Fermilab

TM-1079

0818.000

A MICROPROCESSOR BASED MADC TEST FACILITY

K. Seino

December 1981

1. Introduction

A new multiplexed ADC was designed and constructed back in 1979[1]. However, because of budgetary constraints and manpower shortage, we had not produced any beyond five units. Recently, we constructed fifteen units, and we will build twenty-five or forty-five more units.

Each units has 192 connections at the input. If there are some faulty connections, how can we detect and locate them quickly and efficiently? How can we adjust the offset and gain on the ADC fast and without frustration? These questions motivated the author to implement a microprocessor based test facility. A graphic display greatly eases pains in adjusting the offset and gain of the ADC.

The author thinks this is only the first generation, and he intends to try and improve it through experience.

2. Test Method

2.1 Basic Check

The MADC units are assembled by an outside subcontractor. When the units are delivered, the following basic functions are checked out.

1. Check and adjustment on power supply voltages.
2. Check and troubleshoot on controls and displays.
3. Check on one of four MUX cards.
4. Check and adjustment on amplifiers and sample/hold.
5. Rough adjustment on ADC.

After the basic functional check, the units can be locally and remotely controlled, and they can digitize and display analog signals at least on a few channels with a moderate accuracy.

2.2 Basic Tests

Prior to computer aided tests, we document several characteristics, i.e., (1) rough linearity, (2) DC common mode rejection, (3) AC common mode rejection, and (4) channel cross talk.

The amplifier has been properly adjusted and it performs well with reasonably accurate offset, gain and common mode rejection. The sample/hold and the ADC have been adequately adjusted. The unit as a whole appears to be functioning and it produces reasonable readings. We are now ready to do computer aided tests.

2.3 Computer Aided Tests

The first thing that comes across our mind is that there are many physical connections through the input arrangement, i.e., connectors, cables and MUX cards. There are sixty-four differential channels each of which has Hi, Lo, and shield, and therefore we deal with at least 192 connections. If there is something wrong with any one of these connections, how can we detect and locate it quickly and efficiently?

Another thing is how can we adjust the offset and gain on the ADC?, and how can we know whether the ADC performs as it is specified or not? In order to answer these questions, the author introduced a graphics display. When the operator adjusts the offset and gain, he can watch and tell how much the linearity curve changes as he turns the potentiometer. When he finishes with the adjustment, he can watch and tell how good the linearity is on the graphic display.

2.3.1 Channels and their interaction tests

A channel should be properly selected and isolated from other channels. A signal should come through the channel and it should be accurately digitized. On our test facility, we implemented two test programs, i.e., MINTS1 and MINTS2.

Fig. 2.1 shows how to run the MINTS1 program on a CRT terminal. First, the operator makes certain whether Source B on an input switching unit has normal polarity or not. He then specifies a channel group and a reference value. The input switching unit connects only one channel to the

reference and ties the other channels to ground during a cycle. It switches from one channel to another as the cycle advances from one to another. As it is shown in the figure, Channel 0 is selected in Cycle 1, Channel 1 in Cycle 2, Channel 2 in Cycle 3, etc...., and it continues until Channel 15 is selected in Cycle 16. In this test, only the selected channel should have the voltage. If unselected channels have some voltage, there may be miswiring or short circuits. Or, if the selected channel does not have the voltage, there may be open wires or ground short circuits. If the operator selects the other channel groups, he can test all of the sixty-four channels.

Figure 2.2 shows how to run the MINTS2 program on a CRT terminal. Differing from the previous program, this program allows the operator to specify the channel pattern. In the figure, an alternate pattern (AAAA in Hex) is selected. By manipulating the channel pattern, the operator can further diagnose the problem that has been found in the previous test.

2.3.2 Input protection test

The input circuit of all of the channels has to be protected from accidental application of a high voltage. Each one of the connections that are needed for the sixty-four differential channels is protected by two diodes and a series resistors. For the test, we use a mechanical switch box and a power supply instead of the programmable input switching unit and the DAC. Figure 2.3 shows how to run the MINTS2 program

for this particular test. The operator specifies the channel group, and he can put an arbitrary number on the channel pattern and reference, such as FFFF or XXXX. In the figure, each pair of the input wires of the sixteen channels are connected to a common voltage (Ecm in Figure 2.3). When E_{cm}=+10V, the amplifier is in a linear range and it can reject the voltage that is commonly applied to the Hi and Lo wires of the channels. Therefore, all the readings are nearly zero. When E_{cm} is increased to +15V, the amplifier is in saturation and the input to the multiplexer is approaching the supply voltage. All the readings shown resulted from the imbalance between Hi and Lo connections. When E_{cm} is increased to +17V, protection diodes start to conduct. We repeat these with E_{cm} in negative voltages. If all the readings that have been taken are reasonably small, we deduce that all the protection circuits function properly.

2.3.3 Linearity tests

A graphic display greatly eases pains in adjusting the offset and gain of the ADC and allows us to examine the entire linearity curve at a glance.

(1) Linearity (Regular)

The operator invokes a command file called MLNTS2 on his terminal. He then enters type of linearity, channel number and delay. He selects 'Linearity (Regular)' by entering RLA. In a second or so, the graphic display will have a picture like the ones shown in Fig. 2.4. The horizontal axis shows

the reference voltage in a linear scale, and the vertical axis shows the different between reference and ADC reading. There are twenty-one data points starting from -10V up to +10V. Figure 2.4(a) shows a curve when gain is too high. A half turn adjustment on the gain potentiometer gives us a curve like the one shown in Fig. 2.4(b). If the operator selects repeat mode, the program repeats digitization and updating of display. Therefore, he can adjust the gain potentiometer as he watches how much the linearity curve is affected. He can do the same thing with the offset adjustment. With this graphic display, we now feel that we have a search light in the dark.

(2) Linearity (Differential)

After invoking a command file called MLNTS2, the operator selects 'Linearity (Diff.)' by entering DFL. In a second or so, the graphic display will have a picture like the one shown in Fig. 2.5. The horizontal axis shows the reference voltage in a binary scale, and the vertical axis shows the difference between reference and ADC reading. There are twenty-four data points, i.e., -10.24, -5.12, -2.56,----0----, 2.555, 5.115 and 10.235 (V), or 100000000000, 110000000000, 111000000000, ----------, 000000000000, --------, 000111111111, 001111111111 and 011111111111 (Binary).

3. Hardware

3.1 Microcomputer and Peripherals

We purchased card cages, chassis parts, a power supply and cards in order to construct a Multibus-based microcomputer shown in Fig. 3.1. Monolithic Systems' MSC8004 Z80A based CPU board is the heart of the computer. There ae 32K byte on-board RAM and 32K byte off-board RAM. The Z80A communicates with TEC501 terminal via on-board serial I/0 interface. A home made disk control board handles two double sided disk drives, Shugart Associates' SA850. Matrox Electronic Systems' MSBC-512 graphic control board, which has a resolution of 256x256, drives Shibaden's VM904U video monitor. Kinetic Systems' 5110 interface board drives Kinetic Systems' 3908 crate controller.

3.2 MADC Test Setup

A CAMAC crate based test setup is shown in Fig. 3.2. Analog Devices' DAC1136K high resolution 16 bit DAC is housed in a chassis. It is optically isolated from the digital environment at the input, and it is buffered by Analog Devices' 234K chopper stabilized amplifier at the output. The chassis has its own power supply, which can be left on for stable output. The input switching unit has 16 solid state, single pole-double throw swiches, which connect the MADC channels to either Source A or Source B. Source B can be selected to either Norm or B=-A. When it is selected to B=-A, Source B will have the reversed polarity of and the same amplitude as Source A. The unit is also optically

isolated from its controller and is self-powering. This unit is replaced with a mechanical switch box when the input protection test is carried out. All of the controller cards that reside in the CAMAC crate are home made, except the crate controller.

4. Software

4.1 System Software

CDOS (Cromemco Disk Operating System) is the disk operating system. It is a single-user, single-task operating system for disk file management. CDOS occupies memory from Location 0 through 100H as well as approximately 8K bytes of memory above the user area (High Memory). CDOSGEN allows CDOS to be built around the user's particular hardware configuration and software needs. It supports up to 64K bytes of memory in 1K blocks and any combination of up to four 8" and 5" disk drives. CDOS utility programs are (1) BATCH, (2) DUMP, (3) INITIALIZE, (4) STATUS, (5) WRITE SYSTEM, (6) TRANSFER, (7) SCREEN EDITOR, and (8) TEXT EDITOR.

FORTRAN IV is a complete implementation of ANSI standard FORTRAN X3.9--1966, except that there is no complex data type and that specification statements must appear in a specific order. The FORTRAN IV Compiler produces relocatable code and FORTRAN program modules can be linked with the code produced by the relocatable assembler.

Z80 MACRO Assembler is a two pass assembler which reads source code from a disk file, assembles it, and produces listings and an object file either in relocatable or in Intel hex format.

LINK, relocating linker/loader, loads and links separately assembled modules as desired at run time. These modules may include FORTRAN library routines as well as those generated by the FORTRAN compiler. The linked machine code can then be saved in a disk file for execution.

DEBUG allows machine language programs to be traced, disassembled and patched, and it allows the operator to establish break points, display and alter the Z80 registers, and initiate normal or step by step program execution.

4.2 Program Example

All of our test programs are written in FORTRAN IV. MINTS2.FOR is shown as a program example in Fig. 4.1. There are many subprogram calls in this program. However, in order to illustrate how parameters are passed between the main program and a subprogram, a subprogram named C126DA is shown in Fig. 4.1(b). The subprogram call is made at one line below Line 110 of the main program. The subprogram has two parameters, i.e., ICDI and ICDO. ICDI and ICDO are passed by (HL) and (DE) respectively, between the main program and the subprogram.

At the end of the program, the operator is allowed to intervene and select a mode. He has four choices, i.e., (1) C (Continue) repeats another cycle with the same parameters, (2) N (New) starts another cycle with new parameters, (3) R (Repeat) repeats the number of cycles specified, and (4) S (Stop) stops the execution of program. The author has found this feature is convenient and useful, and he is in the habit of using this

feature in any program that he writes.

5. Conclusion

Computer aided tests on the MADC units have just begun, and our present implementation is far from being complete. We intend to try and improve it by learning from experience. One thing that we know right from the start is that we have an input switching unit for 16 channels but not for 64 channels. We ought to construct one soon.

6. Acknowledgement

I am grateful to Timothy Gierhart and Richard Klecka for their efforts on the construction of the test facility. Thanks are also due Walter Knopf for his contribution of CDOS and utility programs to the test facility.

References

1. A New Multiplexed ADC Unit, K. Seino, TM-931, 1979.

2. MSC8004 User's Manual, Monolithic Systems Corp., 1978.

3. 32K-128K Dynamic RAM Board Manual, Central Data Corp., 1980

4. MSBC-512 Graphic Display Board Manual, Matrox Electronics, 1978.

5. Model 5110 Multibus Adapter Manual, Kinetic Systems Inc., 1980.

6. Model 3908 Crate Controller Manual, Kinetic Systems Inc., 1979.

7. SA850/851 OEM Manual, Shugart Associates, 1977.

8. Series 500 Terminal Manual, TEC Inc., 1978.

9. CDOS Instruction Manual, Cromemco, Inc., 1978.

10. FORTRAN IV Instruction Manual, Cromemco, Inc., 1979.

11. Macro Assembler Instruction Manual, Cromemco, Inc., 1978.

```
XINTS 1
ON INPUT SWITCH UNIT, SOURCE B = NORM ? (Y/N) :Y
SELECT CHANNEL GROUP 1 2 3 4 :1
ENTER REFERENCE RANGING FROM -2048 TO 2047 :2.0.0 0

CYCLE =  1        CHANNEL GROUP = 1
     0         1         2         3         4         5         6         7
    10.00     0.00      0.00      0.00      1.00      0.00      0.00      0.00
     0.00     0.00      0.00      0.00      0.00      0.00      0.00      0.00
     8         9        10        11        12        13        14        15

CYCLE =  2        CHANNEL GROUP = 1
     0         1         2         3         4         5         6         7
     0.00     9.99      0.00      0.00      0.00      0.00      0.00      0.00
     0.00     0.00      0.00      0.00      0.00      0.00      0.00     -.01
     8         9        10        11        12        13        14        15

CYCLE =  3        CHANNEL GROUP = 1
     0         1         2         3         4         5         6         7
     0.00      0.00     9.99       .00      0.00      0.00      0.00      0.00
     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
     8         9        10        11        12        13        14        15

CYCLE =  4        CHANNEL GROUP = 1
     0         1         2         3         4         5         6         7
     0.00      0.00      0.00     9.99       .00      0.00      0.00      0.00
     0.00     -.01      -.00      -.00      -.00      -.00      -.00      -.00
     8         9        10        11        12        13        14        15

CYCLE =  5        CHANNEL GROUP = 1
     0         1         2         3         4         5         6         7
     0.00      0.00      0.00      0.00     9.99      0.00      0.00      0.00
     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
     8         9        10        11        12        13        14        15

CYCLE =  6        CHANNEL GROUP = 1
     0         1         2         3         4         5         6         7
     0.00      0.00      0.00      0.00      0.00     9.99      0.00      0.00
     0.00      0.00      0.00      0.00      0.00      0.00      0.00     - 01
     8         9        10        11        12        13        14        15

CYCLE =  7        CHANNEL GROUP = 1
     0         1         2         3         4         5         6         7
     - 00      0 00      0.00      0.00      0.00      0.00     9.99      0.00
     0 00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
     8         9        10        11        12        13        14        15

CYCLE =  8        CHANNEL GROUP = 1
     0         1         2         3         4         5         6         7
     0.00      0.00      0.00      0.00      0.00      0.00      0.00     9.99
     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
     8         9        10        11        12        13        14        15
```

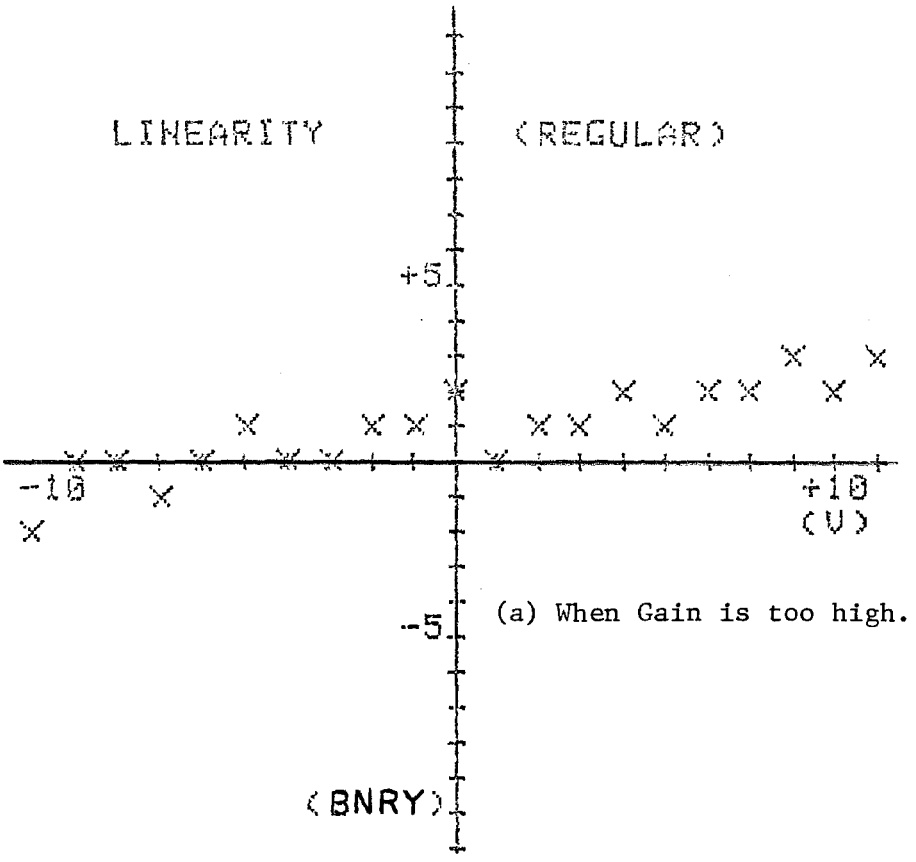
FIG. 2.1 Channels and Their Interaction Test No. 1

```
M I N T E 2

ENTER CHANNEL PATTERN IN HEX : A A A A
SELECT CHANNEL GROUP 1 2 3 4 : 1

ENTER REFERENCE -2048 THRU 2047 : 2 0 0 0



CHANNEL GROUP : 1

     0        1        2        3        4        5        6        7
-9.980    9.910   -9.980    9.920   -9.985    9.815   -9.985    9.920
-9.680    9.900   -9.985    9.910   -9.985    9.910   -9.985    9.920
     8        9       10       11       12       13       14       15
```

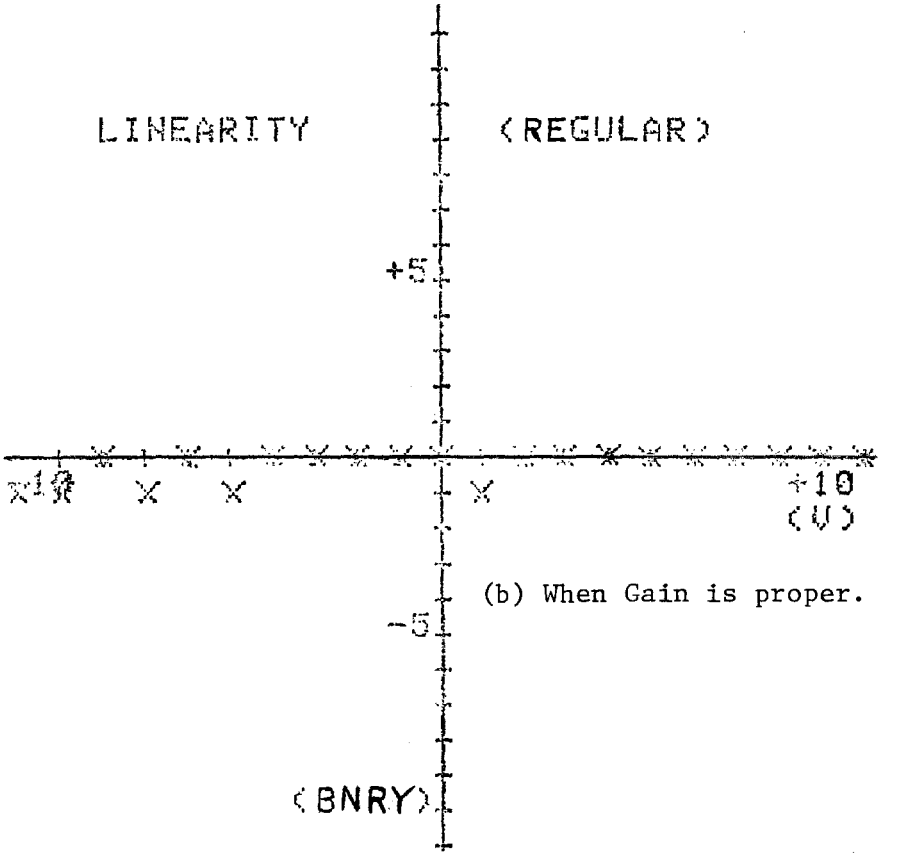
FIG. 2.2 Channels and Their Interaction Test No. 2

```
     M I N.T S.2.

ENTER CHANNEL PATTERN IN HEX .F F F F.
SELECT CHANNEL GROUP 1 2 3 4 :1

ENTER REFERENCE -2048 THRU 2047  X X X X.



CHANNEL GROUP : 1                                   E  =+10V
                                                     cm

    0       1       2       3       4       5       6       7
 -.005   -.005   -.005   -.005   -.005   -.005   0.000   -.005
 -.005   -.005   0.000   -.005   -.005   -.005   0.000   0.000
    8       9      10      11      12      13      14      15

SELECT MODE C N R S :C



CHANNEL GROUP : 1                                   E  =+15V
                                                     cm

    0       1       2       3       4       5       6       7
 .095    .105    .110    .110    .105    .105    .105    .105
 .105    .100    .105    .105    .100    .105    .105    .105
    8       9      10      11      12      13      14      15

SELECT MODE C N R S  C



CHANNEL GROUP   1                                   E  =+17V
                                                     cm

    0       1       2       3       4       5       6       7
 -.035   -.025   -.030   -.030   -.030   -.025   -.030   -.025
 -.030   -.030   -.030   -.030   -.030   -.030   -.030   -.030
    8       9      10      11      12      13      14      15
```

FIG. 2.3 Input Protection Test



(a) When Gain is too high.



(b) When Gain is proper.

FIG. 2.4 Graphic Display of Linearity (Regular)

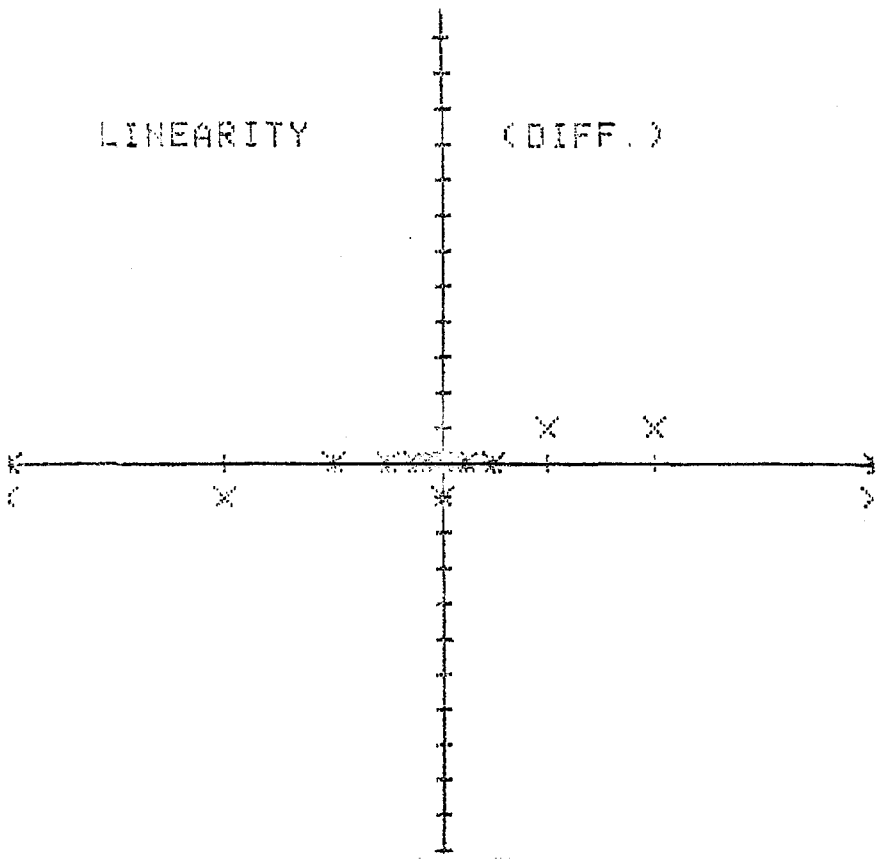


FIG. 2.5 Graphic Display of Linearity (Differential)

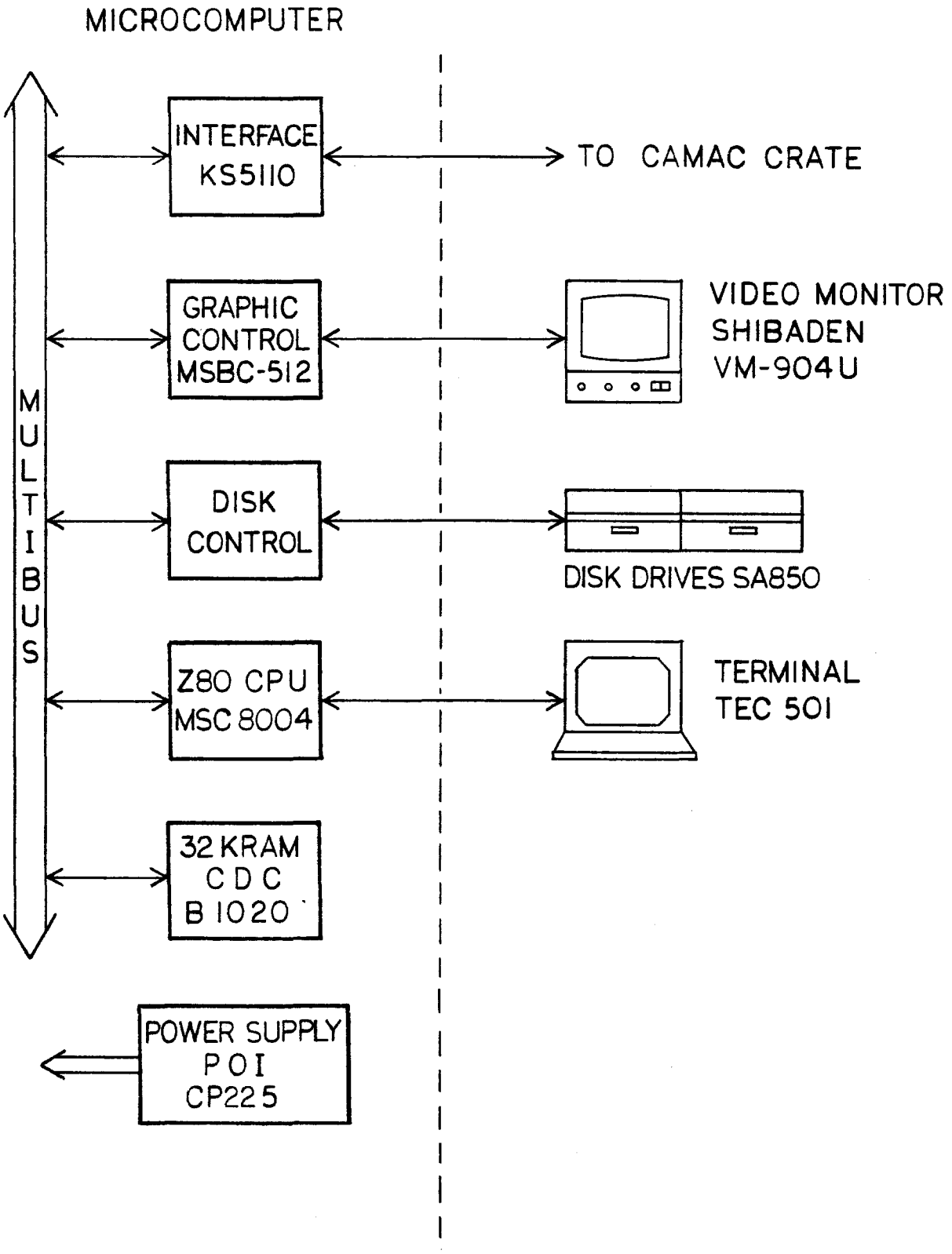


FIG. 3.1 MICROCOMPUTER & PERIPHERALS

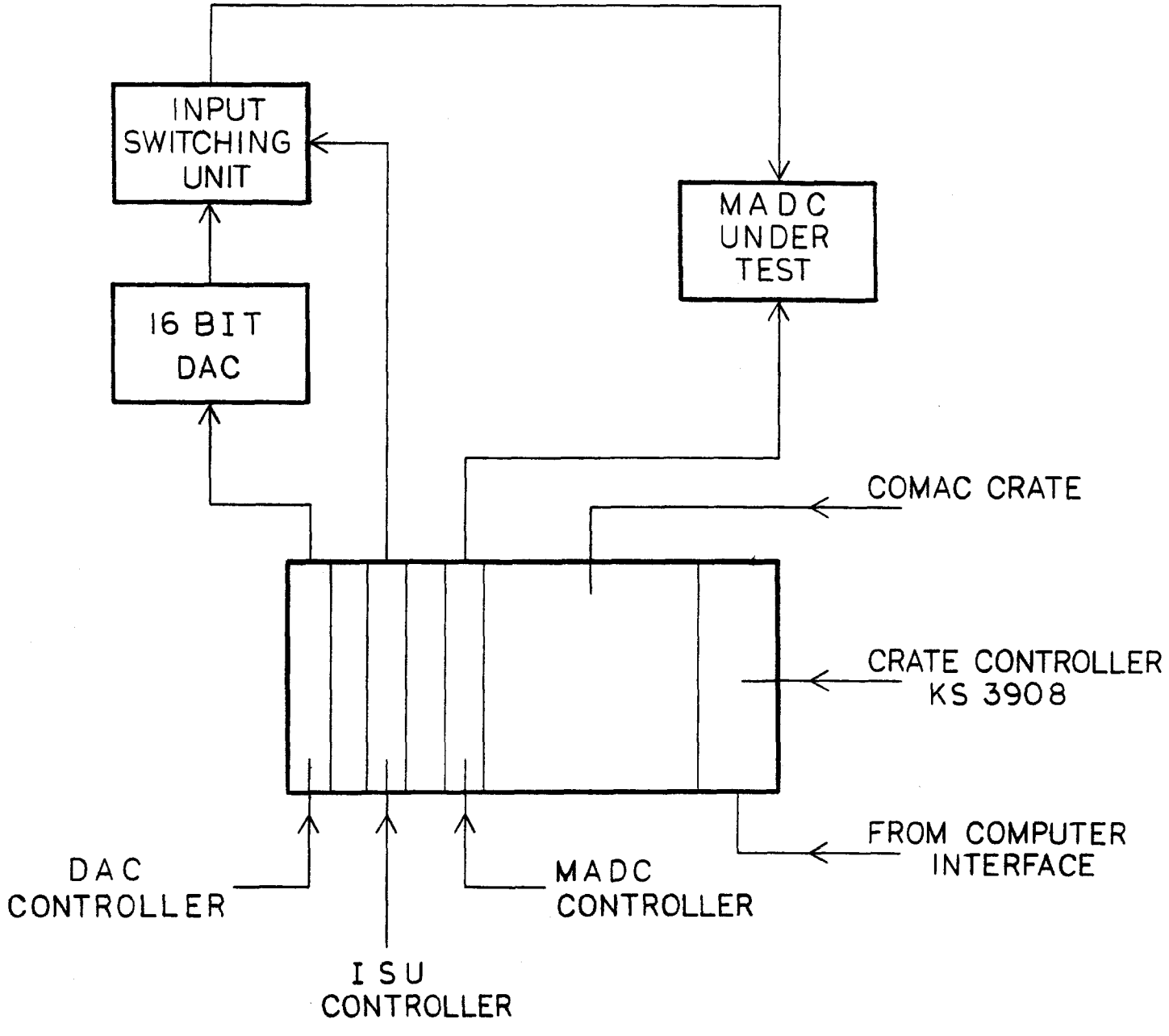


FIG. 3.2 MADC TEST SETUP

```
C
C  MADC INPUT TEST NO.2
C                                K. SEINO, 9/11/81
C ---------------------------------------------------------------
C  SPECIFY CHANNEL PATTERN, CHANNEL GROUP AND DAC REFERENCE VALUE.
C  THE PROGRAM DIGITIZES AND DISPLAYS A GROUP OF 16 CHANNELS WITH
C  THE SPECIFIED PARAMETERS.
C ---------------------------------------------------------------
C
      PROGRAM MINTS2
      DIMENSION ICHW(2),IDAREF(2),ICHPAT(2),CHVOL(16),IADR(2)
      DIMENSION ICIDA(8),ICIDB(8)
      REAL*8 CYCL
      REAL CHVOL
      INTEGER C,N,R,S
      DATA C,N,R,S/'C','N','R','S'/
      DATA ICIDA/0,1,2,3,4,5,6,7/
      DATA ICIDB/8,9,10,11,12,13,14,15/
C
C  INPUT OPERATION PARAMETERS
C
30    LMD=N
      WRITE(5,40)
40    FORMAT(2X,'ENTER CHANNEL PATTERN IN HEX :')
      CALL HEXIN(IHXI)
      ICHPAT(2)=IHXI
      WRITE(5,50)
50    FORMAT(2X,'SELECT CHANNEL GROUP 1 2 3 4 :')
      READ(5,60) ICHGRP
60    FORMAT(I1)
      WRITE(5,70)
70    FORMAT(2X,'ENTER REFERENCE -2048 THRU 2047 :')
      READ(5,80) IREF
80    FORMAT(I5)
C  FORM CAMAC ADDRESSES
90    CALL CDREG(NADR1,0,0,1,0)
      CALL CDREG(NADR2,0,0,5,0)
      CALL CDREG(NADR3,0,0,3,0)
C  CONVERT REFERENCE FROM 12 BIT TO 16 BIT USE
110   ICDI=IREF
      CALL C126DA(ICDI,ICDO)
      IDAREF(2)=ICDO
C  SEND REFERENCE TO DAC
120   CALL CFSA(16,NADR1,IDAREF,QRSP)
C  SEND CHANNEL PATTERN TO INPUT SWITCHING UNIT
130   CALL CFSA(16,NADR3,ICHPAT,QRSP)
C  DELAY 1 SEC
140   CALL DLYB(1000)
      GO TO 180
C  INPUT NUMBER OF CYCLES
150   WRITE(5,160)
160   FORMAT(2X,'ENTER NUMBER OF CYCLES :')
      READ(5,170) CYCL
170   FORMAT(F16.0)
C  CONTINUE OR REPEAT FROM THIS POINT
180   IF(ICHGRP.EQ.1) ICHW(2)=Z'80'
      IF(ICHGRP.EQ.2) ICHW(2)=16+Z'80'
      IF(ICHGRP.EQ.3) ICHW(2)=32+Z'80'
      IF(ICHGRP.EQ.4) ICHW(2)=48+Z'80'
      DO 190 I=1,16
      CALL CFSA(16,NADR2,ICHW,QRSP)
C  DELAY 30 MICRO SEC
      CALL DLYA(3)
C  GET DATA FROM ADC
      CALL CFSA(0,NADR2,IADR,QRSP)
      CHVOL(I)=IADR(2)
C  CONVERT IT FROM INTEGER TO REAL AND THEN DIVID
      CHVOL(I)=CHVOL(I)/200
      ICHW(2)=ICHW(2)+1
190   CONTINUE
C
C  PRINTS RESULTS
C
200   CALL SERAS
      CALL SLFIX(4)
      WRITE(5,210) ICHGRP
210   FORMAT(2X,'CHANNEL GROUP : ',I1)
      CALL SLFIX(1)
      WRITE(5,220) ICIDA
220   FORMAT(6X,8(I1,7X))
      WRITE(5,230) CHVOL
230   FORMAT(2X,8(F8.3)/2X,8(F8.3))
      WRITE(5,240) ICIDB
240   FORMAT(6X,2(I1,7X),6(I2,6X))
      CALL SLFIX(2)
C  DELAY 2 SEC
      CALL DLYB(2000)
C  CHECK MODE
      IF(LMD.NE.R) GO TO 250
      CYCL=CYCL-1
      IF(CYCL.EQ.0) GO TO 250
      GO TO 180
C
C  SET MODE: C=CONTINUE, N=NEW, R=REPEAT, S=STOP
C
250   WRITE(5,260)
260   FORMAT(2X,'SELECT MODE C N R S :')
      READ(5,270) LMD
270   FORMAT(A1)
      IF(LMD.EQ.C) GO TO 180
      IF(LMD.EQ.N) GO TO 30
      IF(LMD.EQ.R) GO TO 150
      IF(LMD.EQ.S) GO TO 280
      GO TO 250
280   STOP
      END
```

FIG. 4.1 (a) Program Example - Main Program

```
;
;  ROUTINE TO CONVERT 12 BIT DATA FOR 16 BIT DAC USE
;
;       FORMAT: C126DA(ICDI,ICDO)
;
;               ICDI            (HL)
;               ICDO            (DE)
        ENTRY   C126DA
;
C126DA: PUSH    AF
        PUSH    BC
        LD      A,(HL)                  ;PUT IT IN A AND B
        INC     HL                      ;FROM (HL)
        LD      B,(HL)
        SLA     A                       ;SHIFT 4 TIMES
        RL      B
        SLA     A
        RL      B
        SLA     A
        RL      B
        SLA     A
        RL      B
        LD      (DE),A                  ;PUT RESULT IN (DE)
        LD      A,B
        INC     DE
        LD      (DE),A
        POP     BC
        POP     AF
        RET
```

FIG. 4.1 (b) Program Example - Subprogram