ornl

ORNL/TM-7363

OAK RIDGE NATIONAL LABORATORY

UNION CARBIDE

A CAMAC Modular Programmable Function Generator

G. W. Turner
S. Suehiro
R. W. Hendricks

OPERATED BY
UNION CARBIDE CORPORATION
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

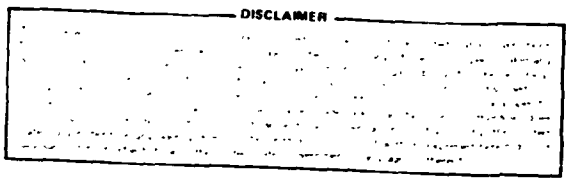
ORNL/TM-7363

Contract No. W-7405-eng-26

Metals and Ceramics Division

A CAMAC MODULAR PROGRAMMABLE FUNCTION GENERATOR

G. W. Turner, S. Suehiro, and R. W. Hendricks



Research sponsored by
the Division of Materials Science
of the National Science Foundation
under grants to the University of Massachusetts
and to the Oak Ridge National Laboratory

and

by the Basic Energy Sciences Program
of the U.S. Department of Energy

Date Published: December 1980

OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37830
operated by
UNION CARBIDE CORPORATION
for the
DEPARTMENT OF ENERGY

CONTENTS

	Page
ABSTRACT	1
INTRODUCTION	1
DESCRIPTION OF FUNCTION GENERATOR	2
Overview	2
Memory and Program Counter	5
Digital-to-Analog Converter	6
Programmable Frequency Clock	7
Programmable Synchronization Registers	8
LAM Control Logic	9
SOFTWARE	10
ACKNOWLEDGMENTS	11
REFERENCES	13
APPENDIX A	15
Circuit Diagrams	17
APPENDIX B	25
Manufacturer's Specifications	27
APPENDIX C	31
Program FGEN Source Listing	33

A CAMAC MODULAR PROGRAMMABLE FUNCTION GENERATOR

G. W. Turner,* S. Suehiro,† and R. W. Hendricks

ABSTRACT

A CAMAC modular programmable function generator has been developed. The device contains a 1024 word by 12-bit memory, a 12-bit digital-to-analog converter with a 600 ns settling time, an 18-bit programmable frequency register, and two programmable trigger output registers. The trigger registers can produce programmed output logic transitions at various (binary) points in the output function curve, and are used to synchronize various other data acquisition devices with the function curve.

INTRODUCTION

In our laboratory small-angle x-ray and neutron scattering experiments are performed on instruments which are operated with pinhole collimation and which are equipped with two-dimensional position sensitive detectors.[1-3] It has been found that the data acquisition rates with these instruments are sufficiently high that dynamic experiments performed during crystallization of polymers in situ in the x-ray beam are feasible.[4] In addition, the high speed and the time-switching capabilities of the data acquisition system available on these instruments[1] lead one of us (RWH) to propose that dynamic experiments performed in situ during the mechanical deformation of polymeric and biological samples could be performed.[1,5,6] For such experiments to be successful, it is necessary to: (1) deform the sample under programmably variable stress-strain relationships, (2) record the stress-strain during the deformation, (3) synchronize the acquisition

*Instrumentation and Controls Division.
†On assignment to Oak Ridge National Laboratory from the Polymer Research Institute, University of Massachusetts, Amherst, MA 01003; permanent address: Department of Polymer Chemistry, Faculty of Engineering, Kyoto University, Kyoto 606, Japan.

of the small-angle scattering diffraction pattern with the deformation, and (4) switch between acquisition arrays in memory at known and programmable times in the deformation cycle.

The availability of a fast-response CAMAC crate controller[7] for the ModComp computers used with our small-angle scattering cameras made a CAMAC data acquisition system a logical choice for the complex control system required by this problem. With the exception of a programmable function generator capable of providing the necessary timing and synchronizing signals, all of the necessary CAMAC modules were commercially available. It is the purpose of this report to describe our development of such a CAMAC programmable function generator.

DESCRIPTION OF FUNCTION GENERATOR

Overview

A block diagram of the programmable function generator is shown in DWG Q-5747-1* and photographs of the completed module are shown in Fig. 1. The CAMAC functions to which it responds are given in Table 1. As may be seen in the block diagram, the module has been designed to store and independently execute a program (CAMAC loaded and controlled) to drive the digital-to-analog converter (DAC) which generates a -10 V to +10 V analog signal. The module has a 1K × 12-bit random access memory (RAM) and an independent program counter (PC) which are used to store and execute the program. An on-board 1 MHz clock is used to shift the stored memory contents (analog waveform) out to the 12-bit DAC. The clock frequency may be divided by a factor which ranges from 1 to 2^{18} and which may be loaded from the dataway. In addition, there are two independent synchronization signals, each of whose frequency may also be loaded from the dataway into separate registers. Logic has also been provided to control and select various LAM (interrupt) signals. A variety of status and control LED indicators have been provided on the front panel as may be seen in Fig. 1(a).

---

* The schematic drawings for the function generator are found as foldouts in Appendix A.

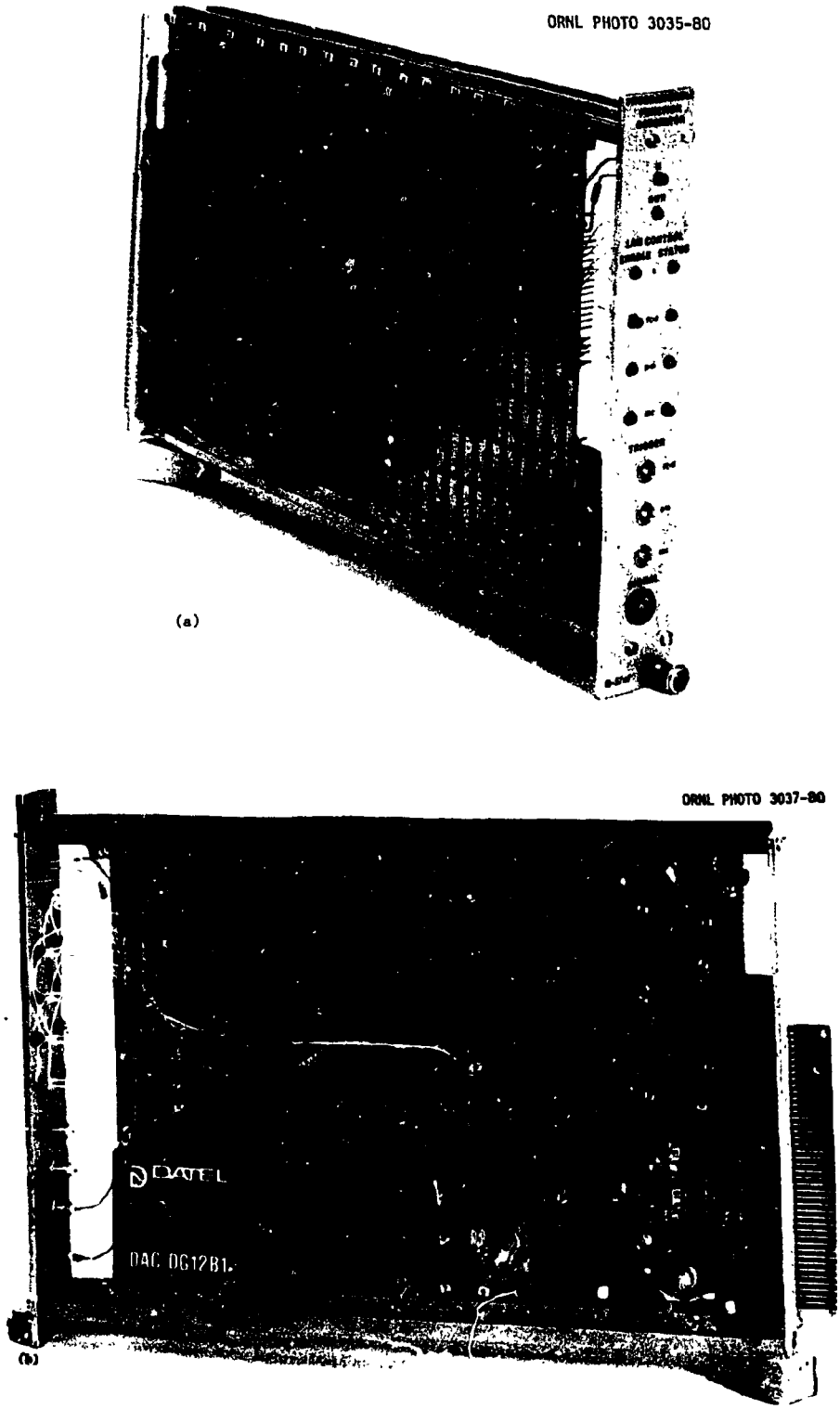


Fig. 1. The ORNL Q-5747 programmable function generator. (a) Front view, (b) wirewrap side.

Table 1. Programmable Function Generator CAMAC Commands

Function	Operation
F(0)A(0)	Read memory; advance memory address; Q = 1 on address overflow
F(16)A(0)	Write memory; advance memory address; Q = 1 on address overflow
F(17)A(0)	Load clock frequency register (W1-W18)
F(17)A(1)	Load trigger registers; (W1-W8 for C = M register; W9-W12 for ADC register)
F(10)A(1)	Clear ADC (C = T) LAM
F(10)A(2)	Clear PCOVFL (PC = 0) LAM
F(10)A(3)	Clear C = M LAM
F(11)A(0)	Module zero; clear PC; disable clock; disable all LAMs; clear all LAMs; clear all registers and latches.
F(24)A(0)	Disable LAM
F(24)A(1)	Disable ADC trigger (C = T) LAM
F(24)A(2)	Disable PCOVFL (PC = 0) LAM
F(24)A(3)	Disable C = M LAM
F(24)A(4)	Disable clock
F(25)A(0)	Execute; start function generator
F(25)A(1)	Advance PC
F(26)A(0)	Enable LAM
F(26)A(1)	Enable ADC trigger (C = T) LAM
F(26)A(2)	Enable PCOVFL (PC = 0) LAM
F(26)A(3)	Enable C = M LAM
Z	Same as F(11)A(0)
C	Same as F(11)A(0)

The various sections and their operation are described in the following paragraphs. The entire module has been constructed on a Standard Engineering Corp. WW-3 prototype wirewrap card. Low power Schottky (LS) TTL logic has been used throughout. (An exception is the CD4001 CMOS chip used in the clock circuit.) This was especially important because with standard TTL the high power consumption of several chips (notably the 'LS193 counters) made the power consumption of the module exceed the CAMAC specification of 2A on the +6 V power supply.

Memory and Program Counter

The Memory is a block of 1024 × 12 bits constructed from twelve Advanced Micro Devices AM9102 random access memory chips* (11A-13B) The AM9102 chips are 1024 bit static N-channel MOS devices organized as 1024 × 1 and have a power saving standby mode. Their access time is 650 ns. The RAMs are bused together on a 12-bit address bus as shown on Q5747-2. The memory is written into from the dataway (W1-W12) with [F(16)A(0)]; write memory and increment program counter. The write lines W1-W12 are enabled to the "data-in" leads of the RAMs by $\overline{\text{WRT}}$. This is a 300 ns negative pulse generated by the retriggerable, resettable monostable multivibrator in 10D when triggered by [F(16)A(0)]. $\overline{\text{WRT}}$ enables the tristate hex bus drivers ('LS367) in 7G and 8G and is also bused to the write enable of the RAM. The memory address (A0-A9) is generated in the program counter as discussed in the next paragraph. The data out (D01-D012) of the RAM is connected directly to the DAC input buffer (D01 LSB) and also to the 'LS367 hex bus drivers (8E and 8F) on Q5747-3. These drivers are enabled by [F(0)A(0)] which gates D01-D012 onto R1-R12 of the dataway. Thus, data may be written from the dataway into memory, read from memory to the DAC, and read from memory onto the dataway.

The program counter (PC) generates the address for the memory. The PC is an incremental up counter built around LS193's (9E, 10A, 10B) as shown on Q5747-3. There are four conditions which may increment the PC: write data [NF(16)A(0)S2′]; read data [NF(0)A(0)S2′], advance PC [NF(25)A(1)S2′], and $\overline{\text{CLK}}$′. (The latter is present only when the module is in the execute mode.) These four signals are ORed since any one should cause the memory address to be incremented. There are two conditions which can clear (zero) the PC. These are module zero [MZ; F(11)A(0)] (on Q5747-2) and execute [EXE; F(25)(A0)]. MZ clears all registers including the PC and the memory. EXE clears the PC in order to force the memory to start at location zero when the function generator is started. This is accomplished by using the leading edge of EXE to clear the PC and the trailing edge to set the execute flag (see Q-5747-2).

---

*AM9102 are plug compatible with Intel 2102, Signetics 2602, Intersil IM7552, and Mostek 4102.

During operation of the module it is necessary to know when the memory goes through a full cycle and to have a signal available for external devices. In addition, if enabled, a LAM may be generated. The end of cycle is detected by NANDing the carry of the second PC counter with the two high order bits of the third counter. This signal presets the overflow flag and is gated onto the Q line of the dataway with N'. The overflow flag is cleared with the next S2'. With this procedure, it is possible to either read or write the entire memory (1024 words) in the Q-stop controlled block transfer mode.[8]

Digital-to-Analog Converter

The digital-to-analog converter is a model DAC-DG12B1 from Datel-Intersil, Inc. The unit is a 12-bit DAC with a fast voltage output and is organized in three sections. These are: a 12-bit strobed digital input buffer, an ultra-fast 12-bit current conversion block with a deglitched switch and a stable Zener reference, and a fast-response operational amplifier. The DAC and associated logic are shown on Q-5747-2. The manufacturers specifications are given in Appendix B.

The required +15 and -15 V supplies are derived from 7815 and 7915 voltage regulators, respectively, which operate off the CAMAC +24 V and -24 V supplies. The +VCC logic supply which used to power the DAC input register is drawn from the +5 V regulator used to power the TTL logic. The DAC may be connected for 0-10 V, -5 to +5 V, or -10 to +10 V output peak to peak. We have implemented the latter by using pins 31 and 32 for the DAC output. The analog ground is connected to pins 29 and 30. Care has been taken to keep the analog ground separate from the digital ground to ensure maximum isolation and noise immunity for the output signal. External trim pots (100 Ω) are provided as shown for the gain and bipolar offset adjustments. The gain adjustment assures a symmetrical plus and minus swing. The bipolar offset adjustment is to ensure that the digital code at the input register is calibrated to give the proper output. The 12-bit input register is connected to the twelve data bits from RAM DO1-DO12. The data are strobed into the register with STB' which is generated as shown on Q-5747-3. The leading edge of CLK' initiates a 650 ns

delay generated by the monostable multivibrator in 10C (on Q-5747-3). This delay is to ensure data D01-D012 are valid to the DAC before actually generating the strobe pulse (RAM worst case access = 650 ns). The 80 ns pulse produced at the end of the delay is the actual strobe. Note that STB' is locked out during a memory read to the dataway [F(0)A(0)] to ensure that the DAC is isolated from the data lines during a read operation.

Programmable Frequency Clock

The basic clock frequency of 1 MHz is provided by a crystal oscillator and a CMOS pulse-shaping circuit as shown in Q-5747-2. In order to ensure that the clock pulses are not clipped on either STOP or START, a 74120 pulse synchronizer is used to gate the clock signal. The clock is gated to three separate counters by a START flag which is derived from the EXE command [F(25)A(0)] through the flip-flop in 6A. The flag is clocked high enabling the synchronizer when EXE is given. Note that a memory read [F(0)A(0)], memory write [F(16)A(0)], disable clock [F(24)A(4)] and master clear (MZ) all clear the flag to ensure that the clock is disabled during these operations. This is important since accidentally operating the PC with CLK and Read or Write simultaneously would be catastrophic. This clock signal (CLK) is the base frequency for the programmable up counters. The preset number (binary divisor) is loaded in 2's complement and latched from the Dataway W1-W18 lines via the three hex D flip-flops (7A-7C) which connect to the counter preset lines. These latches are cleared with MZ and loaded with [F(17)A(0)S1'] (Load Clock Frequency register). It is important to note that the data loaded in the latches are then loaded into the counters at [F(17)A(0)S2']. In this way, when the module is started the first CLK' pulse will appear at the correct time following EXE. Otherwise, the first CLK' pulse would appear after 2^{18} pulses or 262 ms. The counters are used as modulo-N divider counters with the [F(17)A(0)S1'] data used as an offset. The last carry out is used to edge trigger a D flip-flop producing CLK' (programmed clock) on the Q output. CLK' is also used to reload the latched preset number to the counters. The next CLK pulse will clear the flip-flop and the entire cycle will repeat.

The crystal oscillator frequency (1 MHz) can be divided by any binary number between 1 and 2^{18}. Since the PC always counts through 1024 locations, the frequency range of the function generator is from 0.003725 to 976.6 Hz.* Of course, if less resolution is required in the output waveform, more than one period of the desired function could be loaded into the memory in order to increase the output frequency.

Programmable Synchronization Registers

Two additional registers were implemented to allow programmed control of data acquisition. One signal is used to strobe a pair of CAMAC analog-to-digital converters (ADCs) which are connected to the sample displacement linear variable differential transformer (LVDT) signal conditioner and the load cell signal conditioner. In order to obtain accurate stress-strain relationships it is necessary to measure numerous points (e.g., 128) in each function generator cycle. Similarly, in order to synchronize the diffraction pattern data acquisition system with the function generator output, it is necessary at various preprogrammed times to signal either a Signetics 8X300 microcontroller[1,9] or the Time Correlator of the new LASL-ORNL fast digital data acquisition system[10] that they must change the origin of the data acquisition array. The ADC and Time Correlator triggers are strobe pulses, while the 8X300 signal is a latched handshaking signal which is cleared only

*With an 18-bit divisor and a 1 MHz clock, the high frequency repetition rates have little selectability. The available frequencies are $976.6*2^{-n}$ (n = 0,1,2,...262,144) or 976.6, 488.2, 325.5, 244.1, 195.3, 162.8, 139.5 ... . For the present application, where we are concerned with the very low frequency region, such limitations are unimportant. However, in other applications where greater selectability at higher frequencies is necessary, two possibilities exist. If the present dynamic range is satisfactory, one must merely replace the crystal oscillator with one of higher frequency. On the other hand, if a greater dynamic range is required, it is straightforward to provide a 24-bit divisor by adding an 'LS193 counter, an 'LS174 data latch, and a hex 'LS04 inverter.

on acknowledgment by the 8X300. These two signals, labeled C = T (ADC trigger) and C = M (memory map) are generated in a manner similar to CLK'.

The two registers use the divided clock (CLK') as their base frequency. Although sharing the same base frequency the two registers are completely independent. They are loaded simultaneously with their preset numbers (in 2's complement); W1-W8 for C = M and W9-W12 for C = T (7D-7E). The preset number on W1-W12 is latched in the hex D flip-flops with [F(17)A(1)S1'] (Load Trigger Registers). The registers are connected to the preset of 'LS193 up/down binary counters which are used as modulo-N dividers. The data are loaded into the counters with [F(17)(A1)S2'] as with the programmable frequency clock in order to ensure proper synchronization with EXE. The carry out of 9C is used to edge trigger the D flip-flop in 14A which in turn uses $\overline{Q}$ to (a) reload the counters (9A-9C) to preset values, (b) to produce $\overline{C = M}$ for the flip-flop at 6A and (c) to provide the "1" for next edge trigger. The next CLK' (beginning of next cycle) will clear the $\overline{C = M}$ flip-flop (14A). The $\overline{C = M}$ signal is also latched and sent to the status input word of the microprocessor communication module[9] (so it can be used to change the memory map). The flip-flop is cleared when the microprocessor responds with a status acknowledgment. The $\overline{C = T}$ signal (14A and 10D) is generated in the same manner as $\overline{C = M}$ utilizing the logic shown on Q5747-2. The $\overline{C = T}$ is used to trigger an adjustable (150-250 ns) monostable multivibrator in 10D which produces the negative asserted pulse which goes to the ADC trigger.

LAM Control Logic

It is often necessary to generate computer interrupts via the CAMAC LAM (look-at-me) signals when certain events occur. In the present module, depending on experimental conditions, it may be desirable to generate a LAM: (1) when PC = 0, (2) when C = M, or (3) when C = T (i.e., when the address counter overflows, when the ADC trigger is set, or when the microprocessor communication module or Time Correlator trigger is set).

The logic on Q5747-3 performs this task. First, the LAM line itself may be enabled with an [F(26)A(0)S1'] or may be disabled with an

[F(24)A(0)S1' + MZ]. The enable/disable is controlled by an 'LS74 D flip-flop. The three signals which can generate a LAM are handled similarly. They may all be selectively enabled or disabled, and the actual LAM may be selectively cleared (allowing priority servicing). The enable is implemented with [F(26)A(n)S1'], disable is [F(24)A(n)S1'] and clear is [F(10)A(n)S1']. The three subaddresses are organized as follows: A(1) - ADC trigger; A(2) - PC = 0; and A(3) - C = M trigger. The logic is organized similarly for all three with the left bank of D flip-flops used as an enable/disable latch which opens or closes the appropriate gate leading to the LAM logic. The actual signals are latched in the right bank of D flip-flops and remain there until they are selectively cleared. Front panel LEDs indicate which LAM is enabled and whether or not a LAM is present. The LAM status is displayed until cleared by [F(10)A(n)S1'].

SOFTWARE

A function generator driver program called FGEN has been coded in top-down structured FORTRAN[11] using the SFORTRAN precompiler.[12] A source listing is given in Appendix C. This program contains six sections:

(1) a function form generator,
(2) write memory,
(3) verify memory,
(4) set frequency registers,
(5) LAM enables, and
(6) run/stop.

All CAMAC communication with the module is via the standard CAMAC subroutines CFSA, CSSA, and CVSA as defined in the CAMAC tutorial articles[8] and as implemented by King and Hendricks[13] for the ORNL ModComp computers. The various commands to which the program responds are summarized in Table 2. The program response to each command guides the user as to what information is required and the necessary input format. Although this code currently generates sine, triangular or

Table 2. Summary of FGEN Commands

COMMAND*	Purpose
EXIT	Exit from program.
HALT	Stop function generator.
LAM	Enable, disable and clear LAMs.
PARAM	Enter execution parameters (frequency and trigger registers).
RUN	Start function generator in execution.
VERIFY	Read memory and compare with WRITE command.
WRITE	Write desired function curve into memory.

*The command may be abbreviated to the underlined characters. All commands must be terminated by a carriage return.

square waves, clearly any form of input can be easily coded. A typical operating sequence which loads and verifies the memory, loads the counter registers, and then starts the function generator in execution is shown in Example 1. It is important to note that once the generator is started in execution, FGEN is no longer required to remain in core. Thus, core can be utilized by other programs once the unit is running.

Acknowledgments

The authors are indebted to R. T. Roseberry and J. W. Woody, Jr. for their continued interest and encouragement and for their valuable suggestions, to E. Madden for comments on the manuscript, and to F. Sims for his expertise in wirewrapping and assembling the module.

```
JOB
$
ASS 5=TY 6=TY
$
EXE FGEN
PROGRAM FGEN
VERSION A.06   04 MAY 80
WHERE IS FG(I2)?
20
SELECT FUNCTION
WRITE
ENTER WAVE FORM
- 1(COSINE)
- 2(TRIANGULAR)
- 3(SQUARE)
AND VOLTAGE SCALE FACTOR BETWEEN 0.0 AND 10.0 (I1,F5.0)
3 5.
SELECT FUNCTION
VERIFY
VERIFY COMPLETE  8604
SELECT FUNCTION
PARAM
ENTER FREQ, NADC, AND NCM (F6.0,2I5)
(0.0038 <= FREQ <= 977.0   64 <= NADC <= 1024   4 <= NCM <= 1024)
100.    128    4
 FREQUENCY = 108.5069 ( 0.1085069E+03)    NADC =  128    NCM = 4
SELECT FUNCTION
LAM
ENTER - 0(ALL), 1(ADC TRIG), 2(PC=0), 3(C=M)
1
ENTER - C(CLEAR), D(DISABLE), E(ENABLE)
E
SELECT FUNCTION
LAM
ENTER - 0(ALL), 1(ADC TRIG), 2(PC=0), 3(C=M)
0
ENTER - C(CLEAR), D(DISABLE), E(ENABLE)
E
SELECT FUNCTION
RUN
SELECT FUNCTION
HALT
SELECT FUNCTION
EXIT
$
```

Example 1

Operator Response to Program FGEN in Order to Load, Verify, and Start the FUNCTION GENERATOR in Operation.

REFERENCES

1. R. W. Hendricks, J. Appl. Cryst. 11, 15-30 (1978).

2. W. C. Koehler and R. W. Hendricks, "Construction and Operation of a National User-Oriented Small-Angle Neutron Scattering Facility at the Oak Ridge National Laboratory," a proposal accepted for funding by the National Science Foundation, Grant No. DMR-7724458 (1977).

3. W. C. Koehler and R. W. Hendricks, J. Appl. Phys. 50, 1951 (1979).

4. J. M. Schultz, J. S. Lin, and R. W. Hendricks, J. Appl. Cryst. 11, 551-557 (1978).

5. R. W. Hendricks, "Design and Construction of a Specimen Holder and Control System for Dynamic X-Ray and Neutron Small-Angle Scattering Studies of the Plastic Deformation of Polymers," a proposal accepted for funding by the National Science Foundation (February 1979).

6. R. W. Hendricks and S. Suehiro (to be published).

7. P. A. Seeger, A Fast-Response CAMAC Crate Controller, Los Alamos Scientific Laboratory Report, LASL-6605-M, Los Alamos, New Mexico, (December 1976).

8. CAMAC Tutorial Articles, US NIM Committee, Energy Research and Development Report TID-26618 (October 1976).

9. G. W. Turner and R. W. Hendricks, Nucl. Instrum. Meth. 169, 373-380 (1980).

10. R. W. Hendricks, P. A. Seeger, J. W. Scheer, and S. Suehiro, The LASL-ORNL Fast Digital Data Acquisition System, Oak Ridge National Laboratory Report, ORNL/TM-7325 and Los Alamos Scientific Laboratory Report LASL-8260-M (June 1980).

11. E. Yourdon, Techniques of Program Structure and Design, Prentice-Hall, Englewood Cliffs, New Jersey, 1975.

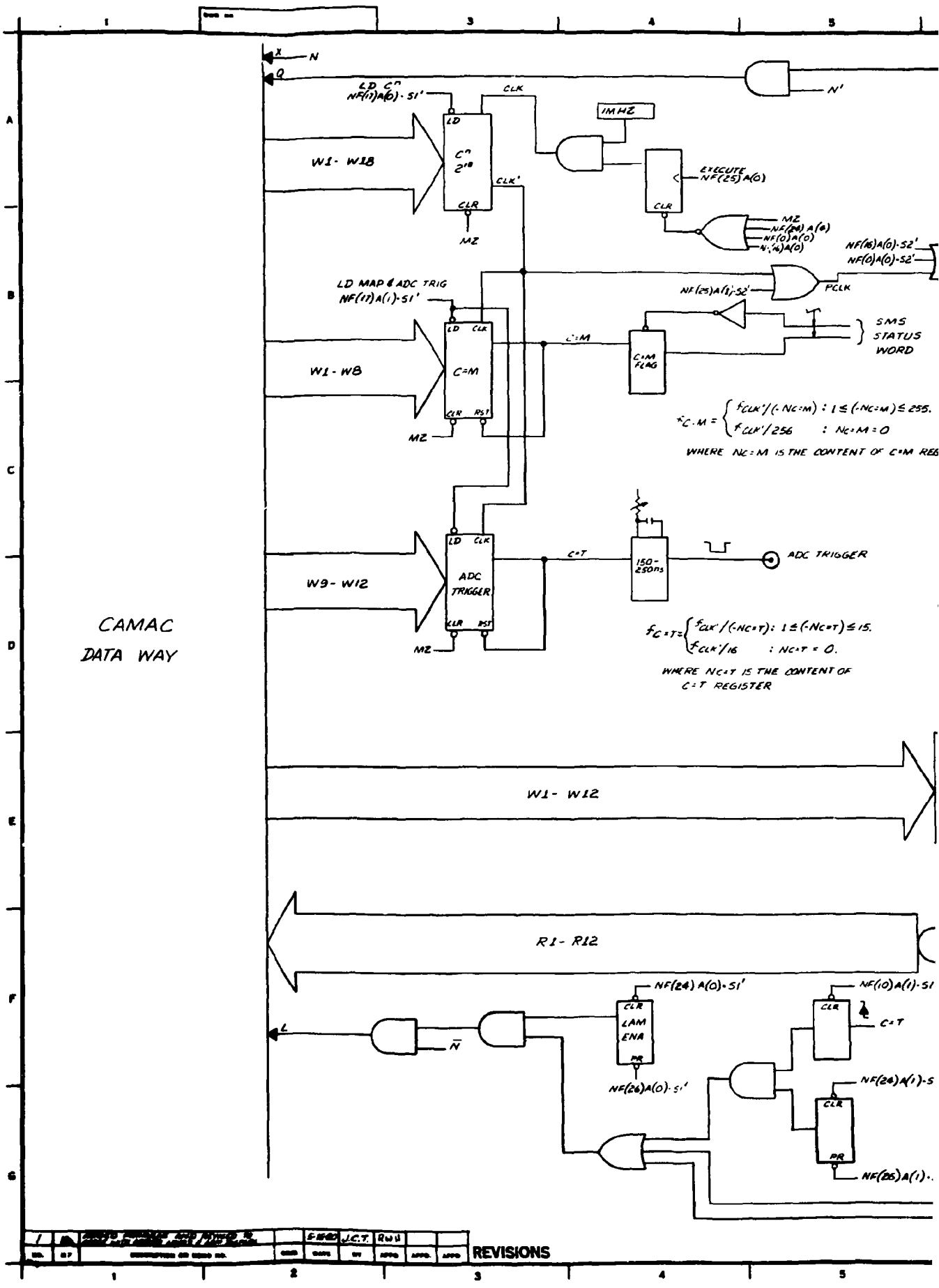
12. SFORTRAN was developed by Lars G. Mossberg of Volvo Flysmotor, A.B., Trollhattan, Sweden, and is distributed in the United States by Software Consulting Services, 901 Whittier Drive, Allentown, PA 18103.

13. S. P. King and R. W. Hendricks, The ORNL 10-m Small-Angle X-Ray Scattering Camera. 8. CAMAC Software, Oak Ridge National Laboratory Report ORNL/TM-6342 V8 (in preparation).

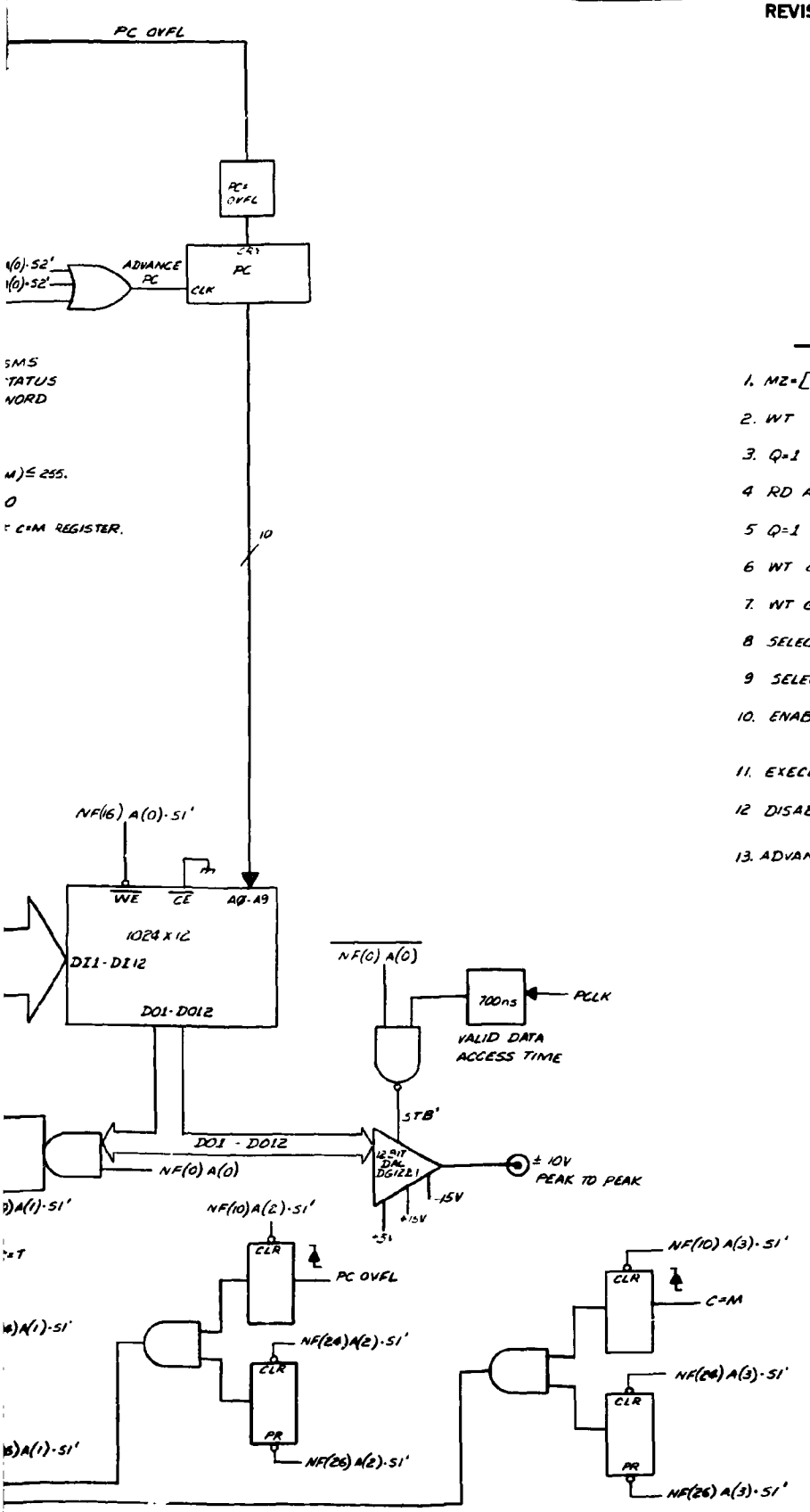
APPENDIX A

Circuit Diagrams

ORNL Model Q-5747 Programmable Function Generator


X
N
Q
N'
LD C"
NF(17)A(0)·S1'
CLK
LD
C"
2"
CLR
CLK'
MZ
W1 - W18
1MHZ
EXECUTE
NF(25)A(0)
CLR
MZ
NF(24)A(0)
NF(0)A(0)
NF(16)A(0)
NF(16)A(0)·S2'
NF(0)A(0)·S2'
LD MAP & ADC TRIG
NF(17)A(1)·S1'
NF(25)A(1)·S2'
PCLK
SMS
STATUS
WORD
LD
CLR
C=M
CLR
RST
MZ
W1 - W8
C=M
C=M
FLAG
$f_{C=M} = \begin{cases} f_{CLK'}/(-Nc=M) : 1 \le (-Nc=M) \le 255. \\ f_{CLK'}/256 \quad : Nc=M = 0 \end{cases}$
WHERE Nc=M IS THE CONTENT OF C=M REG
LD
CLR
ADC
TRIGGER
CLR
RST
MZ
W9 - W12
C=T
150-
250ns
ADC TRIGGER
$f_{C=T} = \begin{cases} f_{CLK'}/(-Nc=T) : 1 \le (-Nc=T) \le 15. \\ f_{CLK'}/16 \quad : Nc=T = 0. \end{cases}$
WHERE Nc=T IS THE CONTENT OF
C=T REGISTER
CAMAC
DATA WAY
W1 - W12
R1 - R12
L
$\bar{N}$
NF(24)A(0)·S1'
CLR
LAM
ENA
PR
NF(26)A(0)·S1'
NF(10)A(1)·S1
CLR
C=T
NF(24)A(1)·S
CLR
PR
NF(25)A(1)·
REVISIONS
J.C.T.
RWJ

REVISIONS



MODE SEQUENCE

1. MZ=[NF(11) A(0)·S1'] + [(Z+C)·S2]
2. WT GRP 1 DATA NF(16) A(0)·S1'
3. Q=1 WHEN PC OVFL
4. RD AND VERIFY GRP 1 DATA, ADVANCE MEMORY ADDR. IS NF(0)A(0)
5. Q=1 WHEN PC OVFL
6. WT GRP 2 DATA - LD C=M (2'S COMPLEMENT W1-W18) NF(17) A(0)·S1'
7. WT GRP 2 DATA - LD C=M & ADC TRIGGER (W1-W12) NF(17) A(1)·S1'
8. SELECTIVELY ENABLE FUNCTION TO GENERATE LAM NF(26) A(n)·S1'
9. SELECTIVELY DISABLE OTHER LAM'S NF(24) A(n)·S1'
10. ENABLE OR DISABLE LAM FLAG; ENA; NF(26) A(0)·S1'
    DIS; NF(24) A(0)·S1'
11. EXECUTE (START DRIVER) NF(25) A(0)
12. DISABLE (HLT DRIVER) NF(24) A(4)
13. ADVANCE PC NF(25) A(1)

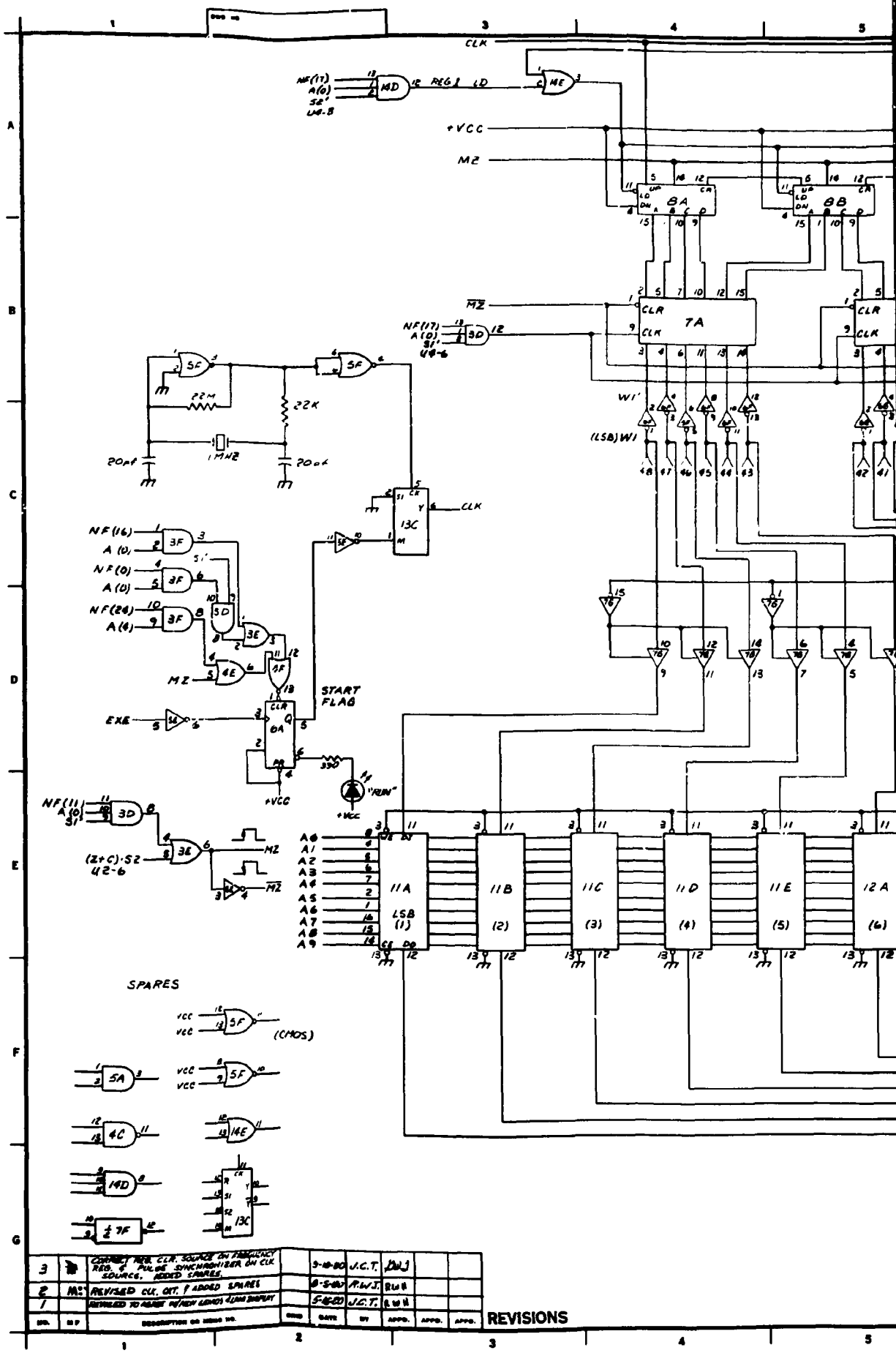
REFERENCE DRAWINGS	DWG. NO.
J.C. TURNER 10-79	
G.W. TURNER 10-79	
COORDINATOR R.T. ROSEBERRY	

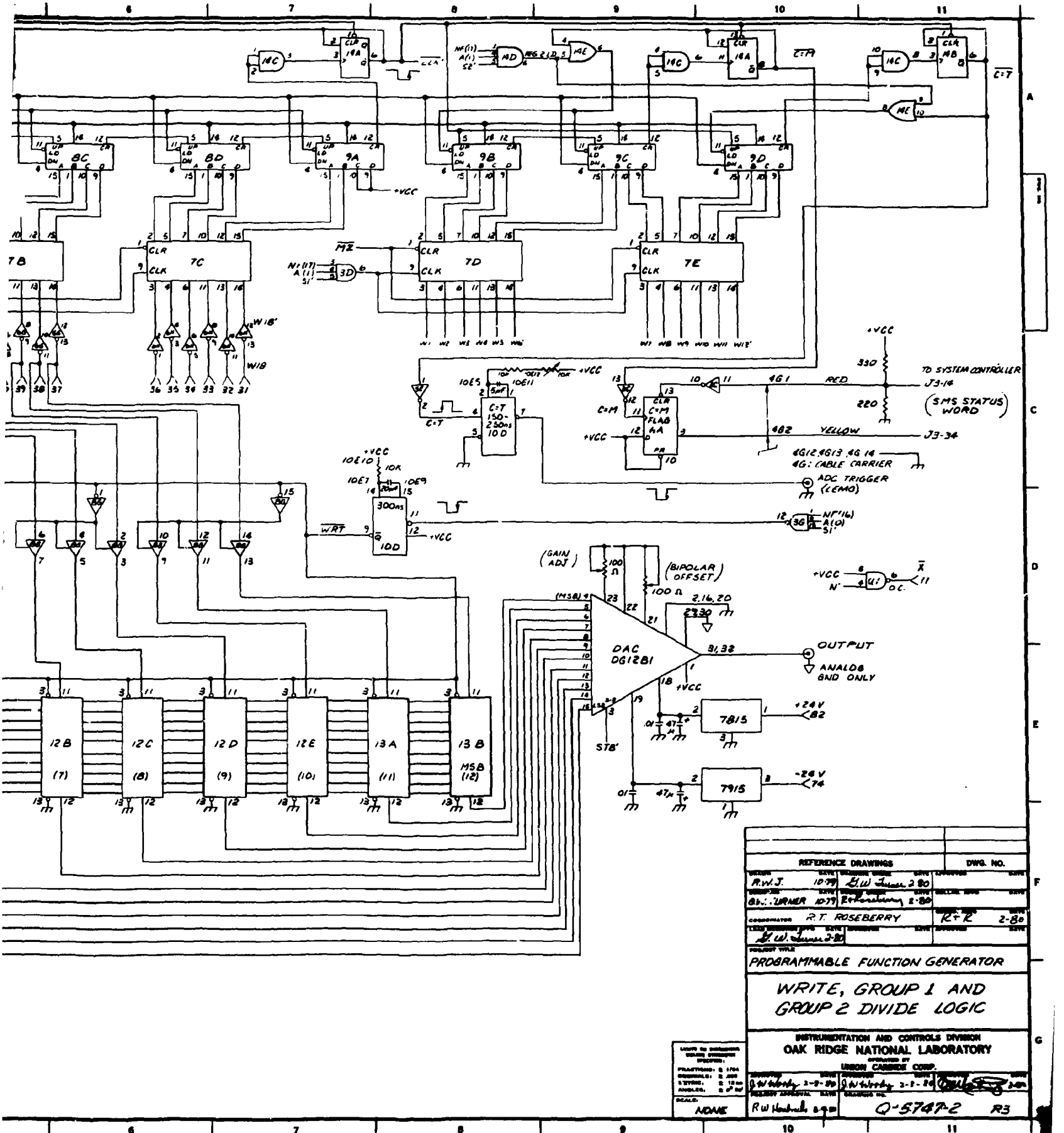
PROGRAMMABLE FUNCTION GENERATOR

BLOCK DIAGRAM

INSTRUMENTATION AND CONTROLS DIVISION
OAK RIDGE NATIONAL LABORATORY
OPERATED BY
UNION CARBIDE CORP.

Q-5747-1 R1


CLK
REG 1 LD
+VCC
MZ
BA
BB
7A
W1'
(LSB) W1
22M
22K
20pf
1 MHZ
20pf
13C
START FLAG
EXE
6A
330
+VCC
"RUN"
(Z+C)·S2
42-6
11A LSB (1)
11B (2)
11C (3)
11D (4)
11E (5)
12A (6)
SPARES
(CMOS)
5A
4C
14E
14D
13C
7F
CORRECT REG. CLR. SOURCE ON FREQUENCY REG. & PULSE SYNCHRONIZER ON CLK SOURCE. ADDED SPARES.
REVISED CKT. DWT. & ADDED SPARES
REVISED TO ADJUST REFRESH LOADS & LAMP DISPLAY
REVISIONS



REFERENCE DRAWINGS	DWG. NO.
DESIGN R.W.J. 10-79	
DRAWN G.W. URNER 10-79	
COORDINATOR R.T. ROSEBERRY	R+R 2-80

PROJECT TITLE

PROGRAMMABLE FUNCTION GENERATOR

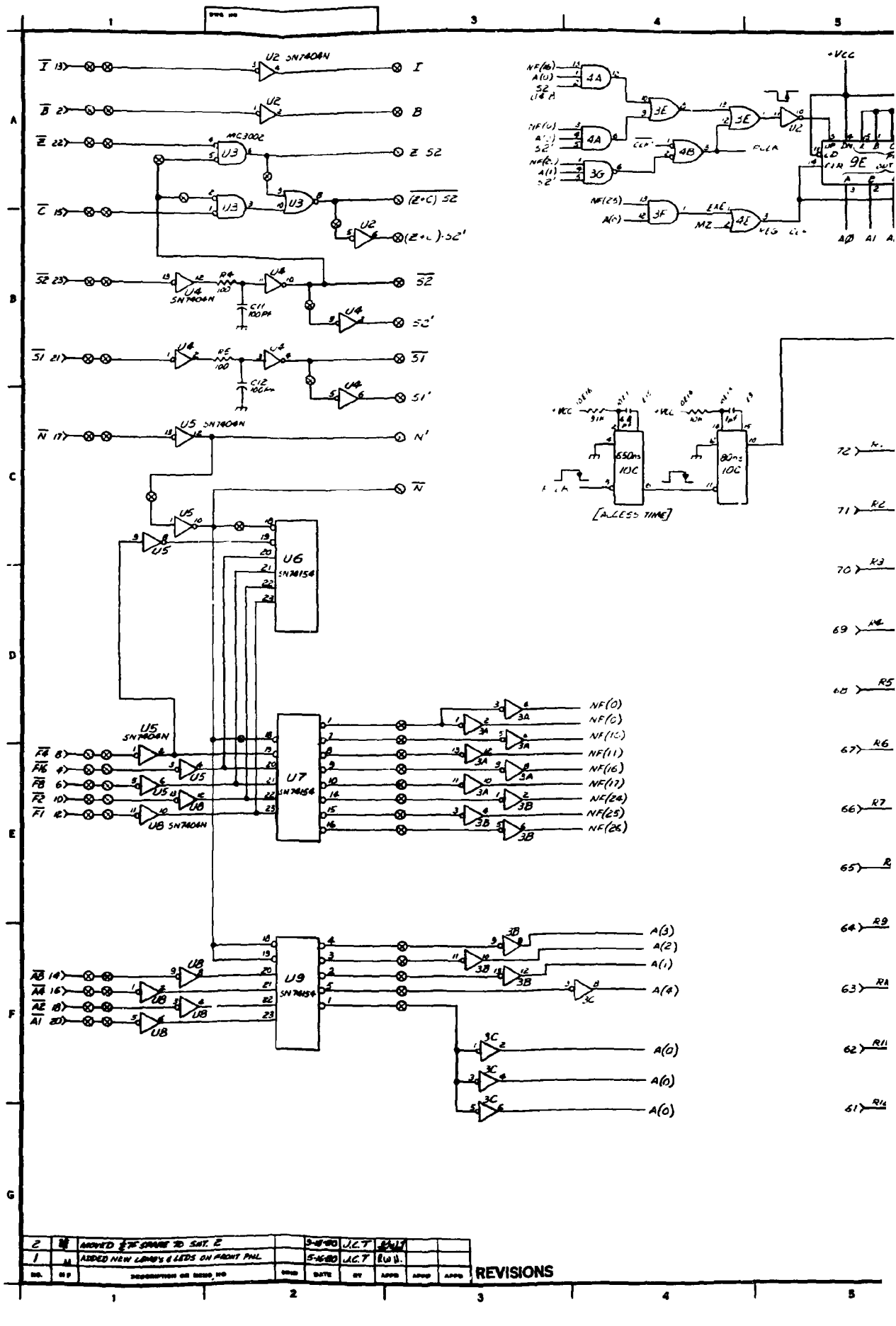
WRITE, GROUP 1 AND GROUP 2 DIVIDE LOGIC

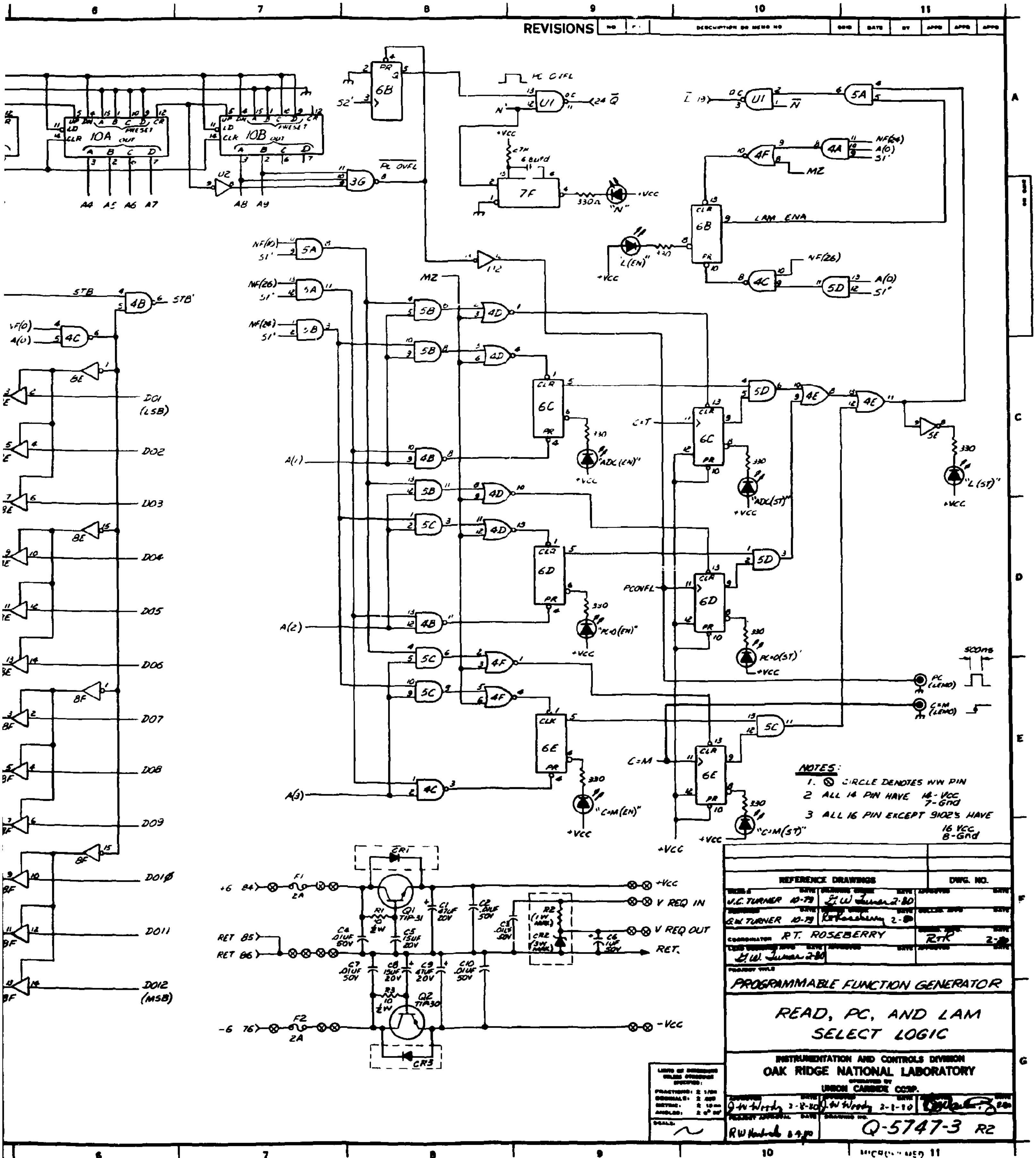
INSTRUMENTATION AND CONTROLS DIVISION
OAK RIDGE NATIONAL LABORATORY
OPERATED BY
UNION CARBIDE CORP.

DRAWN	DESIGNED	CHECKED
G W Urner 2-8-80	G W Urner 2-8-80	

RELEASED	DRAWING NO.
R W Hembree 2-8-80	Q-5747-2 R3

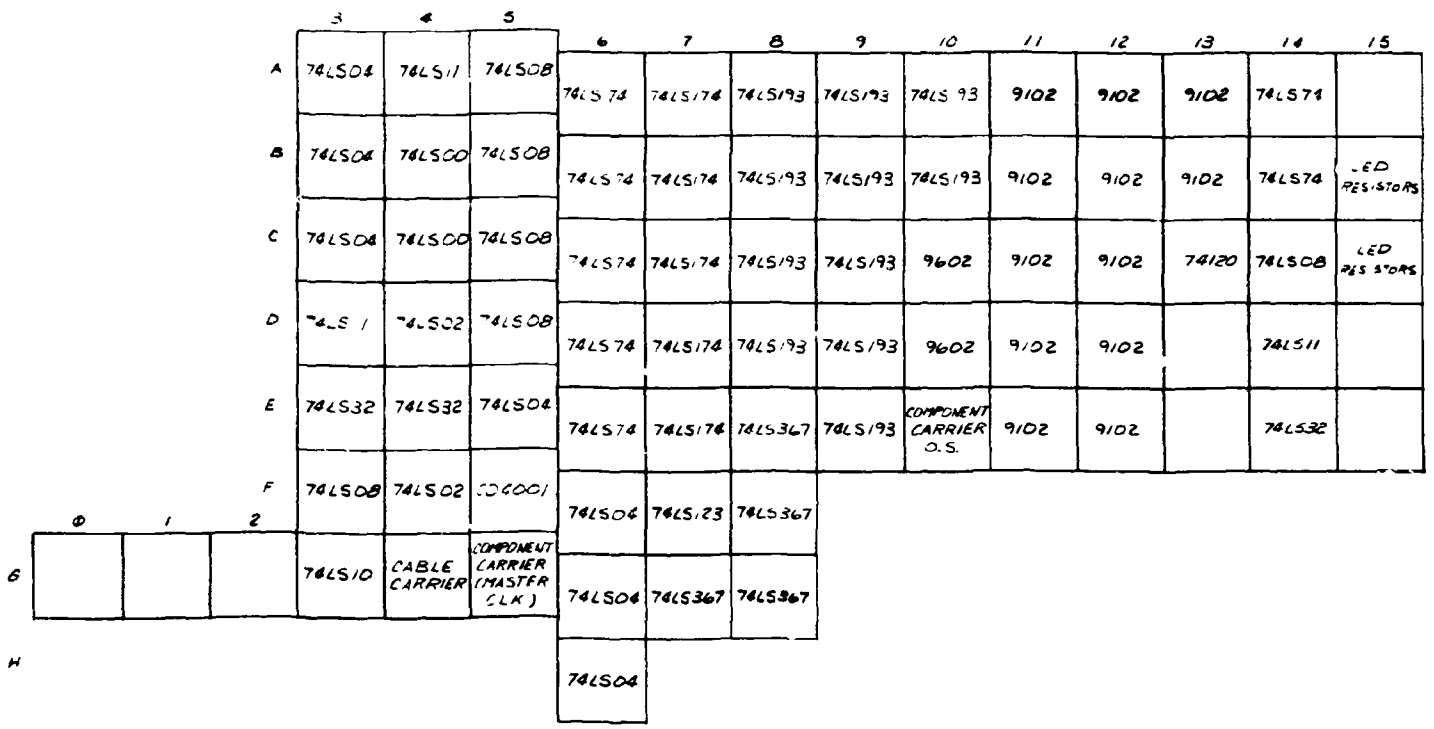
SCALE: NONE


U2 SN7404N
MC3002
U4 SN7404N
U5 SN7404N
U6 SN74154
U7 SN74154
U9 SN74154
U8 SN7404N
4A
3E
5E
4B
3G
3F
4E
9E
[ACCESS TIME]
NF(0)
NF(1)
NF(10)
NF(11)
NF(16)
NF(17)
NF(24)
NF(25)
NF(26)
A(3)
A(2)
A(1)
A(4)
A(0)
A(0)
A(0)
REVISIONS


REVISIONS
10A
10B
PRESET
A4 A5 A6 A7
A8 A9
PC OVFL
PC OVFL
N'
24 Q̄
L 19
LAM ENA
MZ
STB
STB'
DO1 (LSB)
DO2
DO3
DO4
DO5
DO6
DO7
DO8
DO9
DO10
DO11
DO12 (MSB)
A(1)
A(2)
A(3)
C+T
PCOVFL
C=M
"ADC(EN)"
"ADC(ST)"
"PC=0(EN)"
"PC=0(ST)"
"C=M(EN)"
"C=M(ST)"
"L(ST)"
"L(EN)"
"N"
500ns
PC (LEMO)
C=M (LEMO)
+6 84
RET 85
RET 86
-6 76
+Vcc
V REQ IN
V REQ OUT
RET.
-Vcc
NOTES:
1. ⊗ CIRCLE DENOTES WW PIN
2 ALL 14 PIN HAVE 14-Vcc 7-Gnd
3 ALL 16 PIN EXCEPT 9102S HAVE 16 Vcc 8-Gnd
REFERENCE DRAWINGS
DWG. NO.
J.C. TURNER 10-79
G.W. TURNER 10-79
R.T. ROSEBERRY
PROGRAMMABLE FUNCTION GENERATOR
READ, PC, AND LAM SELECT LOGIC
INSTRUMENTATION AND CONTROLS DIVISION
OAK RIDGE NATIONAL LABORATORY
UNION CARBIDE CORP.
Q-5747-3 R2

2	[illegible]	Pulse synchroniz[illegible] shafts for [illegible]		9-[illegible]	J.C.T.	[illegible]		
1	[illegible]	Released to Books 5[illegible], 16B & 16C		6-6-80	R.W.J.	[illegible]		
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

REVISIONS



PARTS LOCATION

VIEW FROM COMPONENT SIDE

REVISIONS	NO	BY	DESCRIPTION OR MEMO NO	CHKD	DATE	BY	APPD	APPD	APPD

9	10	11	12	13	14	15
74LS193	74LS 93	9102	9102	9102	74LS74	
74LS193	74LS193	9102	9102	9102	74LS74	-ED RESISTORS
74LS193	9602	9102	9102	74120	74LS08	LED RES STORS
74LS193	9602	9102	9102		74LS11	
74LS193	COMPONENT CARRIER O.S.	9102	9102		74LS32	

LOCATION
COMPONENT SIDE

REFERENCE DRAWINGS		DWG. NO.
RWJ 10-79	G W Turner 2-80	
G W TURNER 10-79	R T Roseberry 2-80	
COORDINATION R.T. ROSEBERRY		R T R 2-80
G W Turner 2-80		

PROGRAMMABLE FUNCTION GENERATOR

PARTS LOCATION

INSTRUMENTATION AND CONTROLS DIVISION
OAK RIDGE NATIONAL LABORATORY
OPERATED BY
UNION CARBIDE CORP.

G W Woody 2-8-80	G W Woody 2-8-80	2-8-80
R W Thomas 2-8-80	Q-5747-4	R2

NONE

MICROFILMED

APPENDIX B

Manufacturers' Specifications*

DATEL — INTERSIL MODEL DAC-DG12B1

Digital-to-Analog Converter

*Taken with permission from DATEL "Modules for Data Conversion" handbook.

DATEL SYSTEMS, INC.

FAST 12-BIT DEGLITCHED DAC

MODEL DAC-DG12B

FEATURES

- ±2 LSB Max. Glitch
- 600 nsec. Settling Time
- Up to 2.5 MHz Update Rate
- 12 Bit Resolution
- Self-Contained Module

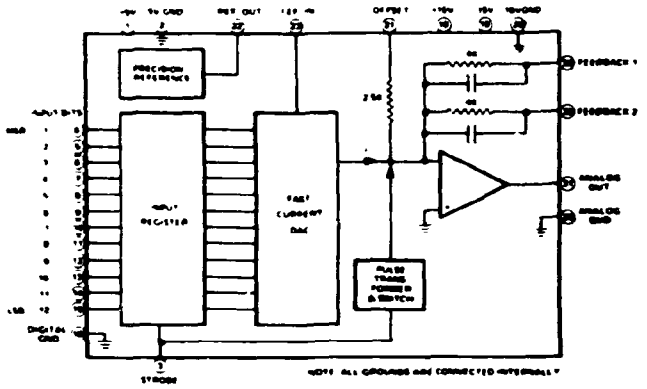
GENERAL DESCRIPTION

Model DAC-DG12B is a deglitched 12-bit D/A converter with a fast voltage output. The maximum output glitch amplitude is ±2 LSB's while settling time for a 10 volt output change is 600 nsec to 1 LSB. For a 10 volt change to 1%, the settling time is 250 nsec and for small output changes it is only 400 nsec permitting update rates as fast as 2.5 MHz.

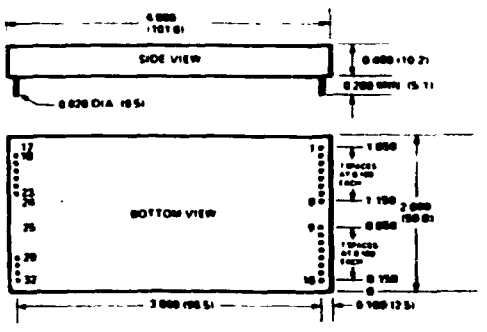
The unique circuit design of the DAC-DG12B realizes both small size and low price at the same time. Unlike other deglitched DAC's which are comprised of several inter-connected modules mounted on a circuit card, the DAC-DG12B is completely self-contained in a compact 4 x 2 x 0.4 inch (102 x 51 x 10 mm) module. It consists of several optimized circuit functions: digital input register, ultra-fast 12-bit current DAC, stable Zener voltage reference, fast deglitching switch, and a fast output operational amplifier.

The DAC-DG12B has three voltage output ranges determined by external pin connection: 0 to −10 V, ±5V and ±10 V. Output current is ±10 mA with output short circuit protection. For higher output current requirements an external current booster amplifier may be connected inside the feedback loop of the output amplifier. There are two input coding options: complementary binary/complementary offset binary or complementary two's complement.

The DAC-DG12B is an ideal device for fast CRT display applications and for other test and measurement applications where monotonic output changes are required.



MECHANICAL DIMENSIONS—INCHES (MM)



INPUT/OUTPUT CONNECTIONS

PIN	FUNCTION
1	+5 V POWER
2	5 V GND
3	STROBE
4	BIT 1 IN (MSB)
5	BIT 2 IN
6	BIT 3 IN
7	BIT 4 IN
8	BIT 5 IN
9	BIT 6 IN
10	BIT 7 IN
11	BIT 8 IN
12	BIT 9 IN
13	BIT 10 IN
14	BIT 11 IN
15	BIT 12 IN (LSB)
16	DIGITAL GND
18	+15 V POWER
19	−15 V POWER
20	15 V GND
21	OFFSET
22	REF OUT
23	REF IN
29	ANALOG GND
30	FEEDBACK 1
31	ANALOG OUT
32	FEEDBACK 2

INPUTS

Resolution	12 bits
Coding, unipolar	Complementary Binary
Coding, bipolar	Complementary Offset Binary ' Complementary Two's Comp '
Input Logic Level, bit ON ("0")	0V to +0.8 V
Input Logic Level, bit OFF ("1")	+2.0 V to +5.5 V
Logic Loading	1 TTL load
Input Strobe Pulse[2]	HI to LO transition causes transfer of data from register to DAC
Input Strobe Loading	2 TTL loads

OUTPUTS

Output Voltage, unipolar[3]	0 to −10 V
Output Voltage, bipolar[3]	±5 V, ±10 V
Output Current, S.C. protected	±20 mA typ., ±10 mA min
Output Impedance, DC	0.05 ohm

PERFORMANCE

Linearity Error	±½ LSB max
Differential Nonlinearity	±½ LSB max
Zero Error, before trimming	±½ LSB max
Gain Tempco	±35 ppm/° C max
Offset Tempco, bipolar	±15 ppm/° C max
Zero Tempco, unipolar	±5 ppm/° C of FS max
Diff. Nonlinearity Tempco	±2 ppm/° C of FS
Monotonicity	0° C to 70° C
Settling Time, 10 V change to 1 LSB	600 nsec typ 700 nsec max
Settling Time, 20 V change to 1 LSB	1.0 μsec typ 1.2 μsec max
Settling Time, 10 V change to 1%	250 nsec max
Settling Time, 20 V change to 1%	550 nsec max
Settling Time, ±4 LSB change	400 nsec
Slew Rate	50 V μsec
Glitch Amplitude[4]	±1 LSB typ ±2 LSB max
Glitch Area	250 mV-nsec
Power Supply Rejection	0.01%/% supply

POWER REQUIREMENT

+15 VDC ±0.5 V @ 50 mA
−15 VDC ±0.5 V @ 35 mA
+ 5 VDC ±0.25 V @ 230 mA

PHYSICAL-ENVIRONMENTAL

Operating Temperature Range	0° C to 70° C
Storage Temperature Range	−55° C to +125° C
Case Size	4 x 2 x 0.4 inches (101.6 x 50.8 x 10.2 mm)
Case Material	Black Diallyl Phthalate per MIL-M-14
Pins	0.020" round, gold plated 0.200 lg. min
Weight	4 oz. max (114 g)

NOTES

1. Because the analog output is inverted, in the bipolar mode the complementary offset binary code is equivalent to offset binary and the complementary two's complement code is equivalent to two's complement. See Technical Note 4.
2. Has same logic levels as data inputs.
3. Determined by external pin connection.
4. Measured with 20 MHz bandwidth oscilloscope at major carry (half scale) and at 7 transitions either side of major carry.

1. The sequence of operations inside the DAC-DG12B after the input strobe changes from HI to LO are
   a. the pulse transformer and switch are activated and turn ON
   b. within 11 nanoseconds (typically) the data in the input register is transferred to the current DAC
   c. during the next 19 nanoseconds (typically) the DAC output current changes
   d. after 30 nanoseconds (typically) from the strobe change the pulse transformer and switch are deactivated turning OFF
   e. the output amplifier begins to change to its new output value
2. A 5 nanosecond minimum setup time is required for the input data to be valid before the input strobe goes from HI to LO. The input strobe then should not go HI again for at least 40 nanoseconds.
3. The maximum update rate for the DAC-DG12B is 25 MHz based on the 400 nanosecond settling time for small output changes (±4 LSB's max.). For 10 V changes to 1% of final value the maximum update rate is 4 MHz and for 10 V changes to within 1 LSB of final value the maximum update rate is 1.6 MHz.
4. From the coding tables it should be noted that each model of the DAC-DG12B has its coding defined in two ways when operating in bipolar mode. For the DAC-DG12B1 the complementary offset binary coding with inverted (negative) analog output is the same as offset binary coding with noninverted (positive) analog output except for an analog shift of 1 LSB. The converter therefore can be externally calibrated for either code. For the DAC-DG12B2 the complementary two's complement coding with inverted (negative) analog output is the same as two's complement coding with noninverted (positive) analog output except for an analog shift of 1 LSB.
5. The DAC-DG12B is internally calibrated at zero for unipolar operation with a zero error of ±½ LSB maximum. In many applications therefore no external zero adjustment is required. For exact calibration the external zero adjustment should be used. The DAC-DG12B2 operates in unipolar mode except that its input code is complementary binary with the MSB inverted.
6. For higher output current drive capability a wideband current booster amplifier with unity voltage gain may be enclosed inside the feedback loop of the output amplifier.

ORDERING INFORMATION

DAC-DG12B —┬—

CODING
1 = COMP BINARY/COMP OFFSET BINARY
2 = COMP TWO's COMPLEMENT

PRICES (1-9)

DAC-DG12B1	$275.00
DAC-DG12B2	$275.00

MATING SOCKETS DILS-2
(2/MODULE) AT $5.00/PAIR

TRIMMING POTENTIOMETERS
TP100 (100 ohms) TP10K (10 K ohms) AT $3.00 EACH

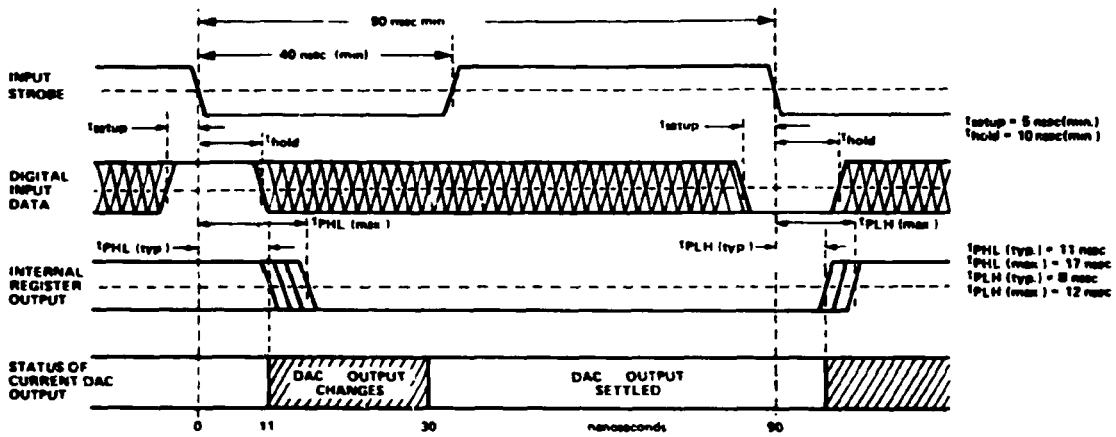
For extended temperature range operation, the following suffixes are added to the model number. Consult factory for pricing.

-EX	−25° C to +85° C operation
-EXX-HS	−55° C to +85° C operation with hermetically sealed semiconductor components

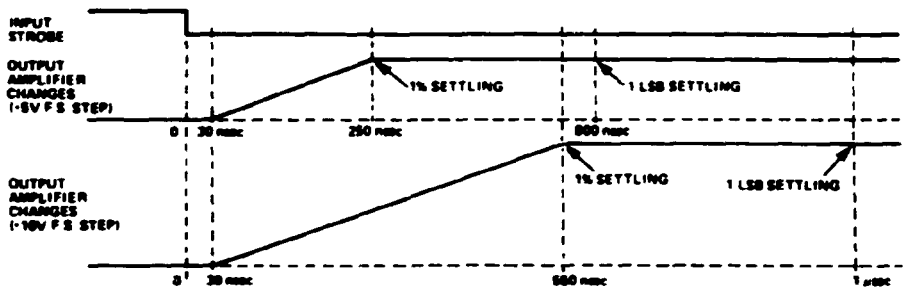
THE DAC-DG12B IS COVERED BY GSA CONTRACT

TIMING DIAGRAMS

INTERNAL TIMING



EXTERNAL TIMING



TYPICAL TABLES

UNIPOLAR OPERATION

SCALE	VOLTAGE RANGE	DAC-DG12B1
	0 TO −10 V	COMP BINARY
0	0 0000	1111 1111 1111
0-1 LSB	0 0024	1111 1111 1110
¼FS	2 5000	1011 1111 1111
½FS	5 0000	0111 1111 1111
¾FS	7 5000	0011 1111 1111
FS - 1 LSB	9 9976	0000 0000 0000

BIPOLAR OPERATION

SCALE	VOLTAGE RANGE		DAC-DAC-DG12B1		DAC-DG12B2	
	±5 V	±10 V	COMP OFFS BIN	OFFSET BINARY	COMP 2's COMP	2'S COMPLEMENT
+FS	+5 0000 V	+10 0000 V	1111 1111 1111		0111 1111 1111	
+FS-1 LSB	+4 9976	+9 9951	1111 1111 1110	1111 1111 1111	0111 1111 1110	0111 1111 1111
0 + 1 LSB	+0 0024	+0 0049	1000 0000 0000	1000 0000 0001	0000 0000 0000	0000 0000 0001
0	0 0000	0 0000	0111 1111 1111	1000 0000 0000	1111 1111 1111	0000 0000 0000
FS + 1 LSB	4 9976	9 9951	0000 0000 0000	0000 0000 0001	1000 0000 0000	1000 0000 0001
FS	5 0000	10 000		0000 0000 0000		1000 0000 0000

CONNECTIONS AND CALIBRATION

CALIBRATION PROCEDURE

Select the desired output voltage range (0 to +10 V, ±5 V or ±10 V) and make the connections shown in the diagrams below. To calibrate refer to the coding tables on the previous page.

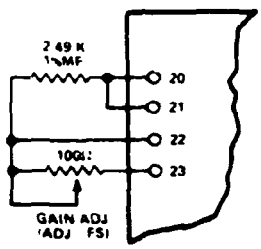
UNIPOLAR OPERATION (0 TO +10 V OUTPUT)

1. **Zero Adjustment.** Set the digital input code to 1111 1111 1111 and adjust the ZERO ADJ. potentiometer to give 0.0000 V output.
2. **Gain Adjustment.** Set the digital input code to 0000 0000 0000 and adjust the GAIN ADJ. potentiometer to give 9.9976 V output.

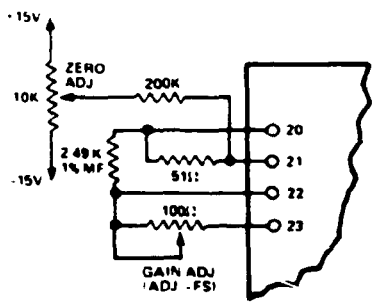
BIPOLAR OPERATION (±5 V OR ±10 V OUTPUT)

1. **Offset Adjustment.** Set the digital input code to 0111 1111 1111 (comp. offset binary) 1000 0000 0000 (offset binary) 1111 1111 1111 (comp. two's complement) or 0000 0000 0000 (two's complement) and adjust the BIPOLAR OFFSET ADJ. potentiometer to give 0.0000 V output.
2. **Gain Adjustment.** Set the digital input code to 0000 0000 0000 (comp. offset binary) 0000 0000 0001 (offset binary) 1000 0000 0000 (comp. two's complement) or 1000 0000 0001 (two's complement) and adjust the GAIN ADJ. potentiometer to give +4.9976 V output for (±5 V range) or 9.9951 V output for (±10 V range).
3. Repeat steps 1 and 2 to recheck adjustments.

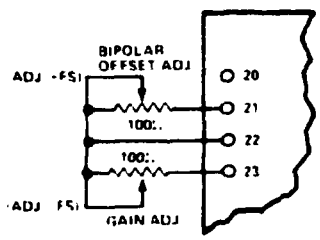
UNIPOLAR OPERATION – NO ZERO ADJ.



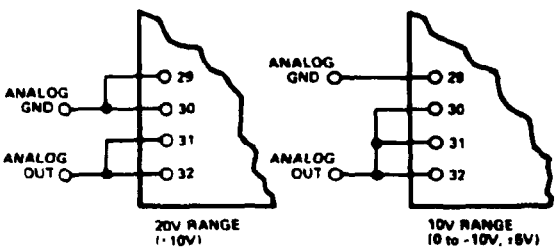
UNIPOLAR OPERATION – WITH ZERO ADJ.



BIPOLAR OPERATION



OUTPUT CONNECTIONS



DATEL SYSTEMS, INC

APPENDIX C

Program FGEN Source Listing

```
1     1        PROGRAM FGEN
1     2  C...
1     3  C...  VERSION A.06   04 MAY 80
1     4  C...
1     5        DIMENSION KFRQ(2),XDATA(2)
1     6        INTEGER ANS(3),DATA(2)
1     7        INTEGER FNC,RD1,CLM,WT1,WT2,CL2,DIS,XEQ,ENB
1     8        INTEGER CDREGF
1     9        LOGICAL IS,ID,EXIT
1    10        DIMENSION IWAVE(1024,3),ICOS(1024),ITRI(1024),ISQU(1024)
1    11        EQUIVALENCE (IWAVE(1,1),ICOS(1)),(IWAVE(1,2),ITRI(1))
1    12       1,(IWAVE(1,3),ISQU(1))
1    13        DATA RD1/0/,CLM/10/,WT1/16/,WT2/17/,CL2/11/,DIS/24/,XEQ/25/,
1    14       1ENB/26/,IB/0/
1    15  C...  CREATE WAVEFORM ( 2'S COMPLEMENT)
1    16        WRITE(6,8000)
1    17        READ(5,8005) N
1    18        DA=3.1415927/512.0
1    19        DOLOOP I=2,513
1    20           ICOSI=IFIX(2047.*COS(DA*FLOAT(I-1)))
1    21           ICOS(I)=ICOSI
1    22           ICOS(1026-I)=ICOSI
1    23           ITRII=1799-7*I
1    24           ITRI(I)=ITRII
1    25           ITRI(1026-I)=ITRII
1    26           ISQU(I)=-2047
1    27           ISQU(1026-I)=2047
1    28        ENDLOOP
1    29        ICOS(1)=2047
1    30        ITRI(1)=1792
1    31        ISQU(1)=-2047
1    32  C...  SET UP CAMAC LREG'S
1    33        IA=0
1    34        LREG=CDREGF(IB,N,IA)
1    35        IA=1
1    36        LREG1=CDREGF(IB,N,IA)
1    37        IA=4
1    38        LREG4=CDREGF(IB,N,IA)
1    39  C...
1    40  C... EXECUTE FUNCTIONS
1    41  C...
1    42        EXIT=.FALSE.
1    43        WHILE(.NOT.EXIT) DO
1    44           WRITE(6,8010)
1    45           READ(5,8015) IFNC
1    46           IF(IFNC.EQ.'WR') THEN
1    47              WRITE(6,8020)
1    48              READ(5,8025) IW,SF
1    49              SF=SF/10.
1    50  C...  LOAD MEMORY
1    51                 CALL RESETM(LREG,ISTAT)
1    52                 DATA(1)=0
1    53                 DOLOOP I=1,1024
1    54                    IDATA =INVOFS(IFIX(SF*FLOAT(IWAVE(I,IW))))
1    55                    CALL CSSA(WT1,LREG,IDATA,ISTAT,IERR)
1    56                 ENDLOOP
1    57              ELSEIF(IFNC.EQ.'VE') THEN
1    58                 CALL RESETM(LREG,ISTAT)
1    59  C...  VERIFY MEMORY
1    60                 DOLOOP I=1,1024
1    61                    CALL CSSA(RD1,LREG,IDATA,ISTAT,IERR)
1    62                    IF(IDATA.GT.2047) IDATA=IDATA-4096
1    63                    IWAV=INVOFS(IFIX(SF*FLOAT(IWAVE(I,IW))))
1    64                    IF (IERR.NE.0.OR.IWAV.NE.IDATA) THEN
1    65                       WRITE(6,8040) I,IWAV,IDATA,ISTAT
1    66                    ENDIF
1    67                 ENDLOOP
1    68                 CALL CSSA(RD1,LREG,IDATA,ISTAT,IERR)
1    69                 WRITE(6,8045) ISTAT
```

```
1   70          ELSEIF(IFNC.EQ.'PA') THEN
1   71  C...  INPUT PARAMTERS
1   72            WRITE(6,8050)
1   73            READ(5,8055) FREQ,NADC,NCM
1   74            WHILE(FREQ.LE.0.0.OR.NADC.LE.0.OR.NCM.LE.0) DO
1   75               WRITE(6,8090)
1   76               READ(5,8055) FREQ,NADC,NCM
1   77            ENDDO
1   78            FRQ=1000./1.024/FREQ
1   79            IF(FRQ.LT.1.) FRQ=1.
1   80            IF(FRQ.GT.262143.) THEN
1   81               FRQ=0.
1   82               FREQX=1./1.024/262.144
1   83            ELSE
1   84               FREQX=1000./1.024/AINT(FRQ)
1   85            ENDIF
1   86            IF(NADC.GT.1024) NADC=1024
1   87            IF(NADC.LT.64) NADC=64
1   88            IF(NCM.GT.1024) NCM=1024
1   89            IF(NCM.LT.4) NCM=4
1   90            WRITE(6,8056) FREQX,FREQX,NADC,NCM
1   91            KFRQ(1)=INT(FRQ/65536.)
1   92            FRQ2=AMOD(FRQ,65536.)
1   93            IF(FRQ2.GE.32768.) FRQ2=FRQ2-65536.
1   94            KFRQ(2)=INT(FRQ2)
1   95            KADC=1024/NADC
1   96            KCM=1024/NCM
1   97  C...  TWO'S COMPLEMENT KCM,KADC,KFRQ
1   98            KCM=256-KCM
1   99            KADC=16-KADC
1  100            IF(KFRQ(1).NE.0) KFRQ(1)=-KFRQ(1)
1  101            IF(KFRQ(2).NE.0) THEN
1  102               KFRQ(2)=-KFRQ(2)
1  103               KFRQ(1)=KFRQ(1)-1
1  104            ENDIF
1  105  C...  WRITE CONTROL REGISTERS
1  106            CALL CFSA(WT2,LREG,KFRQ,ISTAT,IERR)
1  107            IDATA=KADC*256+KCM
1  108            CALL CSSA(WT2,LREG1,IDATA,ISTAT,IERR)
1  109         ELSEIF(IFNC.EQ.'RU') THEN
1  110  C...  START
1  111            CALL CSSA(XEQ,LREG,IDATA,ISTAT,IERR)
1  112         ELSEIF(IFNC.EQ.'HA') THEN
1  113  C...  STOP
1  114            CALL CSSA(DIS,LREG4,IDATA,ISTAT,IERR)
1  115         ELSEIF(IFNC.EQ.'AD') THEN
1  116  C...  ADVANCE PC
1  117            CALL CSSA(XEQ,LREG1,IDATA,ISTAT,IERR)
1  118         ELSEIF(IFNC.EQ.'CA') THEN
1  119  C...  CAMAC FUNCTION
1  120            WRITE(6,8060)
1  121            READ(5,8065) FNC
1  122            WRITE(6,8070)
1  123            READ(5,8075) ANS
1  124            ID=.FALSE.
1  125            IF(ANS(1).EQ.'Y') ID=.TRUE.
1  126            IS=.FALSE.
1  127            IF(ANS(3).EQ.'Y') IS=.TRUE.
1  128            CALL CMCFNC(FNC,ID,IS)
1  129         ELSEIF(IFNC.EQ.'LA') THEN
1  130  C...  CAMAC LAM'S
1  131            WRITE(6,8080)
1  132            READ(5,8025) ILAM
1  133            IA=ILAM
1  134            LREGL=CDREGF(IB,N,IA)
1  135            WRITE(6,8085)
1  136            READ(5,8015) ILC
1  137            IF(ILC.EQ.'C') CALL CSSA(CLM,LREGL,IDATA,ISTAT,IERR)
1  138            IF(ILC.EQ.'D') CALL CSSA(DIS,LREGL,IDATA,ISTAT,IERR)
1  139            IF(ILC.EQ.'E') CALL CSSA(ENB,LREGL,IDATA,ISTAT,IERR)
```

```
1  140          ELSEIF(IFNC.EQ.'EX') THEN
1  141             EXIT=.TRUE.
1  142          ELSEIF(IFNC.EQ.'FN') THEN
1  143             WRITE(6,8100)
1  144          ELSE
1  145  C...  ERROR
1  146             WRITE(6,8090)
1  147          ENDIF
1  148        ENDDO
1  149        STOP
1  150   8000 FORMAT(1H ,'PROGRAM FGEN'/
1  151       1 1H ,'VERSION A.06   04 MAY 80'/
1  152       2 1H ,'WHERE IS FG(I2)?')
1  153   8005 FORMAT(I2)
1  154   8010 FORMAT(1H ,'SELECT FUNCTION')
1  155   8015 FORMAT(A2)
1  156   8020 FORMAT(1H ,'ENTER WAVE FORM'/
1  157       1 1H ,'- 1(COSINE)'/
1  158       2 1H ,'- 2(TRIANGULAR)'/
1  159       3 1H ,'- 3(SQUARE)'/
1  160       4 1H ,'AND VOLTAGE SCALE FACTOR BETWEEN 0.0 AND 10.0 (I1 F5.0)')
1  161   8025 FORMAT(I1,F5.0)
1  162   8030 FORMAT(1H ,'ERROR IN MEMORY LOAD')
1  163   8035 FORMAT(1H ,'WRITE COMPLETE',2X,Z4)
1  164   8040 FORMAT(1H ,'ERROR IN VERIFYING ',3I10.2X,Z4)
1  165   8045 FORMAT(1H ,'VERIFY COMPLETE',2X,Z4)
1  166   8050 FORMAT(1H ,'ENTER FREQ, NADC, AND NCM (F6.0,2I5)'/
1  167       1 1H ,'(0.0038 <= FREQ <= 977.0',
1  168       2'   64 <= NADC <= 1024',
1  169       3'    4 <= NCM <= 1024)')
1  170   8055 FORMAT(F6.0,2I5)
1  171   8056 FORMAT(1H ,' FREQUENCY =',F9.4,' (',E14.7,
1  172       1')    NADC =',I5,'    NCM =',I5)
1  173   8060 FORMAT(1H ,'ENTER CAMAC CONTROL FUNCTION')
1  174   8065 FORMAT(A2)
1  175   8070 FORMAT(1H ,'DO YOU WANT DI AND/OR SI SET?')
1  176   8075 FORMAT(3A1)
1  177   8080 FORMAT(1H ,'ENTER - 0(ALL), 1(ADC TRIG), 2(PC=0), 3(C=M)')
1  178   8085 FORMAT(1H ,'ENTER - C(CLEAR), D(DISABLE), E(ENABLE)')
1  179   8090 FORMAT(1H ,'ILLEGAL -- - TRY AGAIN')
1  180   8100 FORMAT(1H ,'THE FUNCTIONS ARE:'/
1  181       1 1H ,'AD = ADVANCE PC'/
1  182       2 1H ,'CA = CAMAC CONTROL'/
1  183       3 1H ,'EX = EXIT'/
1  184       4 1H ,'HA = HALT GENERATOR'/
1  185       5 1H ,'LA = LAM CONTROL'/
1  186       6 1H ,'PA = INPUT PARAMETERS'/
1  187       7 1H ,'RU = START GENERATOR'/
1  188       8 1H ,'VE = VERIFY MEMORY'/
1  189       9 1H ,'WR = WRITE MEMORY')
1  190        END
```

```
2    1        SUBROUTINE RESETM(LREG,ISTAT)
2    2        INTEGER DATA,CL2
2    3        CL2=11
2    4        CALL CSSA(CL2,LREG,DATA,ISTAT,IERR)
2    5        IF(IERR.NE.0) THEN
2    6           WRITE(6,8     ISTAT
2    7        ENDIF
2    8        RETURN
2    9   8000 FORMAT(1H ,'RESET  ',2X,Z4)
2   10        END
```

```
3    1        FUNCTION INVOFS(K)
3    2  C...  COMPUTE INVERTED OFFSET BINARY OF K(12 BIT INTEGER)
3    3        IF(K.GE.0) THEN
3    4           INVOFS=2047-K
3    5        ELSE
3    6           INVOFS=-2049-K
3    7        ENDIF
3    8        RETURN
3    9        END
```