MODULE REPAIR GUIDE NO. 4

EDITED BY CUSTOMER ENGINEERING DIVISION

February 21, 1978

MODEL 2200VP SYSTEM REPAIR

1. INTRODUCTION

A tester has been designed and built, which along with a series of DEBUG PROMs, will facilitate repair of 2200VP printed circuit boards. In the future, supplements to this MRG will be published containing information dealing specifically with the repair procedure of the individual modules in the 2200VP. Sections 2 and 3 of this MRG cover the installation and operational aspects of the tester. Schematics for the tester assembly can be found at the end of this repair guide.

2. INSTALLATION

2.1 EQUIPMENT REQUIRED

Qty.	Part #	Description
1	190-0718	Tester Assembly with cables

2.2 PROCEDURE

Remove the first Control Memory board (210-6788 PCB in the fourth PCB location from the heat sink) from the CPU and insert it into the location provided on the lightboard (Ref. 210-7139 schematic). Install the seven 24 pin lightboard cables into their appropriate locations on the 210-7138 Interface board (the number on the cable coincides with the number of the jack that it is to be connected to), then install the 7138 in the CPU in the location previously occupied by the 210-6788 PCB. Remove the 210-6790 from the CPU and connect the 16 pin lightboard cable to J1, then reinstall the 6790 in the CPU.

Printed in U.S.A.

WANG LABORATORIES, INC.

ONE INDUSTRIAL AVENUE, LOWELL, MASSACHUSETTS 01851, TEL. (617) 851-4111, TWX 710 343-6769, TELEX 94-7421

NOTE:

Pin 1 of each lightboard cable must plug into pin 1 of each female connector.

Cont. Mem. Interface 210-7138	CABLE #	(PART #)	210-7139
J1	1	(220-3038)	J10
J2	2	(220-3037)	J6
J3	3	(220-3037)	J9
J4	4	(220-3036)	J8
J5	5	(220-3035)	J5
J6	6	(220-3018)	J4
J7	7	(220-3018)	J7
210-6790			
J1	N/A	(220-3000)	J11

Before attempting to operate the system, ensure the power supply regulated voltages are at their proper levels by checking them at the lightboard (connectors 1 and 3 of the 210-6788). Reference 6788 schematic for pin locations.

After the lightboard has been installed and the voltages checked, a simple test should be performed to ensure proper operation. An example of such a test follows:

SWITCH	POSITION
IC/PC	UP
CM/SWITCHES	DOWN
COMPARE	DOWN
RUN/STEP	UP
A_{15}-A_0	DOWN
R_{23}-R_0	HEX 5CXXXX

Toggle ROM bit switches R_2-R_{17} one at a time and observe the memory address indicators A_{15}-A_0. One and only one indicator

should come on for each ROM bit selected. The ROM bits with the corresponding address indicators that they activate are:

R:	8	9	10	11	12	13	14	15	16	17	2	3	4	5	6	7
A:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

A 2200VP diagnostic should also be loaded from the system platter and run to ensure proper operation of the system. For normal operation of the 2200VP, all the toggle switches on the lightboard should be down with the exception of the RUN/STEP switch which should be in the RUN position and the ROM switch (CM/SWITCHES) which should be in the CM position.

3. OPERATION

3.1 LIGHT INDICATORS

Each set of light indicators (labeled accordingly) represent the following:

1) Control memory output (R_{23}-R_0)
2) Memory address (A_{15}-A_0)
3) K register (K_7-K_0)

3.2 ROM BIT AND ROM SWITCHES

Indicators R_{23}-R_0 (see above) can also represent a manually set ROM instruction, to be used in lieu of (supersedes) the control memory outputs. Such manually introduced commands to the CPU are set on the ROM bit switches located directly below the indicators. Placing the switch in the up position activates the corresponding bit. The ROM switch, labeled CM/SWITCHES, must be in the down position to allow the ROM bit switches to control the CPU.

3.3 RUN/STEP AND STEP PUSHBUTTON SWITCHES

With the RUN/STEP switch in the STEP position, the CPU will complete one cycle each time the STEP pushbutton is depressed. When the RUN/STEP switch is in the RUN mode, and the STEP pushbutton is depressed once, the CPU begins to cycle continuously.

3.4 ADDRESS SELECT SWITCH

The memory address indicators (A_{15}-A_0) are capable of displaying either the Data Memory or the Control Memory address. The type displayed is determined by the address select switch, labeled IC/PC, where PC is Data Memory and IC the Control Memory.

3.5 ADDRESS BIT AND COMPARE SWITCHES

The address bit switches located directly beneath indicators A_{15}-A_0 operate in conjunction with the COMPARE switch. Once the type of address has been selected (Ref. Sec. 3.4) and any desired memory address set by placing the corresponding address bit switches in the up position, the COMPARE switch may be turned on (up for compare) allowing the lightboard to stop the microprogram at the designated address when it is reached. The microprogram may be continued by depressing the step pushbutton and it will only STOP if the designated memory address is accessed again. The compare feature allows for internal observation at a certain point in the microprogram without having to step through the preceding instructions of the program.

3.6 WRITE SWITCH

Not applicable to 2200VP at this time.

MODULE REPAIR GUIDE NO. 4.1

EDITED BY CUSTOMER ENGINEERING DIVISION

6789 MEMORY CONTROL REPAIR February 21, 1978

1. INTRODUCTION

This MRG covers the repair procedure for the 210-6789 Memory Control PCB in the 2200VP. Section 2 introduces the use of DEBUG PROMs, section 3 contains the complete debug microprogram with explanation while section 4 is an abbreviated form of the program which is very helpful in troubleshooting.

NOTE:

It is not necessary to install the 210-6788 Control Memory board in the tester when using the PROM microcode for testing. The 210-6788 Control Memory need only be inserted when loading BASIC.

2. GENERAL

A set of DEBUG PROMs is now available which can replace the BOOTSTRAP PROMs on the 210-6789 PCB. These PROMs, however, are not capable of allowing the 2200VP to operate normally. Their primary purpose is to exercise the circuitry of the 6789 memory control board and, if a failure is present, to provide a microprogram that is simple and easy to follow and to identify a hardware problem with the aid of a lightboard. Reference MRG #4 for installation and operational aspects of the lightboard.

The locations for the DEBUG PROMs are as follows:

PCB	Location	PROM #
6789	L27	378-2156
6789	L28	378-2157
6789	L29	378-2158

Printed in U.S.A.

WANG LABORATORIES, INC.

ONE INDUSTRIAL AVENUE, LOWELL, MASSACHUSETTS 01851, TEL. (617) 851-4111, TWX 710 343-6769, TELEX 94-7421

The program will start to run at POWER ON and will stop if an error is detected. To restart the program, key RESET.

3. 210-6789 DEBUG MICROPROGRAM

The program has two major functions. The first concerns the control memory parity error generation circuit (the entire left half of the schematic). At locations 8004 and 8005 of the program, a parity error is purposely generated to insure that error detection is possible. Other than at the two test locations, the circuitry and program operate as they normally would, that is, decoding the ROM bits and setting a hardware trap (address 8000) if parity is incorrect.

There are four valid combinations of ODD 1, ODD 2 and ODD 3. If one and only one is high, or if all three are high, no parity error is generated. Test locations 8004 and 8005 generate two different invalid combinations of ODD 1-3. If a parity error is not detected at these locations, the program will hang up by executing a continuous branch to the incorrect address. If an error is generated at 8004 and 8005, the microprogram will continue on with no error being displayed. If at any address other than the test locations parity is incorrect, a message will be displayed including the address at which the error occurred. EXAMPLE: *** PECM 8103 ***

In order to find the malfunction, it may be helpful to isolate the ROM bit latches (L36-L40) from the rest of the circuitry. This can be done by manually setting the ROM bit switches to perform the instruction that generated the bad parity. The switches are connected to the outputs of the ROM bit latches, therefore, L36-40 will not be involved. If the problem cannot be found in this manner, the program can be rerun and stopped at the address that generated the error and the circuitry traced (Ref. COMPARE function, Section 3.5).

The program's second function is to check the CH, CL register (L21-L24) along with the data memory parity error generation circuitry. To accomplish this, HEX (5A) is written into the first ten data memory locations and then read back into CH, CL. If there is a parity error, a hardware trap (address 8002) will be executed and the address of the bad location displayed. EXAMPLE: *** PEDM 0002 ***. If no parity error is detected, the data that is in CH, CL is checked for integrity. If the data is incorrect, an error message will be displayed specifying whether CH or CL is bad and what the data was. EXAMPLE: *** CL ERROR: 5A WAS 5B ***. Upon completion of HEX (5A) being read correctly, the data is complimented (A5) and the test is rerun.

After CH, CL and PEDM have been checked, HEX (5A) is written into Control Memory at location 0001, and then read back and tested for data integrity. If the data is incorrect, a PECM will be detected at 0001 or an error message will be displayed (*** WCM OR RCM FAILURE ***) stating that there was a failure when either writing to or reading from Control Memory. The purpose of this test is to check LOP (Long Operation). The loop count is incremented after LOP has been successfully tested.

If a PEDM was detected, L41 pin 3 can be grounded, overriding the hardware trap, thus allowing the data integrity check to be performed. This may point out a failing bit in the CH, CL register. If the test runs successfully with PEDM grounded, the failure is in the error generation circuitry itself.

4. ABBREVIATED DEBUG MICROPROGRAM

The following is a sectionalized breakdown of the debug program. Each section is listed in the order that it is performed. The list is only composed of portions of the program that are accessed if the unit is operating correctly (i.e., error routines are not included). The addresses listed are also in the order that they are accessed.

To locate the step in the program that is failing, the COMPARE function should be untilized. An address somewhere in the middle of the program should be selected as the stopping point. If the program does indeed reach and stop at the selected address, it can be assumed that everything is operating properly up to that point in the program. If the address is not reached, simply back track through the program using the addresses listed in the abbreviated program as stopping points until the last accessed step is found. Once the section in which the failure occurs has been found, the complete listing of the microprogram can be referenced to help determine the exact failure location.

1. Power on hardware trap (8003)
   Reset hardware trap (8001)
2. Clear screen (8059-805B, 804C-8052, 805C-8062)
3. Generate bad parity and test (8004, 8000, 8014-8019)
4. Generate bad parity and test (8005, 8000, 8014-801C)
5. Initialize loop count (8063-8066)
6. Clear CH/CL parity bits (8053-8058, 8067)
7. Display "WRITE 5A" (8068-8078)
8. Write data (8079-807F)
9. Display "READ 5A" (8080-808E)
10. Read data (808F-8095)
11. Write data (8096-809A)
12. Display "WRITE A5" (809B-80AD)
13. Display "READ A5" (80AE-80BB)
14. Read data (80BC-80C1)
15. Test write and read Control Memory (80C2-80CD)
16. Increment loop count and display (80CE-80D8)
17. Repeat sequence #7-#16

```
                 8000    1     1              ORG     8000
8000 5C1480      8014    2     2              B       PE24          PECM HARDWARE TRAP
8001 5C5980      8059    3     3              B       MASTINIT      RESET HARDWARE TRAP
8002 DC4A80      804A    4     4              B       PE8           PEDM HARDWARE TRAP
8003 5C5980      8059    5     5              B       MASTINIT      POWERON HARDWARE TRAP
8004 DC0480      8004    6     6    BDPARY    B       *             GENERATE INVALID ODD COMBINATION
8005 5C0580      8005    7     7    BDPARY1   B       *             GENERATE INVALID ODD COMBINATION
8006 208E0F              8     8    OBSSP     MVI     20,K          K=SPACE
8007 E8078D      8007    9     9    OBSTROBE  BFL     8,SH,*        WAIT UNTIL DEVICE IS READY
8008 978200              10    10             CIO     20
8009 5C4E80      804E    11    11             B       DELAY5
800A C80C21      800C    12    12   PRTBYT1   BLER    F2,F1,*+2
800B AC0272              13    13             AI      07,F2,F2
800C 200E02              14    14             MV      F2,K
800D DC0780      8007    15    15             B       OBSTROBE      OUTPUT HEX CHARACTER
800E A0C19F              16    16   PRTBYTE   MVI     39,F1
800F 20C30F              17    17             MVI     30,F3
8010 8C4203              18    18             SHHH    F0,F3,F2      MOVE 3,HIGH OF F0 TO F2
8011 D40A80      800A    19    19             SB      PRTBYT1
8012 084203              20    20             SHHL    F0,F3,F2      MOVE 3,LOW OF F0 TO F2
8013 5C0A80      800A    21    21             B       PRTBYT1
8014 8D800F              22    22   PE24      TSP     ,             FAILING IC TO PC
8015 800FFF              23    23             OR      -,,           DECREMENT FAILING IC'S
8016 0200E8              24    24             MVX     PHPL,F1F0     MOVE TO F1,F0 FOR COMPARE
8017 1D0004      8004    25    25             LPI     BDPARY        LOAD PC'S WITH COMPARE ADDRESS
8018 D81D19      801D    26    26             BNR     F1,PH,*+5     BRANCH () TEST LOCATION
8019 500508      8005    27    27             BER     F0,PL,BDPARY1 BRANCH TO NEXT TEST LOCATION
801A 9D0005      8005    28    28             LPI     BDPARY1       LOAD PCS WITH COMPARE ADDR
801B D81D19      801D    29    29             BNR     F1,PH,*+2     BRANCH () TEST LOCATION
801C 506308      8063    30    30             BER     F0,PL,DLOOP
801D 0208E0              31    31             MVX     F1F0,PHPL     RESTORE FAILING IC LOCATION
801E 21013F              32    32             MVI     C'C',F1       MESSAGE PECM
801F 542080      8020    33    33             SB      SYSPE         SYSTEM PARITY ERROR
8020 21430F              34    34   SYSPE     MVI     C'P',F3
8021 21025F              35    35             MVI     C'E',F2
8022 2100DF              36    36             MVI     C'M',F0
8023 544C80      804C    37    37             SB      ENABLE5
8024 200E1F              38    38             MVI     01,K
8025 540780      8007    39    39             SB      OBSTROBE      HOME CURSOR
8026 208E0F              40    40             MVI     20,K          CHAR=SPACE
8027 A0040F              41    41             MVI     00,F4         ZERO COUNT
8028 540780      8007    42    42   SPACEOUT  SB      OBSTROBE      SPACE
```

```
8029 98C44F          43   43            AC,,1     F4,,F4              BUMP COUNT
802A FC2854  8028    44   44            BNEH      5,F4,SPACEOUT
802B 200E1F          45   45            MVI       01,K                CHAR=HOME
802C 540780  8007    46   46            SB        OBSTROBE            HOME
802D 208EAF          47   47            MVI       C'*',K
802E 540780  8007    48   48            SB        OBSTROBE            *
802F 540780  8007    49   49            SB        OBSTROBE            *
8030 540780  8007    50   50            SB        OBSTROBE            *
8031 D40680  8006    51   51            SB        OBSSP               SPACE
8032 A00E03          52   52            MV        F3,K
8033 540780  8007    53   53            SB        OBSTROBE            P
8034 200E02          54   54            MV        F2,K
8035 540780  8007    55   55            SB        OBSTROBE            E
8036 200E01          56   56            MV        F1,K
8037 540780  8007    57   57            SB        OBSTROBE            EITHER C OR D
8038 A00E00          58   58            MV        F0,K
8039 540780  8007    59   59            SB        OBSTROBE            M
803A D40680  8006    60   60            SB        OBSSP               SPACE
803B 200009          61   61            MV        PH,F0
803C 540E80  800E    62   62            SB        PRTBYTE             DISPLAY CONTENTS
803D A00008          63   63            MV        PL,F0               OF PC'S
803E 540E80  800E    64   64            SB        PRTBYTE
803F D40680  8006    65   65            SB        OBSSP               SPACE
8040 208EAF          66   66            MVI       C'*',K
8041 540780  8007    67   67            SB        OBSTROBE            *
8042 540780  8007    68   68            SB        OBSTROBE            *
8043 540780  8007    69   69            SB        OBSTROBE            *
8044 544680  8046    70   70            SB        CR/LF               CARRIAGE(RETURN)LINEFEED
8045 DC4580  8045    71   71            B         *                   HANG
8046 200EDF          72   72  CR/LF     MVI       0D,K
8047 540780  8007    73   73            SB        OBSTROBE
8048 A00EAF          74   74            MVI       0A,K
8049 DC0780  8007    75   75            B         OBSTROBE
804A A1014F          76   76  PE8       MVI       C'D',F1             D
804B 542080  8020    77   77            SB        SYSPE
804C A00E5F          78   78  ENABLE5   MVI       5,K                 K=005
804D 178C00          79   79            CIO       0C0                 ADDRESS BUS STROBE
804E 544F80  804F    80   80  DELAY5    SB        *+1
804F D45080  8050    81   81            SB        *+1
8050 200F0F          82   82            NOP       ,
8051 200F0F          83   83            NOP       ,
8052 87800F          84   84            SR        ,
```

```
8053 AA4DFD          85   85  CLEARPB  ANDI      9F,SH,SH          SET HALT/STEP OFF
8054 992000  0000    86   86           LPI,W1    0000
8055 193000  0000    87   87           LPI,W2    0000
8056 991000  0000    88   88           LPI,R     0000
8057 AA4DFD          89   89           ANDI      9F,SH,SH
8058 87800F          90   90           SR        ,
8059 220DBD          91   91  MASTINIT ORI       8B,SH,SH          INHIBIT INPUTS,TRAPS
805A AA4DFD          92   92           ANDI      09F,SH,SH
805B 544C80  804C    93   93           SB        ENABLE5           ENABLE CRT
805C A00E5F          94   94           MVI       05,K
805D 178C00          95   95           CIO       0C0               SEND ABS
805E E85E8D  805E    96   96           BFL       8,SH,*            WAIT FOR READY
805F A00E3F          97   97           MVI       03,K
8060 978200          98   98           CIO       20                CLEAR SCREEN
8061 E8618D  8061    99   99           BFL       8,SH,*            WAIT FOR READY
8062 DC0480  8004   100  100           B         BDPARY            BRANCH TO TEST PECM
8063 190000  0000   101  101  DLOOP    LPI       0000              CLEAR PCS
8064 81803F         102  102           TPA       ,03               INITIALIZE LOOP COUNT
8065 2B8CFC         103  103           ANDI      0EF,SL,SL         SET FLAG TO CRT
8066 D45380  8053   104  104           SB        CLEARPB           CLEAR CH/CL PARITY BITS
8067 29CDFD         105  105           ANDI      7F,SH,SH          SET NO TRAP OFF
8068 214E7F         106  106  WRITE5A  MVI       C'W',K
8069 540780  8007   107  107           SB        OBSTROBE          W
806A 214E2F         108  108           MVI       C'R',K
806B 540780  8007   109  109           SB        OBSTROBE          WR
806C 210E9F         110  110           MVI       C'I',K
806D 540780  8007   111  111           SB        OBSTROBE          WRI
806E 214E4F         112  112           MVI       C'T',K
806F 540780  8007   113  113           SB        OBSTROBE          WRIT
8070 210E5F         114  114           MVI       C'E',K
8071 540780  8007   115  115           SB        OBSTROBE          WRITE
8072 208E0F         116  116           MVI       20,K
8073 540780  8007   117  117           SB        OBSTROBE          SPACE
8074 A0CE5F         118  118           MVI       35,K
8075 540780  8007   119  119           SB        OBSTROBE          WRITE 5
8076 A10E1F         120  120           MVI       C'A',K
8077 540780  8007   121  121           SB        OBSTROBE          WRITE 5A
8078 544680  8046   122  122           SB        CR/LF
8079 190000  0000   123  123           LPI       0000              CLEAR PCS
807A A0010F         124  124           MVI       00,F1
807B 2000AF         125  125           MVI       0A,F0             SET COUNT
807C A142AF         126  126           MVI       5A,F2             SET DATA
```

```
807D 002FE2        127   127  FLOOD    OR,W1     +,F2,              WRITE DATA
807E 002FE2        128   128           OR,W1     +,F2,              WRITE DATA
807F D87D08  807D  129   129           BNR       F0,PL,FLOOD        BRANCH () END
8080 190000  0000  130   130           LPI       0000               SET UP READ FOR 0000
8081 214E2F        131   131           MVI       C'R',K
8082 540780  8007  132   132           SB        OBSTROBE           R
8083 210E5F        133   133           MVI       C'E',K
8084 540780  8007  134   134           SB        OBSTROBE           RE
8085 A10E1F        135   135           MVI       C'A',K
8086 540780  8007  136   136           SB        OBSTROBE           REA
8087 A10E4F        137   137           MVI       C'D',K
8088 540780  8007  138   138           SB        OBSTROBE           READ
8089 208E0F        139   139           MVI       20,K
808A 540780  8007  140   140           SB        OBSTROBE           SPACE
808B A0CE5F        141   141           MVI       35,K
808C 540780  8007  142   142           SB        OBSTROBE           READ 5
808D A10E1F        143   143           MVI       C'A',K
808E 540780  8007  144   144           SB        OBSTROBE           READ 5A
808F A01F0F        145   145  READ1    ORI,R     0,,                5A READ INTO CH/CL
8090 01840F        146   146           TPA+2     ,00
8091 8B800F        147   147           TAP       ,00
8092 58E22B  80E2  148   148           BNR       F2,CH,ERRORCH      BR IF ERROR
8093 D8DE2A  80DE  149   149           BNR       F2,CL,ERRORCL      BR IF ERROR
8094 588F08  808F  150   150           BNR       F0,PL,READ1        BR () END
8095 D45380  8053  151   151           SB        CLEARPB            CLEAR CH/CL PARITY BITS
8096 190000  0000  152   152           LPI       0000               CLEAR PCS
8097 A2825F        153   153           MVI       0A5,F2             SET DATA
8098 002FE2        154   154  FLOOD2   OR,W1     +,F2,              WRITE DATA
8099 002FE2        155   155           OR,W1     +,F2,              WRITE DATA
809A 589808  8098  156   156           BNR       F0,PL,FLOOD2       BR () END
809B 190000  0000  157   157           LPI       0000               CLEAR PCS
809C 544680  8046  158   158           SB        CR/LF
809D 214E7F        159   159           MVI       C'W',K
809E 540780  8007  160   160           SB        OBSTROBE           W
809F 214E2F        161   161           MVI       C'R',K
80A0 540780  8007  162   162           SB        OBSTROBE           WR
80A1 210E9F        163   163           MVI       C'I',K
80A2 540780  8007  164   164           SB        OBSTROBE           WRI
80A3 214E4F        165   165           MVI       C'T',K
80A4 540780  8007  166   166           SB        OBSTROBE           WRIT
80A5 210E5F        167   167           MVI       C'E',K
80A6 540780  8007  168   168           SB        OBSTROBE           WRITE
```

```
80A7 208E0F          169  169          MVI     20,K
80A8 540780  8007    170  170          SB      OBSTROBE        SPACE
80A9 A10E1F          171  171          MVI     C'A',K
80AA 540780  8007    172  172          SB      OBSTROBE        WRITE A
80AB A0CE5F          173  173          MVI     35,K
80AC 540780  8007    174  174          SB      OBSTROBE        WRITE A5
80AD 544680  8046    175  175          SB      CR/LF
80AE 214E2F          176  176          MVI     C'R',K
80AF 540780  8007    177  177          SB      OBSTROBE        R
80B0 210E5F          178  178          MVI     C'E',K
80B1 540780  8007    179  179          SB      OBSTROBE        RE
80B2 A10E1F          180  180          MVI     C'A',K
80B3 540780  8007    181  181          SB      OBSTROBE        REA
80B4 A10E4F          182  182          MVI     C'D',K
80B5 540780  8007    183  183          SB      OBSTROBE        READ
80B6 208E0F          184  184          MVI     20,K
80B7 540780  8007    185  185          SB      OBSTROBE        SPACE
80B8 A10E1F          186  186          MVI     C'A',K
80B9 540780  8007    187  187          SB      OBSTROBE        READ A
80BA A0CE5F          188  188          MVI     35,K
80BB 540780  8007    189  189          SB      OBSTROBE        READ A5
80BC A01F0F          190  190  READ2   ORI,R   0,,             A5 READ INTO CH/CL
80BD 01840F          191  191          TPA+2   ,00
80BE 8B800F          192  192          TAP     ,00
80BF 58E22B  80E2    193  193          BNR     F2,CH,ERRORCH   BR IF ERROR
80C0 D8DE2A  80DE    194  194          BNR     F2,CL,ERRORCL   BR IF ERROR
80C1 58BC08  80BC    195  195          BNR     F0,PL,READ2     BR () END
80C2 2140AF          196  196          MVI     5A,F0           SET COMPARE DATA
80C3 A2815F          197  197          MVI     0A5,F1          SET COMPARE DATA
80C4 1B4A5A  5A5A    198  198          LPI     5A5A            SET WRITE DATA
80C5 A14EAF          199  199          MVI     5A,K            SET WRITE DATA
80C6 81800F          200  200          TPA     ,00             STORE DATA
80C7 990001  0001    201  201          LPI     0001            SET WRITE ADDRESS
80C8 54D980  80D9    202  202          SB      WCM             BRANCH TO WRITE TO CONTROL MEMORY
80C9 990001  0001    203  203          LPI     0001            SET READ ADDRESS
80CA 54DC80  80DC    204  204          SB      RCM             BRANCH TO READ CONTROL MEMORY
80CB 59191E  8119    205  205          BNR     F1,K,CMERROR    CHECK DATA
80CC 591909  8119    206  206          BNR     F0,PH,CMERROR   CHECK DATA
80CD D91908  8119    207  207          BNR     F0,PL,CMERROR   CHECK DATA
80CE 190000  0000    208  208          LPI     0000            CLEAR PCS
80CF 8B803F          209  209          TAP     ,03             RETRIEVE LOOP COUNT
80D0 9AC8E8          210  210          ACX,,1  DD,PHPL,PHPL    INCREMENT LOOP COUNT
```

```
80D1 81803F        211  211          TPA       ,03          STORE LOOP COUNT
80D2 2B8CFC        212  212          ANDI      0EF,SL,SL    SET FLAG TO CRT
80D3 D51080  8110  213  213          SB        DISPLOOP     DISPLAY LOOP COUNT
80D4 200EDF        214  214          MVI       0D,K
80D5 540780  8007  215  215          SB        OBSTROBE     CARRIAGE/RETURN
80D6 200E1F        216  216          MVI       01,K
80D7 540780  8007  217  217          SB        OBSTROBE
80D8 DC6880  8068  218  218          B         WRITE5A      BRANCH TO START NEXT LOOP
80D9 05800F        219  219  WCM     TPS       ,            STORE WRITE ADDRESS IN STACK
80DA 8B800F        220  220          TAP       ,00          RECALL WRITE PATTERN
80DB 078400        221  221          SR,WCM    ,
80DC 05800F        222  222  RCM     TPS       ,            STORE READ ADDRESS IN STACK
80DD 878600        223  223          SR,RCM    ,
80DE 81822F        224  224  ERRORCL  TPA+1     ,02          INCREMENT PCS TO CL
80DF A0070A        225  225          MV        CL,F7        STORE FAILURE
80E0 A106CF        226  226          MVI       C'L',F6
80E1 DCE580  80E5  227  227          B         ERROR
80E2 81822F        228  228  ERRORCH  TPA+1     ,02          INCREMENT PCS TO CH
80E3 20070B        229  229          MV        CH,F7        STORE FAILURE
80E4 21068F        230  230          MVI       C'H',F6
80E5 544C80  804C  231  231  ERROR   SB        ENABLE5
80E6 A00E3F        232  232          MVI       03,K
80E7 540780  8007  233  233          SB        OBSTROBE     CLEAR SCREEN
80E8 208EAF        234  234          MVI       C'*',K
80E9 540780  8007  235  235          SB        OBSTROBE     *
80EA 540780  8007  236  236          SB        OBSTROBE     *
80EB 540780  8007  237  237          SB        OBSTROBE     *
80EC D40680  8006  238  238          SB        OBSSP        SPACE
80ED 210E3F        239  239          MVI       43,K
80EE 540780  8007  240  240          SB        OBSTROBE     C
80EF A00E06        241  241          MV        F6,K
80F0 540780  8007  242  242          SB        OBSTROBE     L OR H
80F1 D40680  8006  243  243          SB        OBSSP        SPACE
80F2 210E5F        244  244          MVI       45,K
80F3 540780  8007  245  245          SB        OBSTROBE     E
80F4 214E2F        246  246          MVI       52,K
80F5 540780  8007  247  247          SB        OBSTROBE     R
80F6 540780  8007  248  248          SB        OBSTROBE     R
80F7 210EFF        249  249          MVI       4F,K
80F8 540780  8007  250  250          SB        OBSTROBE     O
80F9 214E2F        251  251          MVI       52,K
80FA 540780  8007  252  252          SB        OBSTROBE     R
```

```
80FB A0CEAF          253   253           MVI     3A,K
80FC 540780  8007    254   254           SB      OBSTROBE    :
80FD D40680  8006    255   255           SB      OBSSP       SPACE
80FE A00002          256   256           MV      F2,F0
80FF 540E80  800E    257   257           SB      PRTBYTE     DATA WRITTEN
8100 D40680  8006    258   258           SB      OBSSP       SPACE
8101 214E7F          259   259           MVI     57,K
8102 540780  8007    260   260           SB      OBSTROBE    W
8103 A10E1F          261   261           MVI     41,K
8104 540780  8007    262   262           SB      OBSTROBE    A
8105 A14E3F          263   263           MVI     53,K
8106 540780  8007    264   264           SB      OBSTROBE    S
8107 D40680  8006    265   265           SB      OBSSP       SPACE
8108 A00007          266   266           MV      F7,F0
8109 540E80  800E    267   267           SB      PRTBYTE     DATA READ
810A D40680  8006    268   268           SB      OBSSP       SPACE
810B 208EAF          269   269           MVI     C'*',K
810C 540780  8007    270   270           SB      OBSTROBE    *
810D 540780  8007    271   271           SB      OBSTROBE    *
810E 540780  8007    272   272           SB      OBSTROBE    *
810F DD0F80  810F    273   273           B       *
8110 544C80  804C    274   274 DISPLOOP  SB      ENABLE5
8111 D40680  8006    275   275           SB      OBSSP
8112 208E3F          276   276           MVI     C'#',K
8113 540780  8007    277   277           SB      OBSTROBE
8114 D40680  8006    278   278           SB      OBSSP
8115 200009          279   279           MV      PH,F0
8116 540E80  800E    280   280           SB      PRTBYTE
8117 A00008          281   281           MV      PL,F0
8118 DC0E80  800E    282   282           B       PRTBYTE
```

```
8119 544C80   804C   1    283  CMERROR  SB     ENABLE5      ENABLE CRT
811A A00E3F          2    284           MVI    03,K
811B 540780   8007   3    285           SB     OBSTROBE     CLEAR SCREEN
811C 208EAF          4    286           MVI    C'*',K
811D 540780   8007   5    287           SB     OBSTROBE     *
811E 540780   8007   6    288           SB     OBSTROBE     *
811F 540780   8007   7    289           SB     OBSTROBE     *
8120 D40680   8006   8    290           SB     OBSSP        SPACE
8121 214E7F          9    291           MVI    57,K
8122 540780   8007   10   292           SB     OBSTROBE     W
8123 210E3F          11   293           MVI    43,K
8124 540780   8007   12   294           SB     OBSTROBE     C
8125 A10EDF          13   295           MVI    4D,K
8126 540780   8007   14   296           SB     OBSTROBE     M
8127 D40680   8006   15   297           SB     OBSSP        SPACE
8128 210EFF          16   298           MVI    4F,K
8129 540780   8007   17   299           SB     OBSTROBE     O
812A 214E2F          18   300           MVI    52,K
812B 540780   8007   19   301           SB     OBSTROBE     R
812C D40680   8006   20   302           SB     OBSSP        SPACE
812D 214E2F          21   303           MVI    52,K
812E 540780   8007   22   304           SB     OBSTROBE     R
812F 210E3F          23   305           MVI    43,K
8130 540780   8007   24   306           SB     OBSTROBE     C
8131 A10EDF          25   307           MVI    4D,K
8132 540780   8007   26   308           SB     OBSTROBE     M
8133 D40680   8006   27   309           SB     OBSSP        SPACE
8134 210E6F          28   310           MVI    46,K
8135 540780   8007   29   311           SB     OBSTROBE     F
8136 A10E1F          30   312           MVI    41,K
8137 540780   8007   31   313           SB     OBSTROBE     A
8138 210E9F          32   314           MVI    49,K
8139 540780   8007   33   315           SB     OBSTROBE     I
813A 210ECF          34   316           MVI    4C,K
813B 540780   8007   35   317           SB     OBSTROBE     L
813C A14E5F          36   318           MVI    55,K
813D 540780   8007   37   319           SB     OBSTROBE     U
813E 214E2F          38   320           MVI    52,K
813F 540780   8007   39   321           SB     OBSTROBE     R
8140 210E5F          40   322           MVI    45,K
8141 540780   8007   41   323           SB     OBSTROBE     E
8142 D40680   8006   42   324           SB     OBSSP        SPACE
```

```
8143 208EAF          43    325          MVI       C'*',K
8144 540780  8007    44    326          SB        OBSTROBE          *
8145 540780  8007    45    327          SB        OBSTROBE          *
8146 540780  8007    46    328          SB        OBSTROBE          *
8147 DD4780  8147    47    329          B         *
```

NUMBER OF LINES IN ERROR = 0 NO. OF WARNINGS = 0 NO. OF SYMBOLS = 28 OVERFLOWS = 24

SYMBOL	VALUE	DEFN	REFERENCES
BDPARY	8004	0006	0025 0100
BDPARY1	8005	0007	0027 0028
CLEARPB	8053	0085	0104 0151
CMERROR	8119	0283	0205 0206 0207
CR/LF	8046	0072	0070 0122 0158 0175
DELAY5	804E	0080	0011
DISPLOOP	8110	0274	0213
DLOOP	8063	0101	0030
ENABLE5	804C	0078	0037 0093 0231 0274 0283
ERROR	80E5	0231	0227
ERRORCH	80E2	0228	0148 0193
ERRORCL	80DE	0224	0149 0194
FLOOD	807D	0127	0129
FLOOD2	8098	0154	0156
MASTINIT	8059	0091	0003 0005
OBSSP	8006	0008	0051 0060 0065 0238 0243 0255 0258 0265 0268 0275 0278 0290 0297 0302 0309 0324
OBSTROBE	8007	0009	0015 0039 0042 0046 0048 0049 0050 0053 0055 0057 0059 0067 0068 0069 0073 0075
			0113 0115 0117 0119 0121 0132 0134 0136 0138 0140 0142 0144 0160 0162 0164 0166
			0174 0177 0179 0181 0183 0185 0187 0189 0215 0217 0233 0235 0236 0237 0240 0242
			0250 0252 0254 0260 0262 0264 0270 0271 0272 0277 0285 0287 0288 0289 0292 0294
			0304 0306 0308 0311 0313 0315 0317 0319 0321 0323 0326 0327 0328
PE24	8014	0022	0002
PE8	804A	0076	0004
PRTBYT1	800A	0012	0019 0021
PRTBYTE	800E	0016	0062 0064 0257 0267 0280 0282
RCM	80DC	0222	0204
READ1	808F	0145	0150
READ2	80BC	0190	0195
SPACEOUT	8028	0042	0044
SYSPE	8020	0034	0033 0077
WCM	80D9	0219	0202
WRITE5A	8068	0106	0218

MODULE REPAIR GUIDE NO.4.2

EDITED BY CUSTOMER ENGINEERING DIVISION

March 29, 1978

210-6793 TEST PROGRAM

1. INTRODUCTION

210-6793 REGISTERS AND I/O TEST PROGRAM

The 210-6793 registers and I/O test program was written to be used as a helpful aid in the troubleshooting and repair of the 6793 module.

The program utilizes "building-block techniques" of testing. That is, test the module in various steps starting with very basic testing of registers, source selectors, etc., and advance onward utilizing what is "known good" to then test other parts of the module in more detail.

Visual display is implemented wherever possible.

The test program is designed to be run exclusively out of PROM for two basic reasons:

1) To eliminate the complexity of having to use the boot-strap/utility PROMs for initial module troubleshooting.
2) Possible module failures could prevent the loading of the test program from disk or other media.

2. INSTALLATION/OPERATION

Please refer to the "Module Repair Guide #4" for information needed to install and operate equipment associated with 2200VP system repair. The only additional equipment would be the following set of debug PROMs associated with the 6793 test program.

Printed in U.S.A.

WANG LABORATORIES, INC.

ONE INDUSTRIAL AVENUE, LOWELL, MASSACHUSETTS 01851, TEL. (617) 851-4111, TWX 710 343-6769, TELEX 94-7421

	PROM	BOARD	LOCATION
#1	378-2187	6789	L27
#2	378-2188	6789	L28
#3	378-2189	6789	L29

3. GENERAL TEST INFORMATION

POWER ON

As power is applied to the system, a micro trap to PROM address 8003 occurs which begins test operation.

START

Inhibit inputs and clear CRT display.

INITIAL TEST SEQUENCE

Initially a "K" is written into the K register and a check is made to insure that it did in fact get written properly. If the K register = "K" (Hex 4B), the letter "K" is displayed on the CRT. This technique is used for all file registers as well. At the successful completion of this test, the following should be displayed on the CRT:

K 0 1 2 3 4 5 6 7

If an error is detected during execution of this routine, the micro program will execute a branch to the beginning of the subtest that failed, resulting in a continuous looping operation. This is done in an effort to aid in troubleshooting.

NOTE:

If a register fails in this test phase, no print out on CRT will occur for that particular register and any registers that follow.

EXAMPLE: File register #2 fails

Print Out = K 0 1

No "2" is printed because of a failure in register #2. No other registers will be tested until the register #2 failure is gone. At this point the program should be looping in a short routine that puts a "2" in register #2 and does a conditional branch test to check that the "2" was written properly.

INPUT SEQUENCE

After K 0 1 2 3 4 5 6 7 is printed on CRT, display stops and the micro program waits for keyboard input. This test is a visual test of input to the K register and associated logic from the keyboard. For every key struck, a character is printed on the CRT (echo mode). It is up to the operator to verify that the key struck is the character printed. By typing the alphabet and numerics 0-9, most possible bit combinations will be displayed. This test is terminated by striking the "RETURN (EXEC)" key.

TEST0-TEST9

After RETURN (EXEC) key is struck, a series of ten tests are run continuously with visual display on CRT. The sequence of events is as follows:

1) Print Test# on CRT.
2) Execute test.
3) If failure results, print error. Go to beginning of test and repeat operation.
4) If no error results, go to next test.
5) After completion of test 9, restart at test 0.

TEST#0 DESCRIPTION

Test 0 is a two-part test. Part 1 tests that the K register and all file registers can be cleared (=0). Part 2 tests the same registers by setting them equal to all ones (FF HEX). If a failure

results, the program will print "ERROR" on the CRT and restart at the beginning of the failing subtest. Prior to part 1, the K register was cleared and checked in a small routine labeled "START1". In this subtest if K register failed, the program will loop attempting to reclear K.

TEST#1 DESCRIPTION

Test 1 is a test of the K register. The register is first cleared and then incremented one step at a time (00-FF HEX) continuously checking that the operation was successful. The K register was previously tested to ensure it could be cleared (all 0) as well as set to all ones (FF). The compare operation uses file register 1 to check the K register; the file register was also tested prior to this test for bits stuck hi or low (set clear). If either K register or file #1 fail, this test will print out "ERROR" and restart.

TEST#2 DESCRIPTION

This is a test of all file registers ability to be incremented from 00-FF (HEX). The K register, which is now known to be good (able to be incremented from 00-FF), is used as the comparison register. The test operates in a chain mode fashion, that is, the K register is incremented and moved into F0. Then F0 is moved to F1, F1 to F2, etc. Finally, file register 7 is compared to the K register to see if all registers were written properly. If a failure results, the program will restart.

TEST#3 DESCRIPTION

This test clears the SH register and the SL register and checks that both registers are, in fact, clear.

NOTE:

The SH register ready bit (SH=08 HEX) is masked out because it is randomly set by I/O devices.

TEST#4 DESCRIPTION

This is a test of the SL register's ability to be stepped from 00 to FF (HEX) one step at a time comparing it to file register 2 which was previously tested. Any error will restart this test.

TEST#5 DESCRIPTION

This is a test of the SH register's ability to be stepped from 00 to FF (HEX) one step at a time, comparing it to file register 0 which was previously tested. A fairly complex mask compare operation is performed due to the ready bit problem. Any error will restart this test.

TEST#6 DESCRIPTION

This test is very similar to Test#2 with the exception that it incorporates the SL register in the chain mode testing.

TEST#7 DESCRIPTION

This test fills file registers 0-7 with several worst case patterns and checks to see that they were written properly.

F0=5A	F4=AA
F1=A5	F5=A5
F2=5A	F6=5A
F3=55	F7=A5

These patterns are generated by multiple shift operations and moved into the appropriate registers.

TEST#8 DESCRIPTION

This test moves four worst case patterns through all eight file registers and checks to see that only one register was altered at a time. The following is an example using F0:

F0 initial value=5A. In the first move F0 goes to an A5, the second move F0 goes to 55, the third move F0 goes to AA and finally the fourth move puts the 5A back in F0. At this time, a check is performed to see if any other registers were altered.

TEST#9 DESCRIPTION

This test writes one location in control memory and reads the same location, checking to see that it was written properly. The purpose of this test is to exercise some previously untested inputs to the registers through source selectors (on the read control memory instruction, registers K, PH, PL are used as input registers and during the write control memory instruction they are used as output registers).

PARITY ERRORS

The program has the ability to detect and display both control memory and data memory parity errors on the CRT. The address that is displayed on the CRT, when either error occurs, is decremented by 1. Thus if a parity error occurs, the print out address is the last valid instruction. The next instruction is the instruction that caused the error.

After the error is printed on the CRT, the program hangs in a branch-on-self routine, waiting for reset to be struck on keyboard.

RESET

If at any time during execution the program gets hung, the reset key may be struck which will restart the program at the very beginning.

```
                8000   1    1          ORG     8000          FORCE PROGRAM START FIRST PROM ADDRESS
 8000 5DE680    81E6   2    2          B       PE24          CONTROL MEMORY PARITY ERROR TRAPS HERE
 8001 DC1080    8010   3    3          B       START         RESET TRAPS HERE
 8002 DDEB80    81EB   4    4          B       PE8           DATA MEMORY PARITY ERROR TRAPS HERE
 8003 DC1080    8010   5    5          B       START         POWER ON TRAPS HERE
 8004 DC0480    8004   6    6          B       *             BRANCH SELF SHOULD NEVER HAVE GOTTEN HERE
 8005 5C0580    8005   7    7          B       *
 8006 5C0680    8006   8    8          B       *
 8007 DC0780    8007   9    9          B       *
 8008 DC0880    8008   10   10         B       *
 8009 5C0980    8009   11   11         B       *
 800A 5C0A80    800A   12   12         B       *
 800B DC0B80    800B   13   13         B       *
 800C 5C0C80    800C   14   14         B       *
 800D DC0D80    800D   15   15         B       *
 800E DC1080    8010   16   16         B       START
 800F 800000           17   17         INSTR   800000
                       18   18    *
                       19   19    *
                       20   20    *
 8010 D56480    8164   21   21    START SB      INIT          CLEAR SCREEN DISPLAY INPUTS
 8011 A10EBF           22   22          MVI     4B,K          PUT HEX "K" IN REG. K
 8012 F811BE    8011   23   23          BNEL    0B,K,*-1      BRANCH BACK 1 IF LOW HALF K NOT = 0B
 8013 7C114E    8011   24   24          BNEH    04,K,*-2      BRANCH BACK 2 IF HIGH HALF K NOT = 04
 8014 556080    8160   25   25          SB      OBSTROBE      PRINT "K" ON CRT
 8015 20C00F           26   26          MVI     30,F0         PUT HEX "0" IN F0
 8016 781500    8015   27   27          BNEL    00,F0,*-1     CHECK F0 LOW =00
 8017 FC1530    8015   28   28          BNEH    03,F0,*-2     CHECK F0 HIGH =03
 8018 A00E00           29   29          MV      F0,K          PUT F0 IN K
 8019 556080    8160   30   30          SB      OBSTROBE      PRINT "0" ON CRT
 801A 20C11F           31   31          MVI     31,F1         PUT "1" IN F1
 801B 781A11    801A   32   32          BNEL    01,F1,*-1     LOW HALF F1=01?
 801C 7C1A31    801A   33   33          BNEH    03,F1,*-2     HIGH HALF F1=03
 801D 200E01           34   34          MV      F1,K          PUT F1 IN K
 801E 556080    8160   35   35          SB      OBSTROBE      PRINT "1" ON CRT
 801F 20C22F           36   36          MVI     32,F2         PUT "2" IN F2
 8020 781F22    801F   37   37          BNEL    02,F2,*-1     LOW F2=02?
 8021 7C1F32    801F   38   38          BNEH    03,F2,*-2     HIGH F2=03?
 8022 200E02           39   39          MV      F2,K          PUT F2 IN K
 8023 556080    8160   40   40          SB      OBSTROBE      PRINT "2" ON CRT
 8024 20C33F           41   41          MVI     33,F3         PUT "3" IN F3
 8025 F82433    8024   42   42          BNEL    03,F3,*-1     LOW F3=03?
 8026 7C2433    8024   43   43          BNEH    03,F3,*-2     HIGH F3=03?
 8027 A00E03           44   44          MV      F3,K          PUT F3 IN K
 8028 556080    8160   45   45          SB      OBSTROBE      PRINT "3" ON CRT
```

```
8029 20C44F       46  46          MVI     34,F4           PUT "4" IN F4
802A 782944 8029  47  47          BNEL    04,F4,*-1       LOW F4 =04?
802B 7C2934 8029  48  48          BNEH    03,F4,*-2       HIGH F4=03?
802C 200E04       49  49          MV      F4,K            PUT F4 IN K
802D 556080 8160  50  50          SB      OBSTROBE        PRINT "4" ON CRT
802E 20C55F       51  51          MVI     35,F5           PUT "5" IN F5
802F F82E55 802E  52  52          BNEL    05,F5,*-1       LOW F5=05?
8030 7C2E35 802E  53  53          BNEH    03,F5,*-2       HIGH F5=03?
8031 A00E05       54  54          MV      F5,K            PUT F5 IN K
8032 556080 8160  55  55          SB      OBSTROBE        PRINT "5" ON CRT
8033 20C66F       56  56          MVI     36,F6           PUT "6" IN F6
8034 F83366 8033  57  57          BNEL    06,F6,*-1       LOW F6=06?
8035 7C3336 8033  58  58          BNEH    03,F6,*-2       HIGH F6=03?
8036 A00E06       59  59          MV      F6,K            PUT F6 IN K
8037 556080 8160  60  60          SB      OBSTROBE        PRINT "6" ON CRT
8038 20C77F       61  61          MVI     37,F7           PUT "7" IN F7
8039 783877 8038  62  62          BNEL    07,F7,*-1       LOW F7=07?
803A 7C3837 8038  63  63          BNEH    03,F7,*-2       HIGH F7=03?
803B 200E07       64  64          MV      F7,K            PUT F7 IN K
803C 556080 8160  65  65          SB      OBSTROBE        PRINT "7" ON CRT
803D 556A80 816A  66  66          SB      CRLF            CARRIAGE RETURN LINE FEED TO CRT
803E A00D0F       67  67  INTEST   MVI     00,SH           CLEAR SH TO ENABLE INPUT
803F 200E1F       68  68          MVI     01,K            ENABLE KEYBOARD INPUT
8040 555980 8159  69  69          SB      ENABLE          SELECT KEYBOARD AS INPUT CHANNEL
8041 68412D 8041  70  70  INPUT   BFL     2,SH,*          WAIT FOR INPUTED CHARECTER
8042 7845DE 8045  71  71          BNEL    0D,K,*+3        CHECK TO SEE IF CHARECTER IS A CARRIAGE RETURN
8043 7C450E 8045  72  72          BNEH    0,K,*+2         IF IT IS THEN GO TO NEXT TEST, IF NOT NEXT KEY
8044 DC4A80 804A  73  73          B       START0          NEXT TEST
8045 A0000E       74  74          MV      K,F0            MOVE CHAR. INPUT TO F0
8046 D55880 8158  75  75          SB      ENABLE5         ENABLE OUTPUT TO CRT
8047 A00E00       76  76          MV      F0,K            PUT CHAR. BACK NOW
8048 556080 8160  77  77          SB      OBSTROBE        PRINT KEY STRUCK
8049 DC3E80 803E  78  78          B       INTEST          GET NEXT KEY
804A D56480 8164  79  79  START0  SB      INIT            INHIBIT INPUTS , DISABLE TRAPS
                  80  80  *
                  81  81  *
                  82  82  *
804B A00E0F       83  83  START1  MVI     00,K            0 KREG AND CHECK THAT IT IS CLEARED
804C 20000F       84  84          MVI     00,F0           0 FILE REG 0
804D FC4B0E 804B  85  85          BNEH    00,K,*-2        CHECK K REG = 0
804E 784B0E 804B  86  86          BNEL    00,K,*-3
804F D84B0E 804B  87  87          BNR     F0,K,*-4        CHECK F0 = 0
8050 9D0050 8050  88  88  TEST0   LPI     *               PUSH TEST ADDRESS ON STACK. TEST REGS.CAN BE CLEAR
8051 05800F       89  89          TPS     ,
8052 20C00F       90  90          MVI     30,F0
```

```
8053 D57A80  817A  91   91          SB      TEST#           PRINT ON CRT TEST # 0
8054 860000        92   92   PART1  XORX    F1F0,F1F0,F1F0  CLEAR F1 AND F0:CLEAR ALL FILE REGS
8055 060222        93   93          XORX    F3F2,F3F2,F3F2  CLEAR F3 AND F2
8056 060444        94   94          XORX    F5F4,F5F4,F5F4  CLEAR F5 AND F4
8057 860666        95   95          XORX    F7F6,F7F6,F7F6  CLEAR F7 AND F6
8058 7D8A00  818A  96   96          BNEH    00,F0,ERROR     CHECK ALL FILE REGS.=0
8059 F98A00  818A  97   97          BNEL    00,F0,ERROR
805A D98A01  818A  98   98          BNR     F0,F1,ERROR     F0=0?
805B 598A12  818A  99   99          BNR     F1,F2,ERROR
805C D98A23  818A  100  100         BNR     F2,F3,ERROR
805D D98A34  818A  101  101         BNR     F3,F4,ERROR
805E D98A45  818A  102  102         BNR     F4,F5,ERROR
805F 598A56  818A  103  103         BNR     F5,F6,ERROR
8060 D98A67  818A  104  104         BNR     F6,F7,ERROR     F7=0?
8061 200F0F        105  105         NOP     ,
8062 200F0F        106  106         NOP     ,
8063 9D0063  8063  107  107  PART2  LPI     *               TEST THAT ALL FILE REGS. AND K SET TO FF
8064 05800F        108  108         TPS     ,
8065 A3CEFF        109  109         MVI     0FF,K           SET K = ALL ONES
8066 798AFE  818A  110  110         BNEL    0F,K,ERROR      LOW K = 0F
8067 FD8AFE  818A  111  111         BNEH    0F,K,ERROR      HIGH K = 0F
8068 A0000E        112  112         MV      K,F0            FF-F0
8069 A00100        113  113         MV      F0,F1           FF-F1
806A 0202E0        114  114         MVX     F1F0,F3F2       FF-F3,F2
806B 8204E2        115  115         MVX     F3F2,F5F4       FF-F4,F5
806C 0206E4        116  116         MVX     F5F4,F7F6       FF-F6,F7
806D D98A0E  818A  117  117         BNR     F0,K,ERROR      CHECK ALL FILE REGISTERS = FF ALL ONES
806E 598A1E  818A  118  118         BNR     F1,K,ERROR
806F 598A2E  818A  119  119         BNR     F2,K,ERROR
8070 D98A3E  818A  120  120         BNR     F3,K,ERROR
8071 598A4E  818A  121  121         BNR     F4,K,ERROR
8072 D98A5E  818A  122  122         BNR     F5,K,ERROR
8073 D98A6E  818A  123  123         BNR     F6,K,ERROR
8074 598A7E  818A  124  124         BNR     F7,K,ERROR
8075 200F0F        125  125         NOP     ,
8076 200F0F        126  126         NOP     ,
8077 9D0077  8077  127  127  TEST1  LPI     *               INCREMENT K AND COMPARE TO F1
8078 A0C01F        128  128         MVI     31,F0
8079 D57A80  817A  129  129         SB      TEST#           PRINT TEST # 1
807A 200F0F        130  130         NOP     ,
807B A00E0F        131  131         MVI     00,K            CLEAR K
807C A0010F        132  132         MVI     00,F1           CLEAR F1
807D 200E01        133  133  LOOP   MV      F1,K            START OF LOOP INCREMENT F1 MOVE IT TO K AND CHECK
807E 598A1E  818A  134  134         BNR     F1,K,ERROR
807F 7881F1  8081  135  135         BNEL    0F,F1,*+2       CHECK FOR LAST PATTERN (256 TIMES?)
```

```
8080 F483F1  8083  136  136        BEQH    0F,F1,*+3
8081 AC0111        137  137        AI      01,F1,F1        INCR. TEST PATTERN
8082 5C7D80  807D  138  138        B       LOOP
8083 200F0F        139  139        NOP     ,
8084 9D0084  8084  140  140  TEST2  LPI     *               TEST ALL FILE REGS. CHAIN MODE INCREMENT 00-FF
8085 05800F        141  141        TPS     ,
8086 A0C02F        142  142        MVI     32,F0
8087 D57A80  817A  143  143        SB      TEST#           PRINT TEST#2 ON CRT
8088 A00E0F        144  144        MVI     00,K
8089 A0000E        145  145  LOOP1  MV      K,F0            INCR AND CHECK ALL FILE REGS.
808A A00100        146  146        MV      F0,F1           MOVE F0-F1-F2-F3-F4-F5-F6-F7 CHECK K=F7
808B 0202E0        147  147        MVX     F1F0,F3F2
808C 8204E2        148  148        MVX     F3F2,F5F4
808D 0206E4        149  149        MVX     F5F4,F7F6
808E 598A7E  818A  150  150        BNR     F7,K,ERROR
808F F891FE  8091  151  151        BNEL    0F,K,*+2        CHECK FOR LAST PATTERN (FF)
8090 7495FE  8095  152  152        BEQH    0F,K,*+5        IF LAST PATTERN GO TO NEXT TEST
8091 AC0E1E        153  153        AI      01,K,K
8092 200F0F        154  154        NOP     ,
8093 200F0F        155  155        NOP     ,
8094 DC8980  8089  156  156        B       LOOP1           NEXT PATTERN TEST
8095 200F0F        157  157        NOP     ,
8096 DC9780  8097  158  158        B       TEST3
                   159  159  *
                   160  160  *
                   161  161  *
                   162  162  *
8097 1D0097  8097  163  163  TEST3  LPI     *               THIS TEST CLEARS SH AND SL AND CHECKS
8098 05800F        164  164        TPS     ,
8099 20C03F        165  165        MVI     33,F0
809A D57A80  817A  166  166        SB      TEST#           PRINT TEST#3
809B 190000  0000  167  167        LPI     00
809C 020CE8        168  168        MVX     PHPL,SHSL       CLEAR ALL BUT READY BIT IN SH
809D 20000F        169  169        MVI     00,F0
809E 598A0C  818A  170  170        BNR     F0,SL,ERROR     SL=0?
809F 689F8D  809F  171  171        BFL     8,SH,*          MASK OUT READY BIT AFTER IT SETS AND TEST SH=0
80A0 F98A8D  818A  172  172        BNEL    08,SH,ERROR
80A1 FD8A0D  818A  173  173        BNEH    00,SH,ERROR
                   174  174  *
80A2 1D00A2  80A2  175  175        LPI     *
80A3 05800F        176  176        TPS     ,
80A4 A0C04F        177  177  TEST4  MVI     34,F0
80A5 D57A80  817A  178  178        SB      TEST#           PRINT TEST#4 ON CRT
80A6 A0020F        179  179        MVI     00,F2           STEP SL 00-FF AND CHECK COMPARE WITH F2
80A7 A00C02        180  180  LOOP2  MV      F2,SL
```

```
80A8 D98A2C  818A  181  181          BNR     F2,SL,ERROR
80A9 F8ABF2  80AB  182  182          BNEL    0F,F2,*+2       CHECK FOR LAST INCREMENT
80AA F4ADF2  80AD  183  183          BEQH    0F,F2,*+3
80AB AC0212        184  184          AI      01,F2,F2
80AC DCA780  80A7  185  185          B       LOOP2           NEXT PATTERN
                   186  186  *
                   187  187  *
80AD 1D00AD  80AD  188  188          LPI     *
80AE 05800F        189  189          TPS     ,
80AF 20C05F        190  190  TEST5    MVI     35,F0
80B0 D57A80  817A  191  191          SB      TEST#           PRINT TEST#5 ON CRT
80B1 A00E0F        192  192          MVI     00,K            TEST SH STEP 00-FF CHECK COMPARE TO F0
80B2 20000F        193  193          MVI     00,F0
80B3 A00D00        194  194  LOOP5    MV      F0,SH
80B4 200E8F        195  195          MVI     08,K            SET UP MASK IN K REG.
80B5 00000E        196  196          OR      F0,K,F0         OR IN MASK TO F0
80B6 2B80F0        197  197          ANDI    0EF,F0,F0       STRIP UNWANTED BITS
80B7 68B78D  80B7  198  198          BFL     8,SH,*          WAIT FOR READY
80B8 D98A0D  818A  199  199          BNR     F0,SH,ERROR     CHECK
80B9 A04000        200  200          ORI     10,F0,F0        SECOND MASK
80BA 78BCF0  80BC  201  201          BNEL    0F,F0,*+2
80BB F4BEF0  80BE  202  202          BEQH    0F,F0,*+3       CHECK FOR LAST PATTERN
80BC AC0010        203  203          AI      01,F0,F0
80BD DCB380  80B3  204  204          B       LOOP5           NEXT PATTERN
80BE 9D00BE  80BE  205  205          LPI     *
80BF 05800F        206  206          TPS     ,
80C0 20C06F        207  207  TEST6    MVI     36,F0
80C1 D57A80  817A  208  208          SB      TEST#           PRINT TEST#6
80C2 20000F        209  209          MVI     00,F0
80C3 A00E00        210  210  LOOP4    MV      F0,K
80C4 D98A0E  818A  211  211          BNR     F0,K,ERROR
80C5 A00100        212  212          MV      F0,F1           THIS IS THE 2ND CHAIN MODE TEST BUT INCLUDES SL
80C6 0202E0        213  213          MVX     F1F0,F3F2
80C7 8204E2        214  214          MVX     F3F2,F5F4
80C8 0206E4        215  215          MVX     F5F4,F7F6
80C9 D98A07  818A  216  216          BNR     F0,F7,ERROR
80CA A00C07        217  217          MV      F7,SL
80CB 598A0C  818A  218  218          BNR     F0,SL,ERROR     CHECK ALL REGS.=
80CC 200F0F        219  219          NOP     ,
80CD 200F0F        220  220          NOP     ,
80CE 78D0F0  80D0  221  221          BNEL    0F,F0,*+2
80CF F4D2F0  80D2  222  222          BEQH    0F,F0,*+3       CHECK FOR LAST PATTERN
80D0 AC0010        223  223          AI      01,F0,F0
80D1 5CC380  80C3  224  224          B       LOOP4           NEXT PATTERN
80D2 200F0F        225  225          NOP     ,
```

```
80D3 1D00D3  80D3   1   227  TEST7   LPI     *
80D4 05800F          2   228          TPS     ,
80D5 A0C07F          3   229          MVI     37,F0
80D6 D57A80  817A    4   230          SB      TEST#          PRINT TEST#7
80D7 2140AF          5   231          MVI     5A,F0          PUT "5A"IN F0
80D8 A2815F          6   232          MVI     0A5,F1         PUT "A5" IN F1
80D9 A0020F          7   233          MVI     00,F2          CLEAR F2
80DA 004201          8   234          SHLL    F0,F1,F2       SHIFT LL---F2="5A"
80DB 798AA2  818A    9   235          BNEL    0A,F2,ERROR    CHECK F2="5A"
80DC FD8A52  818A   10   236          BNEH    05,F2,ERROR
80DD 044301         11   237          SHLH    F0,F1,F3       SHIFT----F3="55"
80DE F98A53  818A   12   238          BNEL    05,F3,ERROR    CHECK F3="55"
80DF 7D8A53  818A   13   239          BNEH    05,F3,ERROR
80E0 884401         14   240          SHHL    F0,F1,F4       SHIFT----F4="AA"
80E1 798AA4  818A   15   241          BNEL    0A,F4,ERROR    CHECK F4="AA"
80E2 FD8AA4  818A   16   242          BNEH    0A,F4,ERROR
80E3 8C4501         17   243          SHHH    F0,F1,F5       SHIFT----F5="A5"
80E4 F98A55  818A   18   244          BNEL    05,F5,ERROR    CHECK F5="A5"
80E5 7D8AA5  818A   19   245          BNEH    0A,F5,ERROR
80E6 804601         20   246          SHLL    F0,F1,F6       SHIFT----F6="5A"
80E7 F98AA6  818A   21   247          BNEL    0A,F6,ERROR    CHECK F6="5A"
80E8 7D8A56  818A   22   248          BNEH    05,F6,ERROR
80E9 0C4701         23   249          SHHH    F0,F1,F7       SHIFT----F7="A5"
80EA 798A57  818A   24   250          BNEL    05,F7,ERROR    CHECK F7="A5"
80EB FD8AA7  818A   25   251          BNEH    0A,F7,ERROR
                    26   252  *
                    27   253  *
                    28   254  *
80EC 1D00EC  80EC   29   255  TEST8   LPI     *
80ED 05800F         30   256          TPS     ,
80EE A0C08F         31   257          MVI     38,F0
80EF D57A80  817A   32   258          SB      TEST#          PRINT TEST#8
80F0 A00001         33   259          MV      F1,F0          F0=5A=A5=55=AA=5A CHECK NO OTHER REGS CHANGE
80F1 D98A01  818A   34   260          BNR     F0,F1,ERROR
80F2 200003         35   261          MV      F3,F0
80F3 598A03  818A   36   262          BNR     F0,F3,ERROR
80F4 A00004         37   263          MV      F4,F0
80F5 D98A04  818A   38   264          BNR     F0,F4,ERROR
80F6 200006         39   265          MV      F6,F0
80F7 598A06  818A   40   266          BNR     F0,F6,ERROR
80F8 55F080  81F0   41   267          SB      FILCHK
                    42   268  *
                    43   269  *
                    44   270  *
80F9 200102         45   271          MV      F2,F1          F1=A5=5A=AA=55=A5
```

```
80FA 598A12  818A   46   272          BNR     F1,F2,ERROR
80FB 200104          47   273          MV      F4,F1
80FC 598A14  818A   48   274          BNR     F1,F4,ERROR
80FD A00103          49   275          MV      F3,F1
80FE D98A13  818A   50   276          BNR     F1,F3,ERROR
80FF A00105          51   277          MV      F5,F1
8100 D98A15  818A   52   278          BNR     F1,F5,ERROR
8101 55F080  81F0   53   279          SB      FILCHK
                    54   280  *
                    55   281  *
                    56   282  *
8102 A00205          57   283          MV      F5,F2          F2=5A=A5=55=AA=5A
8103 D98A25  818A   58   284          BNR     F2,F5,ERROR
8104 A00203          59   285          MV      F3,F2
8105 D98A23  818A   60   286          BNR     F2,F3,ERROR
8106 200204          61   287          MV      F4,F2
8107 598A24  818A   62   288          BNR     F2,F4,ERROR
8108 A00206          63   289          MV      F6,F2
8109 D98A26  818A   64   290          BNR     F2,F6,ERROR
810A 55F080  81F0   65   291          SB      FILCHK
                    66   292  *
                    67   293  *
                    68   294  *
810B 200506          69   295          MV      F6,F5          F5=A5=5A=AA=55=A5
810C 598A56  818A   70   296          BNR     F5,F6,ERROR
810D A00504          71   297          MV      F4,F5
810E D98A54  818A   72   298          BNR     F5,F4,ERROR
810F 200503          73   299          MV      F3,F5
8110 598A53  818A   74   300          BNR     F5,F3,ERROR
8111 A00507          75   301          MV      F7,F5
8112 D98A57  818A   76   302          BNR     F5,F7,ERROR
8113 55F080  81F0   77   303          SB      FILCHK
                    78   304  *
                    79   305  *
                    80   306  *
8114 A00607          81   307          MV      F7,F6          F6=5A=A5=55=AA=5A
8115 D98A67  818A   82   308          BNR     F6,F7,ERROR
8116 200603          83   309          MV      F3,F6
8117 598A63  818A   84   310          BNR     F6,F3,ERROR
8118 A00604          85   311          MV      F4,F6
8119 D98A64  818A   86   312          BNR     F6,F4,ERROR
811A 200600          87   313          MV      F0,F6
811B 598A60  818A   88   314          BNR     F6,F0,ERROR
811C 55F080  81F0   89   315          SB      FILCHK
                    90   316  *
```

```
                        91    317  *
                        92    318  *
811D A00700             93    319       MV        F0,F7              F7=A5=5A=AA=55=A5
811E D98A70   818A      94    320       BNR       F7,F0,ERROR
811F 200704             95    321       MV        F4,F7
8120 598A74   818A      96    322       BNR       F7,F4,ERROR
8121 A00703             97    323       MV        F3,F7
8122 D98A73   818A      98    324       BNR       F7,F3,ERROR
8123 200701             99    325       MV        F1,F7
8124 598A71   818A     100    326       BNR       F7,F1,ERROR
8125 55F080   81F0     101    327       SB        FILCHK
                       102    328  *
                       103    329  *
                       104    330  *
8126 A00304            105    331       MV        F4,F3              F3=55=AA=5A=A5=55
8127 D98A34   818A     106    332       BNR       F3,F4,ERROR
8128 200306            107    333       MV        F6,F3
8129 598A36   818A     108    334       BNR       F3,F6,ERROR
812A A00307            109    335       MV        F7,F3
812B D98A37   818A     110    336       BNR       F3,F7,ERROR
812C 21435F            111    337       MVI       55,F3
812D F98A53   818A     112    338       BNEL      05,F3,ERROR
812E 7D8A53   818A     113    339       BNEH      05,F3,ERROR
812F 55F080   81F0     114    340       SB        FILCHK
                       115    341  *
                       116    342  *
                       117    343  *
8130 A00403            118    344       MV        F3,F4              F4=AA=55=A5=5A=AA
8131 D98A43   818A     119    345       BNR       F4,F3,ERROR
8132 A00405            120    346       MV        F5,F4
8133 D98A45   818A     121    347       BNR       F4,F5,ERROR
8134 A00406            122    348       MV        F6,F4
8135 D98A46   818A     123    349       BNR       F4,F6,ERROR
8136 A284AF            124    350       MVI       0AA,F4
8137 798AA4   818A     125    351       BNEL      0A,F4,ERROR
8138 FD8AA4   818A     126    352       BNEH      0A,F4,ERROR
8139 55F080   81F0     127    353       SB        FILCHK
                       128    354  *
                       129    355  *
813A 1D013A   813A     130    356  TEST9LPI       *
813B 05800F            131    357       TPS       ,
813C 20C09F            132    358       MVI       39,F0
813D D57A80   817A     133    359       SB        TEST#              PRINT TEST#9
813E 9D0146   8146     134    360       LPI       *+8
813F 05800F            135    361       TPS       ,
```

```
8140 994000   1000   136   362          LPI       1000              SET UP CONTROL MEMORY ADDRESS= 1000
8141 05800F          137   363          TPS       ,                 PUSH ADDRESS ON STACK
8142 234E5F          138   364          MVI       0D5,K             SET DATA TO BE WRITTEN IN K
8143 2289AF          139   365          MVI       0AA,PH            SET DATA TO BE WRITTEN IN PH
8144 A1485F          140   366          MVI       55,PL             SET DATA TO BE WRITTEN IN PL
8145 078400          141   367          SR,WCM    ,                 WRITE CONTROL MEMORY ADDRESS 1000
8146 A00E0F          142   368          MVI       00,K              CLEAR K
8147 20090F          143   369          MVI       00,PH             CLEAR PH
8148 A0080F          144   370          MVI       00,PL             CLEAR PL
8149 1D014E   814E     145   371          LPI       *+5
814A 05800F          146   372          TPS       ,
814B 994000   1000     147   373          LPI       1000              PUSH READ ADDRESS ON STACK
814C 05800F          148   374          TPS       ,
814D 878600          149   375          SR,RCM    ,                 READ CONTROL MEMORY ADDRESS 1000
814E 798AAE   818A     150   376          BNEL      0A,K,ERROR        CHECK THAT DATA WRITTEN IS THE SAME AS READ
814F 7D8A2E   818A     151   377          BNEH      02,K,ERROR
8150 F98AA9   818A     152   378          BNEL      0A,PH,ERROR
8151 7D8AA9   818A     153   379          BNEH      0A,PH,ERROR
8152 798A58   818A     154   380          BNEL      05,PL,ERROR
8153 FD8A58   818A     155   381          BNEH      05,PL,ERROR
8154 200F0F          156   382          NOP       ,
8155 200F0F          157   383          NOP       ,
8156 D60180   8201     158   384          SB        SEC1              WAIT APPROXIMATELY ONE SECOND FOR SCREEN DISPLAY
8157 DC4A80   804A     159   385          B         START0            RESTART CONTINUOUS TEST
```

```
8158 A00E5F         1   387  ENABLE5  MVI   5,K          SET CRT ADDR.
8159 178C00         2   388  ENABLE   CIO   0C0          ADDR. STROBE
815A D55B80  815B   3   389  DELAY10  SB    DELAY5
815B 555C80  815C   4   390  DELAY5   SB    *+1
815C D55D80  815D   5   391           SB    *+1
815D 200F0F         6   392           NOP   ,
815E 200F0F         7   393           NOP   ,
815F 87800F         8   394           SR    ,
                    9   395  *
                    10  396  *
8160 E9608D  8160   11  397  OBSTROBE BFL   8,SH,*       WAIT FOR DEVIE READY
8161 D55B80  815B   12  398           SB    DELAY5
8162 978200         13  399           CIO   20
8163 5D5B80  815B   14  400           B     DELAY5
                    15  401  *
                    16  402  *
8164 A00DBD         17  403  INIT     ORI   0B,SH,SH     INHIBIT INPUT
8165 284DFD         18  404           ANDI  1F,SH,SH     SET NO TRAP,40BIT,HALT/STEP OFF
8166 D55880  8158   19  405           SB    ENABLE5
8167 A00E3F         20  406           MVI   03,K
8168 556080  8160   21  407           SB    OBSTROBE     CLEAR CRT
8169 87800F         22  408           SR    ,
                    23  409  *
                    24  410  *
816A D55880  8158   25  411  CRLF     SB    ENABLE5      THIS ROUTINE PRINTS CARRIAGE RETURN LINE FEED
816B 200EDF         26  412           MVI   0D,K
816C 556080  8160   27  413           SB    OBSTROBE     CR
816D A00EAF         28  414           MVI   0A,K
816E 556080  8160   29  415           SB    OBSTROBE     LF
816F 87800F         30  416           SR    ,
                    31  417  *
                    32  418  *
8170 D55880  8158   33  419  SPACE    SB    ENABLE5
8171 208E0F         34  420           MVI   20,K
8172 556080  8160   35  421           SB    OBSTROBE     PRINT A SPACE
8173 87800F         36  422           SR    ,
                    37  423  *
                    38  424  *
8174 D55880  8158   39  425  INITCRT  SB    ENABLE5
8175 200E1F         40  426           MVI   01,K
8176 556080  8160   41  427           SB    OBSTROBE     HOME CUSOR
8177 A00E3F         42  428           MVI   03,K
8178 556080  8160   43  429           SB    OBSTROBE     CLR CRT
8179 87800F         44  430           SR    ,
                    45  431  *
```

```
                    46   432  *
817A D55880  8158   47   433  TEST#    SB       ENABLE5
817B 214E4F         48   434           MVI      54,K
817C 556080  8160   49   435           SB       OBSTROBE      T
817D 210E5F         50   436           MVI      45,K
817E 556080  8160   51   437           SB       OBSTROBE      E
817F A14E3F         52   438           MVI      53,K
8180 556080  8160   53   439           SB       OBSTROBE      S
8181 214E4F         54   440           MVI      54,K
8182 556080  8160   55   441           SB       OBSTROBE      T
8183 208E3F         56   442           MVI      23,K
8184 556080  8160   57   443           SB       OBSTROBE      #
8185 D57080  8170   58   444           SB       SPACE         SPACE
8186 D59880  8198   59   445           SB       PRTBYTE       TEST NUMBER X IS PRINTED
8187 200F0F         60   446           NOP      ,
8188 556A80  816A   61   447           SB       CRLF
8189 87800F         62   448           SR       ,
                    63   449  *
                    64   450  *
818A D55880  8158   65   451  ERROR    SB       ENABLE5
818B 210E5F         66   452           MVI      45,K
818C 556080  8160   67   453           SB       OBSTROBE      E
818D 214E2F         68   454           MVI      52,K
818E 556080  8160   69   455           SB       OBSTROBE      R
818F 214E2F         70   456           MVI      52,K
8190 556080  8160   71   457           SB       OBSTROBE      R
8191 210EFF         72   458           MVI      4F,K
8192 556080  8160   73   459           SB       OBSTROBE      O
8193 214E2F         74   460           MVI      52,K
8194 556080  8160   75   461           SB       OBSTROBE      R
8195 556A80  816A   76   462           SB       CRLF
8196 D60180  8201   77   463           SB       SEC1          WAIT APP. ONE SECOND FOR DISPLAY PURPOSES
8197 87800F         78   464           SR       ,
                    79   465  *
                    80   466  *
                    81   467  *
                    82   468  *
                    83   469  *
                    84   470  *
8198 A00E00         85   471  PRTBYTE  MV       F0,K          THIS IS A ROUTINE TO PRINT FILE REG 0
8199 556080  8160   86   472           SB       OBSTROBE
819A 87800F         87   473           SR       ,
                    88   474  *
                    89   475  *
                    90   476  *
```

```
819B 21430F        91   477 SYSPE    MVI     C'P',F3          THESE ROUTINES PRINT PARITY ERROR ADDR.
819C 21025F        92   478 SYSVE    MVI     C'E',F2
819D 2100DF        93   479          MVI     C'M',F0
819E D55880  8158  94   480 SYSERROR SB      ENABLE5
819F 200E1F        95   481          MVI     01,K
81A0 556080  8160  96   482          SB      OBSTROBE
81A1 208E0F        97   483          MVI     20,K
81A2 A0040F        98   484          MVI     00,F4
81A3 556080  8160  99   485 SPACEOUT SB      OBSTROBE
81A4 98C44F       100   486          AC,,1   F4,,F4
81A5 7DA354  81A3 101   487          BNEH    5,F4,SPACEOUT
81A6 200E1F       102   488          MVI     01,K
81A7 556080  8160 103   489          SB      OBSTROBE
81A8 208EAF       104   490          MVI     C'*',K
81A9 556080  8160 105   491          SB      OBSTROBE
81AA 556080  8160 106   492          SB      OBSTROBE
81AB 556080  8160 107   493          SB      OBSTROBE
81AC D57080  8170 108   494          SB      SPACE
81AD A14E3F       109   495          MVI     C'S',K
81AE 556080  8160 110   496          SB      OBSTROBE
81AF A14E9F       111   497          MVI     C'Y',K
81B0 556080  8160 112   498          SB      OBSTROBE
81B1 A14E3F       113   499          MVI     C'S',K
81B2 556080  8160 114   500          SB      OBSTROBE
81B3 214E4F       115   501          MVI     C'T',K
81B4 556080  8160 116   502          SB      OBSTROBE
81B5 210E5F       117   503          MVI     C'E',K
81B6 556080  8160 118   504          SB      OBSTROBE
81B7 A10EDF       119   505          MVI     C'M',K
81B8 556080  8160 120   506          SB      OBSTROBE
81B9 D57080  8170 121   507          SB      SPACE
81BA 210E5F       122   508          MVI     C'E',K
81BB 556080  8160 123   509          SB      OBSTROBE
81BC 214E2F       124   510          MVI     C'R',K
81BD 556080  8160 125   511          SB      OBSTROBE
81BE 214E2F       126   512          MVI     C'R',K
81BF 556080  8160 127   513          SB      OBSTROBE
81C0 210EFF       128   514          MVI     C'O',K
81C1 556080  8160 129   515          SB      OBSTROBE
81C2 214E2F       130   516          MVI     C'R',K
81C3 556080  8160 131   517          SB      OBSTROBE
81C4 D57080  8170 132   518          SB      SPACE
81C5 A08E8F       133   519          MVI     C'(',K
81C6 556080  8160 134   520          SB      OBSTROBE
81C7 A00E03       135   521          MV      F3,K
```

```
81C8 556080  8160  136  522          SB      OBSTROBE
81C9 200E02        137  523          MV      F2,K
81CA 556080  8160  138  524          SB      OBSTROBE
81CB 200E01        139  525          MV      F1,K
81CC 556080  8160  140  526          SB      OBSTROBE
81CD A00E00        141  527          MV      F0,K
81CE 556080  8160  142  528          SB      OBSTROBE
81CF D57080  8170  143  529          SB      SPACE
81D0 200009        144  530          MV      PH,F0
81D1 D5E080  81E0  145  531          SB      PRTBITE
81D2 A00008        146  532          MV      PL,F0
81D3 D5E080  81E0  147  533          SB      PRTBITE
81D4 208E9F        148  534          MVI     C')',K
81D5 556080  8160  149  535          SB      OBSTROBE
81D6 D57080  8170  150  536          SB      SPACE
81D7 208EAF        151  537          MVI     C'*',K
81D8 556080  8160  152  538          SB      OBSTROBE
81D9 556080  8160  153  539          SB      OBSTROBE
81DA 556080  8160  154  540          SB      OBSTROBE
81DB DD6A80  816A  155  541          B       CRLF
                   156  542  *
                   157  543  *
81DC 49DE21  81DE  158  544  PRTBIT1  BLER    F2,F1,*+2      THESE ROUTINES PRINT CHARS.STORED IN F0
81DD AC0272        159  545          AI      07,F2,F2
81DE 200E02        160  546          MV      F2,K
81DF DD6080  8160  161  547          B       OBSTROBE
81E0 A0C19F        162  548  PRTBITE  MVI     39,F1
81E1 20C30F        163  549          MVI     30,F3
81E2 8C4203        164  550          SHHH    F0,F3,F2
81E3 D5DC80  81DC  165  551          SB      PRTBIT1
81E4 084203        166  552          SHHL    F0,F3,F2
81E5 5DDC80  81DC  167  553          B       PRTBIT1
                   168  554  *
                   169  555  *
81E6 8D800F        170  556  PE24     TSP     ,              FAILING ADDR TO PC'S
81E7 800FFF        171  557          OR      -,,            DECREMENT ADDR BY 1
81E8 21013F        172  558          MVI     C'C',F1        SET UP F1 TO PRINT OUT "PECM"
81E9 D59B80  819B  173  559          SB      SYSPE          PRINT OUT CONTROL MEM PARITY ERROR INFO
81EA 5DEA80  81EA  174  560          B       *              HANG HERE UNTIL RESET IS STRUCK
                   175  561  *
                   176  562  *
81EB 8D800F        177  563  PE8      TSP     ,              FAILING ADDR. TO PC'S
81EC 800FFF        178  564          OR      -,,            DECREMENT ADDR.
81ED A1014F        179  565          MVI     C'D',F1        SET UP F1 TO PRINT OUT "PEDM"
81EE D59B80  819B  180  566          SB      SYSPE          PRINT OUT DATA MEMORY PARITY ERROR
```

```
81EF 5DEF80  81EF   181  567          B       *          HANG HERE WAITING FOR RESET TO BE STRUCK
                    182  568  *
                    183  569  *
81F0 A14EAF         184  570  FILCHK  MVI     5A,K       THIS ROUTINE CHECKS WORST CASE PATT. UNCHAGED
81F1 D98A0E  818A   185  571          BNR     F0,K,ERROR
81F2 A28E5F         186  572          MVI     0A5,K
81F3 598A1E  818A   187  573          BNR     F1,K,ERROR
81F4 A14EAF         188  574          MVI     5A,K
81F5 598A2E  818A   189  575          BNR     F2,K,ERROR
81F6 A14E5F         190  576          MVI     55,K
81F7 D98A3E  818A   191  577          BNR     F3,K,ERROR
81F8 A28EAF         192  578          MVI     0AA,K
81F9 598A4E  818A   193  579          BNR     F4,K,ERROR
81FA A28E5F         194  580          MVI     0A5,K
81FB D98A5E  818A   195  581          BNR     F5,K,ERROR
81FC A14EAF         196  582          MVI     5A,K
81FD D98A6E  818A   197  583          BNR     F6,K,ERROR
81FE A28E5F         198  584          MVI     0A5,K
81FF 598A7E  818A   199  585          BNR     F7,K,ERROR
8200 87800F         200  586          SR      ,
                    201  587  *
                    202  588  *
8201 20000F         203  589  SEC1    MVI     00,F0      THIS ROUTINE FORCES A SERIES OF INST. DELAYS
8202 A00E0F         204  590  RESTRT  MVI     00,K       APP=TO ONE SECOND
8203 AC0E1E         205  591  WAIT1   AI      01,K,K
8204 FA069E  8206   206  592          BNEL    09,K,*+2
8205 7608CE  8208   207  593          BEQH    0C,K,*+3
8206 D60D80  820D   208  594          SB      DELAY50
8207 DE0380  8203   209  595          B       WAIT1
8208 AC0010         210  596  WAIT2   AI      1,F0,F0
8209 7A0BB0  820B   211  597          BNEL    0B,F0,*+2
820A F60C10  820C   212  598          BEQH    1,F0,*+2
820B 5E0280  8202   213  599          B       RESTRT
820C 87800F         214  600          SR      ,
820D 555A80  815A   215  601  DELAY50 SB      DELAY10
820E 555A80  815A   216  602          SB      DELAY10
820F 555A80  815A   217  603          SB      DELAY10
8210 555A80  815A   218  604          SB      DELAY10
8211 555A80  815A   219  605          SB      DELAY10
8212 87800F         220  606          SR      ,
                    221  607  *
                    222  608  *END
```

NUMBER OF LINES IN ERROR = 0 NO. OF WARNINGS = 0 NO. OF SYMBOLS = 48 OVERFLOWS = 36

SYMBOL	VALUE	DEFN	REFERENCES
CRLF	816A	0411	0066 0447 0462 0541
DELAY10	815A	0389	0601 0602 0603 0604 0605
DELAY5	815B	0390	0389 0398 0400
DELAY50	820D	0601	0594
ENABLE	8159	0388	0069
ENABLE5	8158	0387	0075 0405 0411 0419 0425 0433 0451 0480
ERROR	818A	0451	0096 0097 0098 0099 0100 0101 0102 0103 0104 0110 0111 0117 0118 0119 0120 0121 0122 0123 0124 0134 0150 0170 0172 0173 0181 0199 0211 0216 0218 0235 0236 0238 0239 0241 0242 0244 0245 0247 0248 0250 0251 0260 0262 0264 0266 0272 0274 0276 0278 0284 0286 0288 0290 0296 0298 0300 0302 0308 0310 0312 0314 0320 0322 0324 0326 0332 0334 0336 0338 0339 0345 0347 0349 0351 0352 0376 0377 0378 0379 0380 0381 0571 0573 0575 0577 0579 0581 0583 0585
FILCHK	81F0	0570	0267 0279 0291 0303 0315 0327 0340 0353
INIT	8164	0403	0021 0079
INITCRT	8174	0425	
INPUT	8041	0070	
INTEST	803E	0067	0078
LOOP	807D	0133	0138
LOOP1	8089	0145	0156
LOOP2	80A7	0180	0185
LOOP4	80C3	0210	0224
LOOP5	80B3	0194	0204
OBSTROBE	8160	0397	0025 0030 0035 0040 0045 0050 0055 0060 0065 0077 0407 0413 0415 0421 0427 0429 0435 0437 0439 0441 0443 0453 0455 0457 0459 0461 0472 0482 0485 0489 0491 0492 0493 0496 0498 0500 0502 0504 0506 0509 0511 0513 0515 0517 0520 0522 0524 0526 0528 0535 0538 0539 0540 0547
PART1	8054	0092	
PART2	8063	0107	
PE24	81E6	0556	0002
PE8	81EB	0563	0004
PRTBIT1	81DC	0544	0551 0553
PRTBITE	81E0	0548	0531 0533
PRTBYTE	8198	0471	0445
RESTRT	8202	0590	0599
SEC1	8201	0589	0384 0463
SPACE	8170	0419	0444 0494 0507 0518 0529 0536
SPACEOUT	81A3	0485	0487
START	8010	0021	0003 0005 0016
START0	804A	0079	0073 0385
START1	804B	0083	
SYSERROR	819E	0480	
SYSPE	819B	0477	0559 0566

SYMBOL	VALUE	DEFN	REFERENCES
SYSVE	819C	0478	
TEST#	817A	0433	0091 0129 0143 0166 0178 0191 0208 0230 0258 0359
TEST0	8050	0088	
TEST1	8077	0127	
TEST2	8084	0140	
TEST3	8097	0163	0158
TEST4	80A4	0177	
TEST5	80AF	0190	
TEST6	80C0	0207	
TEST7	80D3	0227	
TEST8	80EC	0255	
TEST9	813A	0356	
WAIT1	8203	0591	0595
WAIT2	8208	0596	

MODULE REPAIR GUIDE NO. 4.3
EDITED BY CUSTOMER ENGINEERING DIVISION

210-6790 TEST PROGRAM May 8, 1978

1. INTRODUCTION

210-6790 INSTRUCTION COUNTER TEST PROGRAM

The 210-6790 test program was written to be used as a helpful aid in the troubleshooting and repair of the 6790 module.

The program utilizes many of the same techniques developed in the 210-6793 test program, ie. implementing visual display of test results wherever possible.

The program is designed to be run exclusively from PROM for the same reasons already discussed in the INTRODUCTION section of the 210-6793 test program. Additionally, it is impossible for this program to be run from Control Memory, because many of the test routines utilize Control Memory for program execution.

2. INSTALLATION/OPERATION

The standard 2200VP system repair equipment is utilized for testing the 6790 module. The only additional equipment required is a set of debug PROMs located as follows:

PROM #	LOCATION
378-2174	6789 - L27
378-2175	6789 - L28
378-2176	6789 - L29

The only installation procedure required is to physically remove the boot/utility PROMs from the 6789 and replace them with the 6790 debug PROMs.

Printed in U.S.A.
WANG LABORATORIES, INC.
ONE INDUSTRIAL AVENUE, LOWELL, MASSACHUSETTS 01851, TEL. (617) 851-4111, TWX 710 343-6769, TELEX 94-7421

3. GENERAL TEST INFORMATION

POWER ON

As power is applied to the system, a micro trap to PROM address 8003 occurs which begins test operation.

INITIAL 0

INITIAL 0 is a preliminary test to see if it is possible for the 6790 module to execute a POWER ON RESET, forcing the Instruction Counter to 8003. The Instruction Counter Source Decoder must multiplex the trap address of 8003 through to the IC. The Instruction Counter must then load 8003. The IC Register then inputs 8003, and increments this value to 8004. The micro program executes a subroutine branch to the beginning of this test which pushed IC's = to 8004 on the stack.

There are essentially three parts to this test. Part 1 is a test of the Program Counter's ability to be cleared (set to 0000) and set to all ones (set to FFFF). Part 2 tests that a Subroutine Branch pushed the incremented IC onto the stack. In Part 3, the Instruction Counter is incremented in order to execute the instruction associated with the program. Therefore, as the program progresses through various tests, the Instruction Counter is being exercised with incremental data. A low order IC bit failure will cause unpredictable results when executing the first series of tests.

If an error is detected in this preliminary test, the program will do one of two things. A failure in Part 1 will cause the program to loop on a short series of instructions that either clear or set the PC's. A failure in Part 2 will cause the program to hang at the failing test instruction executing a branch to the same location (the IC doesn't = 8004).

If no error occurs in either test, the CRT is cleared and a "0" is printed on the screen.

NOTE:

If the CRT is clear, but no "0" is displayed on the screen, Part 1 was executed properly, but there is a failure in Part 2. If the CRT does not clear, Part 1 failed. Once a test fails, there will be no further execution of other tests until the failure is cleared. Thus, as long as the failure is present, the program will loop on the error for troubleshooting purposes.

INITIAL 1

INITIAL 1 will be executed ten times. The test is basically the same as Initial 0 Part 2, except that this test exercises IC address 8005 through 800E checking primarily the trap decoder input to the IC Source Selectors, the IC itself, the IC register, and the data path to the stack from the IC Register. As each address is tested, a character is printed on the CRT until the IC is incremented to 800E. At the completion of this test, the CRT should display:

"0123456789:"

If a failure occurs during this test, the program will branch to the same location at which the failure was detected and the CRT display will be missing characters.

INITIAL 2

INITIAL 2 is a short routine that tests the Control Memory parity error traps. There are two instruction locations in PROM that are purposely written to generate bad parity. The program attempts to execute these instructions which will cause a parity error trap to

PROM location 8000. The program then tests to make sure that the known bad parity locations were the locations that caused the parity error.

If an error, other than parity, results from execution of this test, the program will end up branching to the wrong location as a result of an incorrect parity error trap decode. Failure results are unpredictable but will be detected visually because the print out for the next test will not be displayed on the CRT, but the print out for the previous test will still be displayed. Thus, "0123456789:" will be displayed, but "INPUT THE AMOUNT OF CONTROL MEMORY...." will not be.

INTEST

The INTEST routine is used to determine the amount of Control Memory available to the system. Printed on the CRT will be a statement: "INPUT THE AMOUNT OF CONTROL MEMORY (0=16K 1=20K)". At this point, the operator must type either a "0" or "1" depending on the amount of control memory available to the system. Any other character will echo a "?" and the program will loop until a valid response is entered.

TEST#0

Test#0 checks that the PL (low order portion of the PC's) can be incremented from 00 through FF, one step at a time, without failure. If an error occurs in this test, "TEST#0" and "ERROR" are displayed on the CRT and the program will loop until the error is cleared.

TEST#1

Test#1 checks the PH (high order portion of the PC's) in exactly the same manner as Test#0 checks the PL.

TEST#2 through TEST#5

Test#2 through Test#5 are designed to exercise the memory select lines and other various functions associated with Control Memory that are controlled by the 6790 module.

Test#2 writes the last location of a 4K boundary (OFFF) in Control Memory with a test pattern, then reads back the same location and checks to see if data written compares with the data read.

This, location "OFFF" is written with "D5AA55" then read back and checked. If an error occurs, "TEST#" and "ERROR" are printed then the test restarts from the beginning.

Test 3 through Test 5 do exactly the same, except for the addresses they use.

Test 3 uses address "1FFF" (8K boundary, last address)
Test 4 uses address "2FFF" (12K boundary, last address)
Test 5 uses address "3FFF" (16K boundary, last address)

TEST#6

Test#6 is designed to exercise the Control Memory by writing the memory address into the memory location (location 0000 will contain a 0000, location 1023 will contain a 1023, etc.). In order to accomplish this, a Write Control Memory (WCM) and a Read Control Memory (RCM) must be executed (both use the stack in conjunction with the IC's), to read or write into memory. The successful execution of this test will ensure that all IC bits can be used to read and write to Control Memory properly.

This test loads the Instruction Counter and tests that the IC loaded to the proper value, however, it does not test the IC's ability to count up properly and also does not test the ripple carry add through the IC chips and associated logic.

If the incorrect memory size was specified in INTEST, an error can result in this test because the program will try to write to non-existent memory.

If an error is encountered during this test, the program will print out "TEST#6" and "ERROR" and restart at the beginning of the test.

If the system has 16K of control memory, addresses 0000 through 3FFF are tested. If the system has 20K of control memory, addresses 0000 through 4FFF are tested.

TEST#7

Test#7 is a test designed to exercise chips L-27, L-28, L-18, and partially L-29 (Instruction Counter). Previously, the carry during a count operation (ripple carry add) was untested. This test was designed to exercise this feature.

The test writes locations 000F, 00FF, and 0FFF with a "NOOP" instruction, then writes locations 0010, 0100, and 1000 with a SB 8023 (branch to a test routine). The program then forces execution at locations 000F, 00FF and 0FFF respectively. Thus, at location 000F, the instruction "NOOP" is executed. The IC register is incremented to 0010 and the "SB CHKTST" (8023) is executed, which pushes an address of 0011 onto the stack and forces program execution at "CHKTST". "CHKTST" pops the stack and checks to see if a valid address was pushed on the stack in this case "0011". If an error occurs, "TEST#7" and "ERROR" are displayed then the test restarts. If no error occurred, the two remaining addresses are tested then control proceeds to the next test.

TEST#8

Test#8 is designed to be more of a "confidence" type test than an actual troubleshooting aid. It will however loop on test if an error occurs and display "TEST#8" and "ERROR".

At the beginning of TEST #8, all even Control Memory addresses are written with a "NOOP" instruction and all odd locations are written with a "SB TSTADD" (8054). Beginning at location 0000, each of two instruction routines are executed, making sure that the IC's are incremented properly until the end of memory is reached.

NOTE:

This routine also exercises the Program Counter, using it to compare the test address counter with the test addresses. All memory selects and Control Memory functions are being tested by this routine as well as the refresh circuitry. The add carry function of the IC chips is tested heavily during execution of this program as well as most possible bit patterns through the IC.

RESET

When RESET is keyed, the program reruns the preliminary tests and asks for the amount of Control Memory available to the system.

ADDITIONAL INFORMATION

Since the program relies very heavily on Control Memory, the Control Memory microcode diagnostics on the operating system diskette should be run prior to set up and execution of this program. Verify that this test program runs with all "known good" modules in the test system before attempting to repair boards. While this test is not intended to repair Control Memory boards, it will fail if the Control Memory is defective. Therefore, it is important that both Control Memory modules are good.

Because of the somewhat unpredictable refresh type failures, the technician should verify visually with an oscilloscope that the Control Memory and the Data Memory refresh circuitry is operating properly on the 6790 board under test. This can be done by observing the output of L30 pin 3 at the jumper between L50 and L51: a 3 usec negative pulse occurring every 23 usec.

PARITY ERRORS

The program will visually display any unforced parity errors. The address displayed on the CRT when a failure is detected is decremented by one. Therefore, the address displayed is the last valid instruction performed. The next address is the address which caused the parity error.

```
                 8000   1    1              ORG      8000              FORCE PROGRAM START LOC. 0000
8000 DE5580      8255   2    2              B        PE24              CONTROL MEMORY PARITY ERROR TRAP
8001 5C0380      8003   3    3              B        *+2               RESET KEY WAS STRUCK, RESTART PROGRAM
8002 5E6280      8262   4    4              B        PE8               DATA MEMORY PARITY ERROR TRAP
8003 541080      8010   5    5              SB       INITIAL0          GO TO INITIAL TEST 0 IC'S=8004
8004 542380      8023   6    6              SB       INITIAL1          GO TO INITIAL TEST 1 IC'S=8005
8005 542380      8023   7    7              SB       INITIAL1          IC'S=8006
8006 542380      8023   8    8              SB       INITIAL1          IC'S=8007
8007 542380      8023   9    9              SB       INITIAL1          IC'S=8008
8008 542380      8023   10   10             SB       INITIAL1          IC'S=8009
8009 542380      8023   11   11             SB       INITIAL1          IC'S=800A
800A 542380      8023   12   12             SB       INITIAL1          IC'S=800B
800B 542380      8023   13   13             SB       INITIAL1          IC'S=800C
800C 542380      8023   14   14             SB       INITIAL1          IC'S=800D
800D 542380      8023   15   15             SB       INITIAL1          IC'S=800E
800E 5C2B80      802B   16   16             B        INITIAL2          FINISHED INITIAL TESTS, GO ON
800F 800000             17   17             INSTR    800000
                        18   18   *
                        19   19   *
                        20   20   *
8010 860000             21   21   INITIAL0  XORX     F1F0,F1F0,F1F0    CLEAR FILE REGS 1 AND 0
8011 190000      0000   22   22   PART1     LPI      0000              CLEAR PC'S
8012 581008      8010   23   23             BNR      F0,PL,INITIAL0    CHECK PL = TO F0
8013 581019      8010   24   24             BNR      F1,PH,INITIAL0    CHECK PC'S = TO F1
8014 23C0FF             25   25             MVI      0FF,F0            SET F0= TO FF
8015 A3C1FF             26   26             MVI      0FF,F1            SET F1= TO FF
8016 1FCFFF      FFFF   27   27             LPI      0FFFF             SET PC'S= TO FFFF
8017 581008      8010   28   28             BNR      F0,PL,INITIAL0    CHECK PC= TO FF
8018 581019      8010   29   29             BNR      F1,PH,INITIAL0    CHECK PC= TO FF
8019 55E280      81E2   30   30   PART2     SB       INITCRT           CLEAR CRT INHIBIT TRAPS AND INPUTS
801A 20014F             31   31             MVI      04,F1             SET UP F1 TO TEST LOW IC
801B 22020F             32   32             MVI      80,F2             SET UP F2 TO TEST HIGH IC
801C 8D800F             33   33             TSP      ,                 POP STACK PUT IC'S IN PC'S
801D 581D18      801D   34   34             BNR      F1,PL,*           CHECK FOR "04" IN IC LOW IF NOT = HANG HERE
801E D81E29      801E   35   35             BNR      F2,PH,*           CHECK FOR "80" IN IC HIGH IF NOT = HANG HERE
801F 05800F             36   36             TPS      ,                 RESET STACK POINT TO INITIAL1
8020 A0CE0F             37   37             MVI      30,K              PUT A "0" IN K
8021 D5CE80      81CE   38   38             SB       OBSTROBE          PRINT "0" ON CRT
8022 87800F             39   39             SR       ,                 GO TO 8004 THEN ON TO INITIAL1
                        40   40   *
                        41   41   *
8023 AC0E1E             42   42   INITIAL1  AI       1,K,K             INCREMENT K BY 1 TO PRINT NEXT TEST OK
8024 AC0111             43   43             AI       1,F1,F1           STEP F1 TO = IC'S LOW
8025 8D800F             44   44             TSP      ,                 PUT IC'S IN PC'S FOR TEST
8026 D82618      8026   45   45             BNR      F1,PL,*           HANG HERE IF IC'S NOT = TO PC'S (LOW)
```

```
8027 D82729   8027   46   46          BNR      F2,PH,*          HANG HERE IF IC'S NOT = TO PC'S (HIGH)
8028 D5CE80   81CE   47   47          SB       OBSTROBE         PRINT NEXT TEST NUMERIC
8029 05800F          48   48          TPS      ,                PUT ADDR. BACK ON STACK
802A 87800F          49   49          SR       ,                GO ON TO NEXT TEST
                     50   50  *
                     51   51  *
802B 56DE80   82DE   52   52  INITIAL2 SB       BDPARY           EXECUTE A CONTROL MEMORY PARITY ERROR TRAP
                     53   53  *
                     54   54  *
802C 55D880   81D8   55   55  INTEST   SB       CRLF             MOVE CUSOR TO NEXT LINE
802D D68A80   828A   56   56          SB       CMAMNT           DISPLAY "INPUT THE AMOUNT OF CONTROL MEM...."
802E 55D880   81D8   57   57          SB       CRLF
802F A00D0F          58   58          MVI      00,SH            ALLOW INPUTS
8030 200E1F          59   59          MVI      1,K              SET K FOR KEYBOARD INPUT
8031 D5C780   81C7   60   60          SB       ENABLE           ENABLE INPUT CHANNEL
8032 E8322D   8032   61   61  INPUT    BFL      2,SH,*           WAIT FOR INPUT FROM KEYBOARD
8033 FC363E   8036   62   62          BNEH     3,K,PRINT?       INPUTED CHARECTER = TO 3X?
8034 F0390E   8039   63   63          BEQL     0,K,ZERO         INPUTED CHAR. = TO 30?
8035 703C1E   803C   64   64          BEQL     1,K,ONE          INPUTED CHAR. = TO 31?
8036 20C0FF          65   65  PRINT?   MVI      C'?',F0          PUT A "?" IN F0
8037 560680   8206   66   66          SB       PRTBYTE          PRINT ? ON CRT
8038 DC2C80   802C   67   67          B        INTEST           GO GET ANOTHER CHAR. INPUTED
8039 23C6FF          68   68  ZERO     MVI      0FF,F6           SET FILE REG. 6 = TO FF FOR ADDR. TEST
803A A0C7FF          69   69          MVI      3F,F7            SET F7 = TO LAST ADDR. 16K
803B 5C3F80   803F   70   70          B        TEST0
803C 23C6FF          71   71  ONE      MVI      0FF,F6           PUT FF IN F6 FOR ADDR. TEST
803D 2107FF          72   72          MVI      4F,F7            SET F7 = TO LAST ADDR. 20K
803E 5C3F80   803F   73   73          B        TEST0
                     74   74  *
                     75   75  *
803F 9D003F   803F   76   76  TEST0    LPI      *                LOAD PC'S WITH TEST START
8040 05800F          77   77          TPS      ,                PUSH TEST START ON STACK
8041 20C00F          78   78          MVI      30,F0            PUT "0" IN F0
8042 55E880   81E8   79   79          SB       TEST#            PRINT "TEST#0" ON CRT
8043 20000F          80   80          MVI      00,F0            CLEAR K REG.
8044 A0080F          81   81          MVI      00,PL            CLEAR PL
8045 A00800          82   82  LOOPA    MV       F0,PL            PUT K IN PL FOR TEST
8046 D9F808   81F8   83   83          BNR      F0,PL,ERROR      TEST PL = TO K REG.
8047 7849F0   8049   84   84          BNEL     0F,F0,*+2        CHECK FOR LAST PATTERN
8048 F44BF0   804B   85   85          BEQH     0F,F0,*+3        CHECK FOR LAST PATTERN
8049 AC0010          86   86          AI       1,F0,F0          INCREMENT K BY 1
804A DC4580   8045   87   87          B        LOOPA            NEXT PATTERN TEST
                     88   88  *
                     89   89  *
804B 9D004B   804B   90   90  TEST1    LPI      *                PUT TEST ADDR IN PC'S
```

```
804C 05800F        91   91          TPS       ,              PUT TEST START ON STACK
804D A0C01F        92   92          MVI       31,F0          PUT "1" IN F0
804E 55E880  81E8  93   93          SB        TEST#          PRINT "TEST#1" ON CRT
804F A0010F        94   94          MVI       00,F1          CLEAR K REG.
8050 20090F        95   95          MVI       00,PH          CLEAR PH
8051 A00901        96   96   LOOPB  MV        F1,PH          PUT K IN PH FOR TEST
8052 D9F819  81F8  97   97          BNR       F1,PH,ERROR    PH = TO K ?
8053 7855F1  8055  98   98          BNEL      0F,F1,*+2      LAST PATTERN?
8054 F457F1  8057  99   99          BEQH      0F,F1,*+3      LAST PATTERN?
8055 AC0111       100  100          AI        1,F1,F1        INCREMENT K
8056 DC5180  8051 101  101          B         LOOPB
                  102  102   *
                  103  103   *
8057 1D0057  8057 104  104   TEST2  LPI       *              LOAD PC'S WITH TEST ADDR.
8058 05800F       105  105          TPS       ,              PUT START ADDR. ON STACK
8059 A0C02F       106  106          MVI       32,F0          PUT "2" IN F0
805A 55E880  81E8 107  107          SB        TEST#          PRINT "TEST#2" ON CRT
805B 190FFF  0FFF 108  108          LPI       0FFF           LOAD PC'S WITH LAST ADDR 4K MEM.
805C 81800F       109  109          TPA       ,00            PUT TEST ADDR. IN AUX 00
805D D46080  8060 110  110          SB        INITCMT        INTIALIZE AUX REGS. FOR CONTROL MEM. TEST
805E D46580  8065 111  111          SB        CMTEST         GO TO CONTROL MEMORY TEST
805F DC7C80  807C 112  112          B         TEST3          GO TO NEXT TEST
                  113  113   *
                  114  114   *
8060 1D006D  806D 115  115   INITCMT LPI      READ2          PUT START READ ROUTINE IN PC'S
8061 01801F       116  116          TPA       ,01            PUT READ ADDR. IN AUX 01
8062 1D0075  8075 117  117          LPI       CHECK1         PUT COMPARE ROUTINE START IN PC'S
8063 01802F       118  118          TPA       ,02            PUT START COMPARE ROUTINE IN AUX 02
8064 87800F       119  119          SR        ,              RETURN
                  120  120   *
                  121  121   *
8065 0B801F       122  122   CMTEST  TAP       ,01            LOAD PC'S WITH START OF READ ROUTINE
8066 05800F       123  123          TPS       ,              PUT READ START ON STACK
8067 8B800F       124  124          TAP       ,00            PUT TEST ADDR IN PC'S
8068 05800F       125  125          TPS       ,              PUT TEST ADDR. ON STACK
8069 234E5F       126  126          MVI       0D5,K          PUT DATA PATTERN TO BE WRITTEN IN K
806A 2289AF       127  127          MVI       0AA,PH         SET DATA TO BE WRITTEN IN PH
806B A1485F       128  128          MVI       55,PL          SET DATA TO BE WRITTEN IN PL
806C 078400       129  129          SR,WCM    ,              WRITE CONTROL MEMORY AT TEST ADDR.
806D A00E0F       130  130   READ2  MVI       00,K           CLEAR K
806E A0080F       131  131          MVI       00,PL          CLEAR PL
806F 20090F       132  132          MVI       00,PH          CLEAR PH
8070 0B802F       133  133          TAP       ,02            PUT COMPARE ROUTINE ADDR. IN PC'S
8071 05800F       134  134          TPS       ,              PUT COMPARE ROUTINE START ADDR. ON STACK
8072 8B800F       135  135          TAP       ,00            PUT TEST ADDR. IN PC'S
```

```
8073 05800F        136  136          TPS       ,           PUT TEST ADDR. ON STACK
8074 878600        137  137          SR,RCM    ,           READ CONTROL MEMORY
8075 79F8AE  81F8  138  138  CHECK1  BNEL      0A,K,ERROR  CHECK THAT THE DATA WRITTEN IS THE SAME
8076 7DF82E  81F8  139  139          BNEH      02,K,ERROR  AS THE DATA READ
8077 F9F8A9  81F8  140  140          BNEL      0A,PH,ERROR
8078 7DF8A9  81F8  141  141          BNEH      0A,PH,ERROR
8079 79F858  81F8  142  142          BNEL      05,PL,ERROR
807A FDF858  81F8  143  143          BNEH      05,PL,ERROR
807B 87800F        144  144          SR        ,           RETURN TO MAIN PROGRAM
                   145  145  *
                   146  146  *
807C 1D007C  807C  147  147  TEST3   LPI       *           PUT TEST START IN PC'S
807D 05800F        148  148          TPS       ,           PUT TEST START ON STACK
807E 20C03F        149  149          MVI       33,F0
807F 55E880  81E8  150  150          SB        TEST#       PRINT "TEST#3" ON CRT
8080 994FFF  1FFF  151  151          LPI       1FFF        PUT TEST ADDR END 8K IN PC'S
8081 81800F        152  152          TPA       ,00         PUT TEST ADDR. IN AUX 00
8082 D46580  8065  153  153          SB        CMTEST      GO TO WRITE, READ CONTROL MEM TEST
8083 5C8480  8084  154  154          B         TEST4       GO TO NEXT TEST
                   155  155  *
                   156  156  *
8084 9D0084  8084  157  157  TEST4   LPI       *           LOAD TEST START PC'S
8085 05800F        158  158          TPS       ,           PUT TEST START ON STACK
8086 A0C04F        159  159          MVI       34,F0
8087 55E880  81E8  160  160          SB        TEST#       PRINT "TEST#4" ON CRT
8088 998FFF  2FFF  161  161          LPI       2FFF        LOAD PC'S WITH LAST MEM ADDR 12K
8089 81800F        162  162          TPA       ,00         PUT TEST ADDR IN AUX 00
808A D46580  8065  163  163          SB        CMTEST
808B DC8C80  808C  164  164          B         TEST5
                   165  165  *
                   166  166  *
808C 1D008C  808C  167  167  TEST5   LPI       *           PUT TEST START IN PC'S
808D 05800F        168  168          TPS       ,           PUT START ADDR ON STACK
808E 20C05F        169  169          MVI       35,F0
808F 55E880  81E8  170  170          SB        TEST#       PRINT TEST#5 ON CRT
8090 19CFFF  3FFF  171  171          LPI       3FFF        LOAD PC'S WITH LAST ADDR. 16K
8091 81800F        172  172          TPA       ,00         PUT TEST ADDR. IN AUX 00
8092 D46580  8065  173  173          SB        CMTEST
8093 DC9480  8094  174  174          B         TEST6
                   175  175  *
                   176  176  *
                   177  177  *
8094 20C06F        178  178  TEST6   MVI       36,F0
8095 55E880  81E8  179  179          SB        TEST#       PRIINT TEST#6 ON CRT
8096 190000  0000  180  180          LPI       0000        CLEAR PC'S
```

```
8097 8202E8          181  181          MVX     PHPL,F3F2
8098 8204E2          182  182          MVX     F3F2,F5F4       CLEAR FILE REGISTERS
8099 A00E0F          183  183          MVI     00,K            CLEAR K REG.
809A 81800F          184  184          TPA     ,00             PUT PC'S IN AUX REG 0
809B 1D00A7 80A7     185  185 LOOPADD   LPI     INCRPL          LOAD PC'S WITH RETURN ADD FOR WCM
809C 01801F          186  186          TPA     ,01             PUT RETURN ADDR. IN AUX 1
809D 05800F          187  187          TPS     ,               PUSH RETURN ADDR ON STACK
809E 8B800F          188  188          TAP     ,00             GET CM ADDR. FOR WRITE
809F 05800F          189  189          TPS     ,               PUSH CONTROL MEMORY ADDR. ON STACK
80A0 A00E0F          190  190          MVI     00,K            CLEAR K REG. FOR WRITE OPERATION
80A1 200108          191  191          MV      PL,F1           SET UP F1 FOR MASK
80A2 280111          192  192          ANDI    1,F1,F1         MASK OFF ALL BUT BIT 0
80A3 78A501 80A5     193  193          BNEL    0,F1,*+2        SKIP IF BIT 0 = 1
80A4 A20E0E          194  194          ORI     80,K,K          SET PARITY BIT IF EVEN MEMORY ADDR.
80A5 A7CEFE          195  195          XORI    0FF,K,K         COMPLEMENT K BEFORE WRITE
80A6 078400          196  196          SR,WCM  ,               WRITE CONTROL MEMORY, GO TO INCRPL
80A7 AC0818          197  197 INCRPL   AI      1,PL,PL         INCREMENT PL FOR NEXT ADDR.
80A8 78AAF8 80AA     198  198          BNEL    0F,PL,*+2
80A9 74ACF8 80AC     199  199          BEQH    0F,PL,INCRPH    IF PL=FF, THEN GO TO INCREMENT PH
80AA 81800F          200  200          TPA     ,00             PUT INCREMENTED PC IN AUX 0
80AB DC9B80 809B     201  201          B       LOOPADD         EXECUTE NEXT WRITE
80AC AC0919          202  202 INCRPH   AI      1,PH,PH         INCR. PH
80AD D0B079 80B0     203  203          BER     F7,PH,*+3       IF THIS IS LAST ADDR. GO TO READ CM
80AE 81800F          204  204          TPA     ,00             PUT INCREMENTED PC IN AUX 0
80AF DC9B80 809B     205  205          B       LOOPADD         NEXT WRITE CYCLE
80B0 190000 0000     206  206          LPI     0000            CLR PC'S
80B1 8202E8          207  207          MVX     PHPL,F3F2       CLR F3 F2
80B2 A00E0F          208  208          MVI     00,K            CLR K
80B3 81800F          209  209          TPA     ,00             SET AUX 0 = TO PC'S
80B4 1D00BA 80BA     210  210 READCM   LPI     INCRK           LOAD PC'S WITH RETURN ADDR. FOR READ
80B5 01801F          211  211          TPA     ,01             PUT RETURN ADDR IN AUX 1
80B6 05800F          212  212          TPS     ,               PUSH ADDR ON STACK
80B7 8B800F          213  213          TAP     ,00             GET READ ADDR FROM AUX 0
80B8 05800F          214  214          TPS     ,               PUSH ADDR ON STACK
80B9 878600          215  215          SR,RCM  ,               READ CONTROL MEMORY
80BA 29CEFE          216  216 INCRK    ANDI    7F,K,K          MASK OFF PARITY BIT FROM K
80BB 78C90E 80C9     217  217          BNEL    0,K,ERROR0
80BC FCC90E 80C9     218  218          BNEH    0,K,ERROR0      PRINT ERROR ON CRT IF K NOT CLEAR
80BD 58C928 80C9     219  219          BNR     F2,PL,ERROR0    COMPARE COUNTER TO DATA READ
80BE 58C939 80C9     220  220          BNR     F3,PH,ERROR0    COMPARE COUNTER TO DATA READ
80BF AC0212          221  221          AI      1,F2,F2         INCREMENT F2
80C0 F8C2F2 80C2     222  222          BNEL    0F,F2,*+2       CHECK F2 NOT = F
80C1 74C3F2 80C3     223  223          BEQH    0F,F2,INCRF3    CHECK F2 = TO FF
80C2 5CC580 80C5     224  224          B       INCRPC
80C3 AC0313          225  225 INCRF3   AI      1,F3,F3         INCR F3
```

```
80C4 D0CB37   80CB   226   226          BER      F3,F7,TEST7
80C5 8208E2          227   227  INCRPC  MVX      F3F2,PHPL       INCR PC'S
80C6 81800F          228   228          TPA      ,00             PUT INCREMENTED PC'S IN AUX 0
80C7 5CB480   80B4   229   229          B        READCM
                     230   230  *
                     231   231  *
                     232   232  *
80C8 DCCB80   80CB   233   233          B        TEST7
80C9 D5F880   81F8   234   234  ERROR0  SB       ERROR
80CA DC9480   8094   235   235          B        TEST6
```

```
80CB A0C07F          1   237  TEST7   MVI     37,F0
80CC 55E880  81E8    2   238          SB      TEST#       PRINT TEST#7 ON CRT
80CD 19000F  000F    3   239  WRTF    LPI     000F        PUT "F" IN PC'S
80CE 81800F          4   240          TPA     ,00         PUT "F" IN AUX 0
80CF 9D010D  810D    5   241          LPI     RETURN1     PUT RETURN ADDRESS IN PC'S
80D0 01801F          6   242          TPA     ,01         PUT ADDRESS IN AUX 1
80D1 1D00D5  80D5    7   243          LPI     WRTFF       LOAD PC'S WITH NEXT ROUTINE ADDR.
80D2 01802F          8   244          TPA     ,02         PUT ADDR. IN AUX 2
80D3 54FD80  80FD    9   245          SB      INSTACK     INITIAL THREE STACK LOCATIONS FOR TEST
80D4 5D0880  8108   10   246          B       WRTNOP      GO TO WRITE CONTROL MEM. WITH NOP
                    11   247  *
                    12   248  *
80D5 1900FF  00FF   13   249  WRTFF   LPI     00FF        PUT "FF" IN PC'S
80D6 81800F         14   250          TPA     ,00         PUT ADDR. ("FF") IN AUX 0
80D7 9D010D  810D   15   251          LPI     RETURN1     PUT RETURN ADDR. IN PC'S
80D8 01801F         16   252          TPA     ,01         PUT ADDR. IN AUX 1
80D9 9D00DD  80DD   17   253          LPI     WRTFFF      NEXT ROUTINE ADDR. IN PC'S
80DA 01802F         18   254          TPA     ,02         PUT ADDR. IN AUX 2
80DB 54FD80  80FD   19   255          SB      INSTACK     SET UP THREE ADDRS. ON STACK
80DC 5D0880  8108   20   256          B       WRTNOP      WRITE CONTROL MEM WITH NOP
                    21   257  *
                    22   258  *
80DD 190FFF  0FFF   23   259  WRTFFF  LPI     0FFF        PUT "FFF" IN PC'S
80DE 81800F         24   260          TPA     ,00         PUT ADDR. IN AUX 0
80DF 9D010D  810D   25   261          LPI     RETURN1     SET UP RETURN ADDR. IN PC'S
80E0 01801F         26   262          TPA     ,01         PUT RETURN ADDR. IN AUX 1
80E1 1D00E5  80E5   27   263          LPI     WRT10       PUT NEXT ROUTINE ADDR. IN PC'S
80E2 01802F         28   264          TPA     ,02         PUT ADDR. IN AUX 2
80E3 54FD80  80FD   29   265          SB      INSTACK     INITIALIZE 3 STACK LOCATIONS
80E4 5D0880  8108   30   266          B       WRTNOP      WRITE NOP CONTROL MEM ADDR.
                    31   267  *
                    32   268  *
80E5 990010  0010   33   269  WRT10   LPI     0010        SET UP CONT MEM ADDR (10) IN PC'S
80E6 81800F         34   270          TPA     ,00         PUT ADDR. IN AUX 0
80E7 9D0113  8113   35   271          LPI     RETURN2     SET UP RETURN ADDR. IN PC,S
80E8 01801F         36   272          TPA     ,01         PUT ADDR. IN AUX 1
80E9 9D00ED  80ED   37   273          LPI     WRT100      SET NEXT ROUTINE ADDR. IN PC'S
80EA 01802F         38   274          TPA     ,02         PUT ADDR. IN AUX 2
80EB 54FD80  80FD   39   275          SB      INSTACK     INIT. 3 STACK ADDRS.
80EC 5D0E80  810E   40   276          B       WRTSB       WRITE SUBROUTINE BRANCH AT ADDR.
                    41   277  *
                    42   278  *
80ED 990100  0100   43   279  WRT100  LPI     0100        PUT 100 IN PC'S
80EE 81800F         44   280          TPA     ,00         PUT ADDR IN AUX 0
80EF 9D0113  8113   45   281          LPI     RETURN2     PUT RETURN ADDR. IN PC'S
```

```
                    91   327  *
8114 19000F  000F   92   328  RUNF    LPI     000F            PUT F IN PC'S
8115 05800F         93   329          TPS     ,               PUT F ON STACK
8116 190011  0011   94   330          LPI     0011            PUT SB ADDR. +1 IN PC'S
8117 0200E8         95   331          MVX     PHPL,F1F0       MOVE PC'S TO FILE REGS. FOR TEST
8118 87800F         96   332          SR      ,               EXECUTE INSTR. AT LOC."F" IN CM
                    97   333  *
                    98   334  *
8119 1900FF  00FF   99   335  RUNFF   LPI     00FF            LOAD PC'S "FF"
811A 05800F         100  336          TPS     ,               PUT "FF" ON STACK
811B 190101  0101   101  337          LPI     0101            PC'S = SB INST +1
811C 0200E8         102  338          MVX     PHPL,F1F0       MOVE PC'S TO FILE REGS.
811D 87800F         103  339          SR      ,               EXECUTE INSTR. AT LOC. "FF" IN CM
                    104  340  *
                    105  341  *
811E 190FFF  0FFF   106  342  RUNFFF  LPI     0FFF            PC'S = TO "FFF"
811F 05800F         107  343          TPS     ,               PUSH ADDR. ON STACK
8120 194001  1001   108  344          LPI     1001            PC'S = TO SB INSTR. +1
8121 0200E8         109  345          MVX     PHPL,F1F0       MOVE PC'S TO FILE REGS. FOR TEST
8122 87800F         110  346          SR      ,               EXECUTE INSTR. AT LOC. "FFF" IN CM
                    111  347  *
                    112  348  *
8123 8D800F         113  349  CHKTST  TSP     ,               POP ADDR. OF SUB BRANCH +1 OFF STACK
8124 D92A19  812A   114  350          BNR     F1,PH,ERROR3    CHECK F1=PH (RIGHT CM LOCATION ?)
8125 D92A08  812A   115  351          BNR     F0,PL,ERROR3    CHECK RIGHT LOCATION
8126 F12A08  812A   116  352          BEQL    0,PL,ERROR3     MAKE SURE ADDR.= XXX1
8127 F51918  8119   117  353          BEQH    1,PL,RUNFF      IF ADDR.= 0011 GO TO "RUNFF"
8128 711E19  811E   118  354          BEQL    1,PH,RUNFFF     IF ADDR.= 0101 GO TO "RUNFFF"
8129 5D2C80  812C   119  355          B       INIT8           IF NO ERROR AND ADDR. NOT = TO 0101 NEXT TEST
                    120  356  *
                    121  357  *
812A D5F880  81F8   122  358  ERROR3  SB      ERROR           PRINT ERROR ON CRT
812B DCCB80  80CB   123  359          B       TEST7           EXECUTE SAME TEST AGAIN
                    124  360  *
                    125  361  *
812C AC0616         126  362  INIT8   AI      1,F6,F6         STEP MAX ADDR. LOW ORDER BY 1
812D AC0717         127  363          AI      1,F7,F7         STEP MAX ADDR. HIGH ORDER BY 1
                    128  364  *
812E A0C08F         129  365  TEST8   MVI     38,F0
812F 55E880  81E8   130  366          SB      TEST#           PRINT TEST#8 ON CRT
8130 190000  0000   131  367          LPI     0000            LOAD PC'S = TO 0
8131 81800F         132  368          TPA     ,00             INIT ADDR. POINTER
8132 01804F         133  369          TPA     ,04             INIT ADDR. COUNTER
8133 8B800F         134  370  FLVCHK  TAP     ,00             GET ADDR. POINTER
8134 A00508         135  371          MV      PL,F5           PUT PC LOW IN FILE REG 5 FOR TEST
```

```
8135 280515        136  372          ANDI    01,F5,F5          MASK OFF ALL BUT BIT 0
8136 F13D15  813D  137  373          BEQL    01,F5,WRODD       IF ADDR. IS ODD GO WRITE ODD
                   138  374  *
                   139  375  *
8137 9D010D  810D  140  376  WREVEN  LPI     RETURN1           PUT RETURN ADDR. IN PC'S
8138 01801F        141  377          TPA     ,01               SAVE RETURN ADDR. IN AUX 1
8139 9D0143  8143  142  378          LPI     PCCNTL            LOAD PC'S NEXT ROUTINE ADDR.
813A 01802F        143  379          TPA     ,02               SAVE ADDR. IN AUX 2
813B 54FD80  80FD  144  380          SB      INSTACK           SET UP STACK FOR WRITE CONT. MEM.
813C 5D0880  8108  145  381          B       WRTNOP            EVEN ADDR. GO WRITE A NOP HERE
                   146  382  *
                   147  383  *
813D 9D0168  8168  148  384  WRODD   LPI     RETURN3           PUT RETURN ADDR. IN PC'S
813E 01801F        149  385          TPA     ,01               SAVE ADDR. AUX 1
813F 9D0143  8143  150  386          LPI     PCCNTL            LOAD PC'S WITH ADDR. OF NEXT ROUTINE
8140 01802F        151  387          TPA     ,02               SAVE ADDR. IN AUX 2
8141 54FD80  80FD  152  388          SB      INSTACK           SET UP STACK FOR WRITE CONT. MEM.
8142 DD6380  8163  153  389          B       WRTSB1            ODD ADDR. GO WRITE SUB BRANCH "TSTADD"
                   154  390  *
                   155  391  *
8143 8B800F        156  392  PCCNTL  TAP     ,00               GET ADDR. POINTER
8144 000FEF        157  393          OR      +,,               INCREMENT POINTER BY 1
8145 0200E8        158  394          MVX     PHPL,F1F0         MOVE PC'S TO FILE REGS FOR TEST
8146 81800F        159  395          TPA     ,00               SAVE UPDATED ADDR. POINTER
8147 D93306  8133  160  396          BNR     F0,F6,FLVCHK      CHECK TO SEE IF ADDR. = "00"
8148 D14A17  814A  161  397          BER     F1,F7,CMEXEC      SEE IF PC'S = TO LAST VALID CM ADDR.
8149 DD3380  8133  162  398          B       FLVCHK            GO CHECK ODD OR EVEN THEN WRITE
                   163  399  *
                   164  400  *
814A 190000  0000  165  401  CMEXEC  LPI     0000              LOAD PC'S TO = 0
814B 05800F        166  402          TPS     ,                 PUSH 0 ON THE STACK
814C 87800F        167  403          SR      ,                 START PROGRAM EXECUTION AT LOCATION 0
                   168  404  *
                   169  405  *
814D 0B804F        170  406  CMCONT  TAP     ,04               PUT ADDR. COUNTER IN PC'S
814E 000FEF        171  407          OR      +,,               STEP COUNT BY 1
814F 000FEF        172  408          OR      +,,               STEP COUNT BY 1 (TOTAL STEPS = 2)
8150 01804F        173  409          TPA     ,04               SAVE UPDATED COUNT
8151 8B800F        174  410          TAP     ,00               GET NEXT CM ADDR.
8152 05800F        175  411          TPS     ,                 PUSH ADDR. ON STACK
8153 87800F        176  412          SR      ,                 POP STACK START EXECUTION AT CM ADDR.
                   177  413  *
                   178  414  *
8154 8D800F        179  415  TSTADD  TSP     ,                 SAVE UPDATED CM ADDR. RESULTING FROM SB
8155 81800F        180  416          TPA     ,00               STORE ADDR.. AUX 0
```

```
8156 0B804F        181  417          TAP      ,04              GET ADDR. COUNTER
8157 8202E8        182  418          MVX      PHPL,F3F2        MOVE ADDR. COUNTER TO FILE REGS
8158 8B800F        183  419          TAP      ,00              GET UPDATED CM ADDR.
8159 800FFF        184  420          OR       -,,              DECREMENT ADDR. POINTER BY 1
815A 800FFF        185  421          OR       -,,              DECREMENT ADDR. POINTER BY 1 (TOTAL -2)
815B 596139  8161  186  422          BNR      F3,PH,ERROR4     COMPARE TO SEE IF COUNT = TO ADDR.
815C 596128  8161  187  423          BNR      F2,PL,ERROR4     COMPARE TO SEE IF COUNT = TO ADDR.
                   188  424  *
                   189  425  *
815D 8B800F        190  426  TSTEND   TAP      ,00              GET UPDATED CM ADDR. PUT ADDR. IN PC'S
815E 594D68  814D  191  427          BNR      F6,PL,CMCONT     LAST CONT MEM ADDR. LOW ORDER?
815F D16979  8169  192  428          BER      F7,PH,RESMAX     LAST CONT MEM ADDR HIGH ORDER?
8160 DD4D80  814D  193  429          B        CMCONT           NOT LAST ADDR. GO EXECUTE AGAIN
                   194  430  *
                   195  431  *
8161 D5F880  81F8  196  432  ERROR4  SB       ERROR
8162 DD2E80  812E  197  433          B        TEST8
                   198  434  *
                   199  435  *
8163 234E5F        200  436  WRTSB1  MVI      0D5,K            PUT OP CODE FOR SUB BRANCH INSTR IN K
8164 A7CEFE        201  437          XORI     0FF,K,K          COMPLEMENT K FOR CONTROL MEM WRITE
8165 22080F        202  438          MVI      80,PL            PUT REMAINDER OF INST CODE IN PC,S
8166 A1494F        203  439          MVI      54,PH
8167 078400        204  440          SR,WCM   ,                WRITE SUB BRANCH AT CM ADDR.
8168 87800F        205  441  RETURN3  SR       ,                RETURN
                   206  442  *
                   207  443  *
8169 A7C6F6        208  444  RESMAX   XORI     0FF,F6,F6        COMPLEMENT F6 SHOULD BE = TO "FF"
816A F961F6  8161  209  445          BNEL     0F,F6,ERROR4     IS LOW ORDER F6 = TO "F"
816B 7D61F6  8161  210  446          BNEH     0F,F6,ERROR4     IS HIGH ORDER F6 = TO "F"
816C 757247  8172  211  447          BEQH     4,F7,COMPM       CHECK TO SEE IF F7 = TO "04" (16K MEM)
816D FD6157  8161  212  448          BNEH     5,F7,ERROR4      IF F7 NOT = TO "05" (20K MEM) ERROR
816E 2447F7        213  449          XORI     1F,F7,F7         F7 = TO "05" CHANGE TO "4F"
816F 7961F7  8161  214  450          BNEL     0F,F7,ERROR4     IS LOW ORDER F7 = TO "F"
8170 7D6147  8161  215  451          BNEH     4,F7,ERROR4      IS HIGH ORDER F7 = TO "4"
8171 5D7580  8175  216  452          B        *+4              RESTART PROGRAM
8172 25C7F7        217  453  COMPM    XORI     7F,F7,F7         F7 = TO "04" CHANGE TO "3F"
8173 7961F7  8161  218  454          BNEL     0F,F7,ERROR4     IS F7 LOW ORDER = TO "F"
8174 FD6137  8161  219  455          BNEH     3,F7,ERROR4      IS F7 HIGH ORDER = TO "3"
8175 5D7680  8176  220  456          B        TEST9
                   221  457  *
                   222  458  *
```

```
                    1    460  *
                    2    461  *TEST READ WRITE DATA MEMORY (16K)
                    3    462  *
8176 20C09F         4    463  TEST9    MVI      39,F0            PUT A "9" IN F0
8177 55E880  81E8   5    464           SB       TEST#            PRINT "TEST#9" ON CRT
8178 2140AF         6    465           MVI      5A,F0            PUT WRITE PATTERN IN F0 "5A"
8179 A2815F         7    466           MVI      0A5,F1           PUT WRITE PATTERN 2 IN F1 "A5"
817A 190000  0000   8    467           LPI      0000             SET PC'S TO POINT AT FIRST MEMORY LOCATION
817B 558280  8182   9    468           SB       WRTDM            GO WRITE 16K OF DATA MEMORY
817C 190000  0000   10   469           LPI      0000             SET PC'S TO POINT TO FIRST MEMORY LOCATION
817D 567880  8278   11   470           SB       SEC1             WAIT FOR APP. 1 SECOND BEFORE READ
817E 2140AF         12   471           MVI      5A,F0
817F A2815F         13   472           MVI      0A5,F1
8180 D58A80  818A   14   473           SB       READDM           GO READ AND CHECK 16K OF DATA MEMORY
8181 5D9880  8198   15   474           B        TESTA
                    16   475  *
                    17   476  *
8182 802FE0         18   477  WRTDM    OR,W1    +,F0,            WRITE FIRST LOCATION WITH "5A" STEP PC'S
8183 002FE1         19   478           OR,W1    +,F1,            WRITE NEXT LOCATION WITH "A5" STEP PC'S
8184 F98208  8182   20   479           BNEL     0,PL,WRTDM       CHECK TO SEE IF THIS IS LAST ADDR. IN 16K
8185 7D8208  8182   21   480           BNEH     0,PL,WRTDM
8186 798209  8182   22   481           BNEL     0,PH,WRTDM
8187 758949  8189   23   482           BEQH     4,PH,*+2
8188 DD8280  8182   24   483           B        WRTDM            IF NOT THE LAST ADDR. GO WRITE
8189 87800F         25   484           SR       ,                THIS IS THE LAST ADDR. GO READ AND CHECK
                    26   485  *
                    27   486  *
818A 801FEF         28   487  READDM   OR,R     +,,              READ THE FIRST DATA MEMORY LOCATION STEP PC'S
818B A00E0A         29   488           MV       CL,K             DISPLAY DATA MEMORY
818C 59961A  8196   30   489           BNR      F1,CL,ERROR9     READ DATA SHOULD = F1 "A5"
818D 200E0B         31   490           MV       CH,K             DISPLAY DATA MEMORY
818E 59960B  8196   32   491           BNR      F0,CH,ERROR9     READ DATA SHOULD = F0 "5A"
818F 000FEF         33   492           OR       +,,              STEP PC'S TO POINT TO NEXT EVEN LOCATION
8190 798A08  818A   34   493           BNEL     0,PL,READDM      CHECK TO SEE IF LAST LOC. IN 16K OF DM
8191 FD8A08  818A   35   494           BNEH     0,PL,READDM
8192 F98A09  818A   36   495           BNEL     0,PH,READDM
8193 F59549  8195   37   496           BEQH     4,PH,*+2
8194 5D8A80  818A   38   497           B        READDM           IF NOT LAST LOC. NEXT READ AND CHECK
8195 87800F         39   498           SR       ,                LAST ADDR. IN 16K OF DATA MEM., NEXT TEST
                    40   499  *
                    41   500  *
8196 D5F880  81F8   42   501  ERROR9   SB       ERROR            PRINT "ERROR" ON CRT
8197 5D7680  8176   43   502           B        TEST9            RESTART SAME TEST
                    44   503  *
                    45   504  *TEST VARIOUS BRANCH INST. AND +- PC'S
```

```
                        46   505  *
8198 21001F             47   506  TESTA    MVI       41,F0              PUT "A" IN F0
8199 55E880   81E8        48   507           SB        TEST#              PRINT "TEST# A" ON CRT
819A 20000F             49   508           MVI       00,F0              CLEAR F0=0
819B A3C1FF             50   509           MVI       0FF,F1             SET F1 = ALL 1'S
819C 190000   0000      51   510           LPI       0000               CLEAR PC'S
819D D19FE0   819F      52   511           BER       +,F0,*+2           SHOULD BRANCH INCREMENT PC'S
819E 5DC480   81C4      53   512           B         ERRORA             ERROR IF NO BRANCH
819F F9C418   81C4      54   513           BNEL      1,PL,ERRORA        PC SHOULD = 0001
81A0 D9A2F1   81A2      55   514           BNR       -,F1,*+2           SHOULD BRANCH DECREMENT PC'S
81A1 5DC480   81C4      56   515           B         ERRORA             ERROR IF NO BRANCH
81A2 79C408   81C4      57   516           BNEL      0,PL,ERRORA        PC SHOULD = 0000
81A3 190000   0000      58   517           LPI       0000               SET PC'S TO POINT TO MEMORY LOCATION 0000
81A4 802FE0             59   518           OR,W1     +,F0,              WRITE  "00" IN LOCATION 0000
81A5 002FF0             60   519           OR,W1     -,F0,              WRITE  "00" IN LOCATION 0001
81A6 801FEF             61   520           OR,R      +,,                READ MEMORY LOC 0000, 0001 INTO CH CL
81A7 190000   0000      62   521           LPI       0000               SET PC'S = 0000
81A8 51AAC0   81AA      63   522           BER       CL+,F0,*+2         SHOULD BRANCH AND INCREMENT PC'S
81A9 5DC480   81C4      64   523           B         ERRORA             ERROR IF NO BRANCH
81AA F9C418   81C4      65   524           BNEL      1,PL,ERRORA        PC SHOULD = 0001
81AB 51AD80   81AD      66   525           BER       CL-,F0,*+2         SHOULD BRANCH AND DECREMENT PC'S
81AC 5DC480   81C4      67   526           B         ERRORA             ERROR IF NO BRANCH
81AD 79C408   81C4      68   527           BNEL      0,PL,ERRORA        PC SHOULD = 0000
81AE 23C3FF             69   528           MVI       0FF,F3             SET F3 = ALL 1'S
81AF 61B1F3   81B1      70   529           BTL       0F,F3,*+2          SHOULD BRANCH (F3 = FF)
81B0 5DC480   81C4      71   530           B         ERRORA             ERROR IF NO BRANCH OCCURRED
81B1 69B303   81B3      72   531           BFL       00,F3,*+2          SHOULD BRANCH F3 NOT = 0
81B2 5DC480   81C4      73   532           B         ERRORA             ERROR IF NO BRANCH
81B3 71B5F3   81B5      74   533           BEQL      0F,F3,*+2          SHOULD BRANCH F3 = TO FF
81B4 5DC480   81C4      75   534           B         ERRORA             ERROR IF NO BRANCH
81B5 79B703   81B7      76   535           BNEL      00,F3,*+2          SHOULD BRANCH F3 NOT = TO 0
81B6 5DC480   81C4      77   536           B         ERRORA             ERROR IF NO BRANCH
81B7 23C0FF             78   537           MVI       0FF,F0             PUT "FF" IN F0
81B8 A00100             79   538           MV        F0,F1              PUT "FF" IN F1
81B9 20021F             80   539           MVI       01,F2              PUT "1" IN F2
81BA 20030F             81   540           MVI       00,F3              PUT "0" IN F3
81BB 41BD20   81BD      82   541           BLR       F2,F0,*+2          SHOULD BRANCH "1" LESS THAN "FF"
81BC 5DC480   81C4      83   542           B         ERRORA             ERROR IF NO BRANCH
81BD 45BF20   81BF      84   543           BLRX      F3F2,F1F0,*+2      SHOULD BRANCH "1" LESS THAN "FF"
81BE 5DC480   81C4      85   544           B         ERRORA             ERROR IF NO BRANCH
81BF 51C101   81C1      86   545           BER       F0,F1,*+2          SHOULD BRANCH F0 = TO F1
81C0 5DC480   81C4      87   546           B         ERRORA             ERROR IF NO BRANCH
81C1 567880   8278      88   547           SB        SEC1
81C2 55E280   81E2      89   548           SB        INITCRT
81C3 5C3F80   803F      90   549           B         TEST0
```

```
                        91    550  *
                        92    551  *
81C4 D5F880    81F8     93    552  ERRORA   SB       ERROR          PRINT ERROR ON CRT
81C5 5D9880    8198     94    553           B        TESTA          REPEAT TEST
                        95    554  *
```

```
81C6 A00E5F        1   556 ENABLE5  MVI   5,K          SET CRT ADDR.
81C7 178C00        2   557 ENABLE    CIO   0C0          ADDR. STROBE
81C8 55C980  81C9  3   558 DELAY10   SB    DELAY5
81C9 55CA80  81CA  4   559 DELAY5    SB    *+1
81CA D5CB80  81CB  5   560           SB    *+1
81CB 200F0F        6   561           NOP   ,
81CC 200F0F        7   562           NOP   ,
81CD 87800F        8   563           SR    ,
                   9   564 *
                  10   565 *
81CE 69CE8D  81CE 11   566 OBSTROBE BFL   8,SH,*       WAIT FOR DEVIE READY
81CF 55C980  81C9 12   567           SB    DELAY5
81D0 978200       13   568           CIO   20
81D1 DDC980  81C9 14   569           B     DELAY5
                  15   570 *
                  16   571 *
81D2 A00DBD       17   572 INIT      ORI   0B,SH,SH     INHIBIT INPUT
81D3 284DFD       18   573           ANDI  1F,SH,SH     SET NO TRAP,40BIT,HALT/STEP OFF
81D4 55C680  81C6 19   574           SB    ENABLE5
81D5 A00E3F       20   575           MVI   03,K
81D6 D5CE80  81CE 21   576           SB    OBSTROBE     CLEAR CRT
81D7 87800F       22   577           SR    ,
                  23   578 *
                  24   579 *
81D8 55C680  81C6 25   580 CRLF      SB    ENABLE5      THIS ROUTINE PRINTS CARRIAGE RETURN LINE FEED
81D9 200EDF       26   581           MVI   0D,K
81DA D5CE80  81CE 27   582           SB    OBSTROBE     CR
81DB A00EAF       28   583           MVI   0A,K
81DC D5CE80  81CE 29   584           SB    OBSTROBE     LF
81DD 87800F       30   585           SR    ,
                  31   586 *
                  32   587 *
81DE 55C680  81C6 33   588 SPACE     SB    ENABLE5
81DF 208E0F       34   589           MVI   20,K
81E0 D5CE80  81CE 35   590           SB    OBSTROBE     PRINT A SPACE
81E1 87800F       36   591           SR    ,
                  37   592 *
                  38   593 *
81E2 55C680  81C6 39   594 INITCRT   SB    ENABLE5
81E3 200E1F       40   595           MVI   01,K
81E4 D5CE80  81CE 41   596           SB    OBSTROBE     HOME CUSOR
81E5 A00E3F       42   597           MVI   03,K
81E6 D5CE80  81CE 43   598           SB    OBSTROBE     CLR CRT
81E7 87800F       44   599           SR    ,
                  45   600 *
```

```
                        46   601  *
81E8 55C680   81C6      47   602  TEST#    SB       ENABLE5
81E9 214E4F             48   603           MVI      54,K
81EA D5CE80   81CE      49   604           SB       OBSTROBE      T
81EB 210E5F             50   605           MVI      45,K
81EC D5CE80   81CE      51   606           SB       OBSTROBE      E
81ED A14E3F             52   607           MVI      53,K
81EE D5CE80   81CE      53   608           SB       OBSTROBE      S
81EF 214E4F             54   609           MVI      54,K
81F0 D5CE80   81CE      55   610           SB       OBSTROBE      T
81F1 208E3F             56   611           MVI      23,K
81F2 D5CE80   81CE      57   612           SB       OBSTROBE      #
81F3 55DE80   81DE      58   613           SB       SPACE         SPACE
81F4 560680   8206      59   614           SB       PRTBYTE       TEST NUMBER X IS PRINTED
81F5 200F0F             60   615           NOP      ,
81F6 55D880   81D8      61   616           SB       CRLF
81F7 87800F             62   617           SR       ,
                        63   618  *
                        64   619  *
81F8 55C680   81C6      65   620  ERROR    SB       ENABLE5
81F9 210E5F             66   621           MVI      45,K
81FA D5CE80   81CE      67   622           SB       OBSTROBE      E
81FB 214E2F             68   623           MVI      52,K
81FC D5CE80   81CE      69   624           SB       OBSTROBE      R
81FD 214E2F             70   625           MVI      52,K
81FE D5CE80   81CE      71   626           SB       OBSTROBE      R
81FF 210EFF             72   627           MVI      4F,K
8200 D5CE80   81CE      73   628           SB       OBSTROBE      O
8201 214E2F             74   629           MVI      52,K
8202 D5CE80   81CE      75   630           SB       OBSTROBE      R
8203 55D880   81D8      76   631           SB       CRLF
8204 567880   8278      77   632           SB       SEC1          WAIT APP. ONE SECOND FOR DISPLAY PURPOSES
8205 87800F             78   633           SR       ,
                        79   634  *
8206 55C680   81C6      80   635  PRTBYTE  SB       ENABLE5
8207 A00E00             81   636           MV       F0,K          THIS IS A ROUTINE TO PRINT FILE REG 0
8208 D5CE80   81CE      82   637           SB       OBSTROBE
8209 87800F             83   638           SR       ,
                        84   639  *
820A 21430F             85   640  SYSPE    MVI      C'P',F3       THESE ROUTINES PRINT PARITY ERROR ADDR.
820B 21025F             86   641  SYSVE    MVI      C'E',F2
820C 2100DF             87   642           MVI      C'M',F0
820D 55C680   81C6      88   643  SYSERROR SB       ENABLE5
820E 200E1F             89   644           MVI      01,K
820F D5CE80   81CE      90   645           SB       OBSTROBE
```

```
8210 208E0F          91    646            MVI       20,K
8211 A0040F          92    647            MVI       00,F4
8212 D5CE80  81CE    93    648  SPACEOUT  SB        OBSTROBE
8213 98C44F          94    649            AC,,1     F4,,F4
8214 7E1254  8212    95    650            BNEH      5,F4,SPACEOUT
8215 200E1F          96    651            MVI       01,K
8216 D5CE80  81CE    97    652            SB        OBSTROBE
8217 208EAF          98    653            MVI       C'*',K
8218 D5CE80  81CE    99    654            SB        OBSTROBE
8219 D5CE80  81CE   100    655            SB        OBSTROBE
821A D5CE80  81CE   101    656            SB        OBSTROBE
821B 55DE80  81DE   102    657            SB        SPACE
821C A14E3F         103    658            MVI       C'S',K
821D D5CE80  81CE   104    659            SB        OBSTROBE
821E A14E9F         105    660            MVI       C'Y',K
821F D5CE80  81CE   106    661            SB        OBSTROBE
8220 A14E3F         107    662            MVI       C'S',K
8221 D5CE80  81CE   108    663            SB        OBSTROBE
8222 214E4F         109    664            MVI       C'T',K
8223 D5CE80  81CE   110    665            SB        OBSTROBE
8224 210E5F         111    666            MVI       C'E',K
8225 D5CE80  81CE   112    667            SB        OBSTROBE
8226 A10EDF         113    668            MVI       C'M',K
8227 D5CE80  81CE   114    669            SB        OBSTROBE
8228 55DE80  81DE   115    670            SB        SPACE
8229 210E5F         116    671            MVI       C'E',K
822A D5CE80  81CE   117    672            SB        OBSTROBE
822B 214E2F         118    673            MVI       C'R',K
822C D5CE80  81CE   119    674            SB        OBSTROBE
822D 214E2F         120    675            MVI       C'R',K
822E D5CE80  81CE   121    676            SB        OBSTROBE
822F 210EFF         122    677            MVI       C'O',K
8230 D5CE80  81CE   123    678            SB        OBSTROBE
8231 214E2F         124    679            MVI       C'R',K
8232 D5CE80  81CE   125    680            SB        OBSTROBE
8233 55DE80  81DE   126    681            SB        SPACE
8234 A08E8F         127    682            MVI       C'(',K
8235 D5CE80  81CE   128    683            SB        OBSTROBE
8236 A00E03         129    684            MV        F3,K
8237 D5CE80  81CE   130    685            SB        OBSTROBE
8238 200E02         131    686            MV        F2,K
8239 D5CE80  81CE   132    687            SB        OBSTROBE
823A 200E01         133    688            MV        F1,K
823B D5CE80  81CE   134    689            SB        OBSTROBE
823C A00E00         135    690            MV        F0,K
```

```
823D D5CE80   81CE   136   691           SB       OBSTROBE
823E 55DE80   81DE   137   692           SB       SPACE
823F 200009         138   693           MV       PH,F0
8240 D64F80   824F   139   694           SB       PRTBITE
8241 A00008         140   695           MV       PL,F0
8242 D64F80   824F   141   696           SB       PRTBITE
8243 208E9F         142   697           MVI      C')',K
8244 D5CE80   81CE   143   698           SB       OBSTROBE
8245 55DE80   81DE   144   699           SB       SPACE
8246 208EAF         145   700           MVI      C'*',K
8247 D5CE80   81CE   146   701           SB       OBSTROBE
8248 D5CE80   81CE   147   702           SB       OBSTROBE
8249 D5CE80   81CE   148   703           SB       OBSTROBE
824A DDD880   81D8   149   704           B        CRLF
                    150   705   *
                    151   706   *
824B 4A4D21   824D   152   707   PRTBIT1 BLER     F2,F1,*+2       THESE ROUTINES PRINT CHARS.STORED IN F0
824C AC0272         153   708           AI       07,F2,F2
824D 200E02         154   709           MV       F2,K
824E 5DCE80   81CE   155   710           B        OBSTROBE
824F A0C19F         156   711   PRTBITE MVI      39,F1
8250 20C30F         157   712           MVI      30,F3
8251 8C4203         158   713           SHHH     F0,F3,F2
8252 564B80   824B   159   714           SB       PRTBIT1
8253 084203         160   715           SHHL     F0,F3,F2
8254 DE4B80   824B   161   716           B        PRTBIT1
                    162   717   *
                    163   718   *
8255 8D800F         164   719   PE24    TSP      ,               FAILING ADDR TO PC'S
8256 800FFF         165   720           OR       -,,             DECREMENT ADDR BY 1
8257 0200E8         166   721           MVX      PHPL,F1F0       PUT PC'S IN FILE REGS. FOR COMPARE
8258 1D02DE   82DE   167   722           LPI      BDPARY          LOAD PC'S WITH BAD PARITY LOCATION ADDR.
8259 DA5E19   825E   168   723           BNR      F1,PH,*+5       BRANCH IF THIS IS A REAL PARITY ERROR
825A 52DF08   82DF   169   724           BER      F0,PL,BDPARY1   EXECUTE SECOND CMPE TEST
825B 9D02DF   82DF   170   725           LPI      BDPARY1         LOAD PC'S WITH BAD PARITY LOCATION # 2
825C DA5E19   825E   171   726           BNR      F1,PH,*+2       BRANCH IF THIS SI A REAL PARITY ERROR
825D 52DA08   82DA   172   727           BER      F0,PL,RETURN    PARITY TEST OK SET UP TO RETURN TO PROG.
825E 0208E0         173   728           MVX      F1F0,PHPL       PUT PC'S BACK FOR CMPE PRINT OUT
825F 21013F         174   729           MVI      C'C',F1         SET UP F1 TO PRINT OUT "PECM"
8260 560A80   820A   175   730           SB       SYSPE           PRINT OUT CONTROL MEM PARITY ERROR INFO
8261 5E6180   8261   176   731           B        *               HANG HERE UNTIL RESET IS STRUCK
                    177   732   *
                    178   733   *
8262 8D800F         179   734   PE8     TSP      ,               FAILING ADDR. TO PC'S
8263 800FFF         180   735           OR       -,,             DECREMENT ADDR.
```

```
8264 A1014F         181   736          MVI     C'D',F1          SET UP F1 TO PRINT OUT "PEDM"
8265 560A80  820A   182   737          SB      SYSPE            PRINT OUT DATA MEMORY PARITY ERROR
8266 DE6680  8266   183   738          B       *                HANG HERE WAITING FOR RESET TO BE STRUCK
                    184   739  *
                    185   740  *
8267 A14EAF         186   741  FILCHK  MVI     5A,K             THIS ROUTINE CHECKS WORST CASE PATT. UNCHAGED
8268 D9F80E  81F8   187   742          BNR     F0,K,ERROR
8269 A28E5F         188   743          MVI     0A5,K
826A 59F81E  81F8   189   744          BNR     F1,K,ERROR
826B A14EAF         190   745          MVI     5A,K
826C 59F82E  81F8   191   746          BNR     F2,K,ERROR
826D A14E5F         192   747          MVI     55,K
826E D9F83E  81F8   193   748          BNR     F3,K,ERROR
826F A28EAF         194   749          MVI     0AA,K
8270 59F84E  81F8   195   750          BNR     F4,K,ERROR
8271 A28E5F         196   751          MVI     0A5,K
8272 D9F85E  81F8   197   752          BNR     F5,K,ERROR
8273 A14EAF         198   753          MVI     5A,K
8274 D9F86E  81F8   199   754          BNR     F6,K,ERROR
8275 A28E5F         200   755          MVI     0A5,K
8276 59F87E  81F8   201   756          BNR     F7,K,ERROR
8277 87800F         202   757          SR      ,
                    203   758  *
                    204   759  *
8278 20000F         205   760  SEC1    MVI     00,F0            THIS ROUTINE FORCES A SERIES OF INST. DELAYS
8279 A00E0F         206   761  RESTRT  MVI     00,K             APP=TO ONE SECOND
827A AC0E1E         207   762  WAIT1   AI      01,K,K
827B FA7D9E  827D   208   763          BNEL    09,K,*+2
827C 767FCE  827F   209   764          BEQH    0C,K,*+3
827D 568480  8284   210   765          SB      DELAY50
827E 5E7A80  827A   211   766          B       WAIT1
827F AC0010         212   767  WAIT2   AI      1,F0,F0
8280 FA82B0  8282   213   768          BNEL    0B,F0,*+2
8281 768310  8283   214   769          BEQH    1,F0,*+2
8282 5E7980  8279   215   770          B       RESTRT
8283 87800F         216   771          SR      ,
8284 D5C880  81C8   217   772  DELAY50 SB      DELAY10
8285 D5C880  81C8   218   773          SB      DELAY10
8286 D5C880  81C8   219   774          SB      DELAY10
8287 D5C880  81C8   220   775          SB      DELAY10
8288 D5C880  81C8   221   776          SB      DELAY10
8289 87800F         222   777          SR      ,
                    223   778  *
                    224   779  *
                    225   780  *
```

```
                        226   781  *
828A 55C680   81C6      227   782  CMAMNT   SB        ENABLE5
828B 210E9F             228   783           MVI       C'I',K
828C D5CE80   81CE      229   784           SB        OBSTROBE
828D A10EEF             230   785           MVI       C'N',K
828E D5CE80   81CE      231   786           SB        OBSTROBE
828F A14E0F             232   787           MVI       C'P',K
8290 D5CE80   81CE      233   788           SB        OBSTROBE
8291 A14E5F             234   789           MVI       C'U',K
8292 D5CE80   81CE      235   790           SB        OBSTROBE
8293 214E4F             236   791           MVI       C'T',K
8294 D5CE80   81CE      237   792           SB        OBSTROBE
8295 55DE80   81DE      238   793           SB        SPACE
8296 A10E1F             239   794           MVI       C'A',K
8297 D5CE80   81CE      240   795           SB        OBSTROBE
8298 A10EDF             241   796           MVI       C'M',K
8299 D5CE80   81CE      242   797           SB        OBSTROBE
829A 210EFF             243   798           MVI       C'O',K
829B D5CE80   81CE      244   799           SB        OBSTROBE
829C A14E5F             245   800           MVI       C'U',K
829D D5CE80   81CE      246   801           SB        OBSTROBE
829E A10EEF             247   802           MVI       C'N',K
829F D5CE80   81CE      248   803           SB        OBSTROBE
82A0 214E4F             249   804           MVI       C'T',K
82A1 D5CE80   81CE      250   805           SB        OBSTROBE
82A2 55DE80   81DE      251   806           SB        SPACE
82A3 210EFF             252   807           MVI       C'O',K
82A4 D5CE80   81CE      253   808           SB        OBSTROBE
82A5 210E6F             254   809           MVI       C'F',K
82A6 D5CE80   81CE      255   810           SB        OBSTROBE
82A7 55DE80   81DE      256   811           SB        SPACE
82A8 210E3F             257   812           MVI       C'C',K
82A9 D5CE80   81CE      258   813           SB        OBSTROBE
82AA 210EFF             259   814           MVI       C'O',K
82AB D5CE80   81CE      260   815           SB        OBSTROBE
82AC A10EEF             261   816           MVI       C'N',K
82AD D5CE80   81CE      262   817           SB        OBSTROBE
82AE 214E4F             263   818           MVI       C'T',K
82AF D5CE80   81CE      264   819           SB        OBSTROBE
82B0 214E2F             265   820           MVI       C'R',K
82B1 D5CE80   81CE      266   821           SB        OBSTROBE
82B2 210EFF             267   822           MVI       C'O',K
82B3 D5CE80   81CE      268   823           SB        OBSTROBE
82B4 210ECF             269   824           MVI       C'L',K
82B5 D5CE80   81CE      270   825           SB        OBSTROBE
```

```
82B6 55DE80   81DE   271   826          SB       SPACE
82B7 A10EDF          272   827          MVI      C'M',K
82B8 D5CE80   81CE   273   828          SB       OBSTROBE
82B9 210E5F          274   829          MVI      C'E',K
82BA D5CE80   81CE   275   830          SB       OBSTROBE
82BB A10EDF          276   831          MVI      C'M',K
82BC D5CE80   81CE   277   832          SB       OBSTROBE
82BD A08EEF          278   833          MVI      C'.',K
82BE 55DE80   81DE   279   834          SB       SPACE
82BF A08E8F          280   835          MVI      C'(',K
82C0 D5CE80   81CE   281   836          SB       OBSTROBE
82C1 A0CE0F          282   837          MVI      C'0',K
82C2 D5CE80   81CE   283   838          SB       OBSTROBE
82C3 20CEDF          284   839          MVI      C'=',K
82C4 D5CE80   81CE   285   840          SB       OBSTROBE
82C5 20CE1F          286   841          MVI      C'1',K
82C6 D5CE80   81CE   287   842          SB       OBSTROBE
82C7 A0CE6F          288   843          MVI      C'6',K
82C8 D5CE80   81CE   289   844          SB       OBSTROBE
82C9 A10EBF          290   845          MVI      C'K',K
82CA D5CE80   81CE   291   846          SB       OBSTROBE
82CB 55DE80   81DE   292   847          SB       SPACE
82CC 20CE1F          293   848          MVI      C'1',K
82CD D5CE80   81CE   294   849          SB       OBSTROBE
82CE 20CEDF          295   850          MVI      C'=',K
82CF D5CE80   81CE   296   851          SB       OBSTROBE
82D0 20CE2F          297   852          MVI      C'2',K
82D1 D5CE80   81CE   298   853          SB       OBSTROBE
82D2 A0CE0F          299   854          MVI      C'0',K
82D3 D5CE80   81CE   300   855          SB       OBSTROBE
82D4 A10EBF          301   856          MVI      C'K',K
82D5 D5CE80   81CE   302   857          SB       OBSTROBE
82D6 208E9F          303   858          MVI      C')',K
82D7 D5CE80   81CE   304   859          SB       OBSTROBE
82D8 55DE80   81DE   305   860          SB       SPACE
82D9 87800F          306   861          SR       ,
                     307   862  *
                     308   863  *
82DA 0208E0          309   864  RETURN  MVX      F1F0,PHPL     PUT PC'S OF TRAP BACK IN PC'S
82DB AC0818          310   865          AI       1,PL,PL       INCREMENT PC'S TO POINT AT RETURN ADDR.
82DC 05800F          311   866          TPS      ,             PUT PC'S ON STACK
82DD 87800F          312   867          SR       ,             POP STACK FORCE PROGRAM TO RETURN AFTER PECM TEST
                     313   868  *
                     314   869  *
82DE DEDE80   82DE   315   870  BDPARY  B        *             EDIT THESE LOCATIONS TO CONTAIN BAD PARITY
```

3/29/78

```
82DF 5EDF80   82DF   316   871  BDPARY1  B        *          TO FORCE CONTROL MEM PARITY ERROR FOR TESTING
82E0 DC2C80   802C   317   872           B        INTEST
                     318   873  *END
```

NUMBER OF LINES IN ERROR = 0 NO. OF WARNINGS = 0 NO. OF SYMBOLS = 99 OVERFLOWS = 43

SYMBOL	VALUE	DEFN	REFERENCES
BDPARY	82DE	0870	0052 0722
BDPARY1	82DF	0871	0724 0725
CHECK1	8075	0138	0117
CHKTST	8123	0349	
CMAMNT	828A	0782	0056
CMCONT	814D	0406	0427 0429
CMEXEC	814A	0401	0397
CMTEST	8065	0122	0111 0153 0163 0173
COMPM	8172	0453	0447
CRLF	81D8	0580	0055 0057 0616 0631 0704
DELAY10	81C8	0558	0772 0773 0774 0775 0776
DELAY5	81C9	0559	0558 0567 0569
DELAY50	8284	0772	0765
ENABLE	81C7	0557	0060
ENABLE5	81C6	0556	0574 0580 0588 0594 0602 0620 0635 0643 0782
ERROR	81F8	0620	0083 0097 0138 0139 0140 0141 0142 0143 0234 0358 0432 0501 0552 0742 0744 0746 0748 0750 0752 0754 0756
ERROR0	80C9	0234	0217 0218 0219 0220
ERROR3	812A	0358	0350 0351 0352
ERROR4	8161	0432	0422 0423 0445 0446 0448 0450 0451 0454 0455
ERROR9	8196	0501	0489 0491
ERRORA	81C4	0552	0512 0513 0515 0516 0523 0524 0526 0527 0530 0532 0534 0536 0542 0544 0546
FILCHK	8267	0741	
FLVCHK	8133	0370	0396 0398
INCRF3	80C3	0225	0223
INCRK	80BA	0216	0210
INCRPC	80C5	0227	0224
INCRPH	80AC	0202	0199
INCRPL	80A7	0197	0185
INIT	81D2	0572	
INIT8	812C	0362	0355
INITCMT	8060	0115	0110
INITCRT	81E2	0594	0030 0548
INITIAL0	8010	0021	0005 0023 0024 0028 0029
INITIAL1	8023	0042	0006 0007 0008 0009 0010 0011 0012 0013 0014 0015
INITIAL2	802B	0052	0016
INPUT	8032	0061	
INSTACK	80FD	0299	0245 0255 0265 0275 0285 0295 0380 0388
INTEST	802C	0055	0067 0872
LOOPA	8045	0082	0087

SYMBOL	VALUE	DEFN	REFERENCES
LOOPADD	809B	0185	0201 0205
LOOPB	8051	0096	0101
OBSTROBE	81CE	0566	0038 0047 0576 0582 0584 0590 0596 0598 0604 0606 0608 0610 0612 0622 0624 0626 0628 0630 0637 0645 0648 0652 0654 0655 0656 0659 0661 0663 0665 0667 0669 0672 0674 0676 0678 0680 0683 0685 0687 0689 0691 0698 0701 0702 0703 0710 0784 0786 0788 0790 0792 0795 0797 0799 0801 0803 0805 0808 0810 0813 0815 0817 0819 0821 0823 0825 0828 0830 0832 0836 0838 0840 0842 0844 0846 0849 0851 0853 0855 0857 0859
ONE	803C	0071	0064
PART1	8011	0022	
PART2	8019	0030	
PCCNTL	8143	0392	0378 0386
PE24	8255	0719	0002
PE8	8262	0734	0004
PRINT?	8036	0065	0062
PRTBIT1	824B	0707	0714 0716
PRTBITE	824F	0711	0694 0696
PRTBYTE	8206	0635	0066 0614
READ?	806D	0130	0115
READCM	80B4	0210	0229
READDM	818A	0487	0473 0493 0494 0495 0497
RESMAX	8169	0444	0428
RESTRT	8279	0761	0770
RETURN	82DA	0864	0727
RETURN1	810D	0317	0241 0251 0261 0376
RETURN2	8113	0325	0271 0281 0291
RETURN3	8168	0441	0384
RUNF	8114	0328	0293
RUNFF	8119	0335	0353
RUNFFF	811E	0342	0354
SEC1	8278	0760	0470 0547 0632
SPACE	81DE	0588	0613 0657 0670 0681 0692 0699 0793 0806 0811 0826 0834 0847 0860
SPACEOUT	8212	0648	0650
SYSERROR	820D	0643	
SYSPE	820A	0640	0730 0737
SYSVE	820B	0641	
TEST#	81E8	0602	0079 0093 0107 0150 0160 0170 0179 0238 0366 0464 0507
TEST0	803F	0076	0070 0073 0549
TEST1	804B	0090	
TEST2	8057	0104	
TEST3	807C	0147	0112
TEST4	8084	0157	0154
TEST5	808C	0167	0164
TEST6	8094	0178	0174 0235

SYMBOL	VALUE	DEFN	REFERENCES
TEST7	80CB	0237	0226 0233 0359
TEST8	812E	0365	0433
TEST9	8176	0463	0456 0502
TESTA	8198	0506	0474 0553
TSTADD	8154	0415	
TSTEND	815D	0426	
WAIT1	827A	0762	0766
WAIT2	827F	0767	
WREVEN	8137	0376	
WRODD	813D	0384	0373
WRT10	80E5	0269	0263
WRT100	80ED	0279	0273
WRT1000	80F5	0289	0283
WRTDM	8182	0477	0468 0479 0480 0481 0483
WRTF	80CD	0239	
WRTFF	80D5	0249	0243
WRTFFF	80DD	0259	0253
WRTNOP	8108	0312	0246 0256 0266 0381
WRTSB	810E	0320	0276 0286 0296
WRTSB1	8163	0436	0389
ZERO	8039	0068	0063

MODULE REPAIR GUIDE NO.4.4
EDITED BY CUSTOMER ENGINEERING DIVISION

210-6791 TEST PROGRAM May 19, 1978

1. INTRODUCTION

210-6791 STACK AUXILIARY REGISTER TEST PROGRAM

The 210-6791 test program was writen to be used as a helpful aid in the troubleshooting and repair of the 6791 module.

This program is similiar in many respects to the 210-6790 test program, ie. utilizing preliminary tests, advancing onward to more detailed tests, and providing visual display of test results wherever possible.

This program is designed to run exclusively from PROM. It is very important to run this program with all KNOWN GOOD MODULES before attempting to repair any other module.

2. INSTALLATION/OPERATION

The standard 2200VP system repair equipment is utilized for testing the 6791 module. The only additional equipment required is a set of debug PROMS located as follows:

PROM #	LOCATION
378-2204	6789 - L27
378-2205	6789 - L28
378-2206	6789 - L29

The only installation required is to physically remove the boot/utility PROMS from the 6789 and replace them with the 6791 debug PROMS.

Printed in U.S.A.
WANG LABORATORIES, INC.
ONE INDUSTRIAL AVENUE, LOWELL, MASSACHUSETTS 01851, TEL. (617) 851-4111, TWX 710 343-6769, TELEX 94-7421

3. GENERAL TEST INFORMATION

POWER ON

As power is applied to the system, a micro trap to PROM location 8003 occurs which begins test operation.

START

START should clear the CRT display and locate the cursor at home position. This, of course, depends on the ability of the 6791 module to decode and execute both a SB (subroutine branch) and a SR (subroutine return).

There are also several other logic functions occuring during execution of this small subroutine. If the CRT display is not cleared, it is because this routine failed to execute properly.

INITIAL

INITIAL is a short routine that tests the ability of the Program Counter (PH, PL) to be cleared (set to 0000) and set (set to FFFF). Some of the logic functions tested are: the LPI instruction (LOAD PC'S IMMEDIATE) instruction, and the XOP instruction (EXTENDED OPERATION).

If an error occurs during execution, the program will loop in this routine until the error is cleared.

If no error occurs in the routine, a "0" should be printed on the CRT and the program advances to INITIAL 0.

INITIAL 0

INITIAL 0 checks that Auxiliary Register 00 can be cleared (set to 0000) and set (set to FFFF). The TPA (TRANSFER PC'S TO AUXILIARY REGISTER) instruction is tested for the first time during execution of this routine.

If an error occurs, the program will loop until the error is cleared. If no error occurs, the program will print a "1" on the CRT and continue on to INITIAL 1.

INITIAL 1

INITIAL 1 checks that Stack Address 0000 can be written with both all zeros and all ones (0000, FFFF). The TPS (TRANSFER PC'S TO STACK) instruction, and the TSP (TRANSFER STACK TO PC'S) instruction are first executed during this routine.

If an error occurs, the program will loop in this test. If no error occurs, the program prints a "2" on the CRT and continues on to INITIAL 2.

INITIAL 2

INITIAL 2 checks that the XPA (EXCHANGE PC'S WITH AUXILIARY REGISTER) instruction works properly. If an error occurs, the program loops in this test. If no error occurs, the program prints a "3" on the CRT and goes on to INTEST.

INTEST

INTEST is a routine designed to input via keyboard the amount of Control Memory available. If the system has 16K of Control Memory, the operator must type a "0". If the system has 20K of Control Memory, the operator must type a "1". Any response other than 0 or 1 will cause the program to print a "?" and the input statement "INPUT THE AMOUNT OF CONTROL MEMORY....." again.

If, however, the operator typed a "1" for 20K of Control Memory and the system only contained 16K of memory, a Control Memory parity error at location 4000 (PECM 4000) will result. If the operator typed a "0" and the system really had 20K of control memory, the IC will only be incremented to 03FF (hex), resulting in an insufficient test of the Instruction Counter.

After the proper character is typed on the keyboard, the program goes to START 0.

START 0

START 0 clears the CRT and homes the cursor then proceeds to TEST 0.

TEST 0

TEST 0 checks the PL (LOW ORDER PC). The PL is first cleared, then incremented by one, checking that the register incremented properly. This 'increment then test' procedure is carried on until the register equals FF (all ones).

An error in this routine will force the program to restart at TEST 0. If no error occurs, the program continues on to the next test.

NOTE: If an error occurs in any of these tests, the CRT will display:

TEST # X
ERROR

Where X = Test number that failed.

TEST 1

TEST 1 tests the PH from 00 to FF (hex) one step at a time, checking that PH increments properly. The program will restart and loop on error. If no error occurs, the program continues on to the next test.

TEST 2

TEST 2 increments the Program Counter from 0000 to FFFF (hex), one step at a time, checking that the Program Counter did increment properly. The main difference between this routine and the previous two tests is that PH and PL are being considered as one register in this test, rather than seperate registers as in the previous tests. Thus, the 'carry out' from PL to PH is being exercised during this test.

TEST 3

TEST 3 is the same as test 2, except the PC's are now decremented. Initially, the PC's are set to FFFF. The PC's are then decremented and checked until they are equal to 0000.

TEST 4

TEST 4 is the Stack Address test. Stack Address 0000 is written with 0000, Stack Address 0001 is written with 0101 etc., until all the stack addresses are written. The test then reads and checks that all the addresses were written properly.

NOTE: The lower order Program Counter (PL) is duplicated in the high order Program Counter (PH).

TEST 5

TEST 5 first writes all Stack locations with zeroes (0000) and checks that zeroes were written properly, then writes all ones (FFFF) and checks that all ones were written properly.

TEST 6

TEST 6 is the Stack exerciser test. This test uses the following test patterns:

	PH		PL	
1.	0000	0001	0000	0001
2.	0000	0010	0000	0010
3.	0000	0100	0000	0100
4.	0000	1000	0000	1000
5.	0001	0000	0001	0000
6.	0010	0000	0010	0000
7.	0100	0000	0100	0000
8.	1000	0000	1000	0000

Two other patterns are also used - either all 0's or all 1's depending upon which pass the routine is executing.

The Stack is initialized with all 0's in all Stack locations. The first test pattern is written in the first Stack location. The first Stack location is read and checked. The remaining Stack locations are read and checked for the "other pattern". The test continues in this manner until all eight test patterns have been written in each location of the Stack and checked to see that no other location has changed.

Effectively, this test checks the Stack to determine whether each Stack location will:

a) Hold a particular pattern

b) Have any effect on another Stack location

TEST 7

TEST 7 checks that all 32 Auxiliary Registers can be written and read back with four different test patterns. The patterns are:

1.	0000	0000	0000	0000
2.	1111	1111	1111	1111
3.	0101	1010	0101	1010
4.	1010	0101	1010	0101

TEST 8

Test 8 is the Auxiliary Register address/incremental data test. Auxiliary Register 00 is written with "0000". Auxiliary Register 01 is written with "0101". This continues until Auxiliary Register 1F is written with "1F1F". At this time, all the registers are read and compared to see if the correct data was written.

TEST 9

TEST 9 writes to Control Memory location 1000 twice. The first time it is written with all 0's. The K Register and the PC's are then filled with "5A's" (alternating 1's and 0's). Location 1000 is then read back. The K Register and the Program Counter are then checked to

see that each was overwritten properly. In the second pass, location 1000 is written with "FFFF" and checked in the same manner as the first. The overall effect of this test is to check the Control Memory input to the Program Counter via the Program Counter Source Selector.

TEST A

TEST A forces the execution of two instructions, both written with bad parity. The first instruction is an LPI (LOAD PC'S IMMEDIATE). The second instruction is an SB (SUBROUTINE BRANCH). Obviously, both instructions should cause control memory parity errors, but if the parity error signal (PECM) does not reach the 6791 module or the module does not decode it properly, the instruction that caused the parity error will still be executed. In the case of the SB instruction , the program will hang, executing the instruction over and over again if the parity error is not detected.

TEST B

TEST B tests the Instruction Counter (IC) input to the Stack that is used during subroutine branches. The program writes all of Control Memory with two instructions, first a NOOP (NO OPERATION), then with a SB TSTADD (SUBROUTINE BRANCH TO LOCATION 8260). It then begins execution at control memory location 0000. The program executes the NOOP, then executes the SB TSTADD which forces a subroutine branch to PROM location 8260. The program then checks the Stack to see that the right location was pushed during the SB instruction. This program keeps executing in the same manner, until all of Control Memory is addressed.

TEST C

TEST C is the Data Memory Select 1 and 2 (DMS1, DMS2) test. This test exercises a small amount of logic on the 6791 module associated with the selection of a particular Data Memory module. This selection process is totally address dependent. It is a direct function of the high order Program Counter Register (PH) bit 7 (PH7). Depending on the condition of this bit, either DMS1 or DMS2 is asserted.

The micro-program writes two data memory locations with "5A" data patterns, then reads back the same locations checking that they did get written properly. One location is on the first memory board (DMS1), and the other location is on the second memory board (DMS2).

At the completion of this test, the program restarts at Test 0.

RESET

When RESET is keyed, the program re-runs the preliminary tests and asks for the amount of Control Memory available to the system.

PARITY ERRORS

The program will detect and display any unforced parity errors. After the error information is printed on the CRT, the program will hang, branching continuously to the same location. At this point, the RESET key should be struck to restart the program.

```
                8000   1    1            ORG      8000
8000 5F8180     8381   2    2            B        PE24          PARITY ERROR CONT. MEM. TRAP LOCATION
8001 DC1080     8010   3    3            B        START         RESET KEY STRUCK, TRAP LOCATION
8002 DF9280     8392   4    4            B        PE8           PARITY ERROR DATA MEM. TRAP LOCATION
8003 DC1080     8010   5    5            B        START         POWER ON RESET TRAP LOCATION
8004 DC0480     8004   6    6            B        *
8005 5C0580     8005   7    7            B        *
8006 5C0680     8006   8    8            B        *
8007 DC0780     8007   9    9            B        *
8008 DC0880     8008   10   10           B        *
8009 5C0980     8009   11   11           B        *
800A 5C0A80     800A   12   12           B        *
800B DC0B80     800B   13   13           B        *
800C 5C0C80     800C   14   14           B        *
800D DC0D80     800D   15   15           B        *
800E DC1080     8010   16   16           B        START
800F 800000            17   17           INSTR    800000
                       18   18   *
                       19   19   *
                       20   20   *
8010 D6FE80     82FE   21   21   START    SB       INIT          CLEAR CRT DISPLAY DISABLE INPUTS
                       22   22   *
                       23   23   *CHECK PC'S CAN BE CLEARED AND SET FF
                       24   24   *
8011 860000            25   25   INITIAL  XORX     F1F0,F1F0,F1F0  CLEAR FILE REGS. 0 AND 1
8012 190000     0000   26   26            LPI      0000          LOAD PC'S = 0000 (CLEAR)
8013 D81108     8011   27   27            BNR      F0,PL,INITIAL FILE REG. 0 AND PL SHOULD = 00
8014 D81119     8011   28   28            BNR      F1,PH,INITIAL FILE REG. 1 AND PH SHOULD = 00
8015 23C0FF            29   29            MVI      0FF,F0        SET FILE REG. 0 = FF
8016 A00100            30   30            MV       F0,F1         SET FILE REG. 1 = FF
8017 1FCFFF     FFFF   31   31            LPI      0FFFF         SET PC'S = TO FFFF
8018 D81108     8011   32   32            BNR      F0,PL,INITIAL SHOULD BOTH = FF IF NOT HANG HERE
8019 D81119     8011   33   33            BNR      F1,PH,INITIAL SHOULD BOTH = FF IF NOT HANG HERE
801A A0CE0F            34   34            MVI      30,K          PUT "0" IN K REG.
801B 56FA80     82FA   35   35            SB       OBSTROBE      PRINT "0" ON CRT
                       36   36   *
                       37   37   *CHECK AUX. REG. 00 CAN BE CLEARED AND SET TO FF
                       38   38   *
801C 860000            39   39   INITIAL0 XORX     F1F0,F1F0,F1F0  CLEAR FILE REGS. 1 AND
801D 190000     0000   40   40            LPI      0000          CLEAR PC'S
801E 81800F            41   41            TPA      ,00           CLEAR AUX. REG. 00
801F 1B4A5A     5A5A   42   42            LPI      5A5A          PUT GARBAGE IN PC'S
8020 8B800F            43   43            TAP      ,00           PUT AUX. 00 IN PC'S
8021 581C08     801C   44   44            BNR      F0,PL,INITIAL0 SHOULD BE EQUAL IF NOT HANG HERE
8022 D81C18     801C   45   45            BNR      F1,PL,INITIAL0 SHOULD BE EQUAL IF NOT HANG HERE
```

```
8023 23C0FF        46  46          MVI     0FF,F0          PUT FF IN REG 0
8024 A00100        47  47          MV      F0,F1           MOVE FF TO F1
8025 1FCFFF  FFFF  48  48          LPI     0FFFF           PUT FFFF IN PC'S
8026 81800F        49  49          TPA     ,00             PUT FFFF IN AUX 00
8027 1B4A5A  5A5A  50  50          LPI     5A5A            PUT GARBAGE IN PC'S
8028 8B800F        51  51          TAP     ,00             PUT AUX. 00 IN PC'S
8029 581C08  801C  52  52          BNR     F0,PL,INITIAL0  SHOULD BOTH = FF IF NOT HANG HERE
802A D81C18  801C  53  53          BNR     F1,PL,INITIAL0  SHOULD BOTH = FF IF NOT HANG HERE
802B 20CE1F        54  54          MVI     31,K            PUT A "1" IN K REG
802C 56FA80  82FA  55  55          SB      OBSTROBE        PRINT A "1" ON CRT
                   56  56  *
                   57  57  *CHECK STACK 0 CAN BE CLEARED AND SET TO FF
                   58  58  *
802D 860000        59  59  INITIAL1 XORX    F1F0,F1F0,F1F0  CLEAR FILE REGS. 0 AND 1
802E 190000  0000  60  60          LPI     0000            CLEAR PC'S
802F 05800F        61  61          TPS     ,               CLEAR STACK ADDR. 0
8030 1B4A5A  5A5A  62  62          LPI     5A5A            PUT GARBAGE IN PC'S
8031 8D800F        63  63          TSP     ,               PUT STACK CONTENTS IN PC'S
8032 D82D08  802D  64  64          BNR     F0,PL,INITIAL1  SHOULD BE EQUAL IF NOT HANG HERE
8033 D82D19  802D  65  65          BNR     F1,PH,INITIAL1  SHOULD BE EQUAL IF NOT HANG HERE
8034 23C0FF        66  66          MVI     0FF,F0          PUT FF IN FILE REG 0
8035 A00100        67  67          MV      F0,F1           PUT FF IN FILE REG. 1
8036 1FCFFF  FFFF  68  68          LPI     0FFFF           PUT FFFF IN PC'S
8037 05800F        69  69          TPS     ,               PUT FFFF IN STACK ADDR. 0
8038 1B4A5A  5A5A  70  70          LPI     5A5A            PUT GARBAGE IN PC'S
8039 8D800F        71  71          TSP     ,               PUT STACK ADDR 0 IN PC'S
803A D82D08  802D  72  72          BNR     F0,PL,INITIAL1  SHOULD BE = IF NOT HANG HERE
803B D82D19  802D  73  73          BNR     F1,PH,INITIAL1  SHOULD BE = IF NOT HANG HERE
803C 20CE2F        74  74          MVI     32,K            PUT A "2" IN K REG.
803D 56FA80  82FA  75  75          SB      OBSTROBE        PRINT A "2" ON CRT
                   76  76  *
                   77  77  *CHECK XPA INSTR. WORKS PROPERLY
                   78  78  *
803E 190000  0000  79  79  INITIAL2 LPI     0000            CLEAR PC'S
803F 81800F        80  80          TPA     ,00             CLEAR AUX 00
8040 1FCFFF  FFFF  81  81          LPI     0FFFF           PUT TEST PATTERN IN PC'S
8041 03800F        82  82          XPA     ,00             EXCHANGE PC'S WITH AUX 00
8042 20000F        83  83          MVI     00,F0
8043 A00100        84  84          MV      F0,F1           CLEAR FILE REGS. 0 AND 1
8044 A3C2FF        85  85          MVI     0FF,F2
8045 A00302        86  86          MV      F2,F3           SET FILE REGS 2 AND 3 = TO FF
8046 583E08  803E  87  87          BNR     F0,PL,INITIAL2  PC'S SHOULD = 0000 AFTER EXCHANGE
8047 583E19  803E  88  88          BNR     F1,PH,INITIAL2
8048 8B800F        89  89          TAP     ,00             PUT AUX 00 IN PC'S
8049 D83E28  803E  90  90          BNR     F2,PL,INITIAL2  AUX 00 SHOULD = FFFF AFTER EXCHANGE
```

```
804A D83E39   803E   91   91            BNR       F3,PH,INITIAL2
804B A0CE3F          92   92            MVI       33,K              PUT A "3" IN K
804C 56FA80   82FA   93   93            SB        OBSTROBE          PRINT "3" ON CRT
                     94   94   *
                     95   95   *WAIT FOR A "0" OR A "1" FROM KEYBOARD TO CONTINUE
                     96   96   *
804D 570480   8304   97   97   INTEST   SB        CRLF              MOVE CUSOR TO NEXT LINE
804E D68A80   828A   98   98            SB        CMAMNT            DISPLAY "INPUT THE AMOUNT OF CONTROL MEM. ..."
804F 570480   8304   99   99            SB        CRLF
8050 A00D0F         100  100            MVI       00,SH             ALLOW INPUTS
8051 200E1F         101  101            MVI       1,K               SET K FOR KEYBOARD INPUT
8052 56F380   82F3   102  102            SB        ENABLE            ENABLE INPUT CHANNEL
8053 68532D   8053   103  103   INPUT    BFL       2,SH,*            WAIT FOR INPUT FROM KEYBOARD
8054 7C573E   8057   104  104            BNEH      3,K,PRINT?        INPUTED CHARECTER = TO 3X?
8055 F05A0E   805A   105  105            BEQL      0,K,ZERO          INPUT CHAR = TO 30?
8056 F05D1E   805D   106  106            BEQL      1,K,ONE           INPUT CHAR = TO 31?
8057 20C0FF         107  107   PRINT?   MVI       C'?',F0           PUT A "?" IN F0
8058 573280   8332   108  108            SB        PRTBYTE           PRINT ? ON CRT
8059 5C4D80   804D   109  109            B         INTEST            GET NEXT INPUT
805A 23C6FF         110  110   ZERO     MVI       0FF,F6
805B A0C7FF         111  111            MVI       3F,F7             SET F6,F7 = TO LAST ADDR. 16K CONT. MEM.
805C 5C6080   8060   112  112            B         START0
805D 23C6FF         113  113   ONE      MVI       0FF,F6
805E 2107FF         114  114            MVI       4F,F7             SET F6,F7 = TO LAST ADDR. CONT. MEM. 20K
805F 5C6080   80C0   115  115            B         START0
                     116  116   *
                     117  117   *
8060 D6FE80   82FE   118  118   START0   SB        INIT              INHIBIT INPUTS DISABLE TRAPS
                     119  119   *
                     120  120   *TEST PC'S FROM 0000 TO FFFF
                     121  121   *
8061 20C00F         122  122   TEST0    MVI       30,F0             PUT "0" IN F0
8062 D71480   8314   123  123            SB        TEST#             PRINT "TEST# 0" ON CRT
8063 20000F         124  124            MVI       00,F0             CLEAR F0
8064 A0080F         125  125            MVI       00,PL             CLEAR PL
8065 A00800         126  126   LOOPA    MV        F0,PL             PUT TEST PATTERN IN PL
8066 D86C08   806C   127  127            BNR       F0,PL,ERROR0      TEST PATTERN SHOULD = PL OR ERROR
8067 F869F0   8069   128  128            BNEL      0F,F0,*+2         CHECK FOR LAST PATTERN
8068 746BF0   806B   129  129            BEQH      0F,F0,*+3         CHECK FOR LAST PATTERN
8069 AC0010         130  130            AI        1,F0,F0           STEP TEST PATTERN BY ONE
806A 5C6580   8065   131  131            B         LOOPA             NEXT PATTERN
806B DC6E80   806E   132  132            B         TEST1             GO ON TO NEXT TEST
806C D72480   8324   133  133   ERROR0   SB        ERROR             GO PRINT "ERROR" ON CRT
806D DC6180   8061   134  134            B         TEST0             EXECUTE SAME TEST AGAIN
                     135  135   *
```

```
                    136  136  *TEST PC'S (PH) FROM 00 TO FF
                    137  137  *
806E A0C01F         138  138  TEST1    MVI      31,F0          PUT A "1" IN FILE REG. 0
806F D71480  8314   139  139           SB       TEST#          PRINT "TEST#1" ON CRT
8070 A0010F         140  140           MVI      00,F1          CLEAR F1
8071 20090F         141  141           MVI      00,PH          CLEAR PH
8072 A00901         142  142  LOOPB    MV       F1,PH          PUT TEST PATTERN IN PH
8073 587919  8079   143  143           BNR      F1,PH,ERROR1   TEST PATTERN SHOULD = PH OR ERROR
8074 F876F1  8076   144  144           BNEL     0F,F1,*+2      CHECK FOR LAST PATTERN
8075 7478F1  8078   145  145           BEQH     0F,F1,*+3      CHECK FOR LAST PATTERN
8076 AC0111         146  146           AI       1,F1,F1        STEP TEST PATTERN BY 1
8077 5C7280  8072   147  147           B        LOOPB          NEXT TEST PATTERN
8078 5C7B80  807B   148  148           B        TEST2          GO ON TO NEXT TEST
8079 D72480  8324   149  149  ERROR1   SB       ERROR          PRINT "ERROR" ON CRT
807A DC6E80  806E   150  150           B        TEST1          REPEAT SAME TEST
                    151  151  *
                    152  152  *TEST INCREMENT PC'S
                    153  153  *
807B A0C02F         154  154  TEST2    MVI      32,F0          PUT A "2" IN F0
807C D71480  8314   155  155           SB       TEST#          PRINT "TEST#2" ON CRT
807D 860000         156  156           XORX     F1F0,F1F0,F1F0 CLEAR OUT FILE REGS. 0 AND 1
807E 190000  0000   157  157           LPI      0000           CLEAR PC'S
807F 000FEF         158  158  LOOPC    OR       +,,            INCREMENT PC'S
8080 AC0010         159  159           AI       1,F0,F0        INCREMENT F0
8081 788300  8083   160  160           BNEL     0,F0,*+2       CHECK FOR LAST PC INCREMENT
8082 748600  8086   161  161           BEQH     0,F0,INCPH     IF LOW ORDER ADDR. = TO FF THEN INCR HIGH
8083 588A08  808A   162  162  INCTST   BNR      F0,PL,ERROR2   CHECK COUNTER = TO PC'S LOW ORDER
8084 588A19  808A   163  163           BNR      F1,PH,ERROR2   CHECK COUNTER = TO PC,S HIGH ORDER
8085 DC7F80  807F   164  164           B        LOOPC          NEXT INCREMENT
8086 AC0111         165  165  INCPH    AI       1,F1,F1        STEP COUNT HIGH ORDER BY ONE
8087 F88301  8083   166  166           BNEL     0,F1,INCTST    CHECK IF LAST INCREMENT
8088 F48C01  808C   167  167           BEQH     0,F1,TEST3     IF LAST COUNT GO TO NEXT TEST
8089 DC8380  8083   168  168           B        INCTST         NOT LAST COUNT GO CHECK PC INCREMENT
808A D72480  8324   169  169  ERROR2   SB       ERROR          PRINT "ERROR" ON CRT
808B 5C7B80  807B   170  170           B        TEST2          REPEAT SAME TEST
                    171  171  *
                    172  172  * TEST DECREMENT PC'S
                    173  173  *
808C 20C03F         174  174  TEST3    MVI      33,F0          PUT A "3" IN F0
808D D71480  8314   175  175           SB       TEST#          PRINT "TEST#3" ON CRT
808E 23C0FF         176  176           MVI      0FF,F0         PUT "FF" IN F0
808F A00100         177  177           MV       F0,F1          PUT "FF" IN F1
8090 1FCFFF  FFFF   178  178           LPI      0FFFF          PUT "FFFF" IN PC'S
8091 800FFF         179  179  LOOPD    OR       -,,            DECREMENT PC'S
8092 2FC0F0         180  180           AI       0FF,F0,F0      DECREMENT F0
```

```
8093 F895F0  8095  181  181          BNEL    0F,F0,*+2          CHECK FOR LAST DECREMENT (F0=TO 00)
8094 7498F0  8098  182  182          BEQH    0F,F0,DECPH        IF F0 = TO 00 GO DECREMENT F1
8095 D89C08  809C  183  183  DECTST  BNR     F0,PL,ERROR3
8096 D89C19  809C  184  184          BNR     F1,PH,ERROR3
8097 DC9180  8091  185  185          B       LOOPD
8098 2FC1F1        186  186  DECPH   AI      0FF,F1,F1          DECREMENT F1
8099 789501  8095  187  187          BNEL    00,F1,DECTST       CHECK FOR LAST DECREMENT (F0,F1,=TO 00)
809A F49E01  809E  188  188          BEQH    00,F1,TEST4        IF F0,F1 = TO 00 GO TO NEXT TEST
809B 5C9580  8095  189  189          B       DECTST             NEXT DECREMENT OPERATION
809C D72480  8324  190  190  ERROR3  SB      ERROR
809D DC8C80  808C  191  191          B       TEST3              GO ON TO NEXT TEST
                   192  192  *
                   193  193  *STACK ADDRESS TEST
                   194  194  *
809E A0C04F        195  195  TEST4   MVI     34,F0              PUT A "4" IN F0
809F D71480  8314  196  196          SB      TEST#              PRINT "TEST#4" ON CRT
80A0 190000  0000  197  197          LPI     0000               CLEAR PC'S
80A1 A00008        198  198          MV      PL,F0              CLEAR F0
80A2 A1820F        199  199          MVI     60,F2              SET F2 = TO MAXIMUM STACK ADDRESS
80A3 05800F        200  200          TPS     ,                  WRITE "0000" IN STACK ADDR. 00
80A4 000FEF        201  201  LOOPE   OR      +,,                STEP ADDR. BY ONE
80A5 A00008        202  202          MV      PL,F0              SAVE ADDR. IN F0 FOR COMPARE
80A6 05800F        203  203          TPS     ,                  WRITE NEW ADDR. IN STACK ADDR.
80A7 58A420  80A4  204  204          BNR     F2,F0,LOOPE        TEST FOR LAST STACK ADDR.
80A8 8D800F        205  205  CHKSTK  TSP     ,                  POP STACK PUT DATA IN PC'S
80A9 58AE08  80AE  206  206          BNR     F0,PL,ERROR4       DATA READ FROM STACK SHOULD = ADDR. COUNTER
80AA 2FC0F0        207  207          AI      0FF,F0,F0          DECREMENT COUNTER BY ONE
80AB 78A800  80A8  208  208          BNEL    0,F0,CHKSTK        CHECK FOR COUNT = TO 00 LAST READ OPERATION
80AC 74B000  80B0  209  209          BEQH    0,F0,TEST5         IF COUNT = 00 NEXT TEST
80AD DCA880  80A8  210  210          B       CHKSTK             NEXT ADDR. READ
80AE D72480  8324  211  211  ERROR4  SB      ERROR              PRINT ERROR ON CRT
80AF DC9E80  809E  212  212          B       TEST4              RESTART SAME TEST
                   213  213  *
                   214  214  *WRITE STACK ALL 0'S, CHECK, WRITE ALL 1'S, CHECK.
                   215  215  *
80B0 20C05F        216  216  TEST5   MVI     35,F0              PUT A "5" IN F0
80B1 D71480  8314  217  217          SB      TEST#              PRINT "TEST#5" ON CRT
80B2 190000  0000  218  218          LPI     0000               CLEAR PC'S FOR WRITE
80B3 0200E8        219  219  START5  MVX     PHPL,F1F0
80B4 A1820F        220  220          MVI     60,F2              SET MAXIMUM ADDR. TO 60 (HEX VALUE FOR 96 DECIMAL)
80B5 20030F        221  221          MVI     00,F3              SET COUNT TO 0
80B6 A3C4FF        222  222          MVI     0FF,F4             ALL 1'S COMPARITOR
80B7 05800F        223  223  LOOPF   TPS     ,                  WRITE STACK
80B8 AC0313        224  224          AI      1,F3,F3            STEP COUNTER BY ONE
80B9 D8B723  80B7  225  225          BNR     F2,F3,LOOPF        CHECK MAX. LEVEL AGAINST COUNTER
```

```
80BA 05800F        226  226  CHSTK0   TPS      ,              PUT STACK DATA IN PC'S
80BB D8C608  80C6  227  227           BNR      F0,PL,ERROR5   CHECK LOW ORDER DATA = TO TEST PATTERN
80BC D8C619  80C6  228  228           BNR      F1,PH,ERROR5   CHECK HIGH ORDER DATA = TO TEST PATTERN
80BD 2FC3F3        229  229           AI       0FF,F3,F3      DECREMENT COUNTER BY ONE
80BE A00E03        230  230           MV       F3,K           PUT COUNTER IN K REG. FOR VISUAL AIDE
80BF 78BA03  80BA  231  231           BNEL     0,F3,CHSTK0    CHECK IF COUNTER = TO 0
80C0 74C203  80C2  232  232           BEQH     0,F3,ONES      IF COUNT = TO 00 WRITE ONES
80C1 DCBA80  80BA  233  233           B        CHSTK0         NEXT 0'S WRITE
80C2 D8C448  80C4  234  234  ONES     BNR      F4,PL,*+2      CHECK LOW ORDER PC = TO "FF"
80C3 D0C849  80C8  235  235           BER      F4,PH,TEST6    IF PC= "FFFF" DON'T WRITE ONES, GO NEXT TEST
80C4 1FCFFF  FFFF  236  236           LPI      0FFFF          SET PC'S = TO ALL 1'S
80C5 DCB380  80B3  237  237           B        START5         RESTART TEST, THIS TIME WRITE 1'S
80C6 D72480  8324  238  238  ERROR5   SB       ERROR          PRINT ERROR ON CRT
80C7 DCB080  80B0  239  239           B        TEST5          RESTART SAME TEST
                   240  240  *
                   241  241  *
                   242  242  *
                   243  243  *STACK EXCERCISER TEST
                   244  244  *
80C8 20C06F        245  245  TEST6    MVI      36,F0          PRINT TEST#6 ON CRT
80C9 D71480  8314  246  246           SB       TEST#
80CA A0001F        247  247  START1S  MVI      01,F0          INITIAL TEST PATTERN TO 01
80CB 20011F        248  248           MVI      01,F1          INITIAL TEST LEVEL TO 01
80CC A0020F        249  249           MVI      00,F2          INITIAL LEVEL COUNTER TO 00
80CD 21830F        250  250           MVI      60,F3          SET MAX LEVEL TO 96 (HEX 60)
80CE 20041F        251  251           MVI      01,F4          SET F4 = TO CONSTANT OF 1
80CF 20050F        252  252           MVI      00,F5          SET F5 = TO CONSTANT OF 0
80D0 20060F        253  253           MVI      00,F6          INITIAL "OTHER PATTERN" TO 00
                   254  254  *
                   255  255  *WRITE TEST PATTERN AT TEST LEVEL
                   256  256  *
80D1 200204        257  257           MV       F4,F2          MOVE A 1 TO THE LEVEL COUNTER
80D2 50D621  80D6  258  258  WRITE1   BER      F2,F1,WRITE1A  BRANCH IF LEVEL COUNT = TEST LEVEL
80D3 A00806        259  259           MV       F6,PL          PUT OTHER PATTERN IN PC'S
80D4 200906        260  260           MV       F6,PH
80D5 5CD880  80D8  261  261           B        READEND
80D6 A00800        262  262  WRITE1A  MV       F0,PL          MOVE PATTERN TO PC'S
80D7 200900        263  263           MV       F0,PH
80D8 05800F        264  264  READEND  TPS      ,              WRITE PATTERN INTO STACK
80D9 988242        265  265           AC,,0    F4,F2,F2       ADD A 1 TO LEVEL COUNTER
80DA 48D223  80D2  266  266           BLER     F2,F3,WRITE1   TEST TO SEE IF END OF STACK. IF NOT, BRANCH
                   267  267  *
                   268  268  *READ ENTIRE STACK
                   269  269  *
80DB A00203        270  270           MV       F3,F2          MOVE MAX LEVEL TO COUNTER
```

```
80DC 8D800F        271  271  READ1     TSP       ,              PUT STACK IN PC'S
80DD D0E121  80E1  272  272            BER       F2,F1,READTEST IF LEVEL = TEST LEVEL BRANCH
80DE D8F169  80F1  273  273            BNR       F6,PH,ERROR6   IF PH() OTHER PATTERN BRANCH TO ERROR6
80DF 58F168  80F1  274  274            BNR       F6,PL,ERROR6
80E0 DCE380  80E3  275  275            B         STACKCOM
80E1 58F108  80F1  276  276  READTEST BNR       F0,PL,ERROR6   IF PL() TEST PATTERN ERROR6
80E2 D8F109  80F1  277  277            BNR       F0,PH,ERROR6
80E3 0C822F        278  278  STACKCOM SC,,0     F2,,F2         SUBTRACT 1 FROM LEVEL POINTER
80E4 58DC52  80DC  279  279            BNR       F5,F2,READ1    BRANCH IF LEVEL COUNTER ()0
                   280  280  *
                   281  281  *GENERATE NEXT TEST PATTERN
                   282  282  *
80E5 188000        283  283            AC,,0     F0,F0,F0       ADD F0 TO F0 (SHIFT LEFT ONE BIT)
80E6 58D105  80D1  284  284            BNR       F0,F5,WRITE1-1 CHECK TO SEE IF ALL EIGHT PATTERNS DONE
80E7 A0001F        285  285            MVI       01,F0          MOVE INITIAL PATTERN TO F0
                   286  286  *
                   287  287  *INCREMENT TEST LEVEL
                   288  288  *
80E8 988141        289  289  ADDLEVEL AC,,0     F4,F1,F1       ADD A 1 TO TEST LEVEL
80E9 48D113  80D1  290  290            BLER      F1,F3,WRITE1-1 IF NEW TEST LEVEL(= 96 BRANCH
80EA A3C8FF        291  291            MVI       0FF,PL         MOVE FF TO PL (COMPLEMENT)
80EB D0EF68  80EF  292  292            BER       F6,PL,ENDDIAG1 IF OTHER PATTERN = FF THEN DONE
                   293  293  *
                   294  294  *CHANGE PATTERN TO ALL 1'S
                   295  295  *
80EC 23C6FF        296  296            MVI       0FF,F6         MOVE ALL ONE'S TO F6
80ED 200104        297  297            MV        F4,F1          MOVE 1 TO TEST LEVEL
80EE 5CD280  80D2  298  298            B         WRITE1         WRITE OTHER PATTERN AT TEST LEVEL
80EF 20000F        299  299  ENDDIAG1 MVI       00,F0
80F0 DCF480  80F4  300  300            B         TEST7
80F1 D72480  8324  301  301  ERROR6   SB        ERROR
80F2 DCC880  80C8  302  302            B         TEST6
80F3 200F0F        303  303            NOP       ,
```

```
80F4 A0C07F          1   305  TEST7   MVI     37,F0       PUT A "7" IN F0
80F5 D71480  8314    2   306          SB      TEST#       PRINT TEST#7 ON CRT
80F6 190000  0000    3   307          LPI     0000        CLEAR PC'S
80F7 D4FF80  80FF    4   308          SB      START7      WRITE ALL AUX REGS. WITH 0'S AND CHECK
80F8 1FCFFF  FFFF    5   309          LPI     0FFFF       SET PC'S = TO ALL 1'S
80F9 D4FF80  80FF    6   310          SB      START7      WRITE AND READ CHECK AUX REGS. WITH 1'S
80FA 1B4A5A  5A5A    7   311          LPI     5A5A        LOAD PC'S WITH TEST PATTERN
80FB D4FF80  80FF    8   312          SB      START7      WRITE AND CHECK ALL REGS. = TO 5A5A
80FC 1D85A5  A5A5    9   313          LPI     0A5A5       LOAD PC'S WITH TEST PATTERN
80FD D4FF80  80FF   10   314          SB      START7      WRITE, CHECK ALL REGS. = TO "A5A5"
80FE DD6A80  816A   11   315          B       TEST8
                    12   316  *
                    13   317  *
                    14   318  *
80FF 0200E8         15   319  START7  MVX     PHPL,F1F0   SAVE TEST PATTERN IN FILE REGS.
8100 550380  8103   16   320          SB      AUXWRT      WRITE ALL AUX REGS. WITH 0'S
8101 552480  8124   17   321          SB      AUXCHK      READ AND CHECK ALL REGS.
8102 87800F         18   322          SR      ,
                    19   323  *
                    20   324  *
                    21   325  *
                    22   326  *
                    23   327  *
                    24   328  *
8103 81800F         25   329  AUXWRT  TPA     ,00         WRITE ALL AUX REGS. WITH TEST PATTERN
8104 01801F         26   330          TPA     ,01
8105 01802F         27   331          TPA     ,02
8106 81803F         28   332          TPA     ,03
8107 01804F         29   333          TPA     ,04
8108 81805F         30   334          TPA     ,05
8109 81806F         31   335          TPA     ,06
810A 01807F         32   336          TPA     ,07
810B 01808F         33   337          TPA     ,08
810C 81809F         34   338          TPA     ,09
810D 8180AF         35   339          TPA     ,0A
810E 0180BF         36   340          TPA     ,0B
810F 8180CF         37   341          TPA     ,0C
8110 0180DF         38   342          TPA     ,0D
8111 0180EF         39   343          TPA     ,0E
8112 8180FF         40   344          TPA     ,0F
8113 01810F         41   345          TPA     ,10
8114 81811F         42   346          TPA     ,11
8115 81812F         43   347          TPA     ,12
8116 01813F         44   348          TPA     ,13
8117 81814F         45   349          TPA     ,14
```

```
8118 01815F          46   350          TPA     ,15
8119 01816F          47   351          TPA     ,16
811A 81817F          48   352          TPA     ,17
811B 81818F          49   353          TPA     ,18
811C 01819F          50   354          TPA     ,19
811D 0181AF          51   355          TPA     ,1A
811E 8181BF          52   356          TPA     ,1B
811F 0181CF          53   357          TPA     ,1C
8120 8181DF          54   358          TPA     ,1D
8121 8181EF          55   359          TPA     ,1E
8122 0181FF          56   360          TPA     ,1F
8123 87800F          57   361          SR      ,
                     58   362  *
                     59   363  *
                     60   364  *
8124 8B800F          61   365  AUXCHK  TAP     ,00          CHECK ALL AUX REGS. WRITTEN PROPERLY
8125 556580  8165    62   366          SB      PATCHK
8126 0B801F          63   367          TAP     ,01
8127 556580  8165    64   368          SB      PATCHK
8128 0B802F          65   369          TAP     ,02
8129 556580  8165    66   370          SB      PATCHK
812A 8B803F          67   371          TAP     ,03
812B 556580  8165    68   372          SB      PATCHK
812C 0B804F          69   373          TAP     ,04
812D 556580  8165    70   374          SB      PATCHK
812E 8B805F          71   375          TAP     ,05
812F 556580  8165    72   376          SB      PATCHK
8130 8B806F          73   377          TAP     ,06
8131 556580  8165    74   378          SB      PATCHK
8132 0B807F          75   379          TAP     ,07
8133 556580  8165    76   380          SB      PATCHK
8134 0B808F          77   381          TAP     ,08
8135 556580  8165    78   382          SB      PATCHK
8136 8B809F          79   383          TAP     ,09
8137 556580  8165    80   384          SB      PATCHK
8138 8B80AF          81   385          TAP     ,0A
8139 556580  8165    82   386          SB      PATCHK
813A 0B80BF          83   387          TAP     ,0B
813B 556580  8165    84   388          SB      PATCHK
813C 8B80CF          85   389          TAP     ,0C
813D 556580  8165    86   390          SB      PATCHK
813E 0B80DF          87   391          TAP     ,0D
813F 556580  8165    88   392          SB      PATCHK
8140 0B80EF          89   393          TAP     ,0E
8141 556580  8165    90   394          SB      PATCHK
```

```
8142 8B80FF          91   395          TAP       ,0F
8143 556580  8165    92   396          SB        PATCHK
8144 0B810F          93   397          TAP       ,10
8145 556580  8165    94   398          SB        PATCHK
8146 8B811F          95   399          TAP       ,11
8147 556580  8165    96   400          SB        PATCHK
8148 8B812F          97   401          TAP       ,12
8149 556580  8165    98   402          SB        PATCHK
814A 0B813F          99   403          TAP       ,13
814B 556580  8165   100   404          SB        PATCHK
814C 8B814F         101   405          TAP       ,14
814D 556580  8165   102   406          SB        PATCHK
814E 0B815F         103   407          TAP       ,15
814F 556580  8165   104   408          SB        PATCHK
8150 0B816F         105   409          TAP       ,16
8151 556580  8165   106   410          SB        PATCHK
8152 8B817F         107   411          TAP       ,17
8153 556580  8165   108   412          SB        PATCHK
8154 8B818F         109   413          TAP       ,18
8155 556580  8165   110   414          SB        PATCHK
8156 0B819F         111   415          TAP       ,19
8157 556580  8165   112   416          SB        PATCHK
8158 0B81AF         113   417          TAP       ,1A
8159 556580  8165   114   418          SB        PATCHK
815A 8B81BF         115   419          TAP       ,1B
815B 556580  8165   116   420          SB        PATCHK
815C 0B81CF         117   421          TAP       ,1C
815D 556580  8165   118   422          SB        PATCHK
815E 8B81DF         119   423          TAP       ,1D
815F 556580  8165   120   424          SB        PATCHK
8160 8B81EF         121   425          TAP       ,1E
8161 556580  8165   122   426          SB        PATCHK
8162 0B81FF         123   427          TAP       ,1F
8163 556580  8165   124   428          SB        PATCHK
8164 87800F         125   429          SR        ,
                    126   430  *
                    127   431  *
                    128   432  *
8165 D96808  8168   129   433  PATCHK   BNR       F0,PL,ERROR7      TEST PATTERN SHOULD BE = TO AUX. REG.
8166 D96819  8168   130   434          BNR       F1,PH,ERROR7
8167 87800F         131   435          SR        ,
                    132   436  *
                    133   437  *
                    134   438  *
8168 D72480  8324   135   439  ERROR7   SB        ERROR             PRINT "ERROR" ON CRT
```

```
8169 DCF480   80F4   136  440          B        TEST7          REPEAT SAME TEST
                     137  441  *
                     138  442  * THIS TEST WRITES ADDR. OF AUX REG IN REG AND CHECKS
                     139  443  *
816A A0C08F          140  444  TEST8   MVI      38,F0          PUT AN "8" IN F0
816B D71480   8314   141  445          SB       TEST#          PRINT "TEST#8 ON CRT"
816C 190000   0000   142  446          LPI      0000           LOAD PC'S WITH 0'S
816D 81800F          143  447          TPA      ,00            WRITE 0'S IN AUX 00
816E 55ED80   81ED   144  448          SB       AUXINC
816F 01801F          145  449          TPA      ,01            WRITE A 0101 IN AUX 01
8170 55ED80   81ED   146  450          SB       AUXINC
8171 01802F          147  451          TPA      ,02            WRITE A 0202 IN AUX 2
8172 55ED80   81ED   148  452          SB       AUXINC
8173 81803F          149  453          TPA      ,03
8174 55ED80   81ED   150  454          SB       AUXINC
8175 01804F          151  455          TPA      ,04
8176 55ED80   81ED   152  456          SB       AUXINC
8177 81805F          153  457          TPA      ,05
8178 55ED80   81ED   154  458          SB       AUXINC
8179 81806F          155  459          TPA      ,06
817A 55ED80   81ED   156  460          SB       AUXINC
817B 01807F          157  461          TPA      ,07
817C 55ED80   81ED   158  462          SB       AUXINC
817D 01808F          159  463          TPA      ,08
817E 55ED80   81ED   160  464          SB       AUXINC
817F 81809F          161  465          TPA      ,09
8180 55ED80   81ED   162  466          SB       AUXINC
8181 8180AF          163  467          TPA      ,0A
8182 55ED80   81ED   164  468          SB       AUXINC
8183 0180BF          165  469          TPA      ,0B
8184 55ED80   81ED   166  470          SB       AUXINC
8185 8180CF          167  471          TPA      ,0C
8186 55ED80   81ED   168  472          SB       AUXINC
8187 0180DF          169  473          TPA      ,0D
8188 55ED80   81ED   170  474          SB       AUXINC
8189 0180EF          171  475          TPA      ,0E
818A 55ED80   81ED   172  476          SB       AUXINC
818B 8180FF          173  477          TPA      ,0F
818C 55ED80   81ED   174  478          SB       AUXINC
818D 01810F          175  479          TPA      ,10
818E 55ED80   81ED   176  480          SB       AUXINC
818F 81811F          177  481          TPA      ,11
8190 55ED80   81ED   178  482          SB       AUXINC
8191 81812F          179  483          TPA      ,12
8192 55ED80   81ED   180  484          SB       AUXINC
```

```
8193 01813F        181  485          TPA     ,13
8194 55ED80  81ED  182  486          SB      AUXINC
8195 81814F        183  487          TPA     ,14
8196 55ED80  81ED  184  488          SB      AUXINC
8197 01815F        185  489          TPA     ,15
8198 55ED80  81ED  186  490          SB      AUXINC
8199 01816F        187  491          TPA     ,16
819A 55ED80  81ED  188  492          SB      AUXINC
819B 81817F        189  493          TPA     ,17
819C 55ED80  81ED  190  494          SB      AUXINC
819D 81818F        191  495          TPA     ,18
819E 55ED80  81ED  192  496          SB      AUXINC
819F 01819F        193  497          TPA     ,19
81A0 55ED80  81ED  194  498          SB      AUXINC
81A1 0181AF        195  499          TPA     ,1A
81A2 55ED80  81ED  196  500          SB      AUXINC
81A3 8181BF        197  501          TPA     ,1B
81A4 55ED80  81ED  198  502          SB      AUXINC
81A5 0181CF        199  503          TPA     ,1C
81A6 55ED80  81ED  200  504          SB      AUXINC
81A7 8181DF        201  505          TPA     ,1D
81A8 55ED80  81ED  202  506          SB      AUXINC
81A9 8181EF        203  507          TPA     ,1E
81AA 55ED80  81ED  204  508          SB      AUXINC
81AB 0181FF        205  509          TPA     ,1F
                   206  510  *
                   207  511  *
81AC 20000F        208  512          MVI     00,F0    CLEAR COMPARITOR
81AD 8B800F        209  513          TAP     ,00      PUT AUX 00 IN PC'S FOR TEST
81AE 55F080  81F0  210  514          SB      INCCHK   GO TEST FOR CORRECT DATA IN AUX REG 00
81AF 0B801F        211  515          TAP     ,01
81B0 55F080  81F0  212  516          SB      INCCHK
81B1 0B802F        213  517          TAP     ,02
81B2 55F080  81F0  214  518          SB      INCCHK
81B3 8B803F        215  519          TAP     ,03
81B4 55F080  81F0  216  520          SB      INCCHK
81B5 0B804F        217  521          TAP     ,04
81B6 55F080  81F0  218  522          SB      INCCHK
81B7 8B805F        219  523          TAP     ,05
81B8 55F080  81F0  220  524          SB      INCCHK
81B9 8B806F        221  525          TAP     ,06
81BA 55F080  81F0  222  526          SB      INCCHK
81BB 0B807F        223  527          TAP     ,07
81BC 55F080  81F0  224  528          SB      INCCHK
81BD 0B808F        225  529          TAP     ,08
```

```
81BE 55F080   81F0   226   530          SB        INCCHK
81BF 8B809F          227   531          TAP       ,09
81C0 55F080   81F0   228   532          SB        INCCHK
81C1 8B80AF          229   533          TAP       ,0A
81C2 55F080   81F0   230   534          SB        INCCHK
81C3 0B80BF          231   535          TAP       ,0B
81C4 55F080   81F0   232   536          SB        INCCHK
81C5 8B80CF          233   537          TAP       ,0C
81C6 55F080   81F0   234   538          SB        INCCHK
81C7 0B80DF          235   539          TAP       ,0D
81C8 55F080   81F0   236   540          SB        INCCHK
81C9 0B80EF          237   541          TAP       ,0E
81CA 55F080   81F0   238   542          SB        INCCHK
81CB 8B80FF          239   543          TAP       ,0F
81CC 55F080   81F0   240   544          SB        INCCHK
81CD 0B810F          241   545          TAP       ,10
81CE 55F080   81F0   242   546          SB        INCCHK
81CF 8B811F          243   547          TAP       ,11
81D0 55F080   81F0   244   548          SB        INCCHK
81D1 8B812F          245   549          TAP       ,12
81D2 55F080   81F0   246   550          SB        INCCHK
81D3 0B813F          247   551          TAP       ,13
81D4 55F080   81F0   248   552          SB        INCCHK
81D5 8B814F          249   553          TAP       ,14
81D6 55F080   81F0   250   554          SB        INCCHK
81D7 0B815F          251   555          TAP       ,15
81D8 55F080   81F0   252   556          SB        INCCHK
81D9 0B816F          253   557          TAP       ,16
81DA 55F080   81F0   254   558          SB        INCCHK
81DB 8B817F          255   559          TAP       ,17
81DC 55F080   81F0   256   560          SB        INCCHK
81DD 8B818F          257   561          TAP       ,18
81DE 55F080   81F0   258   562          SB        INCCHK
81DF 0B819F          259   563          TAP       ,19
81E0 55F080   81F0   260   564          SB        INCCHK
81E1 0B81AF          261   565          TAP       ,1A
81E2 55F080   81F0   262   566          SB        INCCHK
81E3 8B81BF          263   567          TAP       ,1B
81E4 55F080   81F0   264   568          SB        INCCHK
81E5 0B81CF          265   569          TAP       ,1C
81E6 55F080   81F0   266   570          SB        INCCHK
81E7 8B81DF          267   571          TAP       ,1D
81E8 55F080   81F0   268   572          SB        INCCHK
81E9 8B81EF          269   573          TAP       ,1E
81EA 55F080   81F0   270   574          SB        INCCHK
```

```
81EB 0B81FF          271   575          TAP       ,1F
81EC 55F080   81F0     272   576          SB        INCCHK
                       273   577  *
                       274   578  *
81ED 000FEF            275   579  AUXINC  OR        +,,              INCREMENT PC'S BY 1
81EE A00908            276   580          MV        PL,PH            DUPLICATE LOW PC IN THE HIGH PC
81EF 87800F            277   581          SR        ,
                       278   582  *
                       279   583  *
81F0 59F608   81F6     280   584  INCCHK  BNR       F0,PL,ERROR8     COMPARITOR SHOULD = AUX DATA IN PC'S
81F1 D9F609   81F6     281   585          BNR       F0,PH,ERROR8
81F2 AC0010            282   586          AI        1,F0,F0
81F3 79F500   81F5     283   587          BNEL      0,F0,*+2
81F4 75F820   81F8     284   588          BEQH      2,F0,TEST9       IF COMPARITOR = 20 GO TO NEXT TEST
81F5 87800F            285   589          SR        ,
                       286   590  *
                       287   591  *
81F6 D72480   8324     288   592  ERROR8  SB        ERROR            PRINT "ERROR" ON CRT
81F7 DD6A80   816A     289   593          B         TEST8
                       290   594  *
                       291   595  *
```

```
81F8 20C09F          1   597  TEST9   MVI     39,F0       PUT A "9" IN F0
81F9 D71480  8314    2   598          SB      TEST#       PRINT "TEST#9" ON CRT
81FA 1D0203  8203    3   599          LPI     *+9         PUT SR,WCM ADDR. +1 IN PC'S
81FB 05800F          4   600          TPS     ,           PUSH ADDR. ON STACK
81FC 994000  1000    5   601          LPI     1000        LOAD PC'S WITH ADDR. TO BE WRITTEN
81FD 05800F          6   602          TPS     ,           PUSH ADDR. ON STACK
81FE 220E0F          7   603          MVI     80,K        SET K = TO ALL 0'S EXCEPT FOR PARITY BIT
81FF A7CEFE          8   604          XORI    0FF,K,K     COMPLEMENT K FOR WRITE
8200 A0080F          9   605          MVI     00,PL       SET PL = TO ALL 0'S
8201 20090F          10  606          MVI     00,PH       SET PH = TO ALL 0'S
8202 078400          11  607          SR,WCM  ,           WRITE LOC. 1000 IN CONT MEM WITH ALL 0'S
8203 A14EAF          12  608          MVI     5A,K        PUT GARBAGE IN K
8204 A148AF          13  609          MVI     5A,PL       PUT GARBAGE IN PL
8205 2149AF          14  610          MVI     5A,PH       PUT GARBAGE IN PH
8206 9D020B  820B    15  611          LPI     *+5         LOAD PC'S WITH SR,RCM ADDR. + 1
8207 05800F          16  612          TPS     ,           PUSH ADDR. ON STACK
8208 994000  1000    17  613          LPI     1000        LOAD PC'S WITH ADDR. 1000
8209 05800F          18  614          TPS     ,           PUSH ADDR. ON STACK
820A 878600          19  615          SR,RCM  ,           READ CONTROL MEMORY
820B 7A290E  8229    20  616          BNEL    0,K,ERROR9  CHECK THAT CONT. MEM. DATA OK
820C 7E298E  8229    21  617          BNEH    8,K,ERROR9
820D 7A2908  8229    22  618          BNEL    0,PL,ERROR9
820E FE2908  8229    23  619          BNEH    0,PL,ERROR9
820F FA2909  8229    24  620          BNEL    0,PH,ERROR9
8210 7E2909  8229    25  621          BNEH    0,PH,ERROR9
                     26  622  *
                     27  623  *
8211 9D021A  821A    28  624          LPI     *+9         PUT SR,WCM ADDR. +1 IN PC'S
8212 05800F          29  625          TPS     ,           PUSH ADDR. ON STACK
8213 994000  1000    30  626          LPI     1000        LOAD PC'S WITH ADDR. TO BE WRITTEN
8214 05800F          31  627          TPS     ,           PUSH ADDR. ON STACK
8215 21CEFF          32  628          MVI     7F,K        SET K = TO ALL 1'S EXCEPT FOR PARITY BIT
8216 A7CEFE          33  629          XORI    0FF,K,K     COMPLEMENT K FOR WRITE
8217 A3C8FF          34  630          MVI     0FF,PL      SET PL = TO ALL 1'S
8218 23C9FF          35  631          MVI     0FF,PH      SET PH = TO ALL 1'S
8219 078400          36  632          SR,WCM  ,           WRITE LOC. 1000 IN CONT MEM WITH ALL 1'S
821A A14EAF          37  633          MVI     5A,K        PUT GARBAGE IN K
821B A148AF          38  634          MVI     5A,PL       PUT GARBAGE IN PL
821C 2149AF          39  635          MVI     5A,PH       PUT GARBAGE IN PH
821D 1D0222  8222    40  636          LPI     *+5         LOAD PC'S WITH SR,RCM ADDR. + 1
821E 05800F          41  637          TPS     ,           PUSH ADDR. ON STACK
821F 994000  1000    42  638          LPI     1000        LOAD PC'S WITH ADDR. 1000
8220 05800F          43  639          TPS     ,           PUSH ADDR. ON STACK
8221 878600          44  640          SR,RCM  ,           READ CONTROL MEMORY
8222 7A29FE  8229    45  641          BNEL    0F,K,ERROR9 CHECK THAT CONT. MEM. DATA OK
```

```
8223 7E297E  8229   46   642          BNEH     7,K,ERROR9
8224 7A29F8  8229   47   643          BNEL     OF,PL,ERROR9
8225 FE29F8  8229   48   644          BNEH     OF,PL,ERROR9
8226 FA29F9  8229   49   645          BNEL     OF,PH,ERROR9
8227 7E29F9  8229   50   646          BNEH     OF,PH,ERROR9
8228 DE2B80  822B   51   647          B        TESTA          GO TO NEXT TEST
                    52   648  *
                    53   649  *
8229 D72480  8324   54   650  ERROR9  SB       ERROR
822A 5DF880  81F8   55   651          B        TEST9
                    56   652  *
                    57   653  *TEST FOR PARITY ERROR CONT. MEM.
                    58   654  *
822B 21001F         59   655  TESTA   MVI      41,F0          PUT "A" IN F0
822C D71480  8314   60   656          SB       TEST#          PRINT TEST#A ON CRT
822D D7A980  83A9   61   657          SB       BDPARY         EXECUTE INSTRUCTIONS WRITTEN WITH BAD PARITY
822E 5E2F80  822F   62   658  END     B        INITB          GO TO NEXT TEST
                    63   659  *
                    64   660  *TEST INSTRUCTION COUNTER INPUT TO STACK
                    65   661  *
822F AC0616         66   662  INITB   AI       1,F6,F6
8230 AC0717         67   663          AI       1,F7,F7
8231 21002F         68   664  TESTB   MVI      42,F0
8232 D71480  8314   69   665          SB       TEST#          PRINT TEST#8 ON CRT
8233 190000  0000   70   666          LPI      0000           LOAD PC'S = TO 0
8234 81800F         71   667          TPA      ,00            INIT ADDR. POINTER
8235 01804F         72   668          TPA      ,04            INIT ADDR. COUNTER
8236 8B800F         73   669  FLVCHK  TAP      ,00            GET ADDR. POINTER
8237 A00508         74   670          MV       PL,F5          PUT PC LOW IN FILE REG 5 FOR TEST
8238 280515         75   671          ANDI     01,F5,F5       MASK OFF ALL BUT BIT 0
8239 F24015  8240   76   672          BEQL     01,F5,WRODD    IF ADDR. IS ODD GO WRITE ODD
                    77   673  *
                    78   674  *
823A 1D027E  827E   79   675  WREVEN  LPI      RETURN1        PUT RETURN ADDR. IN PC'S
823B 01801F         80   676          TPA      ,01            SAVE RETURN ADDR. IN AUX 1
823C 9D0246  8246   81   677          LPI      PCCNTL         LOAD PC'S NEXT ROUTINE ADDR.
823D 01802F         82   678          TPA      ,02            SAVE ADDR. IN AUX 2
823E D67F80  827F   83   679          SB       INSTACK        SET UP STACK FOR WRITE CONT. MEM.
823F 5E7980  8279   84   680          B        WRTNOP         EVEN ADDR. GO WRITE A NOP HERE
                    85   681  *
                    86   682  *
8240 9D026B  826B   87   683  WRODD   LPI      RETURN3        PUT RETURN ADDR. IN PC'S
8241 01801F         88   684          TPA      ,01            SAVE ADDR. AUX 1
8242 9D0246  8246   89   685          LPI      PCCNTL         LOAD PC'S WITH ADDR. OF NEXT ROUTINE
8243 01802F         90   686          TPA      ,02            SAVE ADDR. IN AUX 2
```

```
8244 D67F80  827F   91   687          SB      INSTACK          SET UP STACK FOR WRITE CONT. MEM.
8245 DE6680  8266   92   688          B       WRTSB1           ODD ADDR. GO WRITE SUB BRANCH "TSTADD"
                    93   689  *
                    94   690  *
8246 8B800F         95   691  PCCNTL  TAP     ,00              GET ADDR. POINTER
8247 000FEF         96   692          OR      +,,              INCREMENT POINTER BY 1
8248 0200E8         97   693          MVX     PHPL,F1F0        MOVE PC'S TO FILE REGS FOR TEST
8249 81800F         98   694          TPA     ,00              SAVE UPDATED ADDR. POINTER
824A DA3606  8236   99   695          BNR     F0,F6,FLVCHK     CHECK TO SEE IF ADDR. = "00"
824B 524D17  824D  100   696          BER     F1,F7,CMEXEC     SEE IF PC'S = TO LAST VALID CM ADDR.
824C DE3680  8236  101   697          B       FLVCHK           GO CHECK ODD OR EVEN THEN WRITE
                   102   698  *
                   103   699  *
824D 190000  0000  104   700  CMEXEC  LPI     0000             LOAD PC'S TO = 0
824E 05800F        105   701          TPS     ,                PUSH 0 ON THE STACK
824F 87800F        106   702          SR      ,                START PROGRAM EXECUTION AT LOCATION 0
                   107   703  *
                   108   704  *
8250 0B804F        109   705  CMCONT  TAP     ,04              PUT ADDR. COUNTER IN PC'S
8251 000FEF        110   706          OR      +,,              STEP COUNT BY 1
8252 000FEF        111   707          OR      +,,              STEP COUNT BY 1 (TOTAL STEPS = 2)
8253 01804F        112   708          TPA     ,04              SAVE UPDATED COUNT
8254 8B800F        113   709          TAP     ,00              GET NEXT CM ADDR.
8255 05800F        114   710          TPS     ,                PUSH ADDR. ON STACK
8256 87800F        115   711          SR      ,                POP STACK START EXECUTION AT CM ADDR.
                   116   712  *
                   117   713  *
8257 8D800F        118   714  TSTADD  TSP     ,                SAVE UPDATED CM ADDR. RESULTING FROM SB
8258 81800F        119   715          TPA     ,00              STORE ADDR.. AUX 0
8259 0B804F        120   716          TAP     ,04              GET ADDR. COUNTER
825A 8202E8        121   717          MVX     PHPL,F3F2        MOVE ADDR. COUNTER TO FILE REGS
825B 8B800F        122   718          TAP     ,00              GET UPDATED CM ADDR.
825C 800FFF        123   719          OR      -,,              DECREMENT ADDR. POINTER BY 1
825D 800FFF        124   720          OR      -,,              DECREMENT ADDR. POINTER BY 1 (TOTAL -2)
825E 5A6439  8264  125   721          BNR     F3,PH,ERRORB     COMPARE TO SEE IF COUNT = TO ADDR.
825F 5A6428  8264  126   722          BNR     F2,PL,ERRORB     COMPARE TO SEE IF COUNT = TO ADDR.
                   127   723  *
                   128   724  *
8260 8B800F        129   725  TSTEND  TAP     ,00              GET UPDATED CM ADDR. PUT ADDR. IN PC'S
8261 5A5068  8250  130   726          BNR     F6,PL,CMCONT     LAST CONT MEM ADDR. LOW ORDER?
8262 D26C79  826C  131   727          BER     F7,PH,RESMAX     LAST CONT MEM ADDR HIGH ORDER?
8263 DE5080  8250  132   728          B       CMCONT           NOT LAST ADDR. GO EXECUTE AGAIN
                   133   729  *
                   134   730  *
8264 D72480  8324  135   731  ERRORB  SB      ERROR
```

```
8265 5E3180  8231   136  732          B       TESTB
                    137  733  *
                    138  734  *
8266 234E6F         139  735  WRTSB1  MVI     0D6,K          PUT OP CODE FOR SUB BRANCH INSTR IN K
8267 A7CEFE         140  736          XORI    0FF,K,K        COMPLEMENT K FOR CONTROL MEM WRITE
8268 22080F         141  737          MVI     80,PL          PUT REMAINDER OF INST CODE IN PC,S
8269 A1497F         142  738          MVI     57,PH
826A 078400         143  739          SR,WCM  ,              WRITE SUB BRANCH AT CM ADDR.
826B 87800F         144  740  RETURN3 SR      ,              RETURN
                    145  741  *
                    146  742  *
826C A7C6F6         147  743  RESMAX  XORI    0FF,F6,F6      COMPLEMENT F6 SHOULD BE = TO "FF"
826D FA64F6  8264   148  744          BNEL    0F,F6,ERRORB   IS LOW ORDER F6 = TO "F"
826E 7E64F6  8264   149  745          BNEH    0F,F6,ERRORB   IS HIGH ORDER F6 = TO "F"
826F F67547  8275   150  746          BEQH    4,F7,COMPM     CHECK TO SEE IF F7 = TO "04" (16K MEM)
8270 FE6457  8264   151  747          BNEH    5,F7,ERRORB    IF F7 NOT = TO "05" (20K MEM) ERROR
8271 2447F7         152  748          XORI    1F,F7,F7       F7 = TO "05" CHANGE TO "4F"
8272 7A64F7  8264   153  749          BNEL    0F,F7,ERRORB   IS LOW ORDER F7 = TO "F"
8273 7E6447  8264   154  750          BNEH    4,F7,ERRORB    IS HIGH ORDER F7 = TO "4"
8274 DE7880  8278   155  751          B       *+4            RESTART PROGRAM
8275 25C7F7         156  752  COMPM   XORI    7F,F7,F7       F7 = TO "04" CHANGE TO "3F"
8276 7A64F7  8264   157  753          BNEL    0F,F7,ERRORB   IS F7 LOW ORDER = TO "F"
8277 FE6437  8264   158  754          BNEH    3,F7,ERRORB    IS F7 HIGH ORDER = TO "3"
8278 5EDA80  82DA   159  755          B       TESTC          RESTART PROGRAM
                    160  756  *
                    161  757  *
8279 208E0F         162  758  WRTNOP  MVI     20,K           PUT FIRST BYTE OF NOP INST IN K REG.
827A A7CEFE         163  759          XORI    0FF,K,K        COMPLEMENT K FOR CONTROL MEM WRITE
827B A008FF         164  760          MVI     0F,PL          PUT NEXT BYTE IN PL
827C 2009FF         165  761          MVI     0F,PH          PUT LAST BYTE IN PH (200F0F=NOP)
827D 078400         166  762          SR,WCM  ,              WRITE A NOP AT CONT. MEM. ADDR.
827E 87800F         167  763  RETURN1 SR      ,              RETURN
                    168  764  *
                    169  765  *
827F 8D800F         170  766  INSTACK TSP     ,              SAVE STACK
8280 81803F         171  767          TPA     ,03            SAVE PC'S FROM SUB BRANCH
8281 0B802F         172  768          TAP     ,02            PUT NEXT ROUTINE ADDR. IN PC'S
8282 05800F         173  769          TPS     ,              PUSH ADDR. ON STACK
8283 0B801F         174  770          TAP     ,01            PUT SR,WCM ADDR. +1 IN PC'S
8284 05800F         175  771          TPS     ,              PUSH ADDR. ON STACK
8285 8B800F         176  772          TAP     ,00            PUT WRITE ADDR. IN PC'S
8286 05800F         177  773          TPS     ,              PUSH ADDR. ON STACK
8287 8B803F         178  774          TAP     ,03            BRING BACK RETURN ADDR.
8288 05800F         179  775          TPS     ,              PUSH PC'S ON STACK
8289 87800F         180  776          SR      ,              RETURN
```

```
                        181   777  *
                        182   778  *
828A D6F280   82F2      183   779  CMAMNT   SB        ENABLE5
828B 210E9F             184   780           MVI       C'I',K
828C 56FA80   82FA      185   781           SB        OBSTROBE
828D A10EEF             186   782           MVI       C'N',K
828E 56FA80   82FA      187   783           SB        OBSTROBE
828F A14E0F             188   784           MVI       C'P',K
8290 56FA80   82FA      189   785           SB        OBSTROBE
8291 A14E5F             190   786           MVI       C'U',K
8292 56FA80   82FA      191   787           SB        OBSTROBE
8293 214E4F             192   788           MVI       C'T',K
8294 56FA80   82FA      193   789           SB        OBSTROBE
8295 D70A80   830A      194   790           SB        SPACE
8296 A10E1F             195   791           MVI       C'A',K
8297 56FA80   82FA      196   792           SB        OBSTROBE
8298 A10EDF             197   793           MVI       C'M',K
8299 56FA80   82FA      198   794           SB        OBSTROBE
829A 210EFF             199   795           MVI       C'O',K
829B 56FA80   82FA      200   796           SB        OBSTROBE
829C A14E5F             201   797           MVI       C'U',K
829D 56FA80   82FA      202   798           SB        OBSTROBE
829E A10EEF             203   799           MVI       C'N',K
829F 56FA80   82FA      204   800           SB        OBSTROBE
82A0 214E4F             205   801           MVI       C'T',K
82A1 56FA80   82FA      206   802           SB        OBSTROBE
82A2 D70A80   830A      207   803           SB        SPACE
82A3 210EFF             208   804           MVI       C'O',K
82A4 56FA80   82FA      209   805           SB        OBSTROBE
82A5 210E6F             210   806           MVI       C'F',K
82A6 56FA80   82FA      211   807           SB        OBSTROBE
82A7 D70A80   830A      212   808           SB        SPACE
82A8 210E3F             213   809           MVI       C'C',K
82A9 56FA80   82FA      214   810           SB        OBSTROBE
82AA 210EFF             215   811           MVI       C'O',K
82AB 56FA80   82FA      216   812           SB        OBSTROBE
82AC A10EEF             217   813           MVI       C'N',K
82AD 56FA80   82FA      218   814           SB        OBSTROBE
82AE 214E4F             219   815           MVI       C'T',K
82AF 56FA80   82FA      220   816           SB        OBSTROBE
82B0 214E2F             221   817           MVI       C'R',K
82B1 56FA80   82FA      222   818           SB        OBSTROBE
82B2 210EFF             223   819           MVI       C'O',K
82B3 56FA80   82FA      224   820           SB        OBSTROBE
82B4 210ECF             225   821           MVI       C'L',K
```

```
82B5 56FA80  82FA   226   822          SB        OBSTROBE
82B6 D70A80  830A   227   823          SB        SPACE
82B7 A10EDF         228   824          MVI       C'M',K
82B8 56FA80  82FA   229   825          SB        OBSTROBE
82B9 210E5F         230   826          MVI       C'E',K
82BA 56FA80  82FA   231   827          SB        OBSTROBE
82BB A10EDF         232   828          MVI       C'M',K
82BC 56FA80  82FA   233   829          SB        OBSTROBE
82BD A08EEF         234   830          MVI       C'.',K
82BE D70A80  830A   235   831          SB        SPACE
82BF A08E8F         236   832          MVI       C'(',K
82C0 56FA80  82FA   237   833          SB        OBSTROBE
82C1 A0CE0F         238   834          MVI       C'0',K
82C2 56FA80  82FA   239   835          SB        OBSTROBE
82C3 20CEDF         240   836          MVI       C'=',K
82C4 56FA80  82FA   241   837          SB        OBSTROBE
82C5 20CE1F         242   838          MVI       C'1',K
82C6 56FA80  82FA   243   839          SB        OBSTROBE
82C7 A0CE6F         244   840          MVI       C'6',K
82C8 56FA80  82FA   245   841          SB        OBSTROBE
82C9 A10EBF         246   842          MVI       C'K',K
82CA 56FA80  82FA   247   843          SB        OBSTROBE
82CB D70A80  830A   248   844          SB        SPACE
82CC 20CE1F         249   845          MVI       C'1',K
82CD 56FA80  82FA   250   846          SB        OBSTROBE
82CE 20CEDF         251   847          MVI       C'=',K
82CF 56FA80  82FA   252   848          SB        OBSTROBE
82D0 20CE2F         253   849          MVI       C'2',K
82D1 56FA80  82FA   254   850          SB        OBSTROBE
82D2 A0CE0F         255   851          MVI       C'0',K
82D3 56FA80  82FA   256   852          SB        OBSTROBE
82D4 A10EBF         257   853          MVI       C'K',K
82D5 56FA80  82FA   258   854          SB        OBSTROBE
82D6 208E9F         259   855          MVI       C')',K
82D7 56FA80  82FA   260   856          SB        OBSTROBE
82D8 D70A80  830A   261   857          SB        SPACE
82D9 87800F         262   858          SR        ,
                    263   859  *
                    264   860  *DATA MEMORY SELECT (1,2) TEST
                    265   861  *
82DA A1003F         266   862  TESTC    MVI       43,F0        PUT A "C" IN F0
82DB D71480  8314   267   863          SB        TEST#        PRINT "TEST#C" ON CRT
82DC 2140AF         268   864          MVI       5A,F0        PUT PATTERN #1 IN F0
82DD 190000  0000   269   865          LPI       0000         SET UP DATA MEMORY ADDR. FOR WRITE (DMS1)
82DE 56E780  82E7   270   866          SB        WRTDM        GO WRITE "5A" IN LOCATIONS 0 AND 1
```

```
82DF 190000  0000  271  867        LPI     0000          RESET MEMORY ADDRESS
82E0 56EB80  82EB  272  868        SB      READDM        READ AND CHECK DATA FROM MEMORY
82E1 9D0000  8000  273  869        LPI     8000          SET UP NEW MEMORY ADDR. FOR WRITE (DMS2)
82E2 56E780  82E7  274  870        SB      WRTDM         GO WRITE A "5A" IN LOCS. 8000 AND 8001
82E3 9D0000  8000  275  871        LPI     8000          RESET MEMORY ADDRESS
82E4 56EB80  82EB  276  872        SB      READDM        READ AND CHECK DATA FROM MEMORY
82E5 D6FE80  82FE  277  873        SB      INIT          CLEAR SCREEN AND HOME CUSOR
82E6 DC6180  8061  278  874        B       TEST0         RESTART PROGRAM
                   279  875  *
                   280  876  *
82E7 802FE0        281  877  WRTDM  OR,W1   +,F0,         WRITE DATA MEMORY WITH F0, INCR. PC'S 1
82E8 802FE0        282  878        OR,W1   +,F0,         WRITE DATA MEMORY WITH F0
82E9 579780  8397  283  879        SB      SEC1          DELAY ONE SECOND
82EA 87800F        284  880        SR      ,             RETURN
                   285  881  *
                   286  882  *
82EB 2140AF        287  883  READDM  MVI     5A,F0         PUT TEST PATTERN IN F0
82EC A01F0F        288  884        ORI,R   0,,           READ DATA MEMORY
82ED DAF00A  82F0  289  885        BNR     F0,CL,ERRORC  CHECK DATA READ FOR ERROR
82EE 5AF00B  82F0  290  886        BNR     F0,CH,ERRORC  CHECK DATA READ FOR ERROR
82EF 87800F        291  887        SR      ,             RETURN
                   292  888  *
                   293  889  *
82F0 D72480  8324  294  890  ERRORC  SB      ERROR         PRINT "ERROR" ON CRT
82F1 5EDA80  82DA  295  891        B       TEST0         RESTART TEST
```

```
82F2 A00E5F        1    893  ENABLE5  MVI   5,K          SET CRT ADDR.
82F3 178C00        2    894  ENABLE   CIO   0C0          ADDR. STROBE
82F4 56F580 82F5   3    895  DELAY10  SB    DELAY5
82F5 56F680 82F6   4    896  DELAY5   SB    *+1
82F6 D6F780 82F7   5    897           SB    *+1
82F7 200F0F        6    898           NOP   ,
82F8 200F0F        7    899           NOP   ,
82F9 87800F        8    900           SR    ,
                   9    901  *
                   10   902  *
82FA EAFA8D 82FA   11   903  OBSTROBE BFL   8,SH,*       WAIT FOR DEVIE READY
82FB 56F580 82F5   12   904           SB    DELAY5
82FC 978200        13   905           CIO   20
82FD DEF580 82F5   14   906           B     DELAY5
                   15   907  *
                   16   908  *
82FE A00DBD        17   909  INIT     ORI   0B,SH,SH     INHIBIT INPUT
82FF 284DFD        18   910           ANDI  1F,SH,SH     SET NO TRAP,40BIT,HALT/STEP OFF
8300 D6F280 82F2   19   911           SB    ENABLE5
8301 A00E3F        20   912           MVI   03,K
8302 56FA80 82FA   21   913           SB    OBSTROBE     CLEAR CRT
8303 87800F        22   914           SR    ,
                   23   915  *
                   24   916  *
8304 D6F280 82F2   25   917  CRLF     SB    ENABLE5      THIS ROUTINE PRINTS CARRIAGE RETURN LINE FEED
8305 200EDF        26   918           MVI   0D,K
8306 56FA80 82FA   27   919           SB    OBSTROBE     CR
8307 A00EAF        28   920           MVI   0A,K
8308 56FA80 82FA   29   921           SB    OBSTROBE     LF
8309 87800F        30   922           SR    ,
                   31   923  *
                   32   924  *
830A D6F280 82F2   33   925  SPACE    SB    ENABLE5
830B 208E0F        34   926           MVI   20,K
830C 56FA80 82FA   35   927           SB    OBSTROBE     PRINT A SPACE
830D 87800F        36   928           SR    ,
                   37   929  *
                   38   930  *
830E D6F280 82F2   39   931  INITCRT  SB    ENABLE5
830F 200E1F        40   932           MVI   01,K
8310 56FA80 82FA   41   933           SB    OBSTROBE     HOME CUSOR
8311 A00E3F        42   934           MVI   03,K
8312 56FA80 82FA   43   935           SB    OBSTROBE     CLR CRT DISPLAY
8313 87800F        44   936           SR    ,
                   45   937  *
```

```
                        46   938  *
8314 D6F280  82F2       47   939  TEST#    SB       ENABLE5
8315 214E4F             48   940           MVI      54,K
8316 56FA80  82FA       49   941           SB       OBSTROBE      T
8317 210E5F             50   942           MVI      45,K
8318 56FA80  82FA       51   943           SB       OBSTROBE      E
8319 A14E3F             52   944           MVI      53,K
831A 56FA80  82FA       53   945           SB       OBSTROBE      S
831B 214E4F             54   946           MVI      54,K
831C 56FA80  82FA       55   947           SB       OBSTROBE      T
831D 208E3F             56   948           MVI      23,K
831E 56FA80  82FA       57   949           SB       OBSTROBE      #
831F D70A80  830A       58   950           SB       SPACE         SPACE
8320 573280  8332       59   951           SB       PRTBYTE       TEST NUMBER X IS PRINTED
8321 200F0F             60   952           NOP      ,
8322 570480  8304       61   953           SB       CRLF
8323 87800F             62   954           SR       ,
                        63   955  *
                        64   956  *
8324 D6F280  82F2       65   957  ERROR    SB       ENABLE5
8325 210E5F             66   958           MVI      45,K
8326 56FA80  82FA       67   959           SB       OBSTROBE      E
8327 214E2F             68   960           MVI      52,K
8328 56FA80  82FA       69   961           SB       OBSTROBE      R
8329 214E2F             70   962           MVI      52,K
832A 56FA80  82FA       71   963           SB       OBSTROBE      R
832B 210EFF             72   964           MVI      4F,K
832C 56FA80  82FA       73   965           SB       OBSTROBE      O
832D 214E2F             74   966           MVI      52,K
832E 56FA80  82FA       75   967           SB       OBSTROBE      R
832F 570480  8304       76   968           SB       CRLF
8330 579780  8397       77   969           SB       SEC1          WAIT APP. ONE SECOND FOR DISPLAY PURPOSES
8331 87800F             78   970           SR       ,
                        79   971  *
8332 D6F280  82F2       80   972  PRTBYTE  SB       ENABLE5
8333 A00E00             81   973           MV       F0,K          THIS IS A ROUTINE TO PRINT FILE REG 0
8334 56FA80  82FA       82   974           SB       OBSTROBE
8335 87800F             83   975           SR       ,
                        84   976  *
8336 21430F             85   977  SYSPE    MVI      C'P',F3       THESE ROUTINES PRINT PARITY ERROR ADDR.
8337 21025F             86   978  SYSVE    MVI      C'E',F2
8338 2100DF             87   979           MVI      C'M',F0
8339 D6F280  82F2       88   980  SYSERROR SB       ENABLE5
833A 200E1F             89   981           MVI      01,K
833B 56FA80  82FA       90   982           SB       OBSTROBE
```

```
833C 208E0F        91   983            MVI     20,K
833D A0040F        92   984            MVI     00,F4
833E 56FA80 82FA   93   985  SPACEOUT SB      OBSTROBE
833F 98C44F        94   986            AC,,1   F4,,F4
8340 7F3E54 833E   95   987            BNEH    5,F4,SPACEOUT
8341 200E1F        96   988            MVI     01,K
8342 56FA80 82FA   97   989            SB      OBSTROBE
8343 208EAF        98   990            MVI     C'*',K
8344 56FA80 82FA   99   991            SB      OBSTROBE
8345 56FA80 82FA  100   992            SB      OBSTROBE
8346 56FA80 82FA  101   993            SB      OBSTROBE
8347 D70A80 830A  102   994            SB      SPACE
8348 A14E3F       103   995            MVI     C'S',K
8349 56FA80 82FA  104   996            SB      OBSTROBE
834A A14E9F       105   997            MVI     C'Y',K
834B 56FA80 82FA  106   998            SB      OBSTROBE
834C A14E3F       107   999            MVI     C'S',K
834D 56FA80 82FA  108  1000            SB      OBSTROBE
834E 214E4F       109  1001            MVI     C'T',K
834F 56FA80 82FA  110  1002            SB      OBSTROBE
8350 210E5F       111  1003            MVI     C'E',K
8351 56FA80 82FA  112  1004            SB      OBSTROBE
8352 A10EDF       113  1005            MVI     C'M',K
8353 56FA80 82FA  114  1006            SB      OBSTROBE
8354 D70A80 830A  115  1007            SB      SPACE
8355 210E5F       116  1008            MVI     C'E',K
8356 56FA80 82FA  117  1009            SB      OBSTROBE
8357 214E2F       118  1010            MVI     C'R',K
8358 56FA80 82FA  119  1011            SB      OBSTROBE
8359 214E2F       120  1012            MVI     C'R',K
835A 56FA80 82FA  121  1013            SB      OBSTROBE
835B 210EFF       122  1014            MVI     C'O',K
835C 56FA80 82FA  123  1015            SB      OBSTROBE
835D 214E2F       124  1016            MVI     C'R',K
835E 56FA80 82FA  125  1017            SB      OBSTROBE
835F D70A80 830A  126  1018            SB      SPACE
8360 A08E8F       127  1019            MVI     C'(',K
8361 56FA80 82FA  128  1020            SB      OBSTROBE
8362 A00E03       129  1021            MV      F3,K
8363 56FA80 82FA  130  1022            SB      OBSTROBE
8364 200E02       131  1023            MV      F2,K
8365 56FA80 82FA  132  1024            SB      OBSTROBE
8366 200E01       133  1025            MV      F1,K
8367 56FA80 82FA  134  1026            SB      OBSTROBE
8368 A00E00       135  1027            MV      F0,K
```

```
8369 56FA80  82FA  136  1028          SB      OBSTROBE
836A D70A80  830A  137  1029          SB      SPACE
836B 200009        138  1030          MV      PH,F0
836C D77B80  837B  139  1031          SB      PRTBITE
836D A00008        140  1032          MV      PL,F0
836E D77B80  837B  141  1033          SB      PRTBITE
836F 208E9F        142  1034          MVI     C')',K
8370 56FA80  82FA  143  1035          SB      OBSTROBE
8371 D70A80  830A  144  1036          SB      SPACE
8372 208EAF        145  1037          MVI     C'*',K
8373 56FA80  82FA  146  1038          SB      OBSTROBE
8374 56FA80  82FA  147  1039          SB      OBSTROBE
8375 56FA80  82FA  148  1040          SB      OBSTROBE
8376 DF0480  8304  149  1041          B       CRLF
                   150  1042 *
                   151  1043 *
8377 4B7921  8379  152  1044 PRTBIT1  BLER    F2,F1,*+2       THESE ROUTINES PRINT CHARS.STORED IN F0
8378 AC0272        153  1045          AI      07,F2,F2
8379 200E02        154  1046          MV      F2,K
837A DEFA80  82FA  155  1047          B       OBSTROBE
837B A0C19F        156  1048 PRTBITE  MVI     39,F1
837C 20C30F        157  1049          MVI     30,F3
837D 8C4203        158  1050          SHHH    F0,F3,F2
837E D77780  8377  159  1051          SB      PRTBIT1
837F 084203        160  1052          SHHL    F0,F3,F2
8380 5F7780  8377  161  1053          B       PRTBIT1
                   162  1054 *
                   163  1055 *
8381 8D800F        164  1056 PE24      TSP     ,               FAILING ADDR TO PC'S
8382 800FFF        165  1057          OR      -,,             DECREMENT ADDR BY 1
8383 0200E8        166  1058          MVX     PHPL,F1F0       PUT PC'S IN FILE REGS. FOR COMPARE
8384 9D03A9  83A9  167  1059          LPI     BDPARY          LOAD PC'S WITH BAD PARITY LOCATION ADDR.
8385 5B8A19  838A  168  1060          BNR     F1,PH,*+5       BRANCH IF THIS IS A REAL PARITY ERROR
8386 53AA08  83AA  169  1061          BER     F0,PL,BDPARY1   EXECUTE SECOND CMPE TEST
8387 9D03AA  83AA  170  1062          LPI     BDPARY1         LOAD PC'S WITH BAD PARITY LOCATION # 2
8388 5B8A19  838A  171  1063          BNR     F1,PH,*+2       BRANCH IF THIS SI A REAL PARITY ERROR
8389 538E08  838E  172  1064          BER     F0,PL,RETURN    PARITY TEST OK SET UP TO RETURN TO PROG.
838A 0208E0        173  1065          MVX     F1F0,PHPL       PUT PC'S BACK FOR CMPE PRINT OUT
838B 21013F        174  1066          MVI     C'C',F1         SET UP F1 TO PRINT OUT "PECM"
838C D73680  8336  175  1067          SB      SYSPE           PRINT OUT CONTROL MEM PARITY ERROR INFO
838D 5F8D80  838D  176  1068          B       *               HANG HERE UNTIL RESET IS STRUCK
                   177  1069 *
                   178  1070 *
838E 0208E0        179  1071 RETURN  MVX     F1F0,PHPL
838F AC0818        180  1072          AI      1,PL,PL
```

```
8390 05800F          181  1073          TPS     ,
8391 87800F          182  1074          SR      ,
                     183  1075 *
                     184  1076 *
8392 8D800F          185  1077 PE8       TSP     ,          FAILING ADDR. TO PC'S
8393 800FFF          186  1078          OR      -,,        DECREMENT ADDR.
8394 A1014F          187  1079          MVI     C'D',F1    SET UP F1 TO PRINT OUT "PEDM"
8395 D73680    8336  188  1080          SB      SYSPE      PRINT OUT DATA MEMORY PARITY ERROR
8396 5F9680    8396  189  1081          B       *          HANG HERE WAITING FOR RESET TO BE STRUCK
                     190  1082 *
                     191  1083 *
8397 20000F          192  1084 SEC1      MVI     00,F0      CLEAR COUNT
8398 A00E0F          193  1085 RESTRT    MVI     00,K       CLEAR COUNT
8399 AC0E1E          194  1086 WAIT1     AI      1,K,K      STEP COUNT BY 1
839A 7B9C9E    839C  195  1087          BNEL    9,K,*+2
839B F79ECE    839E  196  1088          BEQH    0C,K,*+3   CHECK FOR LOOP COUNT = 200 DECIMAL
839C D7A380    83A3  197  1089          SB      DELAY50    DELAY 50 MICRO-SECONDS
839D 5F9980    8399  198  1090          B       WAIT1      REPEAT LOOP
839E AC0010          199  1091 WAIT2     AI      1,F0,F0    STEP COUNT BY 1
839F FBA1B0    83A1  200  1092          BNEL    0B,F0,*+2
83A0 F7A210    83A2  201  1093          BEQH    1,F0,*+2   CHECK FOR LOOP COUNT = 25 DECIMAL
83A1 DF9880    8398  202  1094          B       RESTRT     EXECUTE WAIT1 LOOP AGAIN
83A2 87800F          203  1095          SR      ,          RETURN
83A3 D6F480    82F4  204  1096 DELAY50   SB      DELAY10
83A4 D6F480    82F4  205  1097          SB      DELAY10
83A5 D6F480    82F4  206  1098          SB      DELAY10
83A6 D6F480    82F4  207  1099          SB      DELAY10
83A7 D6F480    82F4  208  1100          SB      DELAY10
83A8 87800F          209  1101          SR      ,
                     210  1102 *
                     211  1103 *
83A9 1FCFFF    FFFF  212  1104 BDPARY    LPI     0FFFF      EDIT THESE LOCATIONS TO CONTAIN BAD PARITY
83AA D7AA80    83AA  213  1105 BDPARY1   SB      *          TO FORCE CONTROL MEM PARITY ERROR FOR TESTING
83AB DE2E80    822E  214  1106          B       END
                     215  1107 *END
```

NUMBER OF LINES IN ERROR = 0 NO. OF WARNINGS = 0 NO. OF SYMBOLS = 113 OVERFLOWS = 55

SYMBOL	VALUE	DEFN	REFERENCES
ADDLEVEL	80E8	0289	
AUXCHK	8124	0365	0321
AUXINC	81ED	0579	0448 0450 0452 0454 0456 0458 0460 0462 0464 0466 0468 0470 0472 0474 0476 0478 0480 0482 0484 0486 0488 0490 0492 0494 0496 0498 0500 0502 0504 0506 0508
AUXWRT	8103	0329	0320
BDPARY	83A9	1104	0657 1059
BDPARY1	83AA	1105	1061 1062
CHKSTK	80A8	0205	0208 0210
CHSTK0	80BA	0226	0231 0233
CMAMNT	828A	0779	0098
CMCONT	8250	0705	0726 0728
CMEXEC	824D	0700	0696
COMPM	8275	0752	0746
CRLF	8304	0917	0097 0099 0953 0968 1041
DECPH	8098	0186	0182
DECTST	8095	0183	0187 0189
DELAY10	82F4	0895	1096 1097 1098 1099 1100
DELAY5	82F5	0896	0895 0904 0906
DELAY50	83A3	1096	1089
ENABLE	82F3	0894	0102
ENABLE5	82F2	0893	0779 0911 0917 0925 0931 0939 0957 0972 0980
END	822E	0658	1106
ENDDIAG1	80EF	0299	0292
ERROR	8324	0957	0133 0149 0169 0190 0211 0238 0301 0439 0592 0650 0731 0890
ERROR0	806C	0133	0127
ERROR1	8079	0149	0143
ERROR2	808A	0169	0162 0163
ERROR3	809C	0190	0183 0184
ERROR4	80AE	0211	0206
ERROR5	80C6	0238	0227 0228
ERROR6	80F1	0301	0273 0274 0276 0277
ERROR7	8168	0439	0433 0434
ERROR8	81F6	0592	0584 0585
ERROR9	8229	0650	0616 0617 0618 0619 0620 0621 0641 0642 0643 0644 0645 0646
ERRORB	8264	0731	0721 0722 0744 0745 0747 0749 0750 0753 0754
ERRORC	82F0	0890	0885 0886
FLVCHK	8236	0669	0695 0697
INCCHK	81F0	0584	0514 0516 0518 0520 0522 0524 0526 0528 0530 0532 0534 0536 0538 0540 0542 0544 0546 0548 0550 0552 0554 0556 0558 0560 0562 0564 0566 0568 0570 0572 0574 0576
INCPH	8086	0165	0161

SYMBOL	VALUE	DEFN	REFERENCES
INCTST	8083	0162	0166 0168
INIT	82FE	0909	0021 0118 0873
INITB	822F	0662	0658
INITCRT	830E	0931	
INITIAL	8011	0025	0027 0028 0032 0033
INITIAL0	801C	0039	0044 0045 0052 0053
INITIAL1	802D	0059	0064 0065 0072 0073
INITIAL2	803E	0079	0087 0088 0090 0091
INPUT	8053	0103	
INSTACK	827F	0766	0679 0687
INTEST	804D	0097	0109
LOOPA	8065	0126	0131
LOOPB	8072	0142	0147
LOOPC	807F	0158	0164
LOOPD	8091	0179	0185
LOOPE	80A4	0201	0204
LOOPF	80B7	0223	0225
OBSTROBE	82FA	0903	0035 0055 0075 0093 0781 0783 0785 0787 0789 0792 0794 0796 0798 0800 0802 0805 0807 0810 0812 0814 0816 0818 0820 0822 0825 0827 0829 0833 0835 0837 0839 0841 0843 0846 0848 0850 0852 0854 0856 0913 0919 0921 0927 0933 0935 0941 0943 0945 0947 0949 0959 0961 0963 0965 0967 0974 0982 0985 0989 0991 0992 0993 0996 0998 1000 1002 1004 1006 1009 1011 1013 1015 1017 1020 1022 1024 1026 1028 1035 1038 1039 1040 1047
ONE	805D	0113	0106
ONES	80C2	0234	0232
PATCHK	8165	0433	0366 0368 0370 0372 0374 0376 0378 0380 0382 0384 0386 0388 0390 0392 0394 0396 0398 0400 0402 0404 0406 0408 0410 0412 0414 0416 0418 0420 0422 0424 0426 0428
PCCNTL	8246	0691	0677 0685
PE24	8381	1056	0002
PE8	8392	1077	0004
PRINT?	8057	0107	0104
PRTBIT1	8377	1044	1051 1053
PRTBITE	837B	1048	1031 1033
PRTBYTE	8332	0972	0108 0951
READ1	80DC	0271	0279
READDM	82EB	0883	0868 0872
READEND	80D8	0264	0261
READTEST	80E1	0276	0272
RESMAX	826C	0743	0727
RESTRT	8398	1085	1094
RETURN	838E	1071	1064
RETURN1	827E	0763	0675
RETURN3	826B	0740	0683
SEC1	8397	1084	0879 0969

SYMBOL	VALUE	DEFN	REFERENCES
SPACE	830A	0925	0790 0803 0808 0823 0831 0844 0857 0950 0994 1007 1018 1029 1036
SPACEOUT	833E	0985	0987
STACKCOM	80E3	0278	0275
START	8010	0021	0003 0005 0016
START0	8060	0118	0112 0115
START1S	80CA	0247	
START5	80B3	0219	0237
START7	80FF	0319	0308 0310 0312 0314
SYSERROR	8339	0980	
SYSPE	8336	0977	1067 1080
SYSVE	8337	0978	
TEST#	8314	0939	0123 0139 0155 0175 0196 0217 0246 0306 0445 0598 0656 0665 0863
TEST0	8061	0122	0134 0874
TEST1	806E	0138	0132 0150
TEST2	807B	0154	0148 0170
TEST3	808C	0174	0167 0191
TEST4	809E	0195	0188 0212
TEST5	80B0	0216	0209 0239
TEST6	80C8	0245	0235 0302
TEST7	80F4	0305	0300 0440
TEST8	816A	0444	0315 0593
TEST9	81F8	0597	0588 0651
TESTA	822B	0655	0647
TESTB	8231	0664	0732
TESTC	82DA	0862	0755 0891
TSTADD	8257	0714	
TSTEND	8260	0725	
WAIT1	8399	1086	1090
WAIT2	839E	1091	
WREVEN	823A	0675	
WRITE1	80D2	0258	0266 0284 0290 0298
WRITE1A	80D6	0262	0258
WRODD	8240	0683	0672
WRTDM	82E7	0877	0866 0870
WRTNOP	8279	0758	0680
WRTSB1	8266	0735	0688
ZERO	805A	0110	0105

MODULE REPAIR GUIDE NO. 4.5

EDITED BY CUSTOMER ENGINEERING DIVISION

June 22, 1978

210-6792 MODULE REPAIR

SECTION 1 INTRODUCTION

A series of test procedures has been developed in order to effectively troubleshoot and repair the 210-6792 (ALU) module.

If these test procedures are followed the 6792 module will be tested in a logical fashion which should make repair of the module quite simple.

These procedures require the use of an oscilloscope (Tektronix model 465 or equivalent), the standard 2200 VP system repair equipment (190-0718), and a set of program PROMs.

Since the 6792 module has a large amount of system control and timing logic associated with it, the first procedure is to test the ability of the module under test to statically generate the minimum required control/timing signals.

The second procedure will be to force the module to execute a Branch instruction continuously, in order to generate T1-T12 (timing signals) for test analysis.

The third procedure will be to force the module to execute a Subroutine Return which will generate the remaining timing signals (T13-T16) for test analysis.

Finally, as with all the other 2200 VP modules, the PROM program is run to exercise all other logic functions on the module.

Printed in U.S.A.

WANG LABORATORIES, INC.

ONE INDUSTRIAL AVENUE, LOWELL, MASSACHUSETTS 01851, TEL. (617) 851-4111, TWX 710 343-6769, TELEX 94-7421

SECTION 2 INSTALLATION/OPERATION

Please refer to the "Module Repair Guide No. 4A" for information needed to install and operate the equipment associated with 2200 VP system repair. The only additional equipment would be the set of test PROMs associated with the 6792 test program.

	PROM	BOARD	LOCATION
#1	378-2207	6789	L-27
#2	378-2208	6789	L-28
#3	378-2209	6789	L-29

SECTION 3 GENERAL TEST PROCEDURE INFORMATION

3.1. TEST PROCEDURES IN GENERAL

There are, as previously stated, three test procedures. All of these procedures follow the same basic format, this format being:

1. Set up the light board in some particular fashion.

2. Set up the oscilloscope.

3. Perform the scope tests referencing pictorial aids.

NOTE:
Initially, a detailed description of how to set up the light board and oscilloscope is given. After the inital setup only changes to this procedure will be given.

3.2. TEST PROCEDURE #1

Turn power off ot the system and set up the light board as follows:

Set ROM switches = 5C0380

Set CM/SWITCHES = CM

Set Address switches = 8003

Set IC/PC switch = IC

Set COMPARE/NONCOMPARE = COMPARE

Set RUN/STEP = RUN

Place the module under test on an extender module. Apply power to the system. Note the IC address, it should be = to 8003.

Set RUN/STEP = STEP

A. Check that the STEP switch asserts STOP.

Probe: L 26 pin 9 signal should be low

Set CM/SWITCHES = SWITCHES

B. Check that BRANCH asserts as a result of a branch instruction.

Probe: L 68 pin 8 signal should be low

Set CM/SWITCHES = CM

Initial scope setup:

VOLTS/DIV	2 DC (CHANNEL 1)
VERT MODE	CH 1
HORIZ DISPLAY	A
TRIG MODE	NORM
COUPLING	AC
SOURCE	CH 1
SLOPE	+
TIME/DIV	.05 MICRO SEC. (50nS)

C. Test that the MASTER CLOCK signal is generated.

Probe: Jumper near L 23 and 20 MHZ crystal.

Refer to Figure #1 for expected signal.

D. Test I/O CLK signal and associated circuitry.

Set CM/SWITCHES = SWITCHES

Set scope: TIME/DIV 1 MICRO SEC.

Probe: L 62 Pin 11

Refer to Figure #2

Set CM/SWITCHES = CM

Refer to Figure #3

Note:

Throughout these procedures references are made to signals that are being checked on IC pins. Most of these signals are outputs on this module and go to output pins on the motherboard. It is important that these signals are also checked on these pins to eliminate the possibility of having an open etch.

FIGURE 1
MASTER CLOCK

FIGURE 2
I/O CLOCK

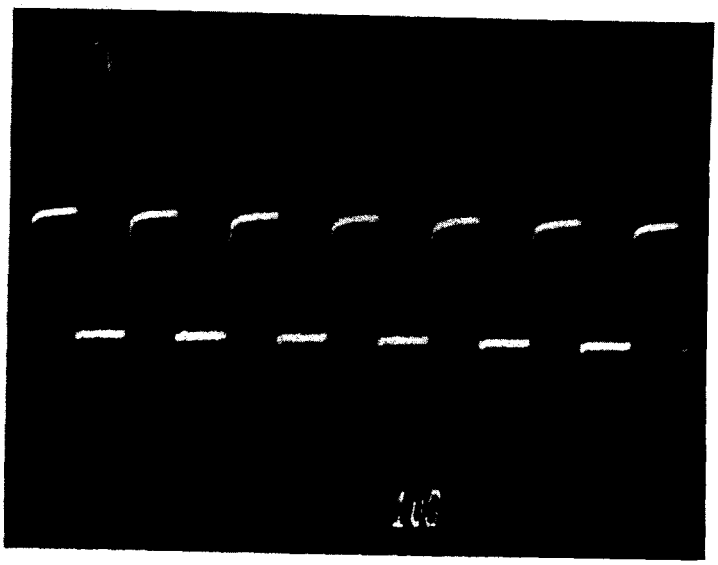
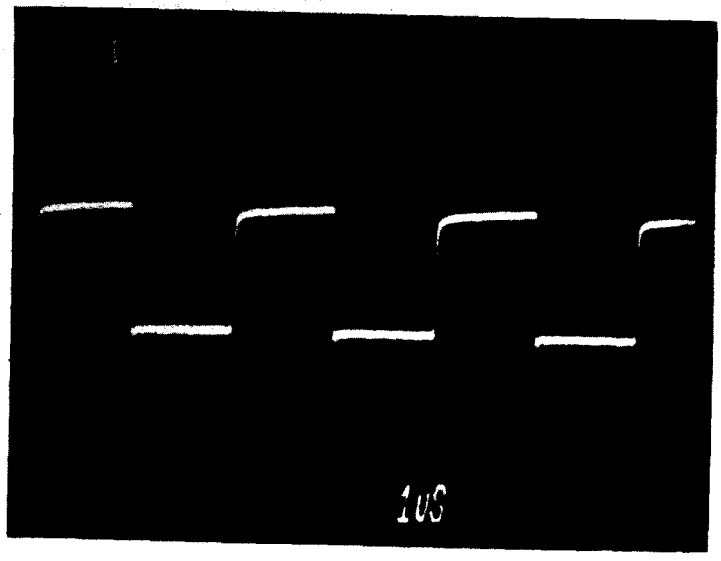


FIGURE 3
I/O CLOCK



E. Test MARCKL and associated circuitry.

Set CM/SWITCHES = SWITCHES

Set scope: TIME/DIV .5 MICRO SEC. (500 nS)

Probe: L 40 pin 3

Refer to Figure #4

Set CM/SWITCHES = CM

Refer to Figure #5

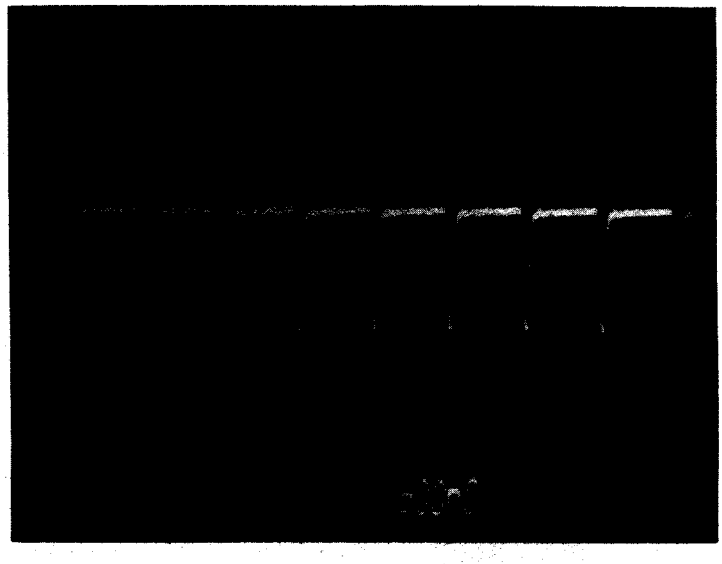


FIGURE 4
MARCLK

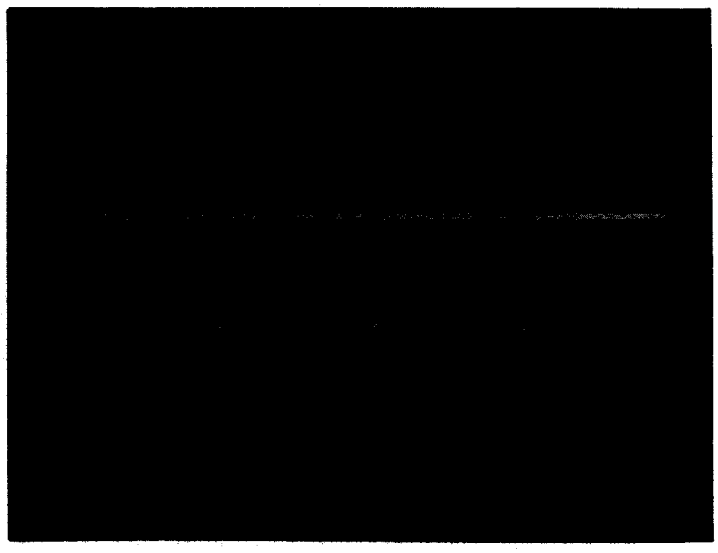


FIGURE 5
MARCLK

F. Test CE and associated circuitry.

Set CM/SWITCHES = SWITCHES

Probe: L 39 pin 6

Refer to Figure #6

Set CM/SWITCHES = CM

Refer to Figure #7

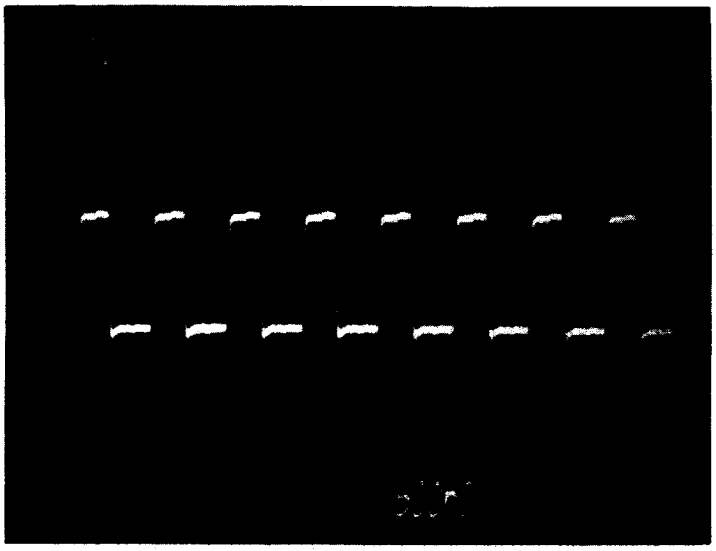


FIGURE 6
CE

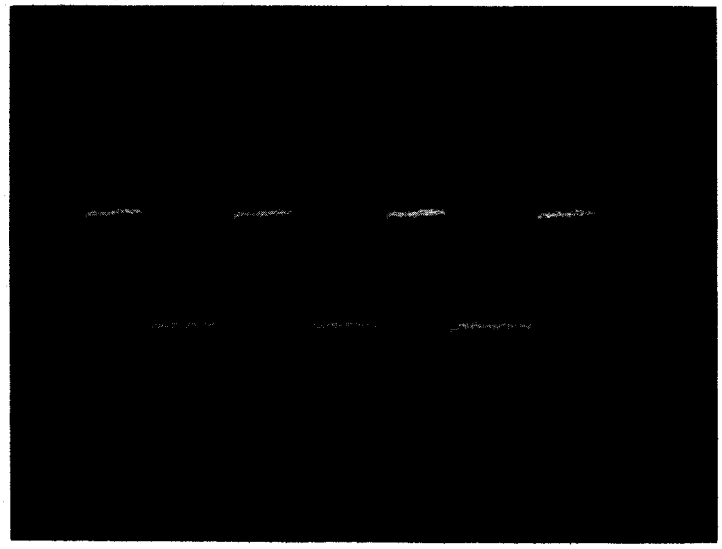


FIGURE 7
CE

G. Test REF and associated circuitry.

Set CM/SWITCHES = SWITCHES

Set scope: TIME/DIV .2 Micro Sec. (200nS)

SLOPE

Probe L 37 pin 6

Refer to Figure #8

Set CM/SWITCHES = CM

Refer to Figure #10

Set scope: TIME/DIV 5 MICRO SEC.

Refer to Figure #11

Set CM/SWITCHES = SWITCHES

Refer to Figure #9

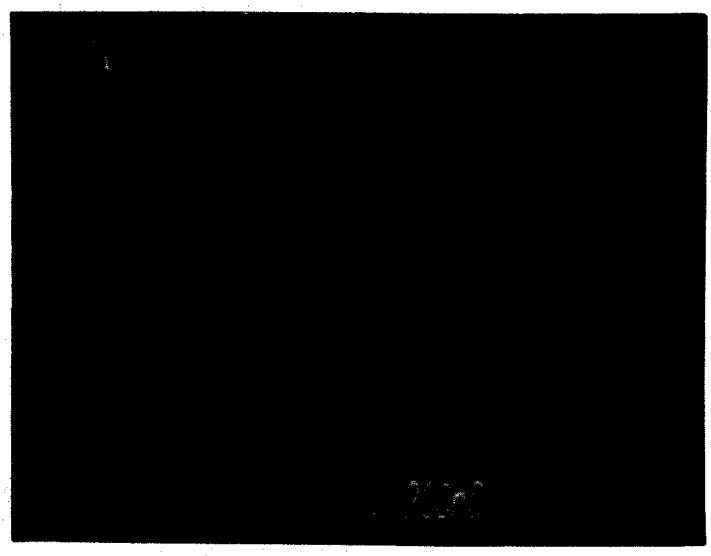


FIGURE 8 REF

FIGURE 9 REF

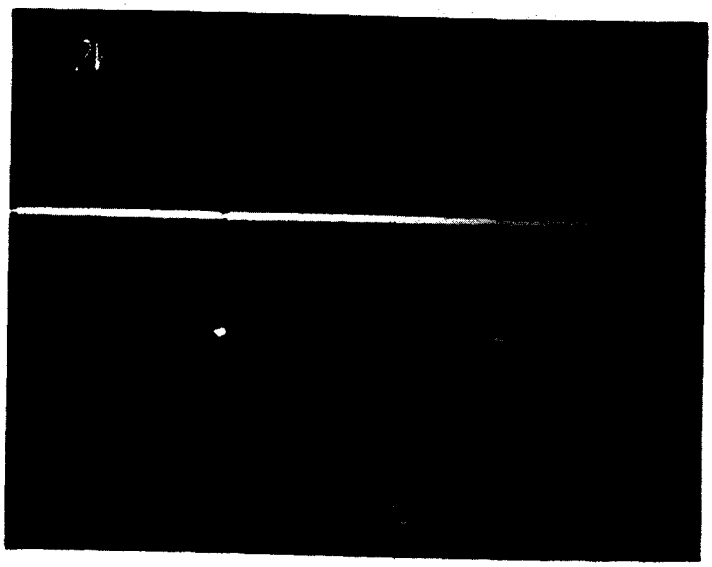


FIGURE 10 REF

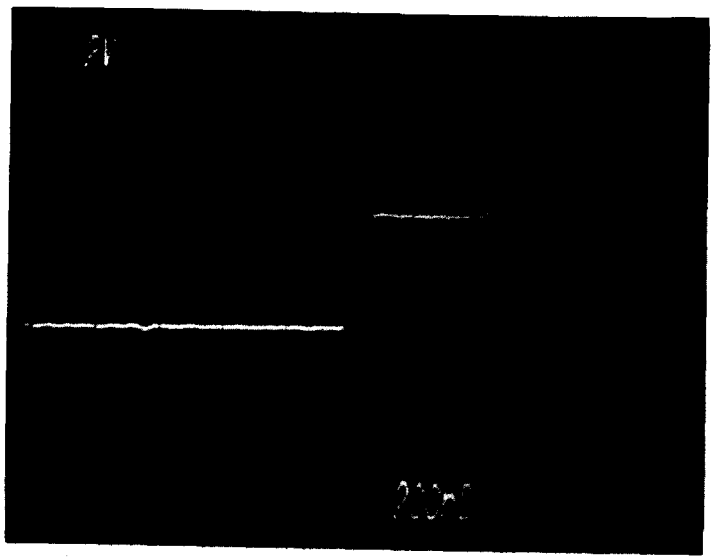
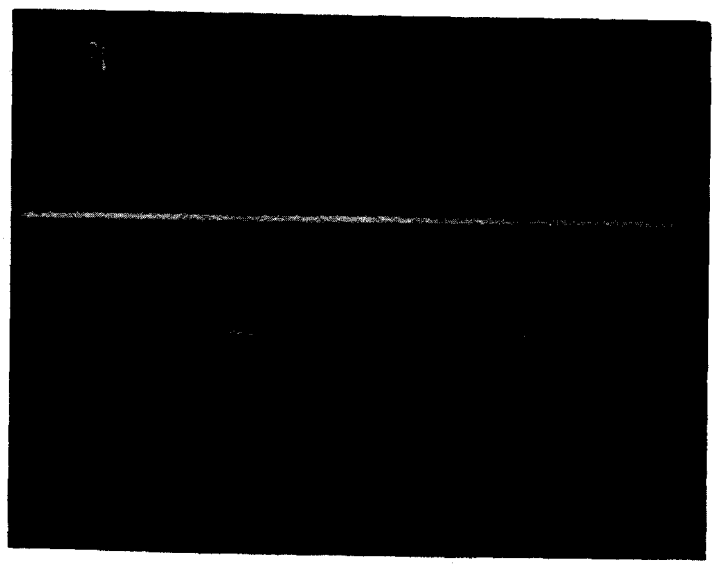


FIGURE 11 REF



3.3. TEST PROCEDURE #2

A power on reset (POR) forced the micro-program to PROM location 8003. In the previous procedure, a match stop occurred halting the program at this location. The instruction at location 8003 will be executed when the STEP switch is pushed. This small program causes the CPU to branch waiting for a key to be struck on the keyboard. Without striking a key on the keyboard, perform the following test procedure:

Change light board:

CM/SWITCHES = CM

COMPARE/NONCOMPARE = NONCOMPARE

STEP switch -- push once

Change initial scope setup as follows:

VERT MODE	ALT
TIME/DIV	.1 MICRO SEC. (100nS)

A. Compare T1 to T2, T3 to T4 etc. until finally comparing T11 to T12.

Probe: CH 1 L 63 pin 3

Probe: CH 2 L 64 pin 3

Refer to Figure #12

NOTE:
There is a 100 nS difference between occurrence of the low going pulses.

Probe: CH 1 L 63 pin 6

Probe: CH 2 L 64 pin 6

Refer to Figure #12

Continue comparing timing pulses in this fashion until finally comparing T11 to T12. (Refer to schematic for chip and pin locations of remaining timing signals).

B. Compare T1 to T12 (full cycle)

Set scope: TIME/DIV .2 MICRO SEC.

Probe: CH 1 L 63 pin 3

Probe: CH 2 L 66 pin 3

Refer to Figure #13

CH 1 T1-T11

CH 2 T2-T12

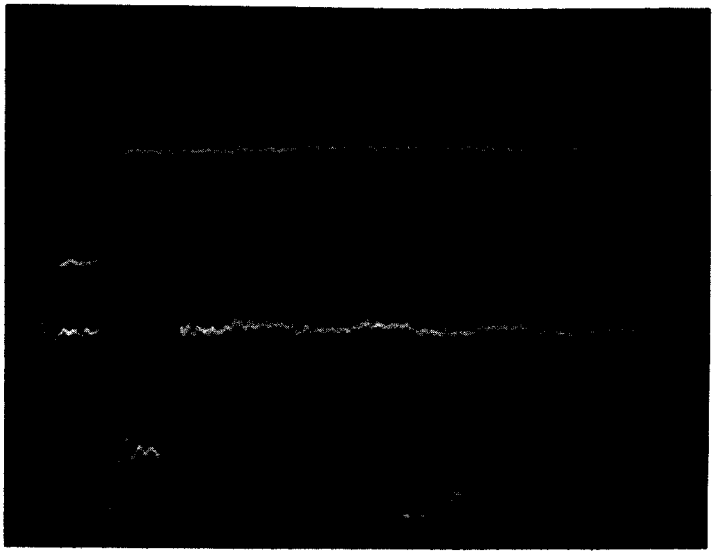


FIGURE 12 T1 COMPARED TO T2

CH 1 T1

CH 2 T12

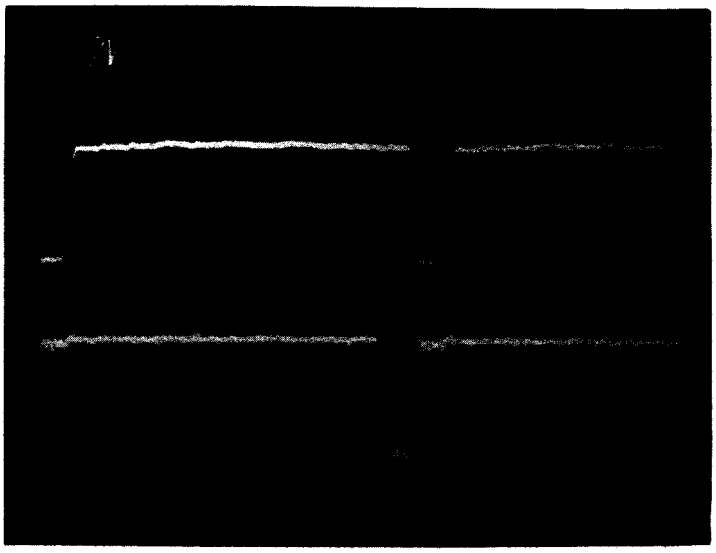


FIGURE 13 T1 COMPARED TO T12

This procedure checks the occurance of T12 in relation to T1 for a full cycle (start of T1 to start of next T1).

3.4. TEST PROCEDURE #3

In order to generate timing pulses T13-T16, a Subroutine Branch instruction is set up on the light board and forced to execute continuously. The ALUCLK signal is also checked in this procedure.

Set up light board as follows:

Set ROM switches = 87800F

Set CM/SWITCHES = CM

Set ADDRESS switches = 8006

Set IC/PC = IC

Set COMPARE/NONCOMPARE = COMPARE

Set RUN/STEP = RUN

Turn system power off then back on again to force Power On Reset. The program should halt at IC address 8006. At this point continue as follows:

Set CM/SWITCHES = SWITCHES

Set COMPARE/NONCOMPARE = NONCOMPARE

Step -- push once

The system should now be executing a continuous Subroutine Branch instruction.

A. Test timing pulses T13-T16

Set scope: TIME/DIV .05 MICRO SEC.

Probe: CH 1 L 65 pin 8

Probe: CH 2 L 66 pin 8

Refer to Figure #14

Note the 50 nS time difference between T13 and T14.

Set scope TIME/DIV .2 MICRO SEC.

Note full cycle = 800 nS.

Set scope: TIME/DIV .05 MICRO SEC.

Probe: CH 1 L 65 pin 11

Probe: CH 2 L 66 pin 11

Refer to Figure #14

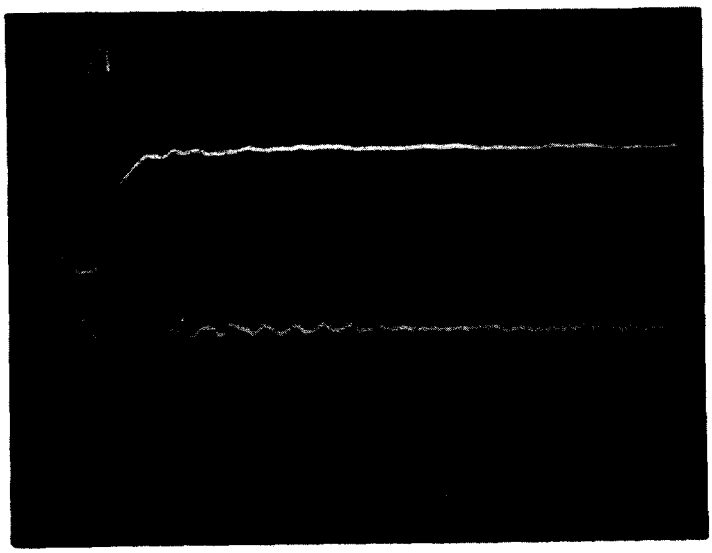
Set scope: TIME/DIV .2 MICRO SEC.

Refer to Figure #15

T13 COMPARED TO T14
T15 COMPARED TO T16

CH 1

CH 2



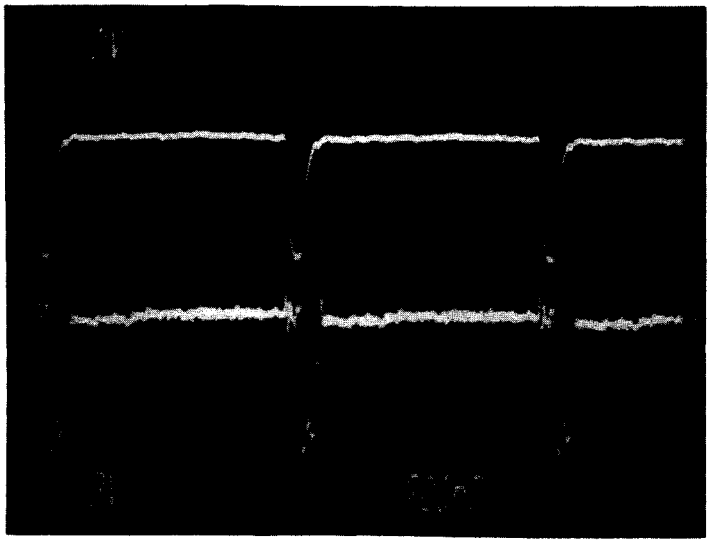
T13 T15

T14 T16

FIGURE 14

FULL CYCLE

CH 1



T13 T15

CH 2

T14 T16

FIGURE 15

B. Test ALUCLK signal

Probe: CH 1 L 26 Pin 6 (CH 1 only)

Refer to Figure #16

Start program execution:

Set COMPARE/NONCOMPARE = COMPARE

Set CM/SWITCHES = CM

Set RUN/STEP = RUN

Step -- push once

Program should now be running.

NOTE:
If a PECM results, push RESET to begin program execution.

ALUCLK

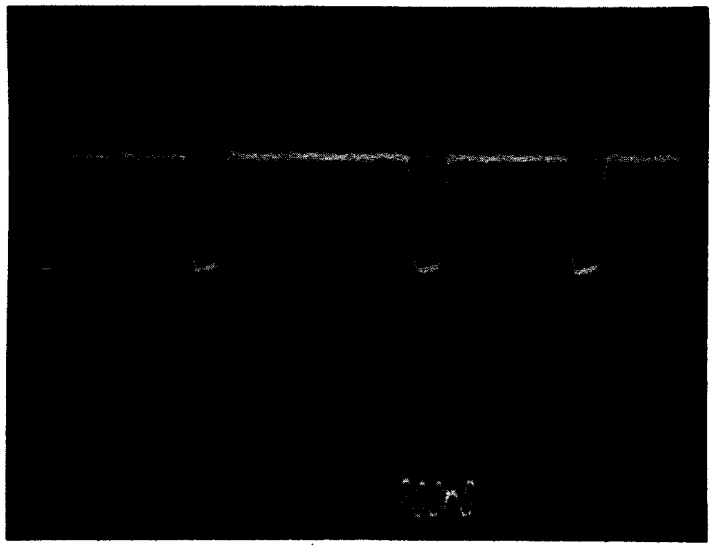


FIGURE 16

SECTION 4 TEST INFORMATION

4.1. POWER ON

As power is applied to the system a micro-trap to PROM address 8003 occurs which begins test operations. The program starts by disabling Data Memory Parity errors and enabling input from the keyboard. The program then waits at address 8006 for a key to be struck before continuing.

4.2. START

This begins a series of preliminary tests which exercise micro-instructions in a very basic fashion. This particular section is broken up into two parts, START and START 1. The START section tests Masked Branch instructions. It is important to note that if the MVI (Move Immediate) instruction does not function properly, erroneous results may occur from these preliminary branch tests.

4.3. START 1

This section of the program tests the Register Branch instructions. At the beginning of the program File Registers 0, 1, and 2 were initialized to a specific value for test purposes. The typical test procedure utilizes these registers in different combinations and with various Branch instructions to test the ability of the module to execute successful branch operations. If a failure is detected the program should branch back to the beginning of the section and restart (loop on error). If no errors are detected the CRT is cleared and a "0" is displayed on the screen.

4.4. INITIAL 1 -- INITIAL 9

These preliminary tests are very similar. Each test begins by setting up File Registers with some particular data pattern then executing a specific instruction several times to produce various results. After each instruction is executed a test is performed to check if the results are correct. If an error occurs the program loops on error, if no error occurs a number will be displayed on the CRT and the program continues on to the next test. At the completion of these initial tests the CRT should be display:

0123456789

4.5. TEST 0 -- TEST 8

These are a series of tests that exercise individual instructions. Previously the preliminary tests performed a very basic check of the ability to execute various instructions. These tests are designed to provide a more in-depth check of the instructions. At the beginning of each test the program displays "TEST #X" where X is the number of the test currently be executed. If an error is detected within any of these tests, "ERROR" is displayed on the CRT and the program restarts at the beginning of the test that fails. The program will loop in this fashion until the error goes away.

4.6. TEST 9

This test exercises various Carry Bit functions. Several different types of instructions are utilized to produce various Carry Bit conditions.

4.7. TEST A

This test exercises the PH and PL Registers ability to multiplex onto the B Bus. Various instructions are performed to force different multiplex conditions.

4.8. TEST B

This test exercises the Data Memory Input Register and Parity circuitry. Data memory locations 0000-00FF are written with their address, read back, and checked to see if they did in fact get written properly. Any single bit failure will cause a parity error (PEDM). At the beginning of this test the parity error circuitry was conditioned to allow parity errors. If a parity error occurs the program will display the failing address on the CRT. If more than one bit fails, and it does not cause a parity error, the program will display "ERROR" and restart at the beginning of the test.

4.9. TEST C

This test exercises the two ALU chips by executing groups of instructions that produce various bit pattern combinations.

4.10. TEST D

This test writes 16K of Control Memory, reads all previously written locations, and verifies that these locations did in fact get written properly. The address of the location is written into the location.

```
Location 0000 = 0000
Location 03AF = 03AF
```

At the completion of TEST D the CRT should display:

```
TEST#0
TEST#1
TEST#2
TEST#3
TEST#4
TEST#5
TEST#6
TEST#7
TEST#8
TEST#9
TEST#A
TEST#B
TEST#C
TEST#D
```

The program will restart at TEST 0 after TEST D has been completed.

4.11. RESET

If for any reason the program gets hung up or a parity error occurs the RESET key may be struck which will restart the program at the beginning of the preliminary tests (START).

SECTION 5 ALU 210-6792 THEORY OF OPERATION

This section provides detailed information concerning the operation of individual circuits within the 210-6792 module.

5.1. Simplified Block Diagram

5.1.1. FUNCTION DECODER

The Function Decoder (L9) specifies the function the ALU is to perform by providing the ALU with a 'word' decoded from instruction bits R14-16, 17-22.

5.1.2. 'C' BUS SOURCE SELECTOR

The 'C' Bus Source Selector (L47, L48, L60, 161) selects the output of one of the ALU sections for output onto the 'C' bus.

5.1.3. BRANCH DECODER

The Branch Decoder (L27, 128, L29, L41, L42, L43, L44, L54, L55, L56, L57, L67, L68, L69, L70) decodes all mask branch and conditional branch functions for the system.

5.1.4. 'B' BUS SOURCE SELECTOR

The 'B' Bus Source Selector (L4, L5, 16) selects either the PL or PH register for output onto the 'B'Bus.

5.1.5. SYSTEM TIMING

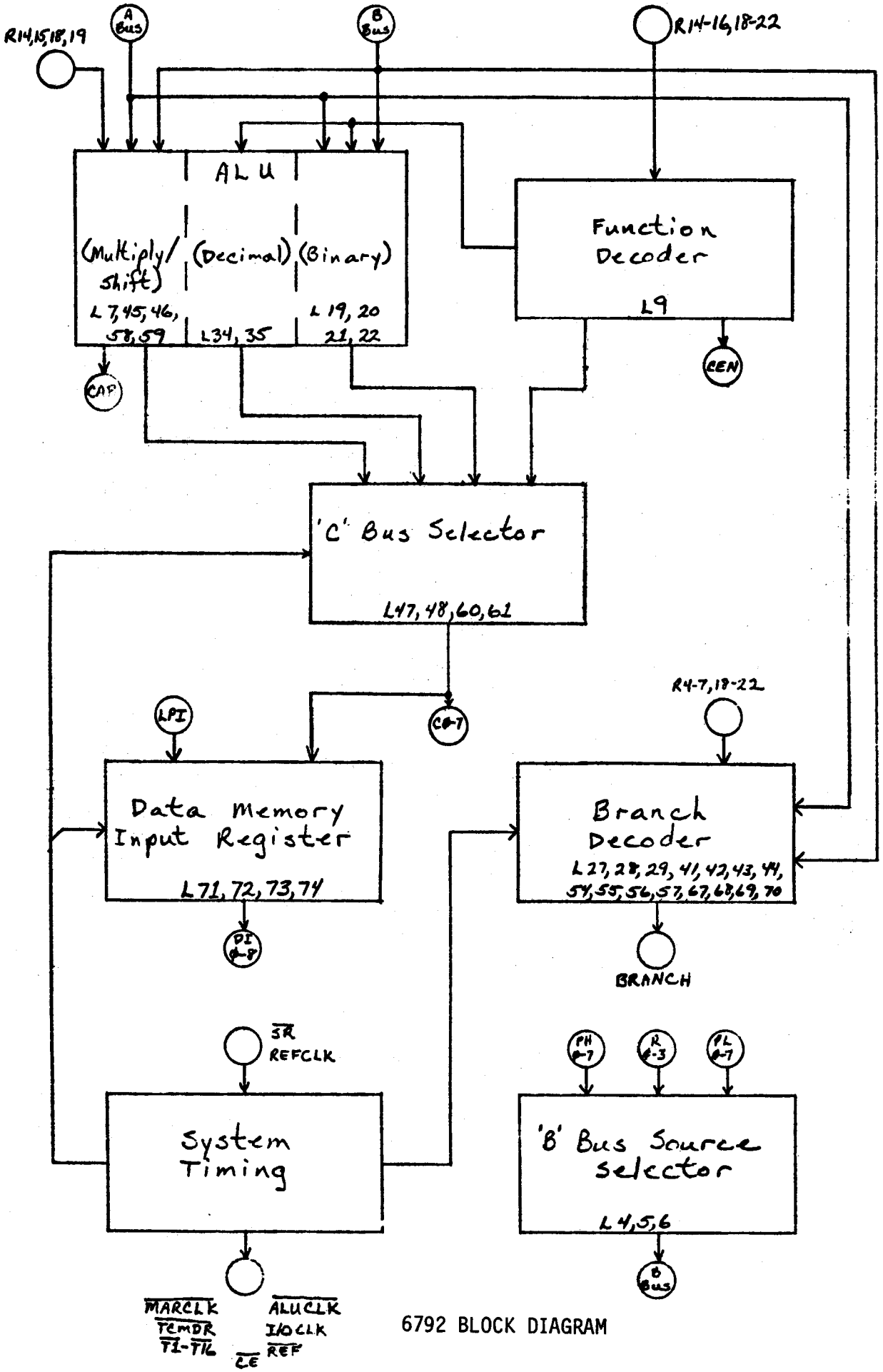
The System Timing utilizes a number of gates, latches, clocks, shift registers and counters to provide all of the basic system clock pulses. All clock signals are derived from a 20 MHz crystal oscillator.

5.1.6. DATA MEMORY INPUT REGISTER

The Data Memory Input Register (L71, L72, L73, L74) latches the output of the 'C'Bus Source Selector. A hardware generated parity bit is determined from the 8 bits of data from the 'C' Bus selector and latched into this register.

5.1.7. ALU

The system ALU is divided into three sections. Binary (L19, 120, L21, L22), Decimal (L34, L35) and Multiply (L7, L45, L46, L58, L59). The binary section performs the arithmetic and logical functions except as follows. The decimal section is used for DAC and DACI when the resultant sum of the low or high order 4 bits from the binary section is greater than 10, and for DSC and DSCI when the resultant difference from the binary section is less than zero (negative). The decimal section converts the output of the binary ALU to BCD. The multiply section is used for M, MI and SHFT instructions and is utilized to establish some system pointers and for control of subscripted variables. The ALU is not utilized for branch instructions or the following mini instructions: LPI; CIO; SR, WCM; SR, RCM.



6792 BLOCK DIAGRAM

5.2. Detailed Theory of Operation

5.2.1. FUNCTION DECODER

The Function Decoder utilizes a 74S188 PROM to decode instruction bits R14-16, 18-20 into ALU functions or words (see figure 4-20). Words 0-15 are self explanitory. Words 16-19 (TRANSFER) enable the binary ALU to pass the 'B' bus data on to the 'C' bus. Words 24-27 (SHIFT) enable the multiply ALU to pass the high or low 4 bits of the A and B register to the 'C' Bus Source Selector.

The C Bus Source Selector utilizes four-74LS257 quad 2 to 1 multiplexers for selecting the binary, decimal or multiply ALU sections for output to the Data Memory Input Register or 'C' bus latch. L47 and L48 are enabled for binary and decimal functions except the multiply and shift instructions. For these, L60 and L61 are enabled. Source selector inputs are selected as follows: 'A' inputs of L47 and L48 are selected for all binary and decimal operations as explained above except for DAC, DACI, DSC and DSCI when the sum is greater than 10 or the difference is less than zero (negative) at which time the B input are selected. The 'A' inputs of L60 and L61 are selected for mutiply operations by DOP being low (and ALU-OP being high). The B inputs (SHO-7) are selected for shift operations (DOP and AL11-OP high).

5.2.2. BRANCH DECODER

The Branch Decoder has two sections. One section for decoding the mask branch instructions and the other for decoding the register branch instructions. Instruction bits R19-22 are decoded by a BCD to Decimal decoder (L67) which decodes the desired branch condition to be tested.

The mask branch section test the condition of the high or low 4 bits of the selected register against the mask field. Instruction bit R18 is used to determine if the high or low 4 bits of the register are to be used. R18 low selects the low 4 bits at pin 1 of L29. If BEQ is decoded, the 4 mask bits are compared with the 4 register bits by L27. If they are equal, pin 6 goes high and with L54 pin 13 already high, BEQ goes low initiating a branch to the specified address. If BNE is decoded, L27 is still used for the compare, and if not equal pin 6 will be low but inverted at L30 pin 2 and with L54 pin 1 high a branch to the specified address occurs. For the BT instruction, the register bits are logically AND'ed with the masked bits by L28. If the result from 128 is equal to the mask field when compared by L42, a branch to the specified address occurs. The BF instruction is similar to BT except that the AND'ed results of the 4 register bits and the mask bits are OR'ed and if none of the register bits coincide with the mask (AND'ed result equals 0000) then BF is enabled and a branch occurs. Any one of BF, BT, BEQ, BNE going active causes L68 pin 8 (BRANCH) to go high.

WORD SELECT TRUTH TABLE

6792 P.C. BOARD

User Def.	Ins. Performed	WORD #	WORD ADR.	Inputs					Outputs								Cn
				A	B	C	D	E	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	
OR	OR	0	00	L	L	L	L	L	L	H	H	H	L	L	L	H	H
XOR	XOR	1	01	H	L	L	L	L	L	H	H	L	L	L	L	H	H
AND	AND	2	02	L	H	L	L	L	H	H	L	H	L	L	L	H	H
SC	SC	3	03	H	H	L	L	L	L	H	H	L	L	L	H	L	$\overline{SH\emptyset}$
DAC	DAC	4	04	L	L	H	L	L	H	L	L	H	L	H	H	L	$\overline{SH\emptyset}$
DSC	DSC	5	05	H	L	H	L	L	L	H	H	L	L	H	H	L	$SH\emptyset$
AC	AC	6	06	L	H	H	L	L	H	L	L	H	L	L	H	L	$\overline{SH\emptyset}$
M	M	7	07	H	H	H	L	L	L	L	L	L	H	L	L	L	H
ORI	ORI	8	08	L	L	L	H	L	L	H	H	H	L	L	L	H	H
XORI	XORI	9	09	H	L	L	H	L	L	H	H	L	L	L	L	H	H
ANDI	ANDI	10	0A	L	H	L	H	L	H	H	L	H	L	L	L	H	H
AI	AI	11	0B	H	H	L	H	L	H	L	L	H	L	L	L	L	H
DACI	DACI	12	0C	L	L	H	H	L	H	L	L	H	L	H	H	L	$\overline{SH\emptyset}$
DSCI	DSCI	13	0D	H	L	H	H	L	L	H	H	L	L	H	H	L	$SH\emptyset$
ACI	ACI	14	0E	L	H	H	H	L	H	L	L	H	L	L	H	L	$\overline{SH\emptyset}$
MI	MI	15	0F	H	H	H	H	L	L	L	L	L	H	L	L	L	H
*TRANSFER	TPA, XPA	16	10	L	L	L	L	H	L	H	L	H	L	L	L	H	H
TRANSFER	TPS, SR	17	11	H	L	L	L	H	L	H	L	H	L	L	L	H	H
TRANSFER	TAP	18	12	L	H	L	L	H	L	H	L	H	L	L	L	H	H
TRANSFER	TSP	19	13	H	H	L	L	H	L	H	L	H	L	L	L	H	H
TRANSFER		20	14	L	L	H	L	H	L	H	L	H	L	L	L	H	H
TRANSFER	ALU not used	21	15	H	L	H	L	H	L	H	L	H	L	L	L	H	H
TRANSFER		22	16	L	H	H	L	H	L	H	L	H	L	L	L	H	H
TRANSFER		23	17	H	H	H	L	H	L	H	L	H	L	L	L	H	H
**SHIFT	SHFT (HbHa=00)	24	18	L	L	L	H	H	L	L	L	L	H	H	L	L	H
SHIFT	SHFT (HbHa=01)	25	19	H	L	L	H	H	L	L	L	L	H	H	L	L	H
SHIFT	SHFT (HbHa=10)	26	1A	L	H	L	H	H	L	L	L	L	H	H	L	L	H
SHIFT	SHFT (HbHa=11)	27	1B	H	H	L	H	H	L	L	L	L	H	H	L	L	H
SHIFT		28	1C	L	L	H	H	H	L	L	L	L	H	H	L	L	H
SHIFT	never addressed	29	1D	H	L	H	H	H	L	L	L	L	H	H	L	L	H
SHIFT		30	1E	L	H	H	H	H	L	L	L	L	H	H	L	L	H
SHIFT		31	1F	H	H	H	H	H	L	L	L	L	H	H	L	L	H
				R_{18}	R_{19}	R_{20}	R_{21} + SHFT DCD	$\overline{R_{21}} \cdot R_{16}$ + SHFT DCD	$S\emptyset$	$S1$	$S2$	$S3$	$\overline{ALUOP}$	DOP	CEN	MODE	

$SHFT\ DCD = R_{14} \cdot \overline{R_{15}} \cdot \overline{R_{16}} \cdot \overline{R_{20}} \cdot \overline{R_{21}} \cdot \overline{R_{22}}$

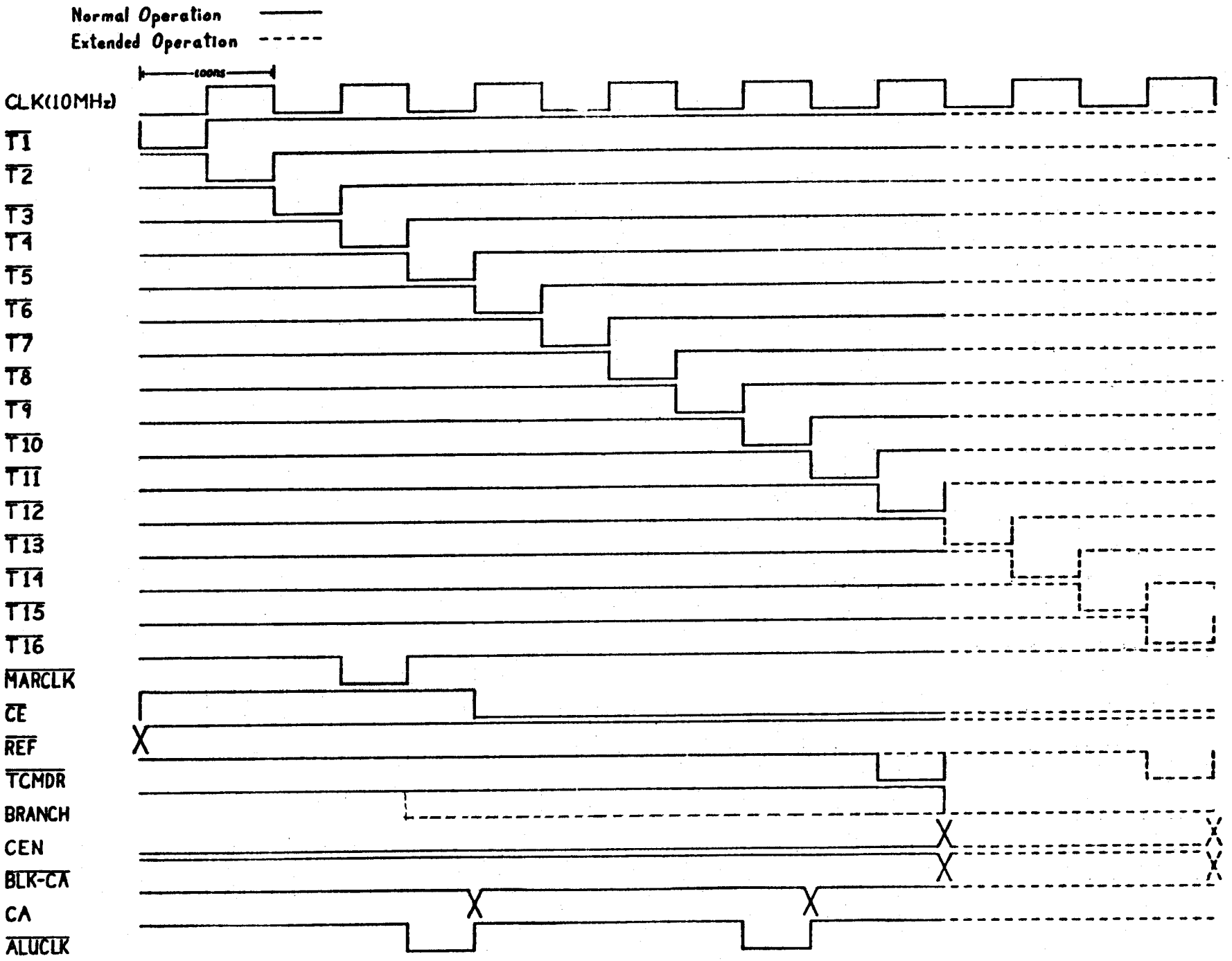
* TRANSFER = 'B' bus to 'C' bus

**SHIFT = High or low 4 bits of A and B registers to C bus

The register branch section tests the conditions between two 8 bit registers, or two 16 bit registers (8 bit register pairs) on an extended operation. Extended operations can be performed only for BLR and BLER instructions. For all of the register branch instructions, a comparison of the registers is performed by L43 and L44. The status of the register comparison is at L44 pins 6 and 7. If the 'A' and 'B' bus registers are equal, pin 6 will be high. If the 'A' bus register is less than the 'B' bus register, pin 7 will be high. If the 'A' bus register is greater than the 'B' bus register, pin 6 and 7 are low. L55 pin 6 will go low when BNR is decoded and register A ≠ register B. L55 pin 8 will go low when BER is decoded and register A = register B. 140 pin 8 will go low when BLR is decoded and register A < register B. L56 pin 8 will go low when BLER is decoded and register A < = register B. For the BLRX (register pair) instruction, BLR will be high at L40 pin 9. If register A < register B, L40 pin 8 will go low causing BRANCH to go high. At T4, L70 pins 3 and 11 are clocked causing L57 pin 11 to go high. At T6 when the register addresses are incremented, the registers are immediately tested by L43 and L44. If register A is again < register B, L40 pin 8 remains low. If register A = register B, L57 pin 10 goes high causing pin 8 to go low which also generates BRANCH. If register A > register B, L57 pin 10 remains low and L40 pin 10 goes low to turn BRANCH off. At T8 BRANCH is used in the instruction counter to load the branch address. For the BLERX (register pair) instruction, BLER will be high at L56 pins 5, 9 and 13. If register A < register B, L56 pins 1 and 10 will go high. As L56 pin 1 goes high, pin 12 goes low to enable BRANCH. At T4, L70 pin 11 is clocked which makes L57 pin 3 go high to make pin 6 go low, also enabling BRANCH. At T6 the next two registers are compared. If register A < register B, L56 pin 12 remains high to keep BRANCH enabled. If register A > register B on the second test, 156 pin 1, pin 4 and pin 10 go low to turn BRANCH off before it can be used by the instruction counter.

5.2.3. SYSTEM TIMING

The system timing originates from a 20 MHz crystal controlled oscillator. There are two speeds for system timing which are developed by L14B. The system operates at the fast speed (20 MHz) whenever ROMS is inactive. This is because pins 1 and 4 of L14B are driven at opposite polarities by the 20 MHz crystal output. This causes L39 pin 11 to be clocked at 20 MHz. The system operates at the slow speed (10 MHz) whenever ROMS is active. This causes pins 1 and 4 of L14B to go high and pin 3 to be clocked at 20 MHz. The output at pin 5 now clocks L39 pin 11 at 10 MHz instead of 20 MHz. The outputs of L39 at pins 8 and 9 are at half the frequency of the input clock. The clock signals at L38 pins 2, 4, 6 and 8 are used for clocking the various counters, latches and gates which develop the system clock and timing signals (refer to the 6792 timing diagram). Note that the time and frequency given for the signals is when the system is operating at the fast speed. Signal relationships remain the same but the time is doubled when the system operates at the slow speed.



6792 P.C. Board TIMING

5.2.4. DATA MEMORY INPUT REGISTER

The Data Memory Input Register has two functions. L72 and L73 latch the data from the 'C' bus selector at T5 for input to Data Memory. At the same time L74 monitors the output of the 'C' bus selector to generate a parity bit. Odd parity is generated from L71 pin 6 at T5. That is, DIP will be clocked low if there is an odd number of high's at the inputs of L74 and clocked high if there is an even number of high's at the inputs of L74. During an LPI instruction, L71 pin 9 is clocked low at T3 which clears the output of the registers to zero, and sets DIP on. If a write is specified in an LPI instruction, the data written will always be zero and will be written at the intial address contained in the PC's.

5.2.5 ALU

The ALU performs arithmetic or logic functions as specified by the decoded 'word'from the function decoder. The 8 bit data inputs from the A and B bus are operated on by the ALU and the resultant 8 bits are output to the 'C' bus selector. L19 and L20 are used for all binary functions and most decimal functions except M, MI and SHIFT instructions (see Fig 4-20). L34 and L35 are used for decimal functions when the resultant sum is greater than 10 or when the resultant difference is less than zero (negative). Its function is to convert the binary result from L19 and L20 to BCD. While the binary and/or decimal ALU sections are being used, the multiply ALU section output is inhibited. When the multiply section is being utilized the binary and decimal outputs are inhibited. The SHFT instruction combines the high or low 4 bits of the 'A' and 'B' Bus for output onto the 'C' bus. The multiply ALU performs binary with multiplication on the high or low 4 bits of the 'A' and 'B' Bus for output onto the 'C' Bus.

During MASK BRANCH, REGISTER BRANCH and BRANCH instructions as well as SR, RCM, SR, WCM, CIO and LPI MINI instructions the ALU outputs are ignored.

Function	S_3-S_0	m	A (A_3-A_0)	B (B_3-B_0)	$\overline{Cn}$ ($\overline{Cin}$/Bin)	F (F_3-F_0)	$\overline{Cn+4}$ ($\overline{C\ out}$/B out)
add (binary only)	HLLH	L	Augend	Addend	Low = Carry in	A Plus B Plus 1	$F>F_{16}$ $\overline{Cn+4}$ = Low (carry out)
					Hi = No Carry in	A Plus B	$F\leq F_{16}$ $\overline{Cn+4}$ = High (no carry out)
Subtract	LHHL	L	Minuend	Subtrahend	Low = No borrow in	A Minus B	$A\geq B$ $\overline{Cn+4}$ = Low (no borrow out) $A<B$ $\overline{Cn+4}$ = High (borrow out)
					Hi = borrow in	A Minus B Minus 1	$A>B$ $\overline{Cn+4}$ = Low (no borrow out) $A\leq B$ $\overline{Cn+4}$ = High (borrow out)
or	HHHL	H	A	B	X	A+B (A or B)	X
xor	LHHL	H	A	B	X	A + B	X
and	HLHH	H	A	B	X	AB (A and B)	X

SECTION 6 REPAIR AIDS

6.1. TROUBLESHOOTING EXAMPLE

To familiarize the repair technician with the operation of the 2200VP tester, a fault will be introduced onto a board, then a step by step procedure for finding the fault described.

Perform the following steps to set up system initially:

1. Install the program PROMS on the 210-6789 module.

2. Remove the 6792 module from the test system.

3. Insert a module extender in the 6792 slot and plug the test module into the extender board.

Perform the following steps to set up the light board:

1. Set CM/SWITCHES = CM

2. Set RUN/STEP = RUN

3. Set IC/PC = IC

4. Set COMPARE/NONCOMPARE = NONCOMPARE

To begin this example it will be necessary to cut pin #1 of I.C. L-25 (74S08) on a previously known good module.

Turn power on to the system. The screen on the CRT should contain "garbage". The IC address should be 8006. The program will be in a loop waiting for a key to be struck on the keyboard.

Strike a key on the keyboard. Normally the first visual response from the program is to clear the CRT and begin printing out test information. In this example the module fails before any CRT activity occurs.

By looking at the IC address lights it can be noted that the program is in a tight loop because the LED's seem to constantly contain the same pattern. By referring to the TEST INFORMATION section of this MODULE REPAIR GUIDE, it becomes apparent that the failure must be occuring within the START or START 1 sections of the program because no "0" is displayed on the CRT.

Set the RUN/STEP switch = STEP and observe the IC address. Push the STEP switch and note the change in the IC address LED's. Continue to push the switch and note the address changes. The program is running in a loop starting at PROM address 8010 and continuing to 8018.

Refer to the listing of the 6792 microprogram found in the back of the "MODULE REPAIR GUIDE". Page 1 contains the program code applicable to this example. The left hand column contains the IC address. Locate IC address 8018. This is the beginning of the Masked Branch Instructions section. The name of the this particular test is START.

Step the program through the loop again by pushing the STEP switch. This allows the execution of one instruction each time the switch is pushed. By writing down the IC addresses the following sequence should result:

8010
8011
8012
8013
8015
8017
8018

Referring to the program listing, read the instruction mnemonics and comments for each of the IC addresses to become familiar with the test.

File Registers 0 and 1 are cleared (= to 00). File Register 2 is set = to FF (all 1's). A series of branch operations are performed testing the various Branch instructions.

In this example, the last IC address before the program restarts is 8018. The instruction at this location caused a branch to START which restarts the test (loop on error). The comment field for this instruction states that an error occured. The actual error occured during execution of the instruction at address 8017. This instruction (BFL) Branch if false (low four bits of register) should have caused the program to branch to location 8019 instead of allowing the branch to START to occur.

The instruction compares it's immediate data field (00 in this case) to File Register 2 which should = all 1's (FF). If the low four register bits specified by corresponding one bits in the immediate data field (M field) are all 0, a branch to the address specified by the instruction occurs. In this example, the instruction should have forced a branch to location 8019. Since this did not occur it can be assumed that the BFL instruction failed.

Set RUN/STEP = RUN
Push STEP switch

The program should be running in a continuous loop again. Refer to the 210-6792 shematic diagram. At coordinates G-3 there is a nand gate L-54. The output of L-54 (pin 6) is labeled BF. With the program looping scope L-54 pin 6. In this example pin 6 stays high (+4v). If the module were functioning properly pin 6 would go low (0v) in this loop.

At the beginning of the program loop the File Registers are initialized to their test values. It is possible to force execution of the BFL instruction using the light board. Doing this makes troubleshooting much easier because only the failing instruction is being executed rather than several instructions when the program is looping on error.

The actual microcode for the BFL instruction that fails can be extracted from the program listing at IC address 8018 (681902). To force single instruction execution set up the light board as follows:

Set RUN/STEP = STEP

Set R switches = 681902

Set CM/SWITCHES = SWITCHES

Set RUN/STEP = RUN

Push STEP switch

The BFL instruction should be running continuously.

Probe L-54 pin 6

BF stays high (branch never will occur)

Probe L-54 input pins 4 and 5.

Pin 4 is low
Pin 5 is high
Pin 4 being low is the problem.

Trace pin 4 of L54 back to L16 (74S260) pin 5
This pin should be high, but is actually low.

Check input pins on 116

Pin 1 is low
Pin 2 is low
Pin 3 is high
Pin 13 is low
pin 3 being high is the problem

Trace pin 3 of L-16 to pin 3 of L-28

L-28 pin 3 should be low at this time but is actually high.
Check input pins 1 and 2.

Pin 1 is floating high (app. 2v) (this pin was cut)
Pin 2 is high
Obviously pin 1 is the problem.

This pin being cut simulates a bad input to L-28 pin 1. Logically this gate sees R5 being high all the time. The logic uses R bits 4, 5, 6, and 7 of the instruction as the Mask field for the comparison. So when the compare of File Register 2 (= F) and the mask field occurs, instead of seeing the mask field = 0, the logic sees a mask field of 2. A BFL instruction comparing F to 2 will not meet the conditions necessary for a branch to occur.

```
                 8000   1    1            ORG      8000
8000 5FAC80      83AC   2    2            B        PE24          CONTROL MEMORY PARITY ERROR
8001 DC1080      8010   3    3            B        START         RESET
8002 DFB580      83B5   4    4            B        PE8           DATA MEMORY PARITY ERROR
8003 220D0F             5    5            MVI      80,SH         ALLOW INPUTS, DISABLE DMPE TRAPS
8004 200E1F             6    6            MVI      1,K           SET UP KEYBOARD AS INPUT CHANNEL
8005 D71E80      831E   7    7            SB       ENABLE        ENABLE INPUT CHANNEL
8006 68062D      8006   8    8            BFL      2,SH,*        WAIT FOR KEYSTROKE
8007 DC1080      8010   9    9            B        START         BEGIN TESTING
8008 DC0880      8008   10   10           B        *
8009 5C0980      8009   11   11           B        *
800A 5C0A80      800A   12   12           B        *
800B DC0B80      800B   13   13           B        *
800C 5C0C80      800C   14   14           B        *
800D DC0D80      800D   15   15           B        *
800E DC1080      8010   16   16           B        START
800F 800000             17   17           INSTR    800000
                        18   18   *
                        19   19   *MASKED BRANCH INSTRUCTIONS
                        20   20   *
8010 20000F             21   21   START    MVI      00,F0         CLEAR FILE REG. 0
8011 A00100             22   22            MV       F0,F1         PUT 0'S IN F1
8012 A3C2FF             23   23            MVI      0FF,F2        PUT ALL 1'S IN FILE REG. 2
8013 E015F2      8015   24   24            BTL      0F,F2,*+2     SHOULD BRANCH HERE OR ERROR
8014 DC1080      8010   25   25            B        START         ERROR OCCURRED
8015 E417F2      8017   26   26            BTH      0F,F2,*+2     SHOULD BRANCH
8016 DC1080      8010   27   27            B        START         ERROR OCCURRED
8017 681902      8019   28   28            BFL      00,F2,*+2     SHOULD BRANCH
8018 DC1080      8010   29   29            B        START         ERROR OCCURRED
8019 6C1B02      801B   30   30            BFH      00,F2,*+2     SHOULD BRANCH
801A DC1080      8010   31   31            B        START         ERROR OCCURRED
801B 701D00      801D   32   32            BEQL     00,F0,*+2     SHOULD BRANCH
801C DC1080      8010   33   33            B        START         ERROR OCCURRED
801D 741F00      801F   34   34            BEQH     00,F0,*+2     SHOULD BRANCH
801E DC1080      8010   35   35            B        START         ERROR OCCURRED
801F F821F0      8021   36   36            BNEL     0F,F0,*+2     SHOULD BRANCH
8020 DC1080      8010   37   37            B        START         ERROR OCCURRED
8021 FC23F0      8023   38   38            BNEH     0F,F0,*+2     SHOULD BRANCH
8022 DC1080      8010   39   39            B        START         ERROR OCCURRED
                        40   40   *
                        41   41   *REGISTER BRANCH INSTRUCTIONS
                        42   42   *
8023 402502      8025   43   43   START1   BLR      F0,F2,*+2     SHOULD BRANCH  0 LESS THAN FF
8024 DC2380      8023   44   44            B        START1        ERROR OCCURRED
8025 482701      8027   45   45            BLER     F0,F1,*+2     SHOULD BRANCH 0 = TO 0
```

```
                8026 DC2380  8023   46   46            B         START1          ERROR OCCURRED
                8027 D02901  8029   47   47            BER       F0,F1,*+2       SHOULD BRANCH
                8028 DC2380  8023   48   48            B         START1          ERROR OCCURRED
                8029 D82B02  802B   49   49            BNR       F0,F2,*+2       SHOULD BRANCH
                802A DC2380  8023   50   50            B         START1          ERROR OCCURRED
                802B 402320  8023   51   51            BLR       F2,F0,START1    SHOULD NOT BRANCH
                802C 5C2D80  802D   52   52            B         *+1
                802D C82320  8023   53   53            BLER      F2,F0,START1    SHOULD NOT BRANCH
                802E DC2F80  802F   54   54            B         *+1
                802F D02320  8023   55   55            BER       F2,F0,START1    SHOULD NOT BRANCH
                8030 DC3180  8031   56   56            B         *+1
                8031 582301  8023   57   57            BNR       F0,F1,START1    SHOULD NOT BRANCH
                8032 5C3380  8033   58   58            B         *+1
                8033 483502  8035   59   59            BLER      F0,F2,*+2       SHOULD BRANCH
                8034 DC2380  8023   60   60            B         START1
                8035 A00302         61   61            MV        F2,F3           F3 = FF
              3 8036 C43802  8038   62   62            BLRX      F0,F2,*+2       SHOULD BRANCH
                8037 DC2380  8023   63   63            B         START1
              3 8038 4C3A00  803A   64   64            BLERX     F0,F0,*+2       SHOULD BRANCH
                8039 DC2380  8023   65   65            B         START1
              3 803A 442322  8023   66   66            BLRX      F2,F2,START1    SHOULD NOT BRANCH
                803B 5C3C80  803C   67   67            B         *+1
              3 803C 4C2320  8023   68   68            BLERX     F2,F0,START1    SHOULD NOT BRANCH
                803D DC3E80  803E   69   69            B         *+1
                803E 572980  8329   70   70            SB        INIT            CLEAR SCREEN DISABLE INPUTS
                803F A0CE0F         71   71            MVI       30,K            PUT A "0" IN K REG
                8040 572580  8325   72   72            SB        OBSTROBE        PRINT A "0" ON CRT
                                    73   73   *
                                    74   74   *REGISTER INSTRUCTIONS:   OR
                                    75   75   *
                8041 23C0FF         76   76   INITIAL1 MVI        0FF,F0          PUT ALL 1'S IN F0
                8042 A0010F         77   77            MVI       00,F1           PUT ALL 0'S IN F1
                8043 200201         78   78            MV        F1,F2           PUT 0'S IN F2
                8044 800201         79   79            OR        F0,F1,F2        OR 1'S WITH 0'S STORE RESULT IN F2 (1'S)
                8045 200E02         80   80            MV        F2,K            PUT F2 IN K FOR DISPLAY
                8046 7841F2  8041   81   81            BNEL      0F,F2,INITIAL1  F2 SHOULD = FF BRANCH IF NOT
                8047 FC41F2  8041   82   82            BNEH      0F,F2,INITIAL1
                8048 20000F         83   83            MV1       00,F0           PUT 0'S IN F0
                8049 A00100         84   84            MV        F0,F1           PUT 0'S IN F1
                804A 800201         85   85            OR        F0,F1,F2        OR 0'S WITH 0'S STORE RESULT IN F2 (0'S)
                804B 200E02         86   86            MV        F2,K            PUT F2 IN K FOR DISPLAY
                804C 784102  8041   87   87            BNEL      0,F2,INITIAL1   F2 SHOULD = 00 BRANCH IF NOT
                804D FC4102  8041   88   88            BNEH      0,F2,INITIAL1
                804E 20CE1F         89   89            MVI       31,K
                804F 572580  8325   90   90            SB        OBSTROBE        PRINT A "1" ON CRT
```

```
                    91   91  *
                    92   92  * XOR INSTRUCTION
                    93   93  *
8050 23C0FF         94   94  INITIAL2 MVI     0FF,F0          PUT ALL 1'S IN F0
8051 040000         95   95          XOR     F0,F0,F0        F0 SHOULD NOW = 00
8052 A00E00         96   96          MV      F0,K            PUT F0 IN K FOR DISPLAY
8053 F85000 8050    97   97          BNEL    0,F0,INITIAL2   BRANCH IF F0 NOT = 00
8054 7C5000 8050    98   98          BNEH    0,F0,INITIAL2
8055 040000         99   99          XOR     F0,F0,F0        F0 SHOULD STILL = 00
8056 A00E00        100  100          MV      F0,K            PUT F0 IN K FOR DISPLAY
8057 F85000 8050   101  101          BNEL    0,F0,INITIAL2   BRANCH IF F0 NOT = 00
8058 7C5000 8050   102  102          BNEH    0,F0,INITIAL2
8059 20CE2F        103  103          MVI     32,K            PUT A "2" IN K
805A 572580 8325   104  104          SB      OBSTROBE        PRINT A "2" ON CRT
                   105  105  *
                   106  106  *AND INSTRUCTION
                   107  107  *
805B 23C0FF        108  108  INITIAL3 MVI     0FF,F0          PUT ALL 1'S IN F0
805C A00100        109  109          MV      F0,F1           PUT ALL 1'S IN F1
805D A0020F        110  110          MVI     00,F2           PUT 0'S IN F2
805E 080201        111  111          AND     F0,F1,F2        F2 SHOULD NOW = FF
805F 200E02        112  112          MV      F2,K            PUT F2 IN K FOR DISPLAY
8060 F85BF2 805B   113  113          BNEL    0F,F2,INITIAL3  BRANCH IF F2 NOT = FF
8061 7C5BF2 805B   114  114          BNEH    0F,F2,INITIAL3
8062 20000F        115  115          MVI     00,F0           PUT 0'S IN F0
8063 080201        116  116          AND     F0,F1,F2        F2 SHOULD BE = 00
8064 200E02        117  117          MV      F2,K            PUT F2 IN K FOR DISPLAY
8065 F85B02 805B   118  118          BNEL    0,F2,INITIAL3   BRANCH IF F2 NOT = 00
8066 7C5B02 805B   119  119          BNEH    0,F2,INITIAL3
8067 A0CE3F        120  120          MVI     33,K            PUT A "3" IN K
8068 572580 8325   121  121          SB      OBSTROBE        PRINT A "3" ON CRT
                   122  122  *
                   123  123  *BINARY SUBTRACT WITH CARRY INSTRUCTION
                   124  124  *
8069 23C0FF        125  125  INITIAL4 MVI     0FF,F0          PUT ALL 1'S IN F0
806A A00100        126  126          MV      F0,F1           PUT ALL 1'S IN F1
806B A3C2FF        127  127          MVI     0FF,F2          PUT 1'S IN F2
806C 8CC201        128  128          SC,,1   F0,F1,F2        SUBTRACT 1'S FROM 1'S, F2 = 00
806D 200E02        129  129          MV      F2,K            PUT F2 IN K FOR DISPLAY
806E 786902 8069   130  130          BNEL    0,F2,INITIAL4   BRANCH IF F2 NOT = TO 00
806F FC6902 8069   131  131          BNEH    0,F2,INITIAL4
8070 A0010F        132  132          MVI     00,F1           PUT 0'S IN F1
8071 8CC102        133  133          SC,,1   F0,F2,F1        SUB. 0'S FROM 1'S, F1 = 1'S
8072 200E01        134  134          MV      F1,K            PUT F1 IN K FOR DISPLAY
8073 7869F1 8069   135  135          BNEL    0F,F1,INITIAL4  F1 SHOULD = FF BRANCH IF NOT =
```

```
8074 FC69F1  8069   136  136          BNEH     0F,F1,INITIAL4
8075 20CE4F         137  137          MVI      34,K              PUT A "4" IN K
8076 572580  8325   138  138          SB       OBSTROBE          PRINT A "4" ON CRT
                    139  139  *
                    140  140  *BINARY ADD WITH CARRY INSTRUCTION
                    141  141  *
8077 A0001F         142  142  INITIAL5 MVI      01,F0             PUT A 1 IN F0
8078 A3C1FF         143  143          MVI      0FF,F1            PUT ALL 1'S IN F1
8079 200201         144  144          MV       F1,F2             PUT 1'S IN F2
807A 188201         145  145          AC,,0    F0,F1,F2          ADD 1 TO FF, F2 = 00
807B 200E02         146  146          MV       F2,K              PUT F2 IN K FOR DISPLAY
807C 787702  8077   147  147          BNEL     0,F2,INITIAL5     BRANCH IF F2 NOT = 00
807D FC7702  8077   148  148          BNEH     0,F2,INITIAL5
807E 188021         149  149          AC,,0    F2,F1,F0          ADD 0'S TO 1'S, F0 = FF
807F A00E00         150  150          MV       F0,K              PUT F0 IN K FOR DISPLAY
8080 F877F0  8077   151  151          BNEL     0F,F0,INITIAL5    BRANCH IF F0 NOT = FF
8081 7C77F0  8077   152  152          BNEH     0F,F0,INITIAL5
8082 A0CE5F         153  153          MVI      35,K              PUT A "5" IN K
8083 572580  8325   154  154          SB       OBSTROBE          PRINT "5" ON CRT
                    155  155  *
                    156  156  *BINARY MULTIPLY INSTRUCTION
                    157  157  *
8084 21405F         158  158  INITIAL6 MVI      55,F0             PUT 55 IN F0
8085 A0C13F         159  159          MVI      33,F1             PUT 33 IN F1
8086 A0020F         160  160          MVI      00,F2             CLEAR F2
8087 1C0201         161  161          MLL      F0,F1,F2          MULT 5*3 = F, F2 SHOULD = F
8088 200E02         162  162          MV       F2,K              DISPLAY RESULT
8089 7884F2  8084   163  163          BNEL     0F,F2,INITIAL6    F2 SHOULD = F BRANCH IF NOT =
808A 1CC201         164  164          MHH      F0,F1,F2          MULT 5*3 = F, F2 SHOULD = FF
808B 200E02         165  165          MV       F2,K              DISPLAY RESULT
808C 7884F2  8084   166  166          BNEL     0F,F2,INITIAL6    F2 SHOULD = F BRANCH IF NOT =
808D A0CE6F         167  167          MVI      36,K              PUT A "6" IN K
808E 572580  8325   168  168          SB       OBSTROBE          PRINT A "6" ON CRT
                    169  169  *
                    170  170  *SHIFT INSTRUCTION
                    171  171  *
808F 2000FF         172  172  INITIAL7 MVI      0F,F0             PUT "0F" IN F0
8090 A3C10F         173  173          MVI      0F0,F1            PUT "F0" IN F1
8091 A0020F         174  174          MVI      00,F2             PUT 0'S IN F2
8092 004201         175  175          SHLL     F0,F1,F2          F2 SHOULD = "0F"
8093 200E02         176  176          MV       F2,K              DISPLAY F2
8094 588F02  808F   177  177          BNR      F0,F2,INITIAL7    BRANCH F2 NOT = "0F"
8095 844201         178  178          SHLH     F0,F1,F2          F2 SHOULD = "00"
8096 200E02         179  179          MV       F2,K              DISPLAY F2
8097 F88F02  808F   180  180          BNEL     0,F2,INITIAL7     BRANCH IF F2 NOT = "00"
```

```
8098 7C8F02  808F  181  181          BNEH     0,F2,INITIAL7
8099 884201        182  182          SHHL     F0,F1,F2          F2 SHOULD = "FF"
809A 200E02        183  183          MV       F2,K              DISPLAY F2
809B F88FF2  808F  184  184          BNEL     0F,F2,INITIAL7    BRANCH IF F2 NOT = "FF"
809C 7C8FF2  808F  185  185          BNEH     0F,F2,INITIAL7
809D 0C4201        186  186          SHHH     F0,F1,F2          F2 SHOULD = "F0"
809E 200E02        187  187          MV       F2,K              DISPLAY F2
809F D88F12  808F  188  188          BNR      F1,F2,INITIAL7    BRANCH IF F2 NOT = "F0"
80A0 20CE7F        189  189          MVI      37,K              PUT A "7" IN K
80A1 572580  8325  190  190          SB       OBSTROBE          PRINT A "7" ON CRT
                   191  191  *
                   192  192  *DECIMAL ADD WITH CARRY INSTRUCTION
                   193  193  *
80A2 A0001F        194  194  INITIAL8 MVI      01,F0             PUT A 1 IN F0
80A3 A0019F        195  195          MVI      09,F1             PUT A 9 IN F1
80A4 A3C2FF        196  196          MVI      0FF,F2            PUT 1'S IN F2
80A5 908201        197  197          DAC,,0   F0,F1,F2          ADD 1+9 = 0 WITH CARRY, F2 = 10
80A6 200E02        198  198          MV       F2,K              DISPLAY F2
80A7 F8A202  80A2  199  199          BNEL     0,F2,INITIAL8     BRANCH IF F2 NOT = TO 10
80A8 FCA212  80A2  200  200          BNEH     1,F2,INITIAL8
80A9 A0016F        201  201          MVI      06,F1             PUT A 6 IN F1
80AA 908201        202  202          DAC,,0   F0,F1,F2          ADD 1+6 = 7, F2 = 7
80AB 200E02        203  203          MV       F2,K              DISPLAY F2
80AC 78A272  80A2  204  204          BNEL     7,F2,INITIAL8     BRANCH IF F2 NOT = 07
80AD 7CA202  80A2  205  205          BNEH     0,F2,INITIAL8
80AE 20CE8F        206  206          MVI      38,K              PUT AN "8" IN K
80AF 572580  8325  207  207          SB       OBSTROBE          PRINT AN "8" ON CRT
                   208  208  *
                   209  209  *DECIMAL SUBTRACT WITH CARRY INSTRUCTION
                   210  210  *
80B0 20009F        211  211  INITIAL9 MVI      09,F0             PUT A 9 IN F0
80B1 A00100        212  212          MV       F0,F1             PUT A 9 IN F1
80B2 A3C2FF        213  213          MVI      0FF,F2            PUT ALL 1'S IN F2
80B3 148201        214  214          DSC,,0   F0,F1,F2          SUB. 9 - 9 = 0, F2 = 0
80B4 200E02        215  215          MV       F2,K              DISPLAY F2
80B5 F8B002  80B0  216  216          BNEL     0,F2,INITIAL9     BRANCH IF F2 NOT = 00
80B6 7CB002  80B0  217  217          BNEH     0,F2,INITIAL9
80B7 A0010F        218  218          MVI      00,F1             PUT 0 IN F1
80B8 148201        219  219          DSC,,0   F0,F1,F2          SUB. 9 - 0 = 9, F2 = 9
80B9 200E02        220  220          MV       F2,K              DISPLAY F2
80BA F8B092  80B0  221  221          BNEL     9,F2,INITIAL9     BRANCH IF F2 NOT = TO 09
80BB 7CB002  80B0  222  222          BNEH     0,F2,INITIAL9
80BC A0CE9F        223  223          MVI      39,K
80BD 572580  8325  224  224          SB       OBSTROBE          PRINT "9" ON CRT
80BE 572F80  832F  225  225          SB       CRLF              CARRIAGE RETURN, LINE FEED
```

```
                    226  226  *
                    227  227  *TEST ORI INSTRUCTION
                    228  228  * NOTE: BNEH AND BNEL ARE BEING USED EXTENSIVELY
                    229  229  *FOR TEST OF ORI RESULTS.
                    230  230  *
80BF 20C00F         231  231  TEST0   MVI     30,F0           PUT A "0" IN F0
80C0 D73F80  833F   232  232          SB      TEST#           PRINT "TEST #0" ON CRT
80C1 A0010F         233  233          MVI     00,F1           PUT 0'S IN FILE REG. 1
80C2 78E101  80E1   234  234          BNEL    0,F1,ERROR0     F1 SHOULD = 00 IF NOT,ERROR
80C3 FCE101  80E1   235  235          BNEH    0,F1,ERROR0
80C4 20011F         236  236          MVI     01,F1           PUT A 1 IN F1
80C5 FCE101  80E1   237  237          BNEH    0,F1,ERROR0     F1 SHOULD = 01 IF NOT, ERROR
80C6 F8E111  80E1   238  238          BNEL    1,F1,ERROR0
80C7 20012F         239  239          MVI     02,F1           F1 = 02
80C8 FCE101  80E1   240  240          BNEH    0,F1,ERROR0
80C9 F8E121  80E1   241  241          BNEL    2,F1,ERROR0
80CA 20014F         242  242          MVI     04,F1           F1 = 04
80CB FCE101  80E1   243  243          BNEH    0,F1,ERROR0
80CC F8E141  80E1   244  244          BNEL    4,F1,ERROR0
80CD 20018F         245  245          MVI     08,F1           F1 = 08
80CE FCE101  80E1   246  246          BNEH    0,F1,ERROR0
80CF F8E181  80E1   247  247          BNEL    8,F1,ERROR0
80D0 20410F         248  248          MVI     10,F1           F1 = 10
80D1 7CE111  80E1   249  249          BNEH    1,F1,ERROR0
80D2 78E101  80E1   250  250          BNEL    0,F1,ERROR0
80D3 20810F         251  251          MVI     20,F1           F1 = 20
80D4 7CE121  80E1   252  252          BNEH    2,F1,ERROR0
80D5 7CE121  80E1   253  253          BNEH    2,F1,ERROR0
80D6 78E101  80E1   254  254          BNEL    0,F1,ERROR0
80D7 21010F         255  255          MVI     40,F1           F1 = 40
80D8 7CE141  80E1   256  256          BNEH    4,F1,ERROR0
80D9 78E101  80E1   257  257          BNEL    0,F1,ERROR0
80DA 22010F         258  258          MVI     80,F1           F1 = 80
80DB 7CE181  80E1   259  259          BNEH    8,F1,ERROR0
80DC 78E101  80E1   260  260          BNEL    0,F1,ERROR0
80DD A0010F         261  261          MVI     00,F1           F1 = 00
80DE FCE101  80E1   262  262          BNEH    0,F1,ERROR0
80DF 78E101  80E1   263  263          BNEL    0,F1,ERROR0
80E0 5CE480  80E4   264  264          B       TEST1
                    265  265  *
                    266  266  *
80E1 574F80  834F   267  267  ERROR0  SB      ERROR           PRINT ERROR ON CRT
80E2 200E01         268  268          MV      F1,K            DISPLAY FAILING F1
80E3 DCBF80  80BF   269  269          B       TEST0           RESTART TEST
                    270  270  *
```

```
                    271   271  *TEST THE XORI INSTRUCTION  (USE BNEL,BNEH FOR TEST)
                    272   272  *
80E4 A0C01F         273   273  TEST1    MVI       31,F0              PUT A "1" IN F0
80E5 D73F80  833F   274   274           SB        TEST#              PRINT "TEST # 1 " ON CRT
80E6 A0010F         275   275           MVI       00,F1              CLEAR F1
80E7 240111         276   276           XORI      01,F1,F1           F1 = 01
80E8 7D0301  8103   277   277           BNEH      0,F1,ERROR1        F1 SHOULD = 01, ERROR IF NOT =
80E9 790311  8103   278   278           BNEL      1,F1,ERROR1
80EA 240121         279   279           XORI      02,F1,F1           F1 = 03
80EB 7D0301  8103   280   280           BNEH      0,F1,ERROR1
80EC F90331  8103   281   281           BNEL      3,F1,ERROR1
80ED 240141         282   282           XORI      04,F1,F1           F1 = 07
80EE 7D0301  8103   283   283           BNEH      0,F1,ERROR1
80EF 790371  8103   284   284           BNEL      7,F1,ERROR1
80F0 240181         285   285           XORI      08,F1,F1           F1 = 0F
80F1 7D0301  8103   286   286           BNEH      0,F1,ERROR1
80F2 F903F1  8103   287   287           BNEL      0F,F1,ERROR1
80F3 244101         288   288           XORI      10,F1,F1           F1 = 1F
80F4 FD0311  8103   289   289           BNEH      1,F1,ERROR1
80F5 F903F1  8103   290   290           BNEL      0F,F1,ERROR1
80F6 248101         291   291           XORI      20,F1,F1           F1 = 3F
80F7 7D0331  8103   292   292           BNEH      3,F1,ERROR1
80F8 F903F1  8103   293   293           BNEL      0F,F1,ERROR1
80F9 250101         294   294           XORI      40,F1,F1           F1 = 7F
80FA FD0371  8103   295   295           BNEH      7,F1,ERROR1
80FB F903F1  8103   296   296           BNEL      0F,F1,ERROR1
80FC 260101         297   297           XORI      80,F1,F1           F1 = FF
80FD F903F1  8103   298   298           BNEL      0F,F1,ERROR1
80FE 7D03F1  8103   299   299           BNEH      0F,F1,ERROR1
80FF A7C1F1         300   300           XORI      0FF,F1,F1          F1 = 00
8100 7D0301  8103   301   301           BNEH      0,F1,ERROR1
8101 F90301  8103   302   302           BNEL      0,F1,ERROR1
8102 DD0680  8106   303   303           B         TEST2
                    304   304  *
                    305   305  *
8103 574F80  834F   306   306  ERROR1   SB        ERROR              PRINT "ERROR" ON CRT
8104 200E01         307   307           MV        F1,K               DISPLAY FAILURE
8105 5CE480  80E4   308   308           B         TEST1              RESTART TEST
                    309   309  *
                    310   310  *
                    311   311  *
```

```
                    1    313  *
                    2    314  *TEST ANDI INSTRUCTION
                    3    315  *
8106 A0C02F         4    316  TEST2    MVI      32,F0          PUT A "2" IN F0
8107 D73F80  833F   5    317           SB       TEST#          PRINT "TEST #2" ON CRT
8108 A3C2FF         6    318           MVI      0FF,F2         PUT 1'S IN FILE REG. 2
8109 280112         7    319           ANDI     01,F2,F1       F1 = 01
810A FD2501  8125   8    320           BNEH     0,F1,ERROR2    F1 SHOULD = 01 IF NOT,ERROR
810B F92511  8125   9    321           BNEL     1,F1,ERROR2
810C 280122        10    322           ANDI     02,F2,F1       F1 = 02
810D FD2501  8125  11    323           BNEH     0,F1,ERROR2    F1 SHOULD = 02 IF NOT, ERROR
810E F92521  8125  12    324           BNEL     2,F1,ERROR2
810F 280142        13    325           ANDI     04,F2,F1       F1 = 04
8110 FD2501  8125  14    326           BNEH     0,F1,ERROR2
8111 F92541  8125  15    327           BNEL     4,F1,ERROR2
8112 280182        16    328           ANDI     08,F2,F1       F1 = 08
8113 FD2501  8125  17    329           BNEH     0,F1,ERROR2
8114 F92581  8125  18    330           BNEL     8,F1,ERROR2
8115 284102        19    331           ANDI     10,F2,F1       F1 = 10
8116 7D2511  8125  20    332           BNEH     1,F1,ERROR2
8117 792501  8125  21    333           BNEL     0,F1,ERROR2
8118 288102        22    334           ANDI     20,F2,F1       F1 = 20
8119 7D2521  8125  23    335           BNEH     2,F1,ERROR2
811A 792501  8125  24    336           BNEL     0,F1,ERROR2
811B 290102        25    337           ANDI     40,F2,F1       F1 = 40
811C 7D2541  8125  26    338           BNEH     4,F1,ERROR2
811D 792501  8125  27    339           BNEL     0,F1,ERROR2
811E 2A0102        28    340           ANDI     80,F2,F1       F1 = 80
811F 7D2581  8125  29    341           BNEH     8,F1,ERROR2
8120 792501  8125  30    342           BNEL     0,F1,ERROR2
8121 A80102        31    343           ANDI     00,F2,F1       F1 = 00
8122 FD2501  8125  32    344           BNEH     0,F1,ERROR2
8123 792501  8125  33    345           BNEL     0,F1,ERROR2
8124 DD2880  8128  34    346           B        TEST3
                   35    347  *
                   36    348  *
8125 574F80  834F  37    349  ERROR2   SB       ERROR          PRINT ERROR ON CRT
8126 200E01        38    350           MV       F1,K           DISPLAY FAILING F1
8127 DD0680  8106  39    351           B        TEST2          RESTART TEST
                   40    352  *
                   41    353  *TEST THE AI INSTRUCTION
                   42    354  *
8128 20C03F        43    355  TEST3    MVI      33,F0          PUT A "3" IN F0
8129 D73F80  833F  44    356           SB       TEST#          PRINT "TEST # 3 " ON CRT
812A A0010F        45    357           MVI      00,F1          CLEAR F1
```

```
812B AC0111          46   358          AI       01,F1,F1          F1 = 01
812C 7D4701  8147    47   359          BNEH     0,F1,ERROR3       F1 SHOULD = 01, ERROR IF NOT =
812D 794711  8147    48   360          BNEL     1,F1,ERROR3
812E AC0121          49   361          AI       02,F1,F1          F1 = 03
812F 7D4701  8147    50   362          BNEH     0,F1,ERROR3
8130 F94731  8147    51   363          BNEL     3,F1,ERROR3
8131 AC0141          52   364          AI       04,F1,F1          F1 = 07
8132 7D4701  8147    53   365          BNEH     0,F1,ERROR3
8133 794771  8147    54   366          BNEL     7,F1,ERROR3
8134 AC0181          55   367          AI       08,F1,F1          F1 = 0F
8135 7D4701  8147    56   368          BNEH     0,F1,ERROR3
8136 F947F1  8147    57   369          BNEL     0F,F1,ERROR3
8137 AC4101          58   370          AI       10,F1,F1          F1 = 1F
8138 FD4711  8147    59   371          BNEH     1,F1,ERROR3
8139 F947F1  8147    60   372          BNEL     0F,F1,ERROR3
813A AC8101          61   373          AI       20,F1,F1          F1 = 3F
813B 7D4731  8147    62   374          BNEH     3,F1,ERROR3
813C F947F1  8147    63   375          BNEL     0F,F1,ERROR3
813D AD0101          64   376          AI       40,F1,F1          F1 = 7F
813E FD4771  8147    65   377          BNEH     7,F1,ERROR3
813F F947F1  8147    66   378          BNEL     0F,F1,ERROR3
8140 AE0101          67   379          AI       80,F1,F1          F1 = FF
8141 F947F1  8147    68   380          BNEL     0F,F1,ERROR3
8142 7D47F1  8147    69   381          BNEH     0F,F1,ERROR3
8143 AC0111          70   382          AI       01,F1,F1          F1 = 00
8144 7D4701  8147    71   383          BNEH     0,F1,ERROR3
8145 F94701  8147    72   384          BNEL     0,F1,ERROR3
8146 5D4A80  814A    73   385          B        TEST4
                     74   386  *
                     75   387  *
8147 574F80  834F    76   388  ERROR3  SB       ERROR             PRINT "ERROR" ON CRT
8148 200E01          77   389          MV       F1,K              DISPLAY FAILURE
8149 DD2880  8128    78   390          B        TEST3             RESTART TEST
                     79   391  *
                     80   392  *TEST THE DACI INSTRUCTION
                     81   393  *
814A A0C04F          82   394  TEST4   MVI      34,F0             PUT A "4" IN F0
814B D73F80  833F    83   395          SB       TEST#             PRINT "TEST # 4 " ON CRT
814C 808FEF          84   396          OR,,0    +,,               CLEAR CARRY
814D A0020F          85   397          MVI      00,F2             CLEAR F2
814E A0010F          86   398          MVI      00,F1             CLEAR F1
814F 300211          87   399          DACI     01,F1,F2          F2 = 01
8150 7D7102  8171    88   400          BNEH     0,F2,ERROR4       F2 SHOULD = 01, ERROR IF NOT =
8151 797112  8171    89   401          BNEL     1,F2,ERROR4
8152 300221          90   402          DACI     02,F1,F2          F2 = 02
```

```
8153 7D7102  8171   91   403            BNEH     0,F2,ERROR4
8154 797122  8171   92   404            BNEL     2,F2,ERROR4
8155 300241         93   405            DACI     04,F1,F2          F2 = 04
8156 7D7102  8171   94   406            BNEH     0,F2,ERROR4
8157 797142  8171   95   407            BNEL     4,F2,ERROR4
8158 300281         96   408            DACI     08,F1,F2          F2 = 08
8159 7D7102  8171   97   409            BNEH     0,F2,ERROR4
815A 797182  8171   98   410            BNEL     8,F2,ERROR4
815B 304201         99   411            DACI     10,F1,F2          F2 = 10
815C FD7112  8171  100   412            BNEH     1,F2,ERROR4
815D F97102  8171  101   413            BNEL     0,F2,ERROR4
815E 308201        102   414            DACI     20,F1,F2          F2 = 20
815F FD7122  8171  103   415            BNEH     2,F2,ERROR4
8160 F97102  8171  104   416            BNEL     0,F2,ERROR4
8161 310201        105   417            DACI     40,F1,F2          F2 = 40
8162 FD7142  8171  106   418            BNEH     4,F2,ERROR4
8163 F97102  8171  107   419            BNEL     0,F2,ERROR4
8164 320201        108   420            DACI     80,F1,F2          F2 = 80
8165 FD7182  8171  109   421            BNEH     8,F2,ERROR4
8166 F97102  8171  110   422            BNEL     0,F2,ERROR4
8167 A0019F        111   423            MVI      09,F1             PUT A "9" IN F1
8168 300111        112   424            DACI     01,F1,F1          F1 = 10
8169 200201        113   425            MV       F1,F2             PUT RESULTS IN F2 FOR DISPLAY IF ERROR
816A F97101  8171  114   426            BNEL     0,F1,ERROR4
816B FD7111  8171  115   427            BNEH     1,F1,ERROR4
816C B24101        116   428            DACI     90,F1,F1          F1 = 00
816D 200201        117   429            MV       F1,F2             PUT F1 IN F2 ERROR DISPLAY
816E F97101  8171  118   430            BNEL     0,F1,ERROR4
816F 7D7101  8171  119   431            BNEH     0,F1,ERROR4
8170 DD7480  8174  120   432            B        TEST5
                   121   433  *
                   122   434  *
8171 574F80  834F  123   435  ERROR4    SB       ERROR             PRINT "ERROR" ON CRT
8172 200E02        124   436            MV       F2,K              DISPLAY FAILURE
8173 5D4A80  814A  125   437            B        TEST4             RESTART TEST
                   126   438  *
                   127   439  *TEST DSCI INSTRUCTION
                   128   440  *
8174 20C05F        129   441  TEST5     MVI      35,F0             PUT A "5" IN F0
8175 D73F80  833F  130   442            SB       TEST#             PRINT "TEST #5" ON CRT
8176 808FEF        131   443            OR,,0    +,,               CLEAR CARRY
8177 A0411F        132   444            MVI      11,F1             PUT INITIAL TEST PATTERN IN F1
8178 364291        133   445            DSCI     99,F1,F2          SUB 1 FROM 9 F2 SHOULD = 88
8179 799C82  819C  134   446            BNEL     8,F2,ERROR5
817A FD9C82  819C  135   447            BNEH     8,F2,ERROR5       BRANCH IF F2 NOT = 88
```

```
817B 808FEF         136  448          OR,,0     +,,              CLEAR CARRY
817C 360281         137  449          DSCI      88,F1,F2         F2 = 77
817D 799C72  819C   138  450          BNEL      7,F2,ERROR5
817E FD9C72  819C   139  451          BNEH      7,F2,ERROR5
817F 808FEF         140  452          OR,,0     +,,              CLEAR CARRY
8180 35C271         141  453          DSCI      77,F1,F2         F2 = 66
8181 F99C62  819C   142  454          BNEL      6,F2,ERROR5
8182 7D9C62  819C   143  455          BNEH      6,F2,ERROR5
8183 808FEF         144  456          OR,,0     +,,
8184 358261         145  457          DSCI      66,F1,F2         F2 = 55
8185 F99C52  819C   146  458          BNEL      5,F2,ERROR5
8186 7D9C52  819C   147  459          BNEH      5,F2,ERROR5
8187 808FEF         148  460          OR,,0     +,,
8188 354251         149  461          DSCI      55,F1,F2         F2 = 44
8189 799C42  819C   150  462          BNEL      4,F2,ERROR5
818A FD9C42  819C   151  463          BNEH      4,F2,ERROR5
818B 808FEF         152  464          OR,,0     +,,
818C 350241         153  465          DSCI      44,F1,F2         F2 = 33
818D F99C32  819C   154  466          BNEL      3,F2,ERROR5
818E 7D9C32  819C   155  467          BNEH      3,F2,ERROR5
818F 808FEF         156  468          OR,,0     +,,
8190 34C231         157  469          DSCI      33,F1,F2         F2 = 22
8191 799C22  819C   158  470          BNEL      2,F2,ERROR5
8192 FD9C22  819C   159  471          BNEH      2,F2,ERROR5
8193 808FEF         160  472          OR,,0     +,,
8194 348221         161  473          DSCI      22,F1,F2         F2 = 11
8195 799C12  819C   162  474          BNEL      1,F2,ERROR5
8196 FD9C12  819C   163  475          BNEH      1,F2,ERROR5
8197 808FEF         164  476          OR,,0     +,,
8198 344211         165  477          DSCI      11,F1,F2         F2 = 00
8199 F99C02  819C   166  478          BNEL      0,F2,ERROR5
819A 7D9C02  819C   167  479          BNEH      0,F2,ERROR5
819B DD9F80  819F   168  480          B         TEST6
                    169  481  *
                    170  482  *
819C 574F80  834F   171  483  ERROR5  SB        ERROR            PRINT "ERROR" ON CRT
819D 200E01         172  484          MV        F1,K             DISPLAY F1
819E DD7480  8174   173  485          B         TEST5            RESTART TEST
                    174  486  *
                    175  487  *TEST ACI INSTRUCTION
                    176  488  *
819F 20C06F         177  489  TEST6   MVI       36,F0            PUT A "6" IN F0
81A0 D73F80  833F   178  490          SB        TEST#            PRINT "TEST # 6 " ON CRT
81A1 A0010F         179  491          MVI       00,F1            CLEAR F1
81A2 808FEF         180  492          OR,,0     +,,              CLEAR CARRY
```

```
81A3 B80111          181  493          ACI      01,F1,F1          F1 = 01
81A4 FDC701  81C7    182  494          BNEH     0,F1,ERROR6       F1 SHOULD = 01, ERROR IF NOT =
81A5 F9C711  81C7    183  495          BNEL     1,F1,ERROR6
81A6 808FEF          184  496          OR,,0    +,,               CLEAR CARRY
81A7 B80121          185  497          ACI      02,F1,F1          F1 = 03
81A8 FDC701  81C7    186  498          BNEH     0,F1,ERROR6
81A9 79C731  81C7    187  499          BNEL     3,F1,ERROR6
81AA 808FEF          188  500          OR,,0    +,,
81AB B80141          189  501          ACI      04,F1,F1          F1 = 07
81AC FDC701  81C7    190  502          BNEH     0,F1,ERROR6
81AD F9C771  81C7    191  503          BNEL     7,F1,ERROR6
81AE 808FEF          192  504          OR,,0    +,,
81AF B80181          193  505          ACI      08,F1,F1          F1 = 0F
81B0 FDC701  81C7    194  506          BNEH     0,F1,ERROR6
81B1 79C7F1  81C7    195  507          BNEL     0F,F1,ERROR6
81B2 808FEF          196  508          OR,,0    +,,
81B3 B84101          197  509          ACI      10,F1,F1          F1 = 1F
81B4 7DC711  81C7    198  510          BNEH     1,F1,ERROR6
81B5 79C7F1  81C7    199  511          BNEL     0F,F1,ERROR6
81B6 808FEF          200  512          OR,,0    +,,
81B7 B88101          201  513          ACI      20,F1,F1          F1 = 3F
81B8 FDC731  81C7    202  514          BNEH     3,F1,ERROR6
81B9 79C7F1  81C7    203  515          BNEL     0F,F1,ERROR6
81BA 808FEF          204  516          OR,,0    +,,
81BB B90101          205  517          ACI      40,F1,F1          F1 = 7F
81BC 7DC771  81C7    206  518          BNEH     7,F1,ERROR6
81BD 79C7F1  81C7    207  519          BNEL     0F,F1,ERROR6
81BE 808FEF          208  520          OR,,0    +,,
81BF BA0101          209  521          ACI      80,F1,F1          F1 = FF
81C0 79C7F1  81C7    210  522          BNEL     0F,F1,ERROR6
81C1 FDC7F1  81C7    211  523          BNEH     0F,F1,ERROR6
81C2 808FEF          212  524          OR,,0    +,,
81C3 B80111          213  525          ACI      01,F1,F1          F1 = 00
81C4 FDC701  81C7    214  526          BNEH     0,F1,ERROR6
81C5 79C701  81C7    215  527          BNEL     0,F1,ERROR6
81C6 DDCA80  81CA    216  528          B        TEST7
                     217  529  *
                     218  530  *
81C7 574F80  834F    219  531  ERROR6   SB       ERROR             PRINT "ERROR" ON CRT
81C8 200E01          220  532          MV       F1,K              DISPLAY FAILURE
81C9 DD9F80  819F    221  533          B        TEST6             RESTART TEST
                     222  534  *
                     223  535  *TEST MI INSTRUCTION
                     224  536  *
81CA A0C07F          225  537  TEST7   MVI      37,F0             PUT "7" IN F0
```

```
81CB D73F80   833F   226   538           SB        TEST#             PRINT "TEST#7" ON CRT
81CC A0020F          227   539           MVI       00,F2             CLEAR F2
81CD A0411F          228   540           MVI       11,F1             PUT 11 IN F1
81CE 3C0211          229   541           MIL       1,F1,F2           MULT. F1 * 1, F2 = 01
81CF BC8211          230   542           MIH       1,F1,F2           MULT. (H) F1 * 1, F2 = 01
81D0 79E712   81E7   231   543           BNEL      1,F2,ERROR7
81D1 7DE702   81E7   232   544           BNEH      0,F2,ERROR7
81D2 3C0221          233   545           MIL       2,F1,F2           F2 = 02
81D3 BC8221          234   546           MIH       2,F1,F2           F2 = 02
81D4 79E722   81E7   235   547           BNEL      2,F2,ERROR7
81D5 7DE702   81E7   236   548           BNEH      0,F2,ERROR7
81D6 3C0241          237   549           MIL       4,F1,F2           F2 = 04
81D7 BC8241          238   550           MIH       4,F1,F2           F2 = 04
81D8 79E742   81E7   239   551           BNEL      4,F2,ERROR7
81D9 7DE702   81E7   240   552           BNEH      0,F2,ERROR7
81DA 3C0281          241   553           MIL       8,F1,F2           F2 = 08
81DB BC8281          242   554           MIH       8,F1,F2           F2 = 08
81DC 79E782   81E7   243   555           BNEL      8,F2,ERROR7
81DD 7DE702   81E7   244   556           BNEH      0,F2,ERROR7
81DE 20012F          245   557           MVI       02,F1             PUT A "02" IN F1
81DF 3C0281          246   558           MIL       8,F1,F2           F2 = 10
81E0 F9E702   81E7   247   559           BNEL      0,F2,ERROR7
81E1 FDE712   81E7   248   560           BNEH      1,F2,ERROR7
81E2 20810F          249   561           MVI       20,F1             PUT "20" IN F1
81E3 BC8281          250   562           MIH       8,F1,F2           F2 = 10
81E4 F9E702   81E7   251   563           BNEL      0,F2,ERROR7
81E5 FDE712   81E7   252   564           BNEH      1,F2,ERROR7
81E6 5DEA80   81EA   253   565           B         TEST8
                     254   566  *
                     255   567  *
81E7 574F80   834F   256   568  ERROR7   SB        ERROR             PRINT "ERROR" ON CRT
81E8 200E02          257   569           MV        F2,K              DISPLAY F2
81E9 DDCA80   81CA   258   570           B         TEST7             REPEAT TEST
                     259   571  *
                     260   572  *TEST EXTENDED INSTRUCTIONS
                     261   573  *
81EA A0C08F          262   574  TEST8    MVI       38,F0             PUT "8" IN F0
81EB D73F80   833F   263   575           SB        TEST#             PRINT "TEST#8" ON CRT
81EC A0070F          264   576           MVI       00,F7             PUT 0'S IN F7
81ED A00607          265   577           MV        F7,F6             PUT 0'S IN F6
81EE 200306          266   578           MV        F6,F3             PUT 0'S IN F3
81EF A00203          267   579           MV        F3,F2             PUT 0'S IN F2
81F0 A3C4FF          268   580           MVI       0FF,F4            PUT 1'S IN F4
81F1 A00504          269   581           MV        F4,F5             PUT 1'S IN F5
81F2 020624          270   582           ORX       F3F2,F5F4,F7F6    OR 0'S WITH 1'S, F7F6 = FFFF
```

```
81F3 A00E06          271  583          MV       F6,K              DISPLAY F6
81F4 5A1D46 821D     272  584          BNR      F4,F6,ERROR8      F6 SHOULD = FF
81F5 200E07          273  585          MV       F7,K              DISPLAY F7
81F6 59E757 81E7     274  586          BNR      F5,F7,ERROR7      F7 SHOULD = FF
81F7 860666          275  587          XORX     F7F6,F7F6,F7F6    F7 AND F6 SHOULD = 00
81F8 A00E06          276  588          MV       F6,K              DISPLAY F6
81F9 DA1D36 821D     277  589          BNR      F3,F6,ERROR8
81FA 200E07          278  590          MV       F7,K              DISPLAY F7
81FB 5A1D37 821D     279  591          BNR      F3,F7,ERROR8
81FC 8A0644          280  592          ANDX     F5F4,F5F4,F7F6    AND 1'S WITH 1'S, F7 AND F6 SHOULD = FF
81FD A00E06          281  593          MV       F6,K              DISPLAY F6
81FE DA1D56 821D     282  594          BNR      F5,F6,ERROR8
81FF 200E07          283  595          MV       F7,K              DISPLAY F7
8200 5A1D57 821D     284  596          BNR      F5,F7,ERROR8
8201 8EC664          285  597          SCX,,1   F7F6,F5F4,F7F6    SUB. 1'S FROM 1'S, F7 AND F6 SHOULD =00
8202 A00E06          286  598          MV       F6,K              DISPLAY F6
8203 DA1D36 821D     287  599          BNR      F3,F6,ERROR8
8204 200E07          288  600          MV       F7,K              DISPLAY F7
8205 5A1D37 821D     289  601          BNR      F3,F7,ERROR8
8206 1A8242          290  602          ACX,,0   F5F4,F3F2,F3F2    ADD 1'S TO 0'S, F3 AND F2 SHOULD = FF
8207 A00E03          291  603          MV       F3,K              DISPLAY F3
8208 DA1D35 821D     292  604          BNR      F3,F5,ERROR8
8209 200E02          293  605          MV       F2,K              DISPLAY F2
820A 5A1D25 821D     294  606          BNR      F2,F5,ERROR8
820B A2429F          295  607          MVI      99,F2             PUT "99" IN F2
820C A00302          296  608          MV       F2,F3             PUT "99 IN F3
820D 8204E2          297  609          MVX      F3F2,F5F4         PUT "99" IN F5 AND F4
820E 23C6FF          298  610          MVI      0FF,F6            PUT 1'S IN F6
820F A00706          299  611          MV       F6,F7             PUT 1'S IN F6
8210 968624          300  612          DSCX,,0  F3F2,F5F4,F7F6    SUB. 9999 FROM 9999, F7F6 = 0000
8211 A00E06          301  613          MV       F6,K              DISPLAY F6
8212 7A1D06 821D     302  614          BNEL     0,F6,ERROR8
8213 FE1D06 821D     303  615          BNEH     0,F6,ERROR8
8214 200E07          304  616          MV       F7,K              DISPLAY F7
8215 5A1D67 821D     305  617          BNR      F6,F7,ERROR8
8216 928224          306  618          DACX,,0  F3F2,F5F4,F3F2    ADD 9999 TO 9999 F3F2 = 9998
8217 A00E03          307  619          MV       F3,K
8218 DA1D53 821D     308  620          BNR      F5,F3,ERROR8
8219 200E02          309  621          MV       F2,K
821A 7A1D82 821D     310  622          BNEL     8,F2,ERROR8
821B 7E1D92 821D     311  623          BNEH     9,F2,ERROR8
821C 5E1F80 821F     312  624          B        TEST9
                     313  625  *
                     314  626  *
821D 574F80 834F     315  627  ERROR8   SB       ERROR
```

6/1/78

821E 5DEA80 81EA 316 628 B TEST8

```
                    1    630  *
                    2    631  *TEST CARRY WITH REGISTER INSTRUCTIONS
                    3    632  *
821F 20C09F         4    633  TEST9    MVI       39,F0            PUT A "9" IN F0
8220 D73F80  833F   5    634           SB        TEST#            PRINT "TEST#9" ON CRT
8221 808FEF         6    635           OR,,0     +,,              SET CARRY BIT = 0
8222 62631D  8263   7    636           BTL       1,SH,ERROR9      ERROR IF CARRY BIT SET (= TO 1)
8223 00CFEF         8    637           OR,,1     +,,              SET CARRY BIT = 1
8224 EA631D  8263   9    638           BFL       1,SH,ERROR9      ERROR IF CARRY BIT CLEAR
8225 048FEF         10   639           XOR,,0    +,,              CLEAR CARRY
8226 62631D  8263   11   640           BTL       1,SH,ERROR9      ERROR IF CARRY SET
8227 84CFEF         12   641           XOR,,1    +,,              SET CARRY
8228 EA631D  8263   13   642           BFL       1,SH,ERROR9      ERROR IF CARRY CLEAR
8229 088FEF         14   643           AND,,0    +,,              CLEAR CARRY
822A 62631D  8263   15   644           BTL       1,SH,ERROR9      ERROR IF CARRY SET
822B 88CFEF         16   645           AND,,1    +,,              SET CARRY
822C EA631D  8263   17   646           BFL       1,SH,ERROR9      ERROR IF CARRY CLEAR
                    18   647  *
                    19   648  *ARITHMETIC CARRY OPERATIONS
                    20   649  *
822D A3C2FF         21   650           MVI       0FF,F2           PUT "FF" IN F2
822E A00302         22   651           MV        F2,F3            F3 = "FF"
822F A00403         23   652           MV        F3,F4            F4 = "FF"
8230 808FEF         24   653           OR,,0     +,,              CLEAR CARRY
8231 8CC423         25   654           SC,,1     F2,F3,F4         FF - FF = 00    F4 SHOULD = 00
8232 200E04         26   655           MV        F4,K             DISPLAY F4
8233 FA6304  8263   27   656           BNEL      0,F4,ERROR9      CHECK F4 = 00
8234 7E6304  8263   28   657           BNEH      0,F4,ERROR9
8235 200E0D         29   658           MV        SH,K             DISPLAY SH (SH 1 CARRY BIT)
8236 EA631D  8263   30   659           BFL       1,SH,ERROR9      CHECK CARRY SET
                    31   660  *
                    32   661  *
8237 20011F         33   662           MVI       01,F1            F1 = 1
8238 A3C2FF         34   663           MVI       0FF,F2           F2 = FF
8239 A00302         35   664           MV        F2,F3            F3 = FF
823A 808FEF         36   665           OR,,0     +,,              CLEAR CARRY
823B 188312         37   666           AC,,0     F1,F2,F3         1 + FF = 00 CARRY SET
823C A00E03         38   667           MV        F3,K             DISPLAY F3
823D 7A6303  8263   39   668           BNEL      0,F3,ERROR9      CHECK F3 = 00
823E FE6303  8263   40   669           BNEH      0,F3,ERROR9
823F 200E0D         41   670           MV        SH,K             DISPLAY SH
8240 EA631D  8263   42   671           BFL       1,SH,ERROR9      CHECK CARRY, SHOULD BE SET
8241 808FEF         43   672           OR,,0     +,,              CLEAR CARRY
8242 A0010F         44   673           MVI       00,F1            F1 = 00
8243 98C312         45   674           AC,,1     F1,F2,F3         00 + FF + 1 (CARRY) = 00 CARRY SET
```

```
8244 A00E03         46   675          MV        F3,K              DISPLAY F3
8245 7A6303  8263   47   676          BNEL      0,F3,ERROR9       CHECK F3 = 00
8246 FE6303  8263   48   677          BNEH      0,F3,ERROR9
8247 200E0D         49   678          MV        SH,K              DISPLAY SH
8248 EA631D  8263   50   679          BFL       1,SH,ERROR9       CHECK CARRY SET
                    51   680  *
                    52   681  *
8249 808FEF         53   682          OR,,0     +,,               CLEAR CARRY
824A 20011F         54   683          MVI       01,F1             F1 = 1
824B A0029F         55   684          MVI       09,F2             F2 = 9
824C 23C3FF         56   685          MVI       0FF,F3            F3 = FF
824D 908312         57   686          DAC,,0    F1,F2,F3          1 + 9 + 0 (CARRY) = 10 CARRY CLEAR
824E A00E03         58   687          MV        F3,K              DISPLAY F3
824F 7A6303  8263   59   688          BNEL      0,F3,ERROR9       CHECK F3 = 10
8250 7E6313  8263   60   689          BNEH      1,F3,ERROR9
8251 62631D  8263   61   690          BTL       1,SH,ERROR9       CHECK FOR CARRY CLEAR
8252 A2429F         62   691          MVI       99,F2             F2 = 99
8253 908312         63   692          DAC,,0    F1,F2,F3          1 + 99 + 0 = 00 CARRY SET
8254 A00E03         64   693          MV        F3,K              DISPLAY F3
8255 7A6303  8263   65   694          BNEL      0,F3,ERROR9       CHECK F3 = 00
8256 FE6303  8263   66   695          BNEH      0,F3,ERROR9
8257 200E0D         67   696          MV        SH,K              DISPLAY SH
8258 EA631D  8263   68   697          BFL       1,SH,ERROR9       CHECK CARRY SET
8259 A0010F         69   698          MVI       00,F1             F1 = 00
825A 808FEF         70   699          OR,,0     +,,               CLEAR CARRY
825B 10C312         71   700          DAC,,1    F1,F2,F3          0 + 99 + 1 = 00 CARRY SET
825C A00E03         72   701          MV        F3,K              DISPLAY F3
825D 7A6303  8263   73   702          BNEL      0,F3,ERROR9       CHECK F3 = 00
825E FE6303  8263   74   703          BNEH      0,F3,ERROR9
825F 200E0D         75   704          MV        SH,K              DISPLAY SH
8260 EA631D  8263   76   705          BFL       1,SH,ERROR9       CHECK CARRY SET
8261 808FEF         77   706          OR,,0     +,,               CLEAR CARRY
8262 DE6680  8266   78   707          B         TESTA
                    79   708  *
                    80   709  *
8263 574F80  834F   81   710  ERROR9  SB        ERROR
8264 200E0D         82   711          MV        SH,K
8265 5E1F80  821F   83   712          B         TEST9
                    84   713  *
                    85   714  *TEST PH AND PL MUX. TO B BUS
                    86   715  *
8266 21001F         87   716  TESTA   MVI       41,F0             PUT AN "A" IN F0
8267 D73F80  833F   88   717          SB        TEST#             PRINT "TEST#A" ON CRT
8268 A141AF         89   718          MVI       5A,F1             F1 = 5A
8269 A2825F         90   719          MVI       0A5,F2            F2 = A5
```

```
826A 200801          91   720          MV       F1,PL              PL = 5A
826B A00902          92   721          MV       F2,PH              PH = A5
826C 5A7618  8276    93   722          BNR      F1,PL,ERRORA       CHECK PL = 5A
826D DA7629  8276    94   723          BNR      F2,PH,ERRORA       CHECK PH = A5
826E A00809          95   724          MV       PH,PL              PL = A5
826F 5A7628  8276    96   725          BNR      F2,PL,ERRORA
8270 A00908          97   726          MV       PL,PH              PH = A5
8271 DA7629  8276    98   727          BNR      F2,PH,ERRORA
8272 A148AF          99   728          MVI      5A,PL              PL = 5A
8273 A00908          100  729          MV       PL,PH              PH = 5A
8274 DA7619  8276    101  730          BNR      F1,PH,ERRORA
8275 DE7880  8278    102  731          B        TESTB
                     103  732  *
                     104  733  *
8276 574F80  834F    105  734  ERRORA  SB       ERROR              PRINT ERROR ON CRT
8277 DE6680  8266    106  735          B        TESTA              RESTART TEST
                     107  736  *
                     108  737  *TEST DATA MEMORY INPUT REGISTER
                     109  738  *
8278 57CE80  83CE    110  739  TESTB   SB       SETPAR             SET GOOD DATA MEM. PARITY NO TRAP OFF
8279 21002F          111  740          MVI      42,F0              PUT A "B" IN F0
827A D73F80  833F    112  741          SB       TEST#              PRINT "TEST#B" ON CRT
827B 190000  0000    113  742          LPI      0000               CLEAR PC'S
827C A00E0F          114  743          MVI      00,K               CLEAR K
827D A0010F          115  744          MVI      00,F1              CLEAR F1
827E 200E01          116  745  LOOPB   MV       F1,K               K = F1
827F 002FEE          117  746          OR,W1    +,K,               WRITE DATA MEM. WITH CONTENTS OF K STEP PC'S
8280 FA82F1  8282    118  747          BNEL     0F,F1,*+2          CHECK FOR LAST WRITE
8281 F684F1  8284    119  748          BEQH     0F,F1,*+3
8282 AC0111          120  749          AI       1,F1,F1            STEP F1
8283 DE7E80  827E    121  750          B        LOOPB              NEXT WRITE
8284 190000  0000    122  751          LPI      0000               CLEAR PC'S
8285 A00E0F          123  752          MVI      00,K               CLEAR K
8286 A0010F          124  753          MVI      00,F1              CLEAR F1
8287 801FEF          125  754  LOOPBR  OR,R     +,,                READ MEMORY STEP PC'S
8288 200E0B          126  755          MV       CH,K               DISPLAY CH
8289 5AAE1B  82AE    127  756          BNR      F1,CH,ERRORB       COMPARE COUNTER TO DATA READ
828A 7A8CF1  828C    128  757          BNEL     0F,F1,*+2          CHECK FOR LAST READ
828B F68EF1  828E    129  758          BEQH     0F,F1,*+3
828C AC0111          130  759          AI       1,F1,F1            STEP COUNTER
828D DE8780  8287    131  760          B        LOOPBR             NEXT READ
                     132  761  *
                     133  762  *
828E 190000  0000    134  763          LPI      0000               CLEAR PC'S
828F 20030F          135  764          MVI      00,F3              CLEAR F3
```

```
8290 A141AF        136  765          MVI       5A,F1          F1 = 5A
8291 A2825F        137  766          MVI       0A5,F2         F2 = 0A5
8292 8C6312        138  767          SHHH,W1   F1,F2,F3       F3 = A5     LOCATION 0000 = A5
8293 A00E03        139  768          MV        F3,K           DISPLAY F3
8294 000FEF        140  769          OR        +,,            STEP PC'S
8295 806312        141  770          SHLL,W1   F1,F2,F3       F3 = 5A     LOCATION 0001 = 5A
8296 A00E03        142  771          MV        F3,K           DISPLAY F3
8297 190000  0000  143  772          LPI       0000           CLEAR PC'S
8298 801FEF        144  773          OR,R      +,,            READ LOCATIONS 0000  AND   0001
8299 200E0B        145  774          MV        CH,K           DISPLAY 0000
829A 7AAE5B  82AE  146  775          BNEL      5,CH,ERRORB    CHECK 0000 = A5
829B FEAEAB  82AE  147  776          BNEH      0A,CH,ERRORB
829C A00E0A        148  777          MV        CL,K           DISPLAY 0001
829D FAAEAA  82AE  149  778          BNEL      0A,CL,ERRORB   CHECK 0001 = 5A
829E 7EAE5A  82AE  150  779          BNEH      5,CL,ERRORB
                   151  780  *
                   152  781  *
829F 190000  0000  153  782          LPI       0000           CLEAR PC'S
82A0 A0020F        154  783          MVI       00,F2          CLEAR F2
82A1 23C11F        155  784          MVI       0F1,F1         F1 = F1
82A2 BC2211        156  785          MIL,W1    1,F1,F2        F2 = 01     LOCATION 0000 = 01
82A3 000FEF        157  786          OR        +,,            STEP PC'S
82A4 3CA211        158  787          MIH,W1    1,F1,F2        F2 = 0F     LOCATION 0001 = 0F
82A5 190000  0000  159  788          LPI       0000           CLEAR PC'S
82A6 801FEF        160  789          OR,R      +,,            READ LOCATIONS 0000 AND 0001
82A7 200E0B        161  790          MV        CH,K           DISPLAY 0000
82A8 FAAE1B  82AE  162  791          BNEL      1,CH,ERRORB    0000 SHOULD = 01
82A9 FEAE0B  82AE  163  792          BNEH      0,CH,ERRORB
82AA A00E0A        164  793          MV        CL,K           DISPLAY 0001
82AB FAAEFA  82AE  165  794          BNEL      0F,CL,ERRORB   0001 SHOULD = 0F
82AC 7EAE0A  82AE  166  795          BNEH      0,CL,ERRORB
82AD 5EB080  82B0  167  796          B         TESTC
                   168  797  *
                   169  798  *
82AE 574F80  834F  170  799  ERRORB  SB        ERROR          PRINT ERROR ON CRT
82AF DE7880  8278  171  800          B         TESTB          RESTART TEST
                   172  801  *
                   173  802  *
82B0 A1003F        174  803  TESTC   MVI       43,F0          PUT A "C" IN F0
82B1 D73F80  833F  175  804          SB        TEST#          PRINT "TEST#C" ON CRT
82B2 A0020F        176  805  TESTC   MVI       00,F2          F2 = 00
82B3 2003FF        177  806          MVI       0F,F3          F3 = 0F
82B4 A3C40F        178  807          MVI       0F0,F4         F4 = F0
82B5 23C5FF        179  808          MVI       0FF,F5         F5 = FF
82B6 000122        180  809          OR        F2,F2,F1       00 ORED WITH 00 = 00
```

```
82B7 5ADF12   82DF   181   810          BNR     F1,F2,ERRORC     F1 = 00 ?
82B8 800123          182   811          OR      F2,F3,F1         00 ORED WITH 0F = 0F
82B9 DADF31   82DF   183   812          BNR     F3,F1,ERRORC     F1 = 0F ?
82BA 000124          184   813          OR      F2,F4,F1         00 ORED WITH F0 = F0
82BB 5ADF41   82DF   185   814          BNR     F4,F1,ERRORC     F1 = F0 ?
82BC 800125          186   815          OR      F2,F5,F1         00 ORED WITH FF = FF
82BD DADF51   82DF   187   816          BNR     F5,F1,ERRORC     F1 = FF ?
                     188   817  *
                     189   818  *
82BE 800132          190   819          OR      F3,F2,F1         0F ORED WITH 00 = 0F
82BF DADF31   82DF   191   820          BNR     F3,F1,ERRORC     F1 = 0F ?
82C0 000142          192   821          OR      F4,F2,F1         F0 ORED WITH 00 = F0
82C1 5ADF41   82DF   193   822          BNR     F4,F1,ERRORC     F1 = F0 ?
82C2 800152          194   823          OR      F5,F2,F1         FF ORED WITH 00 = FF
82C3 DADF51   82DF   195   824          BNR     F5,F1,ERRORC     F1 = FF ?
                     196   825  *
                     197   826  *
82C4 880122          198   827          AND     F2,F2,F1         00 ANDED WITH 00 = 00
82C5 5ADF21   82DF   199   828          BNR     F2,F1,ERRORC     F1 = 00 ?
82C6 880153          200   829          AND     F5,F3,F1         FF ANDED WITH 0F = 0F
82C7 DADF31   82DF   201   830          BNR     F3,F1,ERRORC
82C8 080154          202   831          AND     F5,F4,F1         FF ANDED WITH F0 = F0
82C9 5ADF41   82DF   203   832          BNR     F4,F1,ERRORC
82CA 880155          204   833          AND     F5,F5,F1         FF ANDED WITH FF = FF
82CB DADF51   82DF   205   834          BNR     F5,F1,ERRORC     F5 = FF ?
                     206   835  *
                     207   836  *
82CC 880135          208   837          AND     F3,F5,F1         0F ANDED WITH FF = 0F
82CD DADF31   82DF   209   838          BNR     F3,F1,ERRORC     F1 = 0F ?
82CE 080145          210   839          AND     F4,F5,F1         F0 ANDED WITH FF = F0
82CF 5ADF41   82DF   211   840          BNR     F4,F1,ERRORC     F1 = F0 ?
                     212   841  *
                     213   842  *
82D0 840155          214   843          XOR     F5,F5,F1         FF XORED WITH FF = 00
82D1 5ADF21   82DF   215   844          BNR     F2,F1,ERRORC     F1 = 00 ?
82D2 040154          216   845          XOR     F5,F4,F1         FF XORED WITH F0 = 0F
82D3 DADF31   82DF   217   846          BNR     F3,F1,ERRORC     F1 = 0F ?
82D4 840153          218   847          XOR     F5,F3,F1         FF XORED WITH 0F = F0
82D5 5ADF41   82DF   219   848          BNR     F4,F1,ERRORC     F1 = F0 ?
82D6 040152          220   849          XOR     F5,F2,F1         FF XORED WITH 00 = FF
82D7 DADF51   82DF   221   850          BNR     F5,F1,ERRORC     F1 = FF ?
                     222   851  *
                     223   852  *
82D8 040145          224   853          XOR     F4,F5,F1         F0 XORED WITH FF = 0F
82D9 DADF31   82DF   225   854          BNR     F3,F1,ERRORC     F1 = 0F ?
```

```
82DA 840135          226  855          XOR     F3,F5,F1        OF XORED WITH FF = F0
82DB 5ADF41  82DF    227  856          BNR     F4,F1,ERRORC    F1 = F0 ?
82DC 040125          228  857          XOR     F2,F5,F1        00 XORED WITH FF = FF
82DD DADF51  82DF    229  858          BNR     F5,F1,ERRORC    F1 = FF ?
                     230  859  *
                     231  860  *
82DE DEE280  82E2    232  861          B       TESTD
                     233  862  *
                     234  863  *
82DF 574F80  834F    235  864  ERRORC   SB      ERROR           PRINT "ERROR" ON CRT
82E0 200E01          236  865          MV      F1,K            DISPLAY FAILURE
82E1 5EB080  82B0    237  866          B       TESTC           RESTART TEST
                     238  867  *
                     239  868  *
82E2 21004F          240  869  TESTD   MVI     44,F0
82E3 D73F80  833F    241  870          SB      TEST#           PRIINT TEST#D ON CRT
82E4 23C6FF          242  871          MVI     0FF,F6
82E5 A0C7FF          243  872          MVI     3F,F7           SET UP REGS. FOR 16K MEMORY
82E6 190000  0000    244  873          LPI     0000            CLEAR PC'S
82E7 8202E8          245  874          MVX     PHPL,F3F2
82E8 8204E2          246  875          MVX     F3F2,F5F4       CLEAR FILE REGISTERS
82E9 A00E0F          247  876          MVI     00,K            CLEAR K REG.
82EA 81800F          248  877          TPA     ,00             PUT PC'S IN AUX REG 0
82EB 9D02F7  82F7    249  878  LOOPADD LPI     INCRPL          LOAD PC'S WITH RETURN ADD FOR WCM
82EC 01801F          250  879          TPA     ,01             PUT RETURN ADDR. IN AUX 1
82ED 05800F          251  880          TPS     ,               PUSH RETURN ADDR ON STACK
82EE 8B800F          252  881          TAP     ,00             GET CM ADDR. FOR WRITE
82EF 05800F          253  882          TPS     ,               PUSH CONTROL MEMORY ADDR. ON STACK
82F0 A00E0F          254  883          MVI     00,K            CLEAR K REG. FOR WRITE OPERATION
82F1 200108          255  884          MV      PL,F1           SET UP F1 FOR MASK
82F2 280111          256  885          ANDI    1,F1,F1         MASK OFF ALL BUT BIT 0
82F3 FAF501  82F5    257  886          BNEL    0,F1,*+2        SKIP IF BIT 0 = 1
82F4 A20E0E          258  887          ORI     80,K,K          SET PARITY BIT IF EVEN MEMORY ADDR.
82F5 A7CEFE          259  888          XORI    0FF,K,K         COMPLEMENT K BEFORE WRITE
82F6 078400          260  889          SR,WCM  ,               WRITE CONTROL MEMORY, GO TO INCRPL
82F7 AC0818          261  890  INCRPL   AI      1,PL,PL         INCREMENT PL FOR NEXT ADDR.
82F8 FAFAF8  82FA    262  891          BNEL    0F,PL,*+2
82F9 F6FCF8  82FC    263  892          BEQH    0F,PL,INCRPH    IF PL=FF, THEN GO TO INCREMENT PH
82FA 81800F          264  893          TPA     ,00             PUT INCREMENTED PC IN AUX 0
82FB DEEB80  82EB    265  894          B       LOOPADD         EXECUTE NEXT WRITE
82FC AC0919          266  895  INCRPH   AI      1,PH,PH         INCR. PH
82FD 530079  8300    267  896          BER     F7,PH,*+3       IF THIS IS LAST ADDR. GO TO READ CM
82FE 81800F          268  897          TPA     ,00             PUT INCREMENTED PC IN AUX 0
82FF DEEB80  82EB    269  898          B       LOOPADD         NEXT WRITE CYCLE
8300 190000  0000    270  899          LPI     0000            CLR PC'S
```

```
8301 8202E8          271  900          MVX     PHPL,F3F2       CLR F3 F2
8302 A00E0F          272  901          MVI     00,K            CLR K
8303 81800F          273  902          TPA     ,00             SET AUX 0 = TO PC'S
8304 9D030A    830A  274  903  READCM  LPI     INCRK           LOAD PC'S WITH RETURN ADDR. FOR READ
8305 01801F          275  904          TPA     ,01             PUT RETURN ADDR IN AUX 1
8306 05800F          276  905          TPS     ,               PUSH ADDR ON STACK
8307 8B800F          277  906          TAP     ,00             GET READ ADDR FROM AUX 0
8308 05800F          278  907          TPS     ,               PUSH ADDR ON STACK
8309 878600          279  908          SR,RCM  ,               READ CONTROL MEMORY
830A 29CEFE          280  909  INCRK   ANDI    7F,K,K          MASK OFF PARITY BIT FROM K
830B FB190E    8319  281  910          BNEL    0,K,ERRORD
830C 7F190E    8319  282  911          BNEH    0,K,ERRORD      PRINT ERROR ON CRT IF K NOT CLEAR
830D DB1928    8319  283  912          BNR     F2,PL,ERRORD    COMPARE COUNTER TO DATA READ
830E DB1939    8319  284  913          BNR     F3,PH,ERRORD    COMPARE COUNTER TO DATA READ
830F AC0212          285  914          AI      1,F2,F2         INCREMENT F2
8310 7B12F2    8312  286  915          BNEL    0F,F2,*+2       CHECK F2 NOT = F
8311 F713F2    8313  287  916          BEQH    0F,F2,INCRF3    CHECK F2 = TO FF
8312 DF1580    8315  288  917          B       INCRPC
8313 AC0313          289  918  INCRF3  AI      1,F3,F3         INCR F3
8314 531B37    831B  290  919          BER     F3,F7,RESTART
8315 8208E2          291  920  INCRPC  MVX     F3F2,PHPL       INCR PC'S
8316 81800F          292  921          TPA     ,00             PUT INCREMENTED PC'S IN AUX 0
8317 DF0480    8304  293  922          B       READCM
                     294  923  *
                     295  924  *
                     296  925  *
8318 5F1B80    831B  297  926          B       RESTART
8319 574F80    834F  298  927  ERRORD  SB      ERROR
831A DEE280    82E2  299  928          B       TESTD
                     300  929  *
                     301  930  *
831B 572980    8329  302  931  RESTART SB      INIT
831C DCBF80    80BF  303  932          B       TEST0
```

```
831D A00E5F        1   934  ENABLE5  MVI   5,K          SET CRT ADDR.
831E 178C00        2   935  ENABLE   CIO   0C0          ADDR. STROBE
831F 572080 8320   3   936  DELAY10  SB    DELAY5
8320 D72180 8321   4   937  DELAY5   SB    *+1
8321 D72280 8322   5   938           SB    *+1
8322 200F0F        6   939           NOP   ,
8323 200F0F        7   940           NOP   ,
8324 87800F        8   941           SR    ,
                   9   942  *
                   10  943  *
8325 EB258D 8325   11  944  OBSTROBE BFL   8,SH,*       WAIT FOR DEVIE READY
8326 572080 8320   12  945           SB    DELAY5
8327 978200        13  946           CIO   20
8328 DF2080 8320   14  947           B     DELAY5
                   15  948  *
                   16  949  *
8329 A00DBD        17  950  INIT     ORI   0B,SH,SH      INHIBIT INPUT
832A 284DFD        18  951           ANDI  1F,SH,SH      SET NO TRAP,40BIT,HALT/STEP OFF
832B D71D80 831D   19  952           SB    ENABLE5
832C A00E3F        20  953           MVI   03,K
832D 572580 8325   21  954           SB    OBSTROBE      CLEAR CRT
832E 87800F        22  955           SR    ,
                   23  956  *
                   24  957  *
832F D71D80 831D   25  958  CRLF     SB    ENABLE5       THIS ROUTINE PRINTS CARRIAGE RETURN LINE FEED
8330 200EDF        26  959           MVI   0D,K
8331 572580 8325   27  960           SB    OBSTROBE      CR
8332 A00EAF        28  961           MVI   0A,K
8333 572580 8325   29  962           SB    OBSTROBE      LF
8334 87800F        30  963           SR    ,
                   31  964  *
                   32  965  *
8335 D71D80 831D   33  966  SPACE    SB    ENABLE5
8336 208E0F        34  967           MVI   20,K
8337 572580 8325   35  968           SB    OBSTROBE      PRINT A SPACE
8338 87800F        36  969           SR    ,
                   37  970  *
                   38  971  *
8339 D71D80 831D   39  972  INITCRT  SB    ENABLE5
833A 200E1F        40  973           MVI   01,K
833B 572580 8325   41  974           SB    OBSTROBE      HOME CUSOR
833C A00E3F        42  975           MVI   03,K
833D 572580 8325   43  976           SB    OBSTROBE      CLR CRT DISPLAY
833E 87800F        44  977           SR    ,
                   45  978  *
```

```
                    46   979  *
833F D71D80  831D   47   980  TEST#    SB       ENABLE5
8340 214E4F         48   981           MVI      54,K
8341 572580  8325   49   982           SB       OBSTROBE        T
8342 210E5F         50   983           MVI      45,K
8343 572580  8325   51   984           SB       OBSTROBE        E
8344 A14E3F         52   985           MVI      53,K
8345 572580  8325   53   986           SB       OBSTROBE        S
8346 214E4F         54   987           MVI      54,K
8347 572580  8325   55   988           SB       OBSTROBE        T
8348 208E3F         56   989           MVI      23,K
8349 572580  8325   57   990           SB       OBSTROBE        #
834A D73580  8335   58   991           SB       SPACE           SPACE
834B 575D80  835D   59   992           SB       PRTBYTE         TEST NUMBER X IS PRINTED
834C 200F0F         60   993           NOP      ,
834D 572F80  832F   61   994           SB       CRLF
834E 87800F         62   995           SR       ,
                    63   996  *
                    64   997  *
834F D71D80  831D   65   998  ERROR    SB       ENABLE5
8350 210E5F         66   999           MVI      45,K
8351 572580  8325   67   1000          SB       OBSTROBE        E
8352 214E2F         68   1001          MVI      52,K
8353 572580  8325   69   1002          SB       OBSTROBE        R
8354 214E2F         70   1003          MVI      52,K
8355 572580  8325   71   1004          SB       OBSTROBE        R
8356 210EFF         72   1005          MVI      4F,K
8357 572580  8325   73   1006          SB       OBSTROBE        O
8358 214E2F         74   1007          MVI      52,K
8359 572580  8325   75   1008          SB       OBSTROBE        R
835A 572F80  832F   76   1009          SB       CRLF
835B 57BC80  83BC   77   1010          SB       SEC1            WAIT APP. ONE SECOND FOR DISPLAY PURPOSES
835C 87800F         78   1011          SR       ,
                    79   1012 *
835D D71D80  831D   80   1013 PRTBYTE  SB       ENABLE5
835E A00E00         81   1014          MV       F0,K            THIS IS A ROUTINE TO PRINT FILE REG 0
835F 572580  8325   82   1015          SB       OBSTROBE
8360 87800F         83   1016          SR       ,
                    84   1017 *
8361 21430F         85   1018 SYSPE    MVI      C'P',F3         THESE ROUTINES PRINT PARITY ERROR ADDR.
8362 21025F         86   1019 SYSVE    MVI      C'E',F2
8363 2100DF         87   1020          MVI      C'M',F0
8364 D71D80  831D   88   1021 SYSERROR SB       ENABLE5
8365 200E1F         89   1022          MVI      01,K
8366 572580  8325   90   1023          SB       OBSTROBE
```

```
8367 208E0F        91   1024           MVI       20,K
8368 A0040F        92   1025           MVI       00,F4
8369 572580  8325  93   1026 SPACEOUT  SB        OBSTROBE
836A 98C44F        94   1027           AC,,1     F4,,F4
836B FF6954  8369  95   1028           BNEH      5,F4,SPACEOUT
836C 200E1F        96   1029           MVI       01,K
836D 572580  8325  97   1030           SB        OBSTROBE
836E 208EAF        98   1031           MVI       C'*',K
836F 572580  8325  99   1032           SB        OBSTROBE
8370 572580  8325  100  1033           SB        OBSTROBE
8371 572580  8325  101  1034           SB        OBSTROBE
8372 D73580  8335  102  1035           SB        SPACE
8373 A14E3F        103  1036           MVI       C'S',K
8374 572580  8325  104  1037           SB        OBSTROBE
8375 A14E9F        105  1038           MVI       C'Y',K
8376 572580  8325  106  1039           SB        OBSTROBE
8377 A14E3F        107  1040           MVI       C'S',K
8378 572580  8325  108  1041           SB        OBSTROBE
8379 214E4F        109  1042           MVI       C'T',K
837A 572580  8325  110  1043           SB        OBSTROBE
837B 210E5F        111  1044           MVI       C'E',K
837C 572580  8325  112  1045           SB        OBSTROBE
837D A10EDF        113  1046           MVI       C'M',K
837E 572580  8325  114  1047           SB        OBSTROBE
837F D73580  8335  115  1048           SB        SPACE
8380 210E5F        116  1049           MVI       C'E',K
8381 572580  8325  117  1050           SB        OBSTROBE
8382 214E2F        118  1051           MVI       C'R',K
8383 572580  8325  119  1052           SB        OBSTROBE
8384 214E2F        120  1053           MVI       C'R',K
8385 572580  8325  121  1054           SB        OBSTROBE
8386 210EFF        122  1055           MVI       C'O',K
8387 572580  8325  123  1056           SB        OBSTROBE
8388 214E2F        124  1057           MVI       C'R',K
8389 572580  8325  125  1058           SB        OBSTROBE
838A D73580  8335  126  1059           SB        SPACE
838B A08E8F        127  1060           MVI       C'(',K
838C 572580  8325  128  1061           SB        OBSTROBE
838D A00E03        129  1062           MV        F3,K
838E 572580  8325  130  1063           SB        OBSTROBE
838F 200E02        131  1064           MV        F2,K
8390 572580  8325  132  1065           SB        OBSTROBE
8391 200E01        133  1066           MV        F1,K
8392 572580  8325  134  1067           SB        OBSTROBE
8393 A00E00        135  1068           MV        F0,K
```

```
8394 572580  8325  136  1069          SB      OBSTROBE
8395 D73580  8335  137  1070          SB      SPACE
8396 200009        138  1071          MV      PH,F0
8397 D7A680  83A6  139  1072          SB      PRTBITE
8398 A00008        140  1073          MV      PL,F0
8399 D7A680  83A6  141  1074          SB      PRTBITE
839A 208E9F        142  1075          MVI     C')',K
839B 572580  8325  143  1076          SB      OBSTROBE
839C D73580  8335  144  1077          SB      SPACE
839D 208EAF        145  1078          MVI     C'*',K
839E 572580  8325  146  1079          SB      OBSTROBE
839F 572580  8325  147  1080          SB      OBSTROBE
83A0 572580  8325  148  1081          SB      OBSTROBE
83A1 DF2F80  832F  149  1082          B       CRLF
                   150  1083 *
                   151  1084 *
83A2 4BA421  83A4  152  1085 PRTBIT1  BLER    F2,F1,*+2       THESE ROUTINES PRINT CHARS.STORED IN F0
83A3 AC0272        153  1086          AI      07,F2,F2
83A4 200E02        154  1087          MV      F2,K
83A5 DF2580  8325  155  1088          B       OBSTROBE
83A6 A0C19F        156  1089 PRTBITE  MVI     39,F1
83A7 20C30F        157  1090          MVI     30,F3
83A8 8C4203        158  1091          SHHH    F0,F3,F2
83A9 57A280  83A2  159  1092          SB      PRTBIT1
83AA 084203        160  1093          SHHL    F0,F3,F2
83AB DFA280  83A2  161  1094          B       PRTBIT1
                   162  1095 *
                   163  1096 *
83AC 8D800F        164  1097 PE24     TSP     ,               FAILING ADDR TO PC'S
83AD 800FFF        165  1098          OR      -,,             DECREMENT ADDR BY 1
83AE 21013F        166  1099          MVI     C'C',F1         SET UP F1 TO PRINT OUT "PECM"
83AF 576180  8361  167  1100          SB      SYSPE           PRINT OUT CONTROL MEM PARITY ERROR INFO
83B0 DFB080  83B0  168  1101          B       *               HANG HERE UNTIL RESET IS STRUCK
                   169  1102 *
                   170  1103 *
83B1 0208E0        171  1104 RETURN   MVX     F1F0,PHPL
83B2 AC0818        172  1105          AI      1,PL,PL
83B3 05800F        173  1106          TPS     ,
83B4 87800F        174  1107          SR      ,
                   175  1108 *
                   176  1109 *
83B5 A1014F        177  1110 PE8      MVI     C'D',F1         SET UP F1 TO PRINT OUT "PEDM"
83B6 576180  8361  178  1111          SB      SYSPE           PRINT OUT DATA MEMORY PARITY ERROR
83B7 A20D0D        179  1112          ORI     80,SH,SH        SET NO TRAP
83B8 2ACDFD        180  1113          ANDI    0BF,SH,SH       CLEAR DMPE (40 BIT)
```

```
83B9 991000  0000   181  1114          LPI,R    ZERO          SET UP GOOD PARITY
83BA 29CDFD         182  1115          ANDI     7F,SH,SH      CLEAR NO TRAP
83BB 5FBB80  83BB   183  1116          B        *             WAIT HERE FOR RESET TO BE STRUCK
                    184  1117 *
83BC 20000F         185  1118 SEC1     MVI      00,F0         CLEAR COUNT
83BD A00E0F         186  1119 RESTRT   MVI      00,K          CLEAR COUNT
83BE AC0E1E         187  1120 WAIT1    AI       1,K,K         STEP COUNT BY 1
83BF FBC19E  83C1   188  1121          BNEL     9,K,*+2
83C0 77C3CE  83C3   189  1122          BEQH     0C,K,*+3      CHECK FOR LOOP COUNT = 200 DECIMAL
83C1 57C880  83C8   190  1123          SB       DELAY50       DELAY 50 MICRO-SECONDS
83C2 5FBE80  83BE   191  1124          B        WAIT1         REPEAT LOOP
83C3 AC0010         192  1125 WAIT2    AI       1,F0,F0       STEP COUNT BY 1
83C4 7BC6B0  83C6   193  1126          BNEL     0B,F0,*+2
83C5 F7C710  83C7   194  1127          BEQH     1,F0,*+2      CHECK FOR LOOP COUNT = 25 DECIMAL
83C6 5FBD80  83BD   195  1128          B        RESTRT        EXECUTE WAIT1 LOOP AGAIN
83C7 87800F         196  1129          SR       ,             RETURN
83C8 571F80  831F   197  1130 DELAY50  SB       DELAY10
83C9 571F80  831F   198  1131          SB       DELAY10
83CA 571F80  831F   199  1132          SB       DELAY10
83CB 571F80  831F   200  1133          SB       DELAY10
83CC 571F80  831F   201  1134          SB       DELAY10
83CD 87800F         202  1135          SR       ,
                    203  1136 *
            0000    204  1137 ZERO     EQU      0000
83CE 992000  0000   205  1138 SETPAR   LPI,W1   ZERO          WRITE 00 AT LOCATION 0000
83CF 193000  0000   206  1139          LPI,W2   ZERO          WRITE 00 AT LOCATION 0001
83D0 991000  0000   207  1140          LPI,R    ZERO          READ LOCATIONS 0000/0001
83D1 A8CDFD         208  1141          ANDI     3F,SH,SH      SET NO TRAP AND 40 BITS OFF
83D2 87800F         209  1142          SR       ,             RETURN
                    210  1143 *
                    211  1144 *
                    212  1145 *END
```

NUMBER OF LINES IN ERROR = 0 NO. OF WARNINGS = 4 NO. OF SYMBOLS = 77 OVERFLOWS = 75

SYMBOL	VALUE	DEFN	REFERENCES
CRLF	832F	0958	0225 0994 1009 1082
DELAY10	831F	0936	1130 1131 1132 1133 1134
DELAY5	8320	0937	0936 0945 0947
DELAY50	83C8	1130	1123
ENABLE	831E	0935	0007
ENABLE5	831D	0934	0952 0958 0966 0972 0980 0998 1013 1021
ERROR	834F	0998	0267 0306 0349 0388 0435 0483 0531 0568 0627 0710 0734 0799 0864 0927
ERROR0	80E1	0267	0234 0235 0237 0238 0240 0241 0243 0244 0246 0247 0249 0250 0252 0253 0254 0256 0257 0259 0260 0262 0263
ERROR1	8103	0306	0277 0278 0280 0281 0283 0284 0286 0287 0289 0290 0292 0293 0295 0296 0298 0299 0301 0302
ERROR2	8125	0349	0320 0321 0323 0324 0326 0327 0329 0330 0332 0333 0335 0336 0338 0339 0341 0342 0344 0345
ERROR3	8147	0388	0359 0360 0362 0363 0365 0366 0368 0369 0371 0372 0374 0375 0377 0378 0380 0381 0383 0384
ERROR4	8171	0435	0400 0401 0403 0404 0406 0407 0409 0410 0412 0413 0415 0416 0418 0419 0421 0422 0426 0427 0430 0431
ERROR5	819C	0483	0446 0447 0450 0451 0454 0455 0458 0459 0462 0463 0466 0467 0470 0471 0474 0475 0478 0479
ERROR6	81C7	0531	0494 0495 0498 0499 0502 0503 0506 0507 0510 0511 0514 0515 0518 0519 0522 0523 0526 0527
ERROR7	81E7	0568	0543 0544 0547 0548 0551 0552 0555 0556 0559 0560 0563 0564 0586
ERROR8	821D	0627	0584 0589 0591 0594 0596 0599 0601 0604 0606 0614 0615 0617 0620 0622 0623
ERROR9	8263	0710	0636 0638 0640 0642 0644 0646 0656 0657 0659 0668 0669 0671 0676 0677 0679 0688 0689 0690 0694 0695 0697 0702 0703 0705
ERRORA	8276	0734	0722 0723 0725 0727 0730
ERRORB	82AE	0799	0756 0775 0776 0778 0779 0791 0792 0794 0795
ERRORC	82DF	0864	0810 0812 0814 0816 0820 0822 0824 0828 0830 0832 0834 0838 0840 0844 0846 0848 0850 0854 0856 0858
ERRORD	8319	0927	0910 0911 0912 0913
INCRF3	8313	0918	0916
INCRK	830A	0909	0903
INCRPC	8315	0920	0917
INCRPH	82FC	0895	0892
INCRPL	82F7	0890	0878
INIT	8329	0950	0070 0931
INITCRT	8339	0972	
INITIAL1	8041	0076	0081 0082 0087 0088
INITIAL2	8050	0094	0097 0098 0101 0102
INITIAL3	805B	0108	0113 0114 0118 0119
INITIAL4	8069	0125	0130 0131 0135 0136
INITIAL5	8077	0142	0147 0148 0151 0152
INITIAL6	8084	0158	0163 0166
INITIAL7	808F	0172	0177 0180 0181 0184 0185 0188
INITIAL8	80A2	0194	0199 0200 0204 0205

SYMBOL	VALUE	DEFN	REFERENCES
INITIAL9	80B0	0211	0216 0217 0221 0222
LOOPADD	82EB	0878	0894 0898
LOOPB	827E	0745	0750
LOOPBR	8287	0754	0760
OBSTROBE	8325	0944	0072 0090 0104 0121 0138 0154 0168 0190 0207 0224 0954 0960 0962 0968 0974 0976 0982 0984 0986 0988 0990 1000 1002 1004 1006 1008 1015 1023 1026 1030 1032 1033 1034 1037 1039 1041 1043 1045 1047 1050 1052 1054 1056 1058 1061 1063 1065 1067 1069 1076 1079 1080 1081 1088
PE24	83AC	1097	0002
PE8	83B5	1110	0004
PRTBIT1	83A2	1085	1092 1094
PRTBITE	83A6	1089	1072 1074
PRTBYTE	835D	1013	0992
READCM	8304	0903	0922
RESTART	831B	0931	0919 0926
RESTRT	83BD	1119	1128
RETURN	83B1	1104	
SEC1	83BC	1118	1010
SETPAR	83CE	1138	0739
SPACE	8335	0966	0991 1035 1048 1059 1070 1077
SPACEOUT	8369	1026	1028
START	8010	0021	0003 0009 0016 0025 0027 0029 0031 0033 0035 0037 0039
START1	8023	0043	0044 0046 0048 0050 0051 0053 0055 0057 0060 0063 0065 0066 0068
SYSERROR	8364	1021	
SYSPE	8361	1018	1100 1111
SYSVE	8362	1019	
TEST#	833F	0980	0232 0274 0317 0356 0395 0442 0490 0538 0575 0634 0717 0741 0804 0870
TEST0	80BF	0231	0269 0932
TEST1	80E4	0273	0264 0308
TEST2	8106	0316	0303 0351
TEST3	8128	0355	0346 0390
TEST4	814A	0394	0385 0437
TEST5	8174	0441	0432 0485
TEST6	819F	0489	0480 0533
TEST7	81CA	0537	0528 0570
TEST8	81EA	0574	0565 0628
TEST9	821F	0633	0624 0712
TESTA	8266	0716	0707 0735
TESTB	8278	0739	0731 0800
TESTC	82B0	0803	0796 0866
TESTD	82E2	0869	0861 0928
WAIT1	83BE	1120	1124
WAIT2	83C3	1125	
ZERO	0000	1137	1114 1138 1139 1140