001244          MOV     MASK.B,MASKX    ;MASK PARITY BIT IF SET         ;;++C
020600  113737  001424  023204          MOVB    L04.07,SYNC     ;SET SYNC FOR 04-07
020606  000406                          BR      100$            ;
020610  013737  001406  001244          MOV     MASK.A,MASKX    ;MASK PARITY BIT IF SET         ;;++C
020616  113737  001422  023204  90$:    MOVB    L00.03,SYNC     ;SET SYNC FOR 00-03
020624  113737  023204  023205  100$:   MOVB    SYNC,SYNC+1     ;MAKE SECOND SYNC
020632  012705  023606                  MOV     #TXTAB,R5       ;GET TABLE POINTER
020636  005004                          CLR     R4              ;
020640  112725  000010          101$:   MOVB    #BIT3,(R5)+     ;''INC/BCC'' AND 'MODE 0''
020644  105204                          INCB    R4              ;ALL DONE?
020646  001374                          BNE     101$            ;BR IF NO
020650  012705  023606                  MOV     #TXTAB,R5       ;SET POINTER
020654  005004                          CLR     R4              ;
020656  113704  023204                  MOVB    SYNC,R4         ;SET SYNC CNTRL BYTE
020662  001405                          BEQ     102$            ;BR IF ASYNC LINE CARD.
```

```
2998  020664  042704  177400            BIC     #^C<377>,R4     ;
2999  020670  060405                    ADD     R4,R5           ;
3000  020672  112715  000040            MOVB    #BIT5,(R5)      ;'MODE 1''
3001  020676  012705  023206  102$:     MOV     #TXBAP,R5       ;
3002
3003  020702  005004                    CLR     R4              ;
3004  020704  110425            1$:     MOVB    R4,(R5)+        ;LOAD DATA
3005  020706  105204                    INCB    R4              ;ALL DONE?
3006  020710  022704  000040            CMP     #40,R4          ;
3007  020714  001373                    BNE     1$              ;
3008  020716  013777  007262  160446    MOV     LINE,@DVSRS     ;LOAD LINE NO
3009  020724  105737  023204            TSTB    SYNC            ;IS THIS AN ASYNC CARD?
3010  020730  001006                    BNE     65$             ;BR IF NO
3011  020732  004537  023100            PERFORM ,SETREG         ;
3012  020736     000     001            .BYTE   000,001         ;TXBAP, BYTE CNT
3013  020740  023206                    TXBAP                   ;
3014  020742  077740                    <-40>-BIT15             ;
3015  020744  000405                    BR      66$             ;
3016  020746  004537  023100    65$:    PERFORM ,SETREG         ;
3017  020752     000     001            .BYTE   000,001         ;TX BA, TX BC
3018  020754  023204                    SYNC                    ;SYNC
3019  020756  077736                    <-42>-BIT15             ;MARKED BYTE COUNT
3020  020760  004537  023100    66$:    PERFORM ,SETREG         ;
3021  020764     004     005            .BYTE   004,005         ;RX BA,BC
3022  020766  024206                    RXBA                    ;
3023  020770  077740                    <-40>-BIT15             ;
3024  020772  004537  023100            PERFORM ,SETREG         ;
3025  020776     012     013            .BYTE   012,013         ;
3026  021000  000143                    BIT6+BIT5+BIT1+BIT0     ;
3027  021002  002004                    BIT10+BIT2              ;
3028  021004  004537  023100            PERFORM ,SETREG         ;
3029  021010     016     014            .BYTE   016,014         ;
3030  021012  002000                    BIT10                   ;
3031  021014  000001                    001                     ;IF SYNC LINE CARD; START IN MODE 1
3032  021016  105737  023204            TSTB    SYNC            ;IF ASYNC LINE CARD;
3033  021022  001002                    BNE     .+6             ;SET TX TO MODE 0
3034  021024  005077  160346            CLR     @DVSRA          ;WHICH IS TRUE DDCMP MODE.
3035  021030  004537  023100            PERFORM ,SETREG         ;
3036  021034     010     010            .BYTE   010,010         ;
3037  021036  023206                    TXTAB-400               ;
3038  021040  023206                    TXTAB-400               ;
3039  021042  105737  023204            TSTB    SYNC            ;ASYNC LINE CARD?
3040  021046  001012                    BNE     67$             ;BR IF NOT ASYNC
3041  021050  004537  023144            PERFORM ,LOAD.MODE      ;
3042  021054  015000                    <BIT12+BIT11>+BIT9      ;8 BITS/PER/CHAR.
3043  021056  004537  023144            PERFORM ,LOAD.MODE      ;
3044  021062  020000                    BIT13                   ;RX ENABLE
3045  021064  004537  023144            PERFORM ,LOAD.MODE      ;
3046  021070  072000                    <BIT14+BIT13+BIT12>+BIT10 ;9600 BAJD.
3047  021072  000403                    BR      68$             ;
3048  021074  004537  023144    67$:    PERFORM ,LOAD.MODE      ;MODE FOR CABLE TESTING
3049  021100  030000                    BIT13+BIT12             ;
3050  021102  005277  160254    68$:    INC     @DVSCR          ;SET GC
3051  021106  005005                    CLR     R5              ;
3052  021110  105777  160246    2$:     TSTB    @DVSCR          ;RX BIT7-1?
3053  021114  100404                    BMI     3$              ;YES
```

```
3054  021116  104414                DELAY                   ;WASTE TIME
3055  021120  005205                INC     R5              ;DELAY
3056  021122  001372                BNE     2$              ;
3057  021124  104000                HLT                     ;NO SCR BIT7=1
3058  021126  013705  007262  3$:   MOV     LINE,R5         ;GET LINE NUMBER
3059  021132  000305                SWAB    R5              ;PUT IN HIGH BYTE
3060  021134  052705  050000        BIS     #BIT14+BIT12,R5 ;
3061  021140  017704  160222        MOV     @DVRIC,R4       ;READ RIC
3062                            ;;**************************************************************
3063  021144  143704  001244        BICB    MASKX,R4        ;CLEAR PARITY BIT                ;;++C
3064                                                        ;PARITY BIT IS APPENDED
3065                                                        ;TO HIGH BIT OF CHARACTER
3066                                                        ;WHEN PARITY ENABLED.
3067                            ;;**************************************************************
3068  021150  020504                CMP     R5,R4           ;OK?
3069  021152  001401                BEQ     4$              ;YES
3070  021154  104000                HLT                     ;
3071  021156  005005  4$:   CLR     R5                      ;
3072  021160  005004                CLR     R4              ;
3073  021162  012701  023206        MOV     #TXBAP,R1       ;CHECK DATA!.
3074  021166  012700  024206        MOV     #RXBA,R0        ;
3075  021172  012702  000040        MOV     #40,R2          ;
3076  021176  112004  5$:   MOVB    (R0)+,R4                ;GET RX DATA
3077  021200  042704  177740        BIC     #^C<37>,R4      ;
3078  021204  112105                MOVB    (R1)+,R5        ;GET TX DATA
3079  021206  020504                CMP     R5,R4           ;OK?
3080  021210  001401                BEQ     6$              ;
3081  021212  104000                HLT                     ;RX DATA BAD.!
3082  021214  005302  6$:   DEC     R2                      ;DONE?
3083  021216  001367                BNE     5$              ;
3084  021220  104412                MSTCLR                  ;INIT DV11
3085  021222  104400                SCOPE                   ;SCOPE TEST.
3086
3087
3088
```

```
3089                021224                          LOVE=.
3090                000210                          .=210
3091   000210       000137  021224                          JMP     MANUAL
3092                021224                          .=LOVE
3093   021224       012706  001200          MANUAL: MOV     #STACK,SP
3094   021230       012700  001500                  MOV     #DV.MAP,R0
3095   021234       005020                  1$:     CLR     (R0)+
3096   021236       022700  001740                  CMP     #DV.END,R0
3097   021242       001374                          BNE     1$
3098   021244       104402  022253                  TYPE    ,MXTITLE
3099   021250       004737  022760                  JSR     PC,TKRDY
3100   021254       113737  001272  001301          MOVB    SAVR5,DVNUM
3101   021262       142737  177760  001301          BICB    #^C<17>,DVNUM
3102   021270       112737  000001  001303          MOVB    #1,SAVNUM
3103   021276       012700  001500                  MOV     #DV.MAP,R0
3104   021302       012705  000001          2$:     MOV     #1,R5
3105   021306       104402  022355                  TYPE    ,MXGIVE
3106   021312       113737  001303  001266          MOVB    SAVNUM,SAVR3
3107   021320       104411  023044                  CNVRT   ,XXLIN
3108   021324       104403  022367                  INSTR   ,MXSCR
3109   021330       104405                          PARAM
3110   021332       175000                          175000
3111   021334       175400                          175400
3112   021336       001256                          TEMP5
3113   021340          007     001                  .BYTE   7,1
3114   021342       013720  001256                  MOV     TEMP5,(R0)+
3115   021346       104403  022412                  INSTR   ,MXVEC
3116   021352       104405                          PARAM
3117   021354       000300                          300
3118   021356       000770                          770
3119   021360       001256                          TEMP5
3120   021362          007     001                  .BYTE   7,1
3121   021364       013720  001256                  MOV     TEMP5,(R0)+
3122   021370       113746  001303          65$:    MOVB    SAVNUM,-(SP)
3123   021374       110537  001303                  MOVB    R5,SAVNUM
3124   021400       104402  022544                  TYPE    ,MXGV
3125   021404       113737  001303  001266          MOVB    SAVNUM,SAVR3
3126   021412       104411  023044                  CNVRT   ,XXLIN
3127   021416       112637  001303                  MOVB    (SP)+,SAVNUM
3128   021422       104402  022563                  TYPE    ,MXINST
3129   021426       004737  022760                  JSR     PC,TKRDY
3130   021432       042737  000040  001272          BIC     #40,SAVR5
3131   021440       022737  000131  001272          CMP     #131,SAVR5
3132   021446       001402                          BEQ     .+6
3133   021450       052710  100000                  BIS     #BIT15,(R0)
3134   021454       005710                          TST     (R0)
3135   021456       100532                          BMI     70$
3136   021460       104402  022611                  TYPE    ,MASYNC
3137   021464       004737  022760                  JSR     PC,TKRDY
3138   021470       042737  000040  001272          BIC     #40,SAVR5
3139   021476       022737  000116  001272          CMP     #116,SAVR5
3140   021504       001405                          BEQ     66$
3141   021506       012710  004000                  MOV     #ASYNC,(R0)
3142   021512       005060  000002                  CLR     2(R0)
3143   021516       000512                          BR      70$
3144   021520       104403  022376          66$:    INSTR   ,MXSY1A
```

```
3145  021524  104405                          PARAM
3146  021526  000001                          001
3147  021530  000376                          376
3148  021532  001256                          TEMP5
3149  021534     000     001                  .BYTE   0,1
3150  021536  113710  001256                  MOVB    TEMP5,(R0)
3151  021542  104403  022424                  INSTR   ,MXSY1B
3152  021546  104405                          PARAM
3153  021550  000001                          001
3154  021552  000376                          376
3155  021554  001256                          TEMP5
3156  021556     000     001                  .BYTE   0,1
3157  021560  113760  001256  000002          MOVB    TEMP5,2(R0)
3158  021566  104402  022440                  TYPE    ,MXBITS
3159  021572  004737  022760                  JSR     PC,TKRDY
3160  021576  042737  177770  001272          BIC     #^C<7>,SAVR5
3161  021604  032737  000007  001272  3$:     BIT     #7,SAVR5
3162  021612  001405                          BEQ     4$
3163  021614  062710  000400                  ADD     #400,(R0)
3164  021620  005237  001272                  INC     SAVR5
3165  021624  000767                          BR      3$
3166                                  ;;**********************************************************************************************
3167  021626  104402  022461          4$:     TYPE    ,MPARITY        ;SEE IF PARITY ENABLED          ;;++C
3168  021632  004737  022760                  JSR     PC,TKRDY
3169  021636  042737  000040  001272          BIC     #40,SAVR5
3170  021644  022737  000131  001272          CMP     #'Y,SAVR5       ;IF ANSWER IS YES(Y)
3171  021652  001017                          BNE     5$              ;SET BIT14 OF STAT
3172  021654  052710  040000                  BIS     #PARBIT,(R0)    ;TO ENABLE PARITY
3173  021660  104402  022514                  TYPE    ,MPEVEN         ;SEE IF PARITY EVFN
3174  021664  004737  022760                  JSR     PC,TKRDY
3175  021670  042737  000040  001272          BIC     #40,SAVR5
3176  021676  022737  000131  001272          CMP     #'Y,SAVR5       ;IF EVEN PARITY SELECTED
3177  021704  001002                          BNE     5$              ;SET BIT13 IN STAT
3178  021706  052710  020000                  BIS     #BIT13,(R0)
3179                                  ;;**********************************************************************************************
3180  021712  104402  022643          5$:     TYPE    ,MXSYN
3181  021716  004737  022760                  JSR     PC,TKRDY
3182  021722  042737  000040  001272          BIC     #40,SAVR5
3183  021730  022737  000131  001272          CMP     #131,SAVR5
3184  021736  001402                          BEQ     .+6
3185  021740  052710  010000                  BIS     #BIT12,(R0)
3186  021744  022020                  70$:     CMP     (R0)+,(R0)+
3187  021746  005205                          INC     R5
3188  021750  022705  000005                  CMP     #5,R5
3189  021754  001402                          BEQ     6$
3190  021756  000137  021370                  JMP     65$
3191  021762  105237  001303          6$:     INCB    SAVNUM
3192  021766  123737  001303  001301          CMPB    SAVNUM,DVNUM
3193  021774  101002                          BHI     .+6
3194  021776  000137  021302                  JMP     2$
3195  022002  105037  001300                  CLRB    DVACTV
3196  022006  113737  001301  001303          MOVB    DVNUM,SAVNUM
3197  022014  113701  001301                  MOVB    DVNUM,R1
3198  022020  000241                          CLC
3199  022022  106137  001300                  ROLB    DVACTV
3200  022026  105237  001300                  INCB    DVACTV
```

```
3201  022032  105301                          DECB    R1
3202  022034  001371                          BNE     .-14
3203  022036  113737  001300  001302          MOVB    DVACTV,SAVACT
3204  022044  012710  177777                  MOV     #177777,(R0)
3205  022050  104402  022056                  TYPE    ,MXFIN
3206  022054  000000                          HALT
3207  022056  177777  051440  040524  MXFIN:  .ASCIZ  <377><377>/ START DIAGNOSTICS WITH SW07-1!/<212>
      022121     377  042523  042514  MSEL:   .ASCII  <377>/SELECT LINE(S) XXXXXXXXXXXXXXXX/
      022161     377  020040  020040          .ASCIZ  <377>/                  /
      022202  046377  047111  051505  MLINE:  .ASCIZ  <377>/LINES SELECTED(8):  /<377>
      022231     056  000377          M.CRLF: .ASCIZ  /./<377>
      022234  051777  047111  046107  MSING:  .ASCIZ  <377>/SINGLE LINE: /
      022253                          MXTITLE:
      022253     212  053104  030461          .ASCII  <212>/DV11 MANUAL PARAMETER INPUT PROGRAM./
      022320  042377  030526  023461          .ASCIZ  <377>/DV11'S IN SYSTEM (1 TO 8): /
      022355     212  053104  030461  MXGIVE: .ASCIZ  <212>/DV11 #: /
      022367     377  051503  035122  MXSCR:  .ASCIZ  <377>/CSR: /
      022376  051777  047131  020103  MXSY1A: .ASCIZ  <377>/SYNC "A": /
      022412  053377  041505  047524  MXVEC:  .ASCIZ  <377>/VECTOR: /
      022424  051777  047131  020103  MXSY1B: .ASCIZ  <377>/SYNC "B": /
      022440  041377  052111  026523  MXBITS: .ASCIZ  <377>/BITS-PER-CHAR: /
      022461     377  040520  044522  MPARITY:.ASCIZ  <377>/PARITY ENABLED?(Y OR N): /
      022514  050377  051101  052111  MPEVEN: .ASCIZ  <377>/PARITY EVEN?(Y OR N): /
      022544  046212  047111  020105  MXGV:   .ASCIZ  <212>/LINE CARD #: /
      022563     377  047111  052123  MXINST: .ASCIZ  <377>/INSTALLED?(Y OR N): /
      022611     377  051501  047131  MASYNC: .ASCIZ  <377>/ASYNCHRONOUS ?(Y OR N): /
      022643     377  053524  020117  MXSYN:  .ASCIZ  <377>/TWO SYNCS? (Y OR N): /
      022672  024377  024501  044040  MTURN:  .ASCIZ  <377>/(A) H325/<377>/(B) H861/<377>/TYPE "A" OR "B": /
      022737     377  047515  042504  MVECZ:  .ASCIZ  <377>/MODEM VECTOR: /
                 022760                  .EVEN
      022760  105777  156220          TKRDY:  TSTB    @TKCSR
      022764  100375                          BPL     .-4
      022766  017746  156214                  MOV     @TKDBR,-(SP)
      022772  042716  000200                  BIC     #BIT7,(SP)
      022776  032716  000100                  BIT     #BIT6,(SP)      ;CHAR OR NUMBER
      023002  001402                          BEQ     .+6             ;BR IF NUMBER
      023004  042716  000040                  BIC     #BIT5,(SP)      ;MAKE UPPER CASE
      023010  022716  000015                  CMP     #15,(SP)
      023014  001411                          BEQ     1$
      023016  011637  001272                  MOV     (SP),SAVR5
      023022  105777  156162                  TSTB    @TPCSR
      023026  100375                          BPL     .-4
      023030  011677  156156                  MOV     (SP),@TPDBR
      023034  005726                          TST     (SP)+
      023036  000750                          BR      TKRDY
      023040  005726                  1$:     TST     (SP)+
      023042  000207                          RTS     PC
      023044  000001                  XXLIN:  1
3208  023046     002       001                .BYTE   2,1
3209  023050  001266                          SAVR3
3210
3211  023052                          CKBIT15:
3212  023052  010046                          MOV     R0,-(SP)
3213  023054  005000                          CLR     R0
3214  023056  005777  156306          64$:    TST     @DVLCR
3215  023062  100004                          BPL     65$
```

```
3216  023064  104414                          DELAY
3217  023066  005200                          INC     R0
3218  023070  001372                          BNE     64$
3219  023072  104000                          HLT     0                       ;BIT 15 FAILED TO CLEAR
3220  023074  012600                  65$:    MOV     (SP)+,R0
3221  023076  000207                          RTS     PC
3222  023100  010046                  SETREG: MOV     R0,-(SP)
3223  023102  010146                          MOV     R1,-(SP)
3224  023104  112500                          MOVB    (R5)+,R0
3225  023106  112501                          MOVB    (R5)+,R1
3226  023110  110077  156260                  MOVB    R0,@DVSRSH
3227  023114  012577  156256                  MOV     (R5)+,@DVSRA
3228  023120  042777  000060  156234          BIC     #BIT5+BIT4,@DVSCR
3229  023126  110177  156242                  MOVB    R1,@DVSRSH
3230  023132  012577  156240                  MOV     (R5)+,@DVSRA
3231  023136  012601                          MOV     (SP)+,R1
3232  023140  012600                          MOV     (SP)+,R0
3233  023142  000205                          EXIT
3234
3235  023144                          LOAD.MODE:
3236  023144  012577  156220                  MOV     (R5)+,@DVLCR
3237  023150  052777  100000  156212          BIS     #BIT15,@DVLCR
3238  023156  010046                          MOV     R0,-(SP)
3239  023160  005000                          CLR     R0
3240  023162  005777  156202          1$:     TST     @DVLCR
3241  023166  100004                          BPL     2$
3242  023170  104414                          DELAY
3243  023172  005200                          INC     R0
3244  023174  001372                          BNE     1$
3245  023176  104000                          HLT     0                       ;BIT 15 FAILED TO CLEAR
3246  023200  012600                  2$:     MOV     (SP)+,R0
3247  023202  000205                          EXIT
3248  023204  000001                  SYNC:   .BLKW 1
3249  023206  000400                  TXBAP:  .BLKB   400
3250  023606  000400                  TXTAB:  .BLKB   400
3251  024206  000400                  RXBA:   .BLKB   400
3252  024606  051777  047111  046107  EM1:    .ASCII  <377>/SINGLE LINE CABLE TESTS(DV11 ERROR)/
      024652  051377  053103  020122          .ASCIZ  <377>/RCVR INTERRUPT (BIT 7 OF DVSCR) FAILED TO SET/
      024731     377  040503  046102  EM2:    .ASCIZ  <377>/CABLE TURN AROUND TESTS (MODEM CONTROL ERROR)/
      025010  046777  042117  046505  EM3:    .ASCIZ  <377>/MODEM CONTROL ERROR/
      025035     377  054105  042520  DH4:    .ASCIZ  <377>/EXPECTED  FOUND  REGISTER/
      025070  052777  042516  050130  EM4:    .ASCIZ  <377>/UNEXPECTED MODEM CONTROL INTERUPT./
      025134  046777  042117  046505  EM5:    .ASCIZ  <377>/MODEM CONTROL FAILED TO INTERUPT/
      025176  051377  040505  044504  EM6:    .ASCIZ  <377>/READING MODEM CONTROL CAUSED AT TRAP TO 4./
      025252  042777  050130  041505  DH1:    .ASCIZ  <377>/EXPECTED  FOUND  LINE  DVSCR  MC.CSR/
                                              .EVEN
      025320  000005                  DT1:    5
3253  025322     006      004                 .BYTE 6,4
3254  025324  001272                          SAVR5
3255  025326     006      001                 .BYTE   6,1
3256  025330  001270                          SAVR4
3257  025332     002      004                 .BYTE   2,4
3258  025334  007262                          LINE
3259  025336     006      001                 .BYTE   6,1
3260  025340  001362                          DVSCR
3261  025342     006      001                 .BYTE 6,1
```

```
3262  025344  007300                          MC.CSR
3263  025346  000003                  DT2:    3
3264  025350     006      004                 .BYTE   6,4
3265  025352  001272                          SAVR5
3266  025354     006      001                 .BYTE   6,1
3267  025356  001270                          SAVR4
3268  025360     006      001                 .BYTE   6,1
3269  025362  001266                          SAVR3
3270  025364                          .ERRTAB:
3271  025364  024606                          EM1
3272  025366  025252                          DH1
3273  025370  025320                          DT1
3274  025372  024731                          EM2
3275  025374  025252                          DH1
3276  025376  025320                          DT1
3277  025400  025010                          EM3
3278  025402  025035                          DH4
3279  025404  025346                          DT2
3280  025406  025070                          EM4
3281  025410  000000                          0
3282  025412  000000                          0
3283  025414  025134                          EM5
3284  025416  000000                          0
3285  025420  000000                          0
3286  025422  025176                          EM6
3287  025424  000000                          0
3288  025426  000000                          0
3289  025430  000000                          0
3290  025432  000000                          0
3291  025434  000000                          0
3292          000001                  .END
```

```
ADRCNT= 003443  619* 655* 664#
ALU   = 010000   73#
ASYNC = 004000   81# 3141
AUTO.S  006624 1129#
BCC   = 060000   78#
BINWRD  003746  705* 706  743#
BIT0  = 000001   71# 886  1230 3026
BIT1  = 000002   70# 886  897  1233 3026
BIT10 = 002000   61# 886  3027 3030 3046
BIT11 = 004000   60# 886  1446 3042
BIT12 = 010000   59#  73   75   77   79 3042 3046 3049 3060 3185
BIT13 = 020000   58#  74   75   78   79 3044 3046 3049 3178
BIT14 = 040000   57#  76   77   78   79  526 3046 3060
BIT15 = 100000   56# 3014 3019 3023 3133 3237
BIT2  = 000004   69#  453  886 1234 2973 2981 3027
BIT3  = 000010   68#  886 1235 2971 2991
BIT4  = 000020   67# 1236 3228
BIT5  = 000040   66# 1237 3000 3026 3207 3228
BIT6  = 000100   65# 1197 1238 3026 3207
BIT7  = 000200   64#  520  767  924  945 1197 1239 1449 3207
BIT8  = 000400   63#  886  903
BIT9  = 001000   62#  886 1197 1202 3042
BRB   = 070000   79#
BRW     003014  459  548#
BRX     003016  460  549#
BUSY  = 000020 1215# 1623 1632 1959 1991 2200
CHAR    007266 1244# 1416* 1425 1437*
CHRCNT  003744  703*  707  723*  741#  742
CKBIT1  023052 3211#
CLKX    001242  150#
CLK.A   001416  250# 1047
CLK.B   001417  251# 1052
CLK.C   001420  252# 1057
CLK.D   001421  253# 1062
CLRMUX= 002000 1221# 1937 2082 2115 2192 2253 2300 2347 2394 2584 2638 2692 2746
CLRSCN= 004000 1222# 1958 1990 2199
CNVRT = 104411  209#  478  480  482  484  804  806  862  919 1391 3107 3126
CO    = 000100 1238# 2499 2811
COF   = 040000 1225#
CONVRT= 104410  207#  418  820
COUNT   007270 1245# 1414* 1426* 1428* 1435*
CREAM   001306  171#  387*  994*  995  997* 1002 1003* 1004 1007*
CS    = 000040 1237# 2499 2811
CSF   = 020000 1224#
CSRMAP  006626  412 1131#
CYCLE   005666  462  498  499  984#
DATABP  004276  793*  796  818  821#
DATACL= 104416  219#
DATAHD  004264  792*  814  817#
DELAY = 104414  215# 1913 1948 1965 2002 2046 2135 2213 2276 2323 2370 2417 2464
                2511 2558 2608 2662 2716 2770 2823 2877 2931 3054 3216 3242
DEVADR  003440  617*  652  662#
DH1     025252 3252# 3272 3275
DH4     025035 3252# 3278
DONE  = 000200 1218# 1540 1542 1543 1547 1548 1550 1651 1695 1717 1738 1759 1780
               1801 1822 1843 1864 2152 2230
```

```
DSR    = 000020          1236#  2546
DT1      025320          3252#  3273   3276
DT2      025346          3263#  3279
DVACTV   001300           165#   434*   435    984    989   1163*  1169*  1170*  1174   1193   3195*  3199*  3200*
                         3203
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   3214   3236*  3237*  3240
DVNSR    001402           238#  1033*  1034*  1035
DVNUM    001301           166#   383    488   1135*  1156*  1157   1164   1166   3100*  3101*  3192   3196   3197
DVRIC    001366           232#  1021*  1022*  1023   3061
DVRLVL   001354           227#  1038*  1039*  1040
DVRVEC   001352           226#   505   1009*  1038
DVSCR    001362           230#   502    880*   893*   897*  1008*  1019   1326   1402   1410   3050*  3052   3228*
                         3260
DVSCRH   001364           231#  1019*  1020*  1021
DVSFR    001400           237#  1031*  1032*  1033
DVSRA    001376           236#   884   1029*  1030*  1031   3034*  3227*  3230*
DVSRS    001372           234#   883   1025*  1026*  1027   3008*
DVSRSH   001374           235#  1027*  1028*  1029   3226*  3229*
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   3096
DV.MAP   001500           171    282#   387    414    997   1007   1131   1136   1185   3094   3103
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     024606           3252#   3271
EM2     024731           3252#   3274
EM3     025010           3252#   3277
EM4     025070           3252#   3280
EM5     025134           3252#   3283
EM6     025176           3252#   3286
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#
EXERCI  007274           1247#   1418*   1421*   1439*
EXIT  = 000205             81#   3233    3247
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    1350    1404    3108    3115    3144    3151
INSTR2  003236            593     605#
INTENA= 000100           1217#   1513    1515    1516    1520    1521    1523    1672    1692    1716    1737    1758    1779
                         1800    1821    1842    1863    2128    2206
KBISR   010276           1330    1451    1456#
LIGHT   000174            110#    404
LIGHTS  001200            121#    404*    475*
LIMITS  003364            632     643#
LINE    007262           1242#   1354    1357*   1419*   1420*   1440*   2252    2299    2346    2393    2441    2488    2535
                         2587*   2641*   2695*   2749*   2971    2973    2981    3008    3058    3258
LINENA= 000001           1230#   2091    2121    2256    2264    2444    2452    2460    2491    2499    2507    2538    2546
                         2554    2588    2596    2803    2811    2819    2857    2865    2873    2911    2919    2927
LOAD.M  023144           3041    3043    3045    3048    3235#
LOBITS  003442            618*    647     663#    664
LOCK    001220            137#    542*    556     558     800    1879*   1905*   1935*   1957*   1984*   2028*   2071*   2113*
                         2168*   2583*   2637*   2691*   2745*   2800*   2854*   2908*
LOGICA  002560            107     492#
LOKFLG  001312            178#
LOLIM   003434            615*    645     660#
LOVE  = 021224           3089#   3092
LPCNT   001224            139#    537*    538     541*
LSTERR  001234            143#    390*    472*    524*    778     780*    864*
```

```
L00.03  001422   255#  1010*  1045  2170  2987
L04.07  001424   256#  1012*  1050  2173  2984
L08.11  001426   257#  1014*  1055  2176  2979
L12.15  001430   258#  1016*  1060  2179  2976
MAINT = 001000  1220#  1567   1569  1570  1574  1575  1577  1940  1993  1996  2035  2119
MANUAL  021224  3091   3093#
MAR18 = 007502  1349#  1461
MASKX   001244   151#  2975*  2978*  2983*  2986*  3063
MASK.A  001406   245#  1046   2986
MASK.B  001410   246#  1051   2983
MASK.C  001412   247#  1056   2978
MASK.D  001414   248#  1061   2975
MASTEK  005400   802    960#
MASYNC  022611  3136   3207#
MCRLF   005104   571    694    798   799   807   948   960#  1069  1087
MCSRX   005330   477    960#
MC.CSR  007300  1249#  1326*  1327*  1328  1488  1512  1539  1566  1593  1619  1648*  1651*  1669*
                1672*  1690*  1692*  1695*  1712*  1716*  1717*  1733*  1737*  1738*  1754*  1758*  1759*
                1775*  1779*  1780*  1796*  1800*  1801*  1817*  1821*  1822*  1838*  1842*  1843*  1859*
                1863*  1864*  1880   1906   1936   1985   2029   2072   2114   2190   2250*  2253*  2255*
                2257*  2260   2269*  2273*  2297*  2300*  2302*  2304*  2307   2316*  2320*  2344*  2347*
                2349*  2351*  2354   2363*  2367*  2391*  2394*  2396*  2398*  2401   2410*  2414*  2439*
                2443*  2445*  2448   2457*  2462*  2486*  2490*  2492*  2495   2504*  2509*  2533*  2537*
                2539*  2542   2551*  2556*  2579*  2584*  2586*  2589*  2592   2601*  2605*  2633*  2638*
                2640*  2643*  2646   2655*  2659*  2687*  2692*  2694*  2697*  2700   2709*  2713*  2741*
                2746*  2748*  2751*  2754   2763*  2767*  2796*  2802*  2804*  2807   2816*  2821*  2850*
                2856*  2858*  2861   2870*  2875*  2904*  2910*  2912*  2915   2924*  2929*  3262
MC.LSR  007302  1250#  1328*  1329*  1492  2076*  2084   2091*  2092   2098*  2121*  2195*  2256*  2259
                2275*  2277   2303*  2306  2322*  2324   2350*  2353   2369*  2371   2397*  2400   2416*
                2418   2444*  2447   2463*  2465  2491*  2494   2510*  2512   2538*  2541   2557*  2559
                2588*  2591   2607*  2609  2642*  2645   2661*  2663   2696*  2699   2715*  2717   2750*
                2753   2769*  2771   2803*  2806  2822*  2824   2830*  2857*  2860   2876*  2878   2884*
                2911*  2914   2930*  2932  2938*
MC.LVL  007306  1252#  1411*  1412*  1650*  1671*  1693*  1715*  1736*  1757*  1778*  1798*  1819*  1840*
                1861*  2126*  2205*
MC.VEC  007304  1251#  1408   1411   1649*  1670*  1691*  1714*  1735*  1756*  1777*  1797*  1818*  1839*
                1860*  2125*  2204*
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#
MLINE   022202  1387   3207#
MLOCK   005301   455    960#
MNEW    005402   426    960#
MPARIT  022461  3167   3207#
MPASSX  005344   481    960#
MPEVEN  022514  3173   3207#
MPFAIL  005107   861    960#
MQM     005100   601    960#   1092
MR      005171   463    960#
MRESET= 004000    81#   880    893
MSEL    022121  1362   3207#
MSING   022234  1350   3207#
MSTCLR- 104412   211#   865    3084
```

```
MTITLE  001000          119#    408
MTSTN   005366          803     960#    1071
MTSTPC  005267          960#
MTURN   022672         1336    3207#
MVECX   005336          479     960#
MVECZ   022737         1404    3207#
MXBITS  022440         3158    3207#
MXFIN   022056         3205    3207#
MXGIVE  022355         3105    3207#
MXGV    022544         3124    3207#
MXINST  022563         3128    3207#
MXSCR   022367         3108    3207#
MXSYN   022643         3180    3207#
MXSY1A  022376         3144    3207#
MXSY1B  022424         3151    3207#
MXTITL  022253         3098    3207#
MXVEC   022412         3115    3207#
M.CRLF  022231         1395    3207#
NEXT    001216          136#    544     834    1324*   1443*   1475*   1511*   1538*   1565*   1592*   1618*   1646*   1667*
                       1688*   1711*   1732*   1753*   1774*   1795*   1816*   1837*   1858*   1878*   1904*   1934*   1983*
                       2027*   2070*   2112*   2167*   2188    2245*   2248    2292*   2295    2339*   2342    2386*   2389
                       2434*   2437    2481*   2484    2528*   2531    2574*   2577    2628*   2631    2682*   2685    2736*
                       2739    2791*   2794    2845*   2848    2899*   2902    2965*   2968
NPR     = 040000         76#
NS      = 000010       1235#    2397    2405    2538    2546    2557
PARAM   = 104405        201#    1072    1351    1405    3109    3116    3145    3152
PARAM1  003304          621#     638
PARBIT= 040000           81#    1116    3172
PARERR  003360          624      626     628    637#     644     646     648
PASCNT  001230          141#     384*    474*    475     508
PERFOR= 004537           81#    3011    3016    3020    3024    3028    3035    3041    3043    3045    3048
PFTAB   004470          862      868#
POINTE  007264         1243#    1413*   1414    1415*   1416    1417*   1428    1434*   1435    1436*   1437    1438*
POPRO   = 012600         55#     828
POP1SP= 005726           55#
POP2SP= 022626           55#     546    1655    1676    1699    1720    1741    1762    1783    1805    1826    1847    1868
                        2146    2224
PS      - 177776         53#     380*    449*   1196*   1325*   1333*   1388*   1647*   1652*   1668*   1673*   1689*   1694*
                       1713*   1734*   1755*   1776*   1799*   1820*   1841*   1862*   1882*   1907*   1938*   1987*   2031*
                       2074*   2117*   2126    2129*   2131*   2141*   2194*   2205    2207*   2209*   2219*   2251*   2298*
                       2345*   2392*   2440*   2487*   2534*   2580*   2634*   2688*   2742*   2797*   2851*   2905*
PUSHR0= 010046           55#     825
PUSH1S= 005746           55#
PUSH2S= 024646           55#
QV.FLG  001313          179#     386*    487*    535    1424*
RAM     = 020000         74#
RAMCLR= 104413          213#     866    2970
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
                       1432*   1433    1442*   1443    1454    1500    2188*   2189    2248*   2249    2295*   2296    2342*
                       2343    2389*   2390    2437*   2438    2484*   2485    2531*   2532    2577*   2578    2631*   2632
                       2685*   2686    2739*   2740    2794*   2795    2848*   2849    2902*   2903    2968*   2969
```

```
RING  = 000200  1239#  2452  2865
RINGF = 100000  1226#
ROMCLK= 104415   217#
RS    = 000004  1234#  2350  2358  2491  2499  2510  2696  2704  2857  2865  2876
RUN     001304   169#   388*   989   992*   993*  1000*  1001*
RXBA    024206  3022   3074  3251#
SAVACT  001302   167#   429  1174*  3203*
SAVNUM  001303   168#   383*   485*   488*  1167*  3102*  3106  3122  3123*  3125  3127*  3191*  3192
                3196*
SAVPC   001276   164#   669*   840
SAVR0   001260   157#   678*   683
SAVR1   001262   158#   677*   684
SAVR2   001264   159#   676*   685
SAVR3   001266   160#   675*   686  1385*  1392*  1393  3106*  3125*  3209  3269
SAVR4   001270   161#   674*   687  2132*  2138*  2210*  2216*  3256  3267
SAVR5   001272   162#   673*   688  1338  1342  3100  3130*  3131  3138*  3139  3160*  3161  3164*
                3169*  3170  3175*  3176  3182*  3183  3207*  3254  3265
SAVSP   001274   163#
SAV05 = 104406   203#   782
SCNENA= 000040  1216#  1594  1596  1597  1601  1602  1604  1620  1627  2130  2152  2208  2230
SCOPE - 104400   191#  1497  1527  1554  1581  1608  1636  1657  1678  1700  1722  1743  1764
                1785  1806  1827  1848  1869  1894  1922  1974  2017  2061  2101  2156  2234
                2281  2328  2375  2422  2469  2516  2563  2617  2671  2725  2779  2833  2887
                2941  3085
SCOP1   104401   193#  1890  1918  1953  1970  2011  2055  2089  2096  2144  2151  2222  2229
                2613  2667  2721  2775  2828  2882  2936
SECRX = 000020  1232#  2919
SECRXF= 010000  1223#
SECTX = 000010  1231#  2750  2758  2911  2919  2930
SELECT  007272  1246#  1345*  1356*  1360*  1363  1364*  1372*  1381*  1383  1386  1418  1439
SERV.G  004640   523   770   914#   915
SETREG  023100  3011  3016  3020  3024  3028  3035  3222#
SPACNT= 003745   704*   725   728*   742#
SSWR    000176   112#   405
STACK = 001200    54#   381   450   835   857  1499  3093
STAT    001236   148#
STEP  - 000400  1219#  1912  1941  1947  1964  1995  2001  2036  2040  2045  2077  2120  2196
                2269  2316  2363  2410  2457  2504  2551  2601  2655  2709  2763  2816  2870
                2924
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  1347
SW00  = 000001    34#   424
SW01  = 000002    33#  1064
SW02  = 000004    32#
SW03  = 000010    31#
SW04  = 000020    30#
SW05  = 000040    29#
SW06  = 000100    28#  1347
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     023204            2976*  2979*  2984*  2987*  2988*  2996   3009   3018   3032   3039   3248#
SYNCX    001240             149#
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*
SYNC2B   001434             261#  1013*
SYNC2C   001436             262#  1015*
SYNC2D   001440             263#  1017*
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#
TABLE    010330            1413   1434   1465#
TEMP     005562             708    858*   859*   971#
TEMP1    001246             152#   414*   415    420*   962   1193*  1194*
TEMP2    001250             153#   415*   416    964
TEMP3    001252             154#  1363*  1372   1997*  2013*  2034*  2037*  2041*  2057*  2080*  2083   2097*
TEMP4    001254             155#
TEMP5    001256             156#  3112   3114   3119   3121   3148   3150   3155   3157
TESTER   010064            1420#  1423   1441   2965
TKCSR    001204             127#   529    588    914    921    942   1332*  1365   1446   1480*  1483   1484   3207
TKDBR    001206             128#   519    531    590    596    766    923    944   1367   1448   1457   3207
TKRDY    022760            1337   3099   3129   3137   3159   3168   3174   3181   3207#
TLAST    020450            1090   3277#
TOTAL    007276            1248#  2185*  2202   2581   2635   2689   2743   2798   2852   2906
TPCSR    001210             129#   572    594    773    928   1476   3207
```

```
TPDBR   001212   130#   574*   596*   775*   930*  1368*  3207*
TRMRDY= 000002  1233#  2303   2311   2444   2452   2463   2642   2650   2803   2811   2822
TRPOK   003762   754#
TSTNO   001226   140#   391*   843    870    1075   1082   1084   1323*  1474*  1510*  1537*  1564*  1591*
                1617*  1645*  1666*  1687*  1710*  1731*  1752*  1773*  1794*  1815*  1836*  1857*  1877*
                1903*  1933*  1982*  2026*  2069*  2111*  2166*  2244*  2291*  2338*  2385*  2433*  2480*
                2527*  2573*  2627*  2681*  2735*  2790*  2844*  2898*  2964*
TST1    007310  1078   1094   1323#
TST10   011600  1618   1645#
TST11   011666  1646   1666#
TST12   011754  1667   1687#
TST13   012050  1688   1710#
TST14   012140  1711   1731#
TST15   012230  1732   1752#
TST16   012320  1753   1773#
TST17   012410  1774   1794#
TST2    010770  1324   1442   1474#
TST20   012476  1795   1815#
TST21   012564  1816   1836#
TST22   012652  1837   1857#
TST23   012740  1858   1877#
TST24   013026  1878   1903#
TST25   013124  1904   1933#
TST26   013326  1934   1982#
TST27   013510  1983   2026#
TST3    011126  1475   1510#
TST30   013674  2027   2069#
TST31   014052  2070   2111#
TST32   014304  2112   2166#
TST33   014650  2167   2244#
TST34   015050  2245   2291#
TST35   015250  2292   2338#
TST36   015450  2339   2385#
TST37   015650  2386   2433#
TST4    011220  1511   1537#
TST40   016046  2434   2480#
TST41   016244  2481   2527#
TST42   016442  2528   2573#
TST43   016666  2574   2627#
TST44   017112  2628   2681#
TST45   017336  2682   2735#
TST46   017562  2736   2790#
TST47   020004  2791   2844#
TST5    011312  1538   1564#
TST50   020226  2845   2898#
TST51   020450  2899   2964#  3277
TST6    011404  1565   1591#
TST7    011476  1592   1617#
TTST    002702   456*   457*   459*   460*   527#
TURFLG  007260  1241#  1340*  1344*  2186   2246   2293   2340   2387   2435   2482   2529   2575   2629
                2683   2737   2792   2846   2900   2966
TWOSYN= 010000    81#
TXB4P   023206  3001   3013   3073   3249#
TXTAB   023606  2989   2994   3037   3038   3250#
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    1336    1362    1387    1395    1425
                       3098    3105    3124    3128    3136    3158    3167    3173    3180    3205
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#
XHEAD   005461          413     960#
XPASS   002620          482     506#
XSTATQ  005506          419     960#
XTSTN   004374          804     841#
XVEC    002612          480     503#
XXLIN   023044         1391    3107    3126    3207#
$CRAP   177777            1#    1466#   1470#   1503#   1506#   1530#   1533#   1557#   1560#   1584#   1587#   1610#   1613#
                       1638#   1641#   1659#   1662#   1680#   1683#   1703#   1706#   1724#   1727#   1745#   1748#   1766#
                       1769#   1787#   1790#   1808#   1811#   1829#   1832#   1850#   1853#   1870#   1873#   1896#   1899#
                       1923#   1929#   1975#   1978#   2018#   2022#   2062#   2065#   2103#   2107#   2157#   2162#   2235#
                       2240#   2282#   2287#   2329#   2334#   2376#   2381#   2424#   2429#   2471#   2476#   2518#   2523#
                       2564#   2569#   2618#   2623#   2672#   2677#   2726#   2731#   2781#   2786#   2835#   2840#   2889#
                       2894#   2942#   2960#
$E    - 000053            1#    1324    1325#   1475    1476#   1511    1512#   1538    1539#   1565    1566#   1592    1593#
                       1618    1619#   1646    1647#   1667    1668#   1688    1689#   1711    1712#   1732    1733#   1753
                       1754#   1774    1775#   1795    1796#   1816    1817#   1837    1838#   1858    1859#   1878    1880#
                       1904    1906#   1934    1936#   1983    1985#   2027    2029#   2070    2072#   2112    2114#   2167
                       2169#   2245    2246#   2292    2293#   2339    2340#   2386    2387#   2434    2435#   2481    2482#
                       2528    2529#   2574    2575#   2628    2629#   2682    2683#   2736    2737#   2791    2792#   2845
                       2846#   2899    2900#   2966#
$N    - 000051            1#    1321    1325#   1466    1472    1476#   1503    1508    1512#   1530    1535    1539#   1557
                       1562    1566#   1584    1589    1593#   1610    1615    1619#   1638    1643    1647#   1659    1664
                       1668#   1680    1685    1689#   1703    1708    1712#   1724    1729    1733#   1745    1750    1754#
                       1766    1771    1775#   1787    1792    1796#   1808    1813    1817#   1829    1834    1838#   1850
                       1855    1859#   1870    1875    1880#   1896    1901    1906#   1923    1931    1936#   1975    1980
                       1985#   2018    2024    2029#   2062    2067    2072#   2103    2109    2114#   2157    2164    2169#
                       2235    2242    2246#   2282    2289    2293#   2329    2336    2340#   2376    2383    2387#   2424
                       2431    2435#   2471    2478    2482#   2518    2525    2529#   2564    2571    2575#   2618    2625
                       2629#   2672    2679    2683#   2726    2733    2737#   2781    2788    2792#   2835    2842    2846#
                       2889    2896    2900#   2942    2962    2966#   3277#
$Y      000017              1#    182#    191     193#    195#    197#    199#    201#    203#    205#    207#    209#    211#
                        213#    215#    217#    219#    221#
.       025436             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    1249#   1250#
                       1252#   1349    1371    1431    1447    1465#   1477    1490    1494    1518    1525    1545    1552
                       1572    1579    1599    1606    1625    1634    1960    1992    2134    2139    2183    2201    2212
                       2217    2279    2326    2373    2420    2467    2514    2561    2611    2665    2719    2773    2826
                       2880    2934    3033    3089    3090#   3092#   3132    3184    3193    3202    3207#   3248#   3249#
```

		3250#	3251#		
.BEGIN	002332	449#			
.CNVRT	003542	210	695#		
.CONVR	003536	208	694#		
.DATAC	004576	220	900#		
.DELAY	004476	216	871#		
.EOP	002436	471#	1432		
.ERRTA	025364	790	3270#		
.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#		
.SAVO5	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   1491   1495   1498   1519   1526   1546   1553   1573   1580   1600   1607   1626   1635
        1656   1677   1697   1721   1742   1763   1784   1803   1824   1845   1866   1889   1917   1952   1969
        2010   2054   2088   2095   2140   2143   2150   2218   2221   2228   2268   2280   2315   2327   2362
        2374   2409   2421   2456   2468   2503   2515   2550   2562   2600   2612   2654   2666   2708   2720
        2762   2774   2815   2827   2869   2881   2923   2935   3057   3070   3081   3219   3245
INTS    1465#  1786   1807   1828   1849
MUXS1      1#  2235   2282   2329   2376   2564   2618   2672   2726
MUXS2      1#  2423   2470   2517   2780   2834   2888
NOINT   1466#  1702   1723   1744   1765
$BUFFE     1#    966
$CK15      1#
$CK150     1#   3211
$CLR.T     1#
$CYCLE     1#    975
$EGOLF  1502#  1529   1556   1583
$EOP       1#    465
$FINI      1#   3277
$GETFL     1#
$GETPA     1#   1070
$HEADE     1#
$MSG       1#    960
$PFAIL     1#    844
$RAMCL     1#    871
$RXSHI     1#
$SCOPE     1#    512
$SETLI     1#
$SETSC     1#
$SETSY     1#
$SET.T     1#
$SILOI     1#
$SIMBC     1#
$TRPDF     1#    191    193    195    197    199    201    203    205    207    209    211    213    215    217
         219
$TSTN      1#   1321   1472   1508   1535   1562   1589   1615   1643   1664   1685   1708   1729   1750   1771
        1792   1813   1834   1855   1875   1901   1931   1980   2024   2067   2109   2164   2242   2289   2336
        2383   2431   2478   2525   2571   2625   2679   2733   2788   2842   2896   2962
$TXSHI     1#
$VARIA     1#    117
$XZ        1#   1466   1470   1503   1506   1530   1533   1557   1560   1584   1587   1610   1613   1638   1641
        1659   1662   1680   1683   1703   1706   1724   1727   1745   1748   1766   1769   1787   1790   1808
        1811   1829   1832   1850   1853   1870   1873   1896   1899   1923   1929   1975   1978   2018   2022
        2062   2065   2103   2107   2157   2162   2235   2240   2282   2287   2329   2334   2376   2381   2424
        2429   2471   2476   2518   2523   2564   2569   2618   2623   2672   2677   2726   2731   2781   2786
        2835   2840   2889   2894   2942   2960
```

. ABS. 025436 000

ERRORS DETECTED: 0

CZDVEC,CZDVEC/SOL/CRF/DOC-CZDVEC.MAC,CZDVEC.P11
RUN-TIME: 26 39 4 SECONDS
RUN-TIME RATIO: 330/69 4.7

CZDVEC.P11 19-MAR-79 09:06 CROSS REFERENCE TABLE -- MACRO NAMES VE MACY SEQ 0102

CORE USED: 18K (35 PAGES)

DOCUMENT PAGES: 83