

MULTIANALYZER PROGRAMS

PULSE HEIGHT ANALYSIS

PDP-5 AND PDP-8

MULTIANALYZER PROGRAMS

PULSE HEIGHT ANALYSIS

PDP-5 AND PDP-8

Copyright 1965 by Digital Equipment Corporation

CONTENTS

	<u>Page</u>
INTRODUCTION	iv
ONE DIMENSIONAL DISPLAY AND ANALYSIS	1
Operation	1
Commands	1
Clock	3
Program Description	4
Page Zero Routines	6
Page One Routines	8
Page Two Routines	10
Page Three Routines	11
Page Four Routines	12
Page Five Routines	13
Page Six Routines	14
Program Listing	15
MULTIPARAMETER DISPLAY AND ANALYSIS	25
Operation	25
Commands	25
Sample Output	29
Data Taking	30
Clock	30
Program Description	30
Page Zero Routines	35
Page One Routines	36
Page Two Routines	37
Page Three Routines	38
Page Four Routines	39
Page Five Routines	40
Page Six Routines	41
Program Listing	42

INTRODUCTION

The PDP-5 and PDP-8 Multianalyzers are designed for very high speed data input and display. To facilitate the use of this equipment, two basic programs have been written, one for single parameter analysis and one for two parameter analysis. The specifications and operating instructions for these programs are enclosed.

The great strength of computer-based analyzer work is that the instrument's operation may be tailored to each experiment by simple alteration of the programs. The programs described herein are written in modular form to facilitate their alteration. A complete symbolic listing and a detailed description of the modular units that comprise each program are included.

As variations on these programs are written at Digital Equipment Corporation, they will be sent out to all users. User alterations are invited for submission to DECUS, the Digital Equipment Computer Users Society. Submissions should be addressed to the secretary of the society at Digital's address.

PROGRAM 1

ONE DIMENSIONAL DISPLAY AND ANALYSIS

The purpose of this program is to read in and analyze 1024-channel energy spectra data. The data is stored in locations 3000 to 4700 and 5000 to 6777 (octal), while the program is stored in locations 0001 to 1467 (octal).

OPERATION

The one dimensional pulse height analysis program receives and executes commands from the keyboard. These commands start and stop data taking and determine into which data region it goes, display the data with markers, allow areas of interest on the display screen to be expanded, integrate between markers, write out data, punch out data, and control background subtraction.

COMMANDS

There are eleven legal commands. Three of these commands ("zero," "subtract," "display") must be followed by the digit 1 or 2 to identify the appropriate data region. All commands except "expand," "increase lower marker," "decrease upper marker," and "integrate" are executed after the receipt of a carriage return (↵). All commands may be terminated by either striking the rubout key or by typing in a new command. A rubout causes the program to ignore anything previously typed on that line, rings the bell, and returns to wait for a command. Since the program interrupt is on during the execution of all commands involving the display, the rubout key can be used to terminate the command, or a new command may be typed. All illegal commands are ignored, the bell rings, and the program waits for a legal command. The legal commands are:

- N) Disable the data taking routine. Program control returns to wait for a command. When the program is first loaded, the data taking routine is enabled; thus, it is suggested that the first command given be an N.

- X.) Enable the data taking routine, reset the clock, and return to wait for a command. This command should be followed by the display command as most commands disable the data taking routine.
- Zd.) Zero the data region specified by the digit d (this digit should be either a 1 or 2). The data taking routine is disabled by this command. Control returns to wait for a command.
- Sd.) Subtract data region 2 from data region 1, and leave the result in the data region specified by the digit d. The data taking routine is disabled. Control returns to wait for a command.
- Dd.) Display the data region specified by the digit d with markers at channel 0000 and at channel 1023. All 1024 channels are displayed with a fourfold interleave to reduce flicker. When the $C(SR) = 0$, the display is 512 counts full scale. When the switches are set to 2, full scale is 1024 counts. With the switches set to 4, full scale is 2048 counts; with the switches set at 8, (0010_g) , full scale is 4096 counts. The interrupt is turned on and if the data taking routine is enabled, data is accepted and stored in the region being displayed. One of three conditions will stop the data taking process:
1. a clock overflow;
 2. a channel overflow; and
 3. the receipt of a keyboard command. The keyboard command might be N or any other command which disables the data taking routine.

Clock Overflow - When the clock (see below) has received the maximum number of counts, the message "CLOCK FULL" is typed, the data taking routine is disabled, and the display continues.

Register Overflow - When a channel has received 4096 counts, the channel number and the word "FULL" are typed, the data taking routine is disabled, and the display continues.

- Dd xxxx) Same as Dd) except that the lower marker is specified by the 4-digit decimal number typed. Note that the format of this command must be as specified. The upper marker is at 1023.
- Dd xxxx xxxx) Same as Dd xxxx) except that the upper marker is specified by the second 4-digit decimal number typed. Note that the format of this command must be as specified.
- E Expand the area of the display between the markers. Print the factor of expansion. This command disables the data taking routine. Note that the expanded display is not interleaved, and if the expansion factor is less than four, the display will flicker.
- W) Print out the number of counts in the channels specified by the markers. The left-hand column of the printout consists of channel numbers, while the ten columns on the right are the number of counts in that channel and the nine succeeding channels.
- P) Punch the data specified by the markers in BIN format. The data taking routine is disabled. The display continues.
- I) Integrate the curve between the markers (including the channel numbers represented by the markers) and print (decimal) the result. The data taking routine is disabled and the display continues.
- Ld) Increase the lower marker by the single digit d. Print out the new value; expand and print the expansion factor. The data taking routine is disabled.
- Ud) Decrease the upper marker by the single digit d. Print out the new value; expand and print the expansion factor. The data taking routine is disabled.

CLOCK

The clock sets the ADC flag and the clock flag, which is tested by a SNC instruction. This instruction is "skip if not clock." Thus, if the program is being run on a PDP-5 configuration

not equipped with the clock, SNC (6701) must be changed to an unconditional skip (7410). This is done by changing the register whose tag is ON (absolute address = 0153). The clock is a double precision counter which is reset by the command X) to the values found in the registers whose tags are CLCON1 and CLCON2 (absolute addresses = 0366, 0367). These registers may be changed by the operator of the program to give different counting times.

PROGRAM DESCRIPTION

The One Dimensional Display program is written in a somewhat modular form. The routines that handle the commands are fairly isolated so that they may be removed or altered. However, many of these routines use subroutines such as ORIGIN or PRINT1. These should not be removed unless all of the routines that they service are suitably altered. To conserve storage space, several of the routines use the same registers. For example, both of the routines SWEEP1 and INPUT4 use the register PRODCY; therefore, some care must be exercised when trying to change parts of the program.

The handling of the program interrupts deserves special mention. There are two states of the interrupt program:

1. Accept data from the analog-to-digital converter, and ignore interrupts from the teleprinter.
2. Ignore interrupts from the analog-to-digital converter, and handle interrupts from the teleprinter.

The state of the interrupt handling routine is determined by the contents of location 0002, the entrance point after the occurrence of an interrupt acknowledge. If the program is set to ignore ADC data, location 0002 will contain a jump to a routine that clears the ADC flag and tests for either the teleprinter flag or the keyboard flag. When the program is set to accept data, location 0002 contains an IOT instruction (6701), skip on not clock. This tests to see if the clock caused the interrupt. If not, the ADC flag and the keyboard flags are tested and handled accordingly.

When interrupts are handled in this fashion, the program interrupt can be on at all times when the program is not waiting for a command. In this version of the program, data cannot be

accepted when the display is expanded. This is due to the fact that the expansion factor is printed with the PNTDEC subroutine which uses the interrupt printing routine. If the printing of the expansion factor were eliminated, the state of the interrupt program could be left unchanged.

It is believed that this method of handling the program interrupts is one of the most efficient for this type of application. It results in a minimum "dead-time" for the analog-to-digital converter.

Page Zero Routines

- READ:** Read one character from the keyboard and print the same character. If the character is a rubout (377), return to BEGIN (0200). This rings the bell and starts the command search over again. If the character typed in is not a rubout, the subroutine exists with C(AC)=0 and with the character typed in register HOLD1 (0116). Both the keyboard flag and the teleprinter flag are cleared.
- PRINT1:** Prints the contents of the accumulator. Exists with the teleprinter flag cleared and with C(AC)=0.
- INTER:** The entrance point of the program interrupt. If the data taking routine is enabled, the first instruction (0002) will be SNC (6701). If the clock flag has caused the interrupt, program sequence goes to SETCLK (0020) which indexes the clock (double precision) and returns to the interrupted program via CONTIN+1 (0013). If the clock has received the maximum number of counts, program sequence goes to CKFULL by way of CLKFUL which is a pointer. If the clock flag did not originate the interrupt, the analog-to-digital converter flag is tested. If the ADC flag caused the interrupt, C(AC) are stored in SAVEAC and the converter is read and reset. The proper data region is determined by adding the contents of SETTING. Sequence goes to ADC which increments the appropriate register and returns to the interrupted program via CONTIN. If there is an overflow, program sequence goes to OVRFLO by means of FULL, which is a pointer. If neither the clock nor the ADC flag caused the interrupt, the program jumps to OTHER.
- OTHER:** Clears the teleprinter flag and tests for the keyboard flag. If found, the program jumps to CMMAND by means of RETURN+1. If not, control returns to the interrupted program by means of CONTIN.
- CONTIN:** Restores the accumulator from SAVEAC, turns on the program interrupt and returns to the program which was interrupted.

SETCLK: See INTER.

ADC: See INTER.

TTINTR: Used when the data taking routine is disabled. This is accomplished by putting (JMP TTINTR) in location 0002. TTINTR starts by clearing the ADC flag and testing for the teleprinter flag. If found, sequence returns to the program which originated the type command (TLS). If the interrupt was not caused by the teleprinter flag, the keyboard flag is tested by OTHER+1.

PRNTR: Is used for printing with the interrupt turned on. It initiates printing (TLS), clears the AC and returns to the display program which was last on by means of CONTIN.

Page One Routines

- BEGIN:** Rings the bell, prints CR, LF, and goes to CMMAND.
- CMMAND:** This routine looks for and interprets commands for the program to execute. It does this by keeping two tables, one of the commands and one of the addresses of the commands. If no legal command is found, control returns to BEGIN.
- TABLE1:** The table of the addresses of the entrance points for the commands.
- WRITE:** This is the routine that prints the contents of the various channels. It starts with a JMS READ which looks for the carriage return that tells it to proceed. It then disables the data taking routing by storing the word (JMP TTINTR) in location 0002. It picks up the limits for the printout from registers LOMARK and HIMARK. GO1 prints out this channel number, and GO2 prints out the contents of that channel and the nine succeeding ones. The routine terminates when the contents of the last channel have been printed. The sequence then proceeds to RE1. All printing in this routine is done by using the routine PNTDEC which is a binary-to-decimal conversion and printout. This uses the program interrupt. PNTDEC first must be set to convert 4 decimal digits. This is done by the instructions preceding GO1.
- RE1:** Prints a carriage return and a line feed using the program interrupt. Sequence remains in the main display loop.
- ENABLE:** Accepts a carriage return, puts the instruction SNC in location 0002, resets the clock, prints a line feed, and returns to CMMAND.
- OVRFLO:** Sequence goes here when a register has overflowed. This routine disables the data taking routine, computes the channel number that overflowed, prints it, and goes to PRFULL which prints the word FULL.
- CLCON1:** This is the low order word to which the clock is reset. This may be varied by the operator with the deposit key.

CLCON2: This is the high order word to which the clock is reset. This may be varied by the operator with the deposit key.

CLCON1 and CLCON2 are set to the double precision complement of the number of clock pulses to be accepted.

Page Two Routines

- SWEEP1:** This is the beginning of the major display loop. This loop has several parameters which change depending on the type of display that is shown.
- SCALE:** This closed routine puts scale markers on the display.
- MARK:** This routine puts the vertical markers on the display.
- LOWER:** This routine is entered when the command Ld is interpreted by CMMAND. It reads the digit d, adds it to the current value of the lower marker and goes to PMARK.
- UPPER:** This routine is entered when the command Ud is interpreted by CMMAND. It reads the digit d, subtracts it from the current value of the upper marker, and goes to PMARK.
- PMARK:** This is used by both UPPER and LOWER to print the current value of the marker and return control to EXPAND.

Page Three Routines

- CLDATA:** This routine goes to **ORIGIN** which determines the appropriate data region by setting the contents of the register **SETTING** to either 3000 or 5000. This routine then clears 2000₈ registers starting at the location specified by **SETTING**.
- PNTDEC:** This routine converts binary numbers to decimal and prints the results using the program interrupt. It first must be set to convert either double precision numbers (for the **INTEG** routine) or single precision.
- ADD:** A routine used by **PNTDEC**. It is basically a double precision addition.
- SBTRCT:** This routine is entered when the command **S** is interpreted. It goes to **ORIGIN** and then subtracts data region 2 from data region 1, leaving the result in the region specified by the contents of **SETTING**. Note that if the difference is negative, it is set to 0000.
- INTEG:** This routine is entered when the command **I** is interpreted. It adds the contents of the registers specified by the contents of **LOMARK** and **HIMARK** (the markers) and then prints out the total. Note that this is a double precision operation. **PNTDEC** is set for double precision. The integration includes the markers.

Page Four Routines

- DSPLAY:** This is the routine that is entered when the command D is interpreted. It first sets the markers to 0000 and 1023 and then accepts the origin setting from ORIGIN. If the next character is a Δ , control proceeds; if not, control transfers to INPUT4 which reads in a 4-digit decimal number and stores it in LOMARK. If the next character is a CR, control proceeds, if not, the next 4-digit decimal number is read in and stored in HIMARK. An LF is printed, and the parameters are set for the display SWEEP1. These parameters consist of the interleaving factors (INLEAV, XCONST, YCONST, COUNT8), the marker and scale factors, (MASKX, DSPLCE, MINLO, MINHI), and the address of the first point to be displayed (YORG).
- INPUT4:** Reads four decimal digits and converts them to a binary word which is in the AC when the routine exits.
- PRFULL:** Prints the word "FULL."
- CKFULL:** Disables the data taking routine and prints the words "CLOCK FULL."
- LEADIN:** Punches leader-trailer code for the punch routine.
- ORIGIN:** Reads in one digit. If the digit is a 1, it exits with the contents of SETTING = 3000. If the digit is a 2, it exits with the contents of SETTING = 5000.
- DSABLE:** Disables the data taking routine.

Page Five Routines

- EXPAND:** This routine expands the display when requested either by the command E or by either of the commands U, L. It operates by taking the difference between the markers and multiplying by 2 until the number of points is greater than 1024. The routine then expands the distance between the scale markers, recenters the display, displaces the scale markers if necessary, sets all of the operating parameters of the routine DSPLAY, prints the expansion factor, and then goes to DSPLAY. The data taking routine is disabled.
- SHIFT:** This changes all of the operating parameters for the SWEEP1 routine. It is a part of the EXPAND routine.
- STOP:** When shifting has gone far enough, sequence goes to STOP which sets all the parameters for the display, calls PNTDEC to print the expansion factor, and goes to SWEEP1.

Page Six Routines

- PUNCH:** Turns off the data taking routine and punches, in binary format, the contents of the specified areas of the data region. PUNCH calls the routine LEADIN to punch leader-trailer, BINPCH to punch the 12-bit word.
- BINPCH:** Punch the 12-bit word in the AC in binary format. This subroutine calls CHECK.
- CHECK:** Computes the checksum which is the last thing punched on the tape before the trailer code is punched.
- MEANSD:** This is an expandable command. Sequence transfers here when the key "M" is struck. At the present, it causes an error indication; however, any routine could be inserted here.

ONE DIMENSIONAL DISPLAY PROGRAM LISTING

ORGZ 1		PRINT1: 0
	SWEEP1	TLS
INTER:	SNC	TSF
	JMP SETCLK	JMP *-1
	ASF	TCF
	JMP OTHER	CLA
	DCA SAVEAC	JMP @PRINT1
	ARR	MIN10: 7770
	TAD SETTING	MIN20: 7760
	JMP ADC	EQUAL: 275
CONTIN:	TAD SAVEAC	ZERO: 260
	ION	SPACE: 240
	JMP @1	LNFEED: 0212
INDEX1:	0	CARRTN: 0215
INDEX2:	0	PE: 0305
INDEX3:	0	SKIP: 7410
SETCLK:	ACF	M4HUN: 7400
	ISZ CLOCK1	M1THOU: 7000
	JMP CONTIN+1	M2THOU: 6000
	ISZ CLOCK2	P1THOU: 1000
	JMP CONTIN+1	P2THOU: 2000
	JMP @CLKFUL	P3THOU: 3000
OTHER:	TCF	MINUS4: 7774
	KSF	MRBOUT: 7401
	SKP	PLUS4: 0004
	JMP @RETURN+1	MASK15: 0017
	JMP CONTIN+1	MASK63: 0077
READ:	0	MASKX: 0
	CLA	MINHI: 0
	KSF	MINLO: 0
	JMP *-1	LOMARK: 0
	KRB	HIMARK: 0
	TLS	HOLD1: 0
	TSF	HOLD2: 0
	JMP *-1	HIGH: 0
	TCF	LOW: 0
	DCA HOLD1	COUNT1: 0
	TAD HOLD1	COUNT2: 0
	TAD MRBOUT	COUNT3: 0
	SNA:CLA	COUNT4: 0
	JMP @ERROR	COUNT5: 0
	JMP @READ	COUNT6: 0
ADC:	DCA HOLD2	COUNT7: 0
	ISZ @HOLD2	COUNT8: 0
	JMP CONTIN	SETTING: 0
	JMP @FULL	

ONE DIMENSIONAL DISPLAY PROGRAM LISTING (continued)

CONST6: 0	ORG 200
CONST7: 0	BEGIN: KCC
CONST8: 0	TCF
INLEAV: 0	TAD BELL
PRODC: 0	JMS PRINT1
YCONST: 0	TAD CARRTN
YORG: 0	JMS PRINT1
XCONST: 0	TAD LNFEED
XAXIS: 0	JMS PRINT1
DSPLCE: 0	CMMAND: JMS READ
SAVEAC: 0	TAD ADRSC3
CLOCK1: 0	DCA INDEX3
CLOCK2: 0	TAD TABLE2+2
CLKFUL: CKFULL	DCA COUNT1
ADRSC6: TABLE3+7	TAD ADRSC1
ADRSC7: TABLE3-1	DCA HOLD2
ON: SNC	TAD HOLD1
OFF: JMP TTINTR	TAD @INDEX3
ERROR: BEGIN	SNA
EXPPT: EXPAND	JMP .+5
FULL: OVRFLO	ISZ HOLD2
RESUME: RE1	ISZ COUNT1
OUTPUT: PNTDEC	JMP .-5
ORGPT: ORIGIN	JMP BEGIN
RETURN: CMMAND-2	TAD @HOLD2
CMMAND	DCA HOLD2
	JMP @HOLD2
PRNTIR: 0	ADRSC3: TABLE2-1
TLS	TABLE2: 7474
CLA	7766
JMP CONTIN	7764
TTINTR: ACF	0003
TSF	0007
JMP OTHER+1	0013
	7762
DCA SAVEAC	0012
TCF	7761
JMP @PRNTIR	0013
	7770
	0011

ONE DIMENSIONAL DISPLAY PROGRAM LISTING (continued)

```
BELL:      0207
ADRSC1:    TABLE1
TABLE1:    DISPLAY
           DSABLE
           CLDATA
           WRITE
           PUNCH
           EXPAND
           SBTRCT
           INTEG
           ENABLE
           MEANSD
           UPPER
           LOWER

WRITE:     JMS READ
           TAD OFF
           DCA INTER
           TAD LOMARK
           DCA COUNT5
           TAD LNFEED
           JMS PRNTIR
           TAD HIMARK
           CMA:IAC
           DCA COUNT6
           TAD MINUS4
           DCA CONST6
           TAD ADRSC6
           DCA CONST7
           DCA HIGH

G01:      TAD TABLE2+1
           DCA COUNT7
           TAD COUNT5
           DCA LOW
           JMS @OUTPUT

G02:      TAD COUNT5
           TAD SETTNG
           DCA LOW
           TAD @LOW
           DCA LOW
           JMS @OUTPUT
           TAD COUNT6
           TAD COUNT5
           SNA:CLA
           JMP RE1
           ISZ COUNT5
           ISZ COUNT7
           JMP G02
           TAD CARRTN
           JMS PRNTIR
           TAD LNFEED
           JMS PRNTIR
           JMP G01

RE1:      TAD CARRTN
           JMS PRNTIR
           TAD LNFEED
           JMS PRNTIR
           JMP CONTIN

ENABLE:    JMS READ
           TAD ON
           DCA INTER
           TAD CLCON1
           DCA CLOCK1
           TAD CLCON2
           DCA CLOCK2
           JMP @RETURN

OVRFLO:   TAD OFF
           DCA INTER
           TAD SETTNG
           CMA:IAC
           TAD HOLD2
           DCA LOW
           DCA HIGH
           TAD MINUS4
           DCA CONST6
           TAD ADRSC6
           DCA CONST7
           JMS @OUTPUT
           JMP @.+1
           PRFULL

CLCON1:   0
CLCON2:   0
```

ONE DIMENSIONAL DISPLAY PROGRAM LISTING (continued)

```

ORG 400
SWEEP1:  TAD SKIP
          DCA SWIT5
          ION
          TAD SKIP3
          DCA SWIT6
          TAD M1THOU
          DCA SWIT3
          CLA!OSR
          CLL!RTR
          SNL
          JMP .+5
          CLA
          TAD SKIP
          DCA SWIT3
          JMP .+15
          RAR
          SNL
          JMP .+5
          CLA
          TAD M1THOU
          DCA SWIT5
          JMP .+6
          RAR
          SNL!CLA
          JMP .+3
          TAD M1THOU
          DCA SWIT6
          DCA COUNT3
          TAD CONST8
          DCA COUNT1
SWEEP2:  TAD INLEAV
          DCA COUNT2
          TAD YORG
          TAD COUNT3
          DCA PRODC
          TAD COUNT3
          DCA XAXIS

SWEEP3:  TAD XAXIS
          DXL
          TAD DSPLCE
          AND MASKX
          SNA!CLA
          JMP SCALE
          TAD XAXIS
          TAD MINLO
          SNA
          JMP MARK-3
          TAD MINHI
          SNA!CLA
          JMP MARK-3
          TAD SKIP
          DCA SWIT4
          DYS
          TAD XAXIS
          TAD XCONST
          DCA XAXIS
          TAD @PRODC

SWIT3:   NOP
          CLL!RAL

SWIT5:   SKP
          CLL!RTR

SWIT6:   JMP .+3
          RAR
          CLL!RTR
          DYS

SWIT4:   SKP
          JMP MARK
          CLA
          TAD PRODC
          TAD YCONST
          DCA PRODC
          ISZ COUNT2
          JMP SWEEP3
          ISZ COUNT3
          ISZ COUNT1
          JMP SWEEP2
          JMP SWEEP1

SKIP3:   JMP SWIT4-1

```

ONE DIMENSIONAL DISPLAY PROGRAM LISTING (continued)

SCALE:	CLL:IAC	ORG 600
	RTL	CLDATA: JMS @ORGPT
	DYS	JMS READ
	RTL	CMA
	DYS	TAD SETTING
	CLA	DCA INDEX1
	JMP SWEEP3+6	TAD OFF
	TAD M1THOU	DCA INTER
	DCA SWIT4	TAD M2THOU
	JMP SWIT3-5	DCA COUNT1
MARK:	CLA	DCA @INDEX1
	TAD P2THOU	ISZ COUNT1
	CLL	JMP .-2
	TAD MIN20	JMP @RETURN
	SNL	
	JMP SWIT4+2	PNTDEC: 0
	DYS	TAD CONST6
	JMP .-5	DCA COUNT8
LOWER:	JMS READ	TAD CONST7
	TAD HOLD1	DCA INDEX2
	AND MASK15	TAD @INDEX2
	TAD LOMARK	DCA HSUB
	DCA LOMARK	TAD @INDEX2
	TAD LOMARK	DCA LSUB
	JMP PMARK	DCA CHRCTR
UPPER:	JMS READ	JMS ADD
	TAD HOLD1	TAD CHRCTR
	AND MASK15	TAD ZERO
	CMA:IAC	JMS PRNTIR
	TAD HIMARK	ISZ COUNT8
	DCA HIMARK	JMP .-12
	TAD HIMARK	TAD MINUS4
PMARK:	DCA LOW	IAC
	DCA HIGH	DCA COUNT8
	TAD OFF	ISZ COUNT8
	DCA INTER	SKP
	TAD MINUS4	JMP @PNTDEC
	DCA CONST6	TAD SPACE
	TAD ADRSC6	JMS PRNTIR
	DCA CONST7	JMP .-5
	TAD EQUAL	
	JMS PRNTIR	
	JMS @OUTPUT	
	TAD CARRTN	
	JMS PRNTIR	
	TAD LNFEED	
	JMS PRNTIR	
	TAD PE	
	JMS PRNTIR	
	JMP @EXPPT	

ONE DIMENSIONAL DISPLAY PROGRAM LISTING (continued)

```
ADD:      0
          CLL
          TAD LOW
          TAD LSUB
          DCA TEMPL
          RAL
          TAD HIGH
          TAD HSUB
          SNL
          JMP .+6
          DCA HIGH
          TAD TEMPL
          DCA LOW
          ISZ CHRCTR
          JMP ADD+1
          CLA
          JMP @ADD

HSUB:    0
LSUB:    0
TEMPL:   0
CHRCTR:  0
TABLE3: 3166
         4600
         7413
         6700
         7747
         4540
         7775
         4360
         7777
         6030
         7777
         7634
         7777
         7766
         7777
         7777

SBTRCT:  JMS @ORGPT
          CMA
          TAD SETNG
          DCA INDEX1
          CMA
          TAD P3THOU
          DCA INDEX2
          TAD OFF
          DCA INTER
          TAD P3THOU
          CMA
          DCA INDEX3
          JMS READ
          TAD M2THOU
          DCA COUNT1
          CLL
          TAD @INDEX3
          CMA! IAC
          TAD @INDEX2
          SNL
          CLA
```

ONE DIMENSIONAL DISPLAY PROGRAM LISTING (continued)

INTEG: DCA @INDEX1
ISZ COUNT1
JMP .-10
JMP @RETURN
TAD OFF
DCA INTER
CMA
TAD SETTNG
TAD LOMARK
DCA INDEX2
DCA HIGH
TAD EQUAL
JMS PRNTIR
TAD HIMARK
CMA
TAD LOMARK
DCA COUNT8
CLL
TAD @INDEX2
SZL
ISZ HIGH
ISZ COUNT8
JMP .-5
DCA LOW
TAD MIN10
DCA CONST6
TAD ADRSC7
DCA CONST7
JMS PNTDEC
JMP @RESUME

ORG 1000
DSPLAY: TAD ADRSC2
DCA COUNT2
DCA @COUNT2
ISZ COUNT2
TAD M2THOU
CMA
DCA @COUNT2
TAD ADRSC2
DCA COUNT2
JMS @ORGPT
JMS READ
TAD HOLD1
TAD MINCR
SNA:CLA
JMP .+5
JMS INPUT4
DCA @COUNT2
ISZ COUNT2
JMP .-10
TAD LNFEED
JMS PRINT1
TAD PLUS4
DCA XCONST
TAD SETTNG
DCA YORG
TAD PLUS4
DCA YCONST
TAD LOMARK
CMA:IAC
DCA MINLO
TAD HIMARK
CMA:IAC
TAD LOMARK
DCA MINHI
TAD M4HUN
DCA INLEAV
TAD MINUS4
DCA CONST8
TAD MASK63
DCA MASKX
DCA DSPLCE
JMP @DSPLPT

ADRSC2: LOMARK
MINCR: 7563
DSPLPT: SWEEP1

ONE DIMENSIONAL DISPLAY PROGRAM LISTING (continued)

```
INPUT4: 0
        TAD MINUS4
        DCA COUNT1
        DCA PRODC1
        JMS READ
        TAD HOLD1
        AND MASK15
        TAD PRODC1
        ISZ COUNT1
        SKP
        JMP @INPUT4
        CLL:RAL
        DCA PRODC1
        TAD PRODC1
        RTL
        TAD PRODC1
        DCA PRODC1
        JMP INPUT4+4
PRFULL: TAD ADRSC4
        DCA INDEX2
        JMP .+6
CKFULL: DCA SAVEAC
        TAD OFF
        DCA INTER
        TAD ADRSC5
        DCA INDEX2
        TAD @INDEX2
        SNA
        JMP .+3
        JMS PRNTIR
        JMP .-4
        JMP @RESUME

TABLE6: 0303
        0314
        0317
        0303
        0313
        0240
        0306
        0325
        0314
        0314
        0215
        0212
        0000
ADRSC4: TABLE6+4
ADRSC5: TABLE6-1

LEADIN: 0
        TAD COD200
        CMA:IAC
        DCA COUNT6
        TAD COD200
        JMS PRNTIR
        ISZ COUNT6
        JMP .-3
        JMP @LEADIN
COD200: 0200

ORIGIN: 0
        JMS READ
        TAD HOLD1
        AND MASK15
        CLL:RAR
        SNL:CLA
        IAC
        RTR
        TAD P1THOU
        DCA SETTING
        JMP @ORIGIN

DSABLE: JMS READ
        TAD OFF
        DCA INTER
        JMP @RETURN
```

ONE DIMENSIONAL DISPLAY PROGRAM LISTING (continued)

```

ORG 1200
EXPAND: TAD MASK63
        DCA MASKX
        TAD M2THOU
        DCA INLEAV
        CLA:IAC
        DCA XCONST
        TAD LOMARK
        CMA:IAC
        TAD HIMARK
        SMA
        JMP .+11
        CLA
        TAD LOMARK
        DCA HOLD1
        TAD HIMARK
        DCA LOMARK
        TAD HOLD1
        DCA HIMARK
        JMP .-14
        DCA MINHI
        TAD MASKX
        CMA
        DCA DSPLCE
        TAD LOMARK
        SNA
        JMP .+5
        TAD DSPLCE
        SMA
        JMP .-2
        CMA:IAC
        DCA DSPLCE
        TAD MINHI
        IAC
        CLL:RAR
        SZL
        IAC
        CLL:RAL
        DCA HOLD1
        TAD HOLD1
        DCA YCONST
SHIFT:  TAD YCONST
        CLL:RAL
        DCA TEMP
        TAD TEMP
        TAD M2THOU
        SMA:CLA
        JMP STOP
        TAD TEMP
        DCA YCONST
        TAD XCONST
        CLL:RAL
        DCA XCONST
        TAD MASKX
        CLL:CML
        RAL
        DCA MASKX
        TAD INLEAV
        CLL:CML
        RAR
        DCA INLEAV
        TAD MINHI
        CLL:RAL
        DCA MINHI
        TAD DSPLCE
        CLL:RAL
        DCA DSPLCE
        JMP SHIFT
STOP:   CMA
        DCA CONST8
        TAD INLEAV
        TAD HOLD1
        CLL:CML
        RAR
        TAD LOMARK
        TAD SETTING
        DCA YORG
        TAD M2THOU
        TAD YCONST
        CLL:CML
        RAR
        DCA MINLO
        TAD MINHI
        CMA:IAC
        DCA MINHI
        TAD DSPLCE
        CMA:IAC
        TAD MINLO
        DCA DSPLCE
        IAC
        DCA YCONST
        TAD MINUS4
        DCA CONST6
        TAD ADRSC6
        DCA CONST7
        TAD XCONST
        DCA LOW
        DCA HIGH
        TAD OFF
        DCA INTER
        JMS @OUTPUT
        JMP @RESUME
TEMP:   0

```

ONE DIMENSIONAL DISPLAY PROGRAM LISTING (continued)

```
ORG 1400
PUNCH: JMS READ
        TAD OFF
        DCA INTER
        TAD LNFEED
        JMS PRNTIR
        JMS @LEADER
        TAD LOMARK
        TAD SETTNG
        DCA COUNT5
        TAD HIMARK
        TAD SETTNG
        CMA:IAC
        DCA COUNT6
        DCA LOW
        TAD COUNT5
        CLL:CML
        JMS BINPCH
        TAD @COUNT5
        CLL
        JMS BINPCH
        TAD COUNT5
        TAD COUNT6
        SNA:CLA
        JMP .+3
        ISZ COUNT5
        JMP .-10
        CLL
        TAD LOW
        JMS BINPCH
        JMS @LEADER
        JMP @RESUME
LEADER: LEADIN

        BINPCH: 0
                DCA COUNT8
                TAD COUNT8
                RTR
                RTR
                RTR
                AND MASK7
                JMS CHECK
                JMS PRNTIR
                TAD COUNT8
                AND MASK6
                JMS CHECK
                JMS PRNTIR
                JMP @BINPCH
        MASK6: 0077
        MASK7: 0177
        CHECK: 0
                DCA COUNT7
                TAD COUNT7
                TAD LOW
                DCA LOW
                TAD COUNT7
                JMP @CHECK
        MEANSD: JMP @ERROR
        END
```

PROGRAM 2

MULTIPARAMETER DISPLAY AND ANALYSIS

The purpose of this program is to input and display two-parameter energy and spectra data. The data is stored in locations 1432 to 7631 (octal), while the program is stored in locations 0001 to 1430 (octal).

OPERATION

The multiparameter pulse height analysis program receives and executes commands from the keyboard. These commands start and stop data taking, control the displays, and control writing and punching of the data. The Multiparameter Display Program is completely flexible with respect to matrix shape. The only requirement is that the size (i.e. the total number of elements in the matrix) be less than 6200_8 (3200_{10}).

COMMANDS

There are 13 legal commands. All commands except for "write," "increase vertical marker," and "increase horizontal marker," are executed after the receipt of a carriage return. A rubout causes the program to ignore anything previously typed on that line, rings the bell, and returns to wait for a legal command. The rubout key can be used to terminate any command, or a new command may be typed. All illegal commands are ignored, the bell rings, and the program waits for a legal command.

Command	Interrupt	Data Taking Routine	Comment
N)	OFF	DISABLED	DISABLE the DATA TAKING routine. Program control returns to wait for a command.

Command	Interrupt	Data Taking Routine	Comment
X)	OFF	ENABLED	ENABLE the DATA TAKING routine, reset the clock, and return to wait for a command. Actual data taking begins with the next command that turns on the program interrupt.
I)	ON	UNCHANGED	This command displays the ISOMETRIC of the data. If the data taking routine is enabled, data will be accepted after the receipt of the carriage return. Full scale is 4096 counts when $C(SR) \neq 0$. When $C(SR) = 0$, full scale is 1024 counts. If the scope is equipped with the intensity option, the isometric display will be marked as specified by the three marker commands.
T)	ON	UNCHANGED	This command causes the data coming in to be displayed (TWINKLE BOX) in a 64 x 64 matrix, or in any other size specified by the operator (see program description). This allows visual integration of the nature of the data. Only those words whose value is less than the highest acceptable size are added into the data region. All others are rejected; however, they are displayed in the "twinkle" display. (Note: the data taking rate is considerably slower with this display than it is with the others. It is useful for aligning detectors, etc.)
Z)	OFF	UNCHANGED	ZERO the data region after the receipt of the carriage return. Control returns to wait for the next command.
C xxxx xxxx)	ON	UNCHANGED	This is the DIFFERENTIAL CONTOUR display with the lower cutoff level specified by the first 4-digit decimal number and with the higher cutoff level specified by the second 4-digit decimal number. This will display the data matrix, intensifying only those points having more counts than the lower level, and fewer counts than the upper level (i.e. those points for which

Command	Interrupt	Data Taking Routine	Comment
			LOW LEVEL < COUNT < HIGH LEVEL). The contour displays can have markers on them as specified by the MARK, change HORIZONTAL, and change VERTICAL commands.
C xxxx)	ON	UNCHANGED	This is the INTEGRAL CONTOUR display with the lower cutoff level specified by the 4-digit decimal number. The upper cutoff level is unchanged. This will display the data matrix, intensifying only those points which have more counts than the specified cutoff level (i.e. those points for which LOW LEVEL ≤ COUNT < HIGH LEVEL).
C)	ON	UNCHANGED	Returns display to CONTOUR. It does not change the cutoff levels set by previous contour commands.
M xx)	OFF	UNCHANGED	Set the HORIZONTAL MARKER as specified by the 2-digit decimal number. Program control returns to wait for a command.
M xx xx)	OFF	UNCHANGED	Set the HORIZONTAL MARKER as specified by the first 2-digit decimal number, and set the VERTICAL MARKER as specified by the second 2-digit decimal number. Program control returns to wait for a command.
H	ON	UNCHANGED	Increase the HORIZONTAL MARKER by one. This will change the markers on the isometric and contour displays, and will change the slice under observation.
V	ON	UNCHANGED	Same as H except that it affects the VERTICAL MARKER.
SH)	ON	UNCHANGED	SLICE HORIZONTAL. The slice is taken along the horizontal marker. This command used in conjunction with the "H" command allows successive cross sections of the data matrix to be examined.

Command	Interrupt	Data Taking Routine	Comment
SV) ↙	ON	UNCHANGED	SLICE VERTICAL. The slice is taken along the vertical marker.
P) ↙	ON	DISABLED	PUNCH the entire data region in binary format. The display that was on when the command was received remains on.
Wd	ON	DISABLED	WRITE SLICES of the data region. The most recent slice command determines whether horizontal or vertical slices are to be written. The single digit d determines the number of successive slices to be printed. If d is a 0, ten slices are printed. The first slice is displayed on the scope during the execution of this command. Printout includes the coordinates of the number of counts. (See Sample Output on the following page.)
R) ↙	OFF	UNCHANGED	REVERSE the axes. Does not reverse the axes on the twinkle display, but does on all others. Markers are horizontal or vertical with respect to the scope, not with respect to the orientation of the data matrix. Program control returns to wait for a command. Current orientation can be determined by giving a write command, as print format includes the coordinates of the points. This command allows areas of the isometric display to be seen that were previously obscured.

WO
M=0026, 0027, 0028, 0029, 0030, 0031, 0032, 0033, 0034, 0035
F=

0000	0421	0563	0792	0929	0668	0373	0161	0099	0087	0106
0001	0119	0233	0409	0510	0444	0181	0090	0039	0036	0029
0002	0173	0385	0794	1122	0909	0479	0181	0075	0053	0044
0003	0147	0389	0762	1061	0939	0449	0165	0056	0035	0023
0004	0149	0346	0728	1089	0901	0469	0159	0055	0027	0015
0005	0162	0338	0831	1116	0907	0503	0137	0049	0022	0013
0006	0134	0354	0822	1186	0976	0498	0160	0042	0025	0014
0007	0176	0380	0785	1156	1043	0536	0171	0055	0021	0017
0008	0143	0369	0839	1251	1046	0500	0141	0050	0026	0018
0009	0149	0386	0905	1328	1036	0555	0201	0060	0034	0009
0010	0176	0478	0905	1442	1179	0648	0203	0070	0027	0023
0011	0186	0431	0929	1382	1251	0647	0213	0064	0030	0028
0012	0173	0423	0956	1405	1131	0642	0213	0061	0017	0020
0013	0170	0341	0860	1179	1072	0570	0170	0061	0034	0019
0014	0159	0339	0781	1101	0992	0523	0182	0048	0022	0014
0015	0148	0351	0705	1078	0907	0447	0166	0049	0021	0015
0016	0111	0347	0684	1060	0879	0505	0161	0049	0013	0014
0017	0143	0333	0667	1091	0845	0497	0131	0043	0018	0009
0018	0167	0311	0692	1052	0898	0464	0153	0039	0007	0012
0019	0108	0310	0687	1035	0872	0446	0146	0038	0021	0012
0020	0105	0272	0584	0905	0747	0369	0118	0041	0014	0013
0021	0079	0230	0407	0616	0531	0251	0081	0030	0013	0008
0022	0049	0133	0279	0425	0373	0179	0064	0018	0010	0009
0023	0034	0098	0188	0291	0245	0108	0040	0012	0006	0006
0024	0035	0085	0148	0232	0186	0101	0029	0011	0011	0007
0025	0041	0085	0170	0228	0212	0113	0035	0011	0003	0006
0026	0041	0135	0240	0352	0306	0159	0041	0018	0005	0002
0027	0078	0201	0421	0621	0509	0297	0105	0029	0011	0007
0028	0185	0440	0820	1202	1081	0560	0175	0048	0023	0014
0029	0285	0690	1562	2193	1954	0956	0354	0089	0019	0022
0030	0442	1037	2306	3199	2879	1411	0461	0143	0054	0019
0031	0528	1204	2527	3656	3316	1710	0547	0165	0059	0015
0032	0365	0985	2056	2872	2521	1347	0392	0105	0043	0024
0033	0227	0554	1123	1632	1333	0761	0245	0075	0010	0012
0034	0104	0220	0485	0685	0609	0302	0086	0024	0007	0007
0035	0035	0096	0191	0253	0242	0124	0044	0006	0006	0001
0036	0013	0043	0084	0121	0087	0050	0014	0008	0001	0005
0037	0009	0015	0034	0056	0038	0024	0010	0007	0001	0001
0038	0009	0012	0022	0037	0037	0015	0010	0001	0000	0003
0039	0007	0012	0027	0038	0026	0015	0002	0001	0002	0000
0040	0003	0015	0017	0029	0033	0010	0002	0005	0003	0003
0041	0005	0009	0024	0029	0020	0019	0009	0002	0000	0002
0042	0002	0009	0020	0031	0022	0009	0008	0001	0000	0001
0043	0007	0014	0021	0031	0027	0010	0003	0002	0003	0000
0044	0006	0009	0027	0014	0029	0010	0006	0002	0000	0001
0045	0002	0010	0014	0034	0025	0007	0004	0000	0001	0000
0046	0002	0010	0017	0024	0026	0008	0006	0002	0002	0001
0047	0005	0010	0020	0025	0021	0006	0004	0003	0000	0001
0048	0005	0010	0021	0021	0020	0007	0002	0000	0000	0001
0049	0007	0004	0017	0017	0025	0009	0005	0000	0000	0001
0050	0005	0008	0019	0019	0016	0006	0002	0002	0001	0000
0051	0004	0007	0019	0014	0021	0007	0003	0002	0000	0000
0052	0003	0008	0014	0030	0022	0005	0003	0001	0001	0000
0053	0001	0005	0011	0014	0017	0007	0003	0001	0000	0001
0054	0003	0008	0012	0026	0021	0017	0006	0000	0000	0001
0055	0004	0011	0019	0023	0012	0008	0002	0001	0000	0000
0056	0003	0009	0012	0019	0022	0009	0004	0002	0003	0002
0057	0001	0004	0015	0022	0022	0010	0003	0001	0000	0001
0058	0004	0004	0010	0032	0013	0010	0007	0001	0000	0002
0059	0002	0014	0012	0021	0017	0009	0002	0000	0000	0002
0060	0004	0009	0017	0023	0020	0008	0007	0002	0001	0000
0061	0003	0006	0019	0015	0012	0008	0002	0000	0001	0001
0062	0002	0005	0007	0018	0010	0003	0007	0000	0001	0002
0063	0001	0006	0009	0015	0010	0003	0004	0000	0001	0000

Sample Output

DATA TAKING

Data is accepted by the program when two conditions are met:

1. The data taking routine is enabled by the command X_↓).
2. The program interrupt is turned on by one of the display commands.

Data taking is stopped by one of three conditions:

1. A command is given that either disables the data taking routine or turns off the program interrupt i.e.: X_↓), N_↓), M xx_↓), M xx xx_↓), Z_↓), Wd, P_↓), and R_↓).
2. The clock receives a preset number of counts.
3. A register receives 4096 counts. This last condition will cause the data taking routine to be disabled and the coordinates of the overflowed matrix element will be printed followed by the word "FULL."

CLOCK

The clock is a double precision counter that is reset to the values found in CLCON1 (0276) and in CLCON2 (0277). CLCON1 is the low order clock constant and CLCON2 is the high order clock constant. These constants are the double-precision 2's complement of the number of clock pulses to be counted. The command that enables the data taking routine also resets the clock to these values (the command X_↓)). During data taking, the clock is indexed each time the analog-to-digital converter generates a clock pulse. The number of clock pulses is proportional to the "live time" of the system. When the clock has received the maximum number of counts as determined by the clock constants, the message "CLOCK FULL" is printed and the data taking routine is disabled. The display continues.

PROGRAM DESCRIPTION

The general layout of the Two-Dimensional Display program is very similar to that of the One-Dimensional program. The program interrupts are handled in a similar manner, as are keyboard commands. This program is constructed in a modular form that allows routines to be easily removed. The entry point to each routine to be removed is merely changed to (JMP @ ERROR).

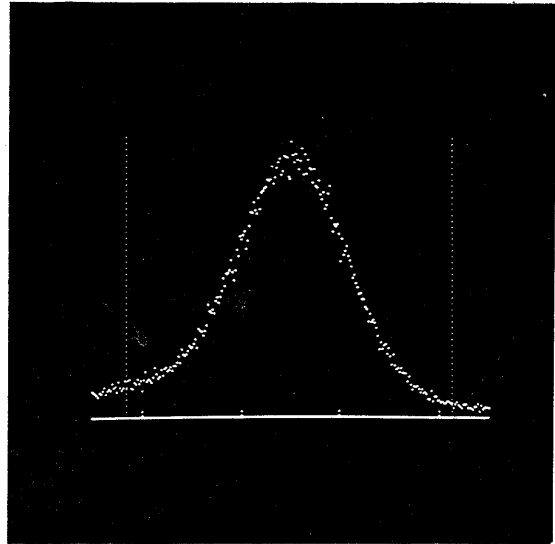
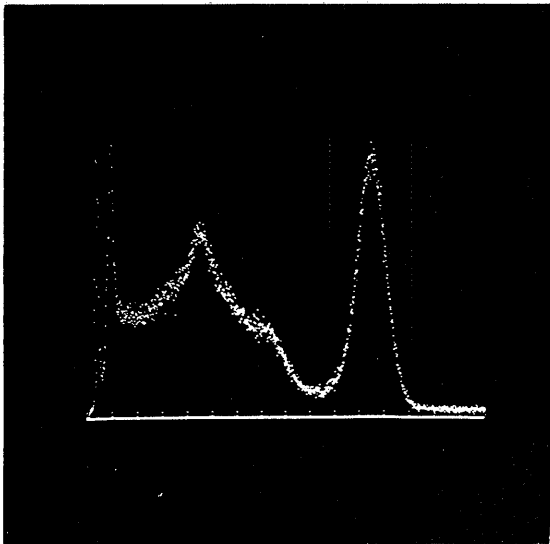
Some care must be exercised here since some routines call other routines for operating parameters. SLICE, CONTUR, and ISO all depend on HMARK and VMARK which specify, respectively, the position of the horizontal marker and the vertical marker. These markers are changed by the commands M, V, H. Thus, VERT and HORZ could readily be removed; but if MARK were removed, the display routines would have to be altered. In addition, WRITE uses parameters that are set up by the routine SLICE. Thus, SLICE can be removed only if WRITE is removed or altered. The PUNCH routines are entirely independent using only certain "universal" constants found on page zero.

There are four registers on page zero that control all displays, CONST1, CONST2, CONST3, and CONST4, which represent respectively:

1. Number of elements in Y direction
2. Number of elements in X direction
3. (Address X_i - Address X_{i+1}) for constant Y
4. (Address Y_i - Address Y_{i+1}) for constant X.

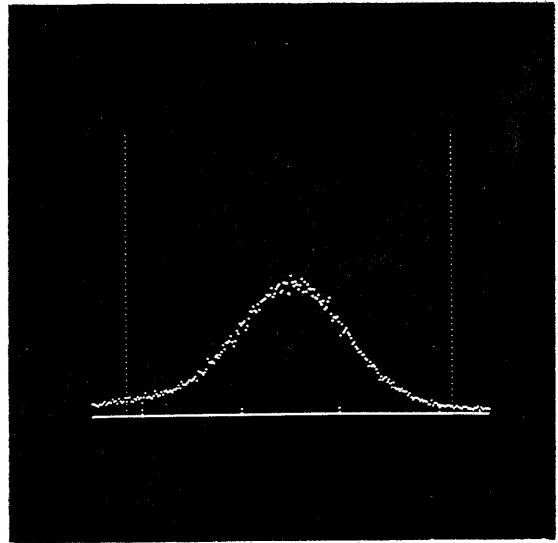
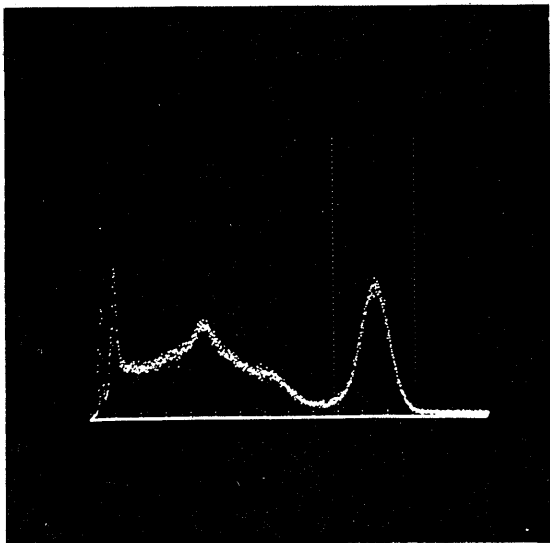
These four constants are initialized when the program is started at location 0200 (octal). The initial values depend on the contents of XSIZE and YSIZE. The contents of these registers (absolute address: 0167, 0170 respectively) are set by the operator to correspond to the particular experiment configuration. In order for the displays to represent data from the analog-to-digital converter, XSIZE must be an integral power of 2. In other words, the X coordinate corresponds to the rightmost bits from the analog-to-digital converter. Since storage space is limited to 6200_8 positions (3200_{10}), words from the converter that are larger than this value are rejected. Matrix reversal is accomplished by interchanging CONST1, CONST2, and CONST3, CONST4. In addition, CONST5, CONST6, are interchanged in order to keep track of the current orientation of the matrix.

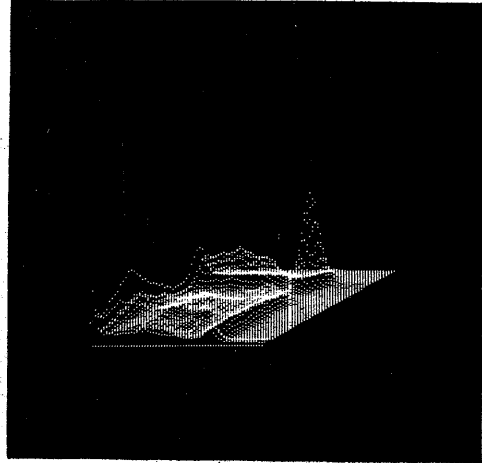
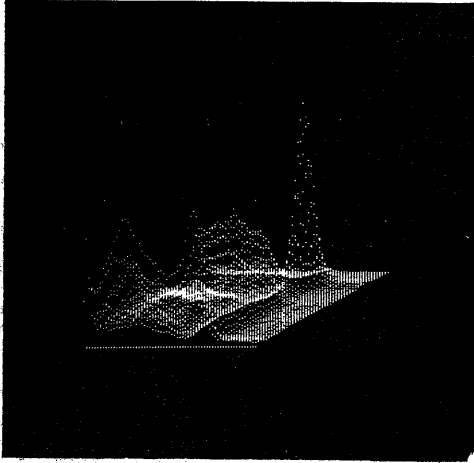
There are three other registers that control the displays. PLUS7 and PLUS4 contain, respectively, the X increase on the isometric display and the Y increase on the isometric display. For matrix sizes other than the one shown in the accompanying photographs, the values should be changed. Their absolute addresses are, respectively, 0171, 0172. The register PLUSX (absolute address = 0173) determines the spacing between points on the SLICE and CONTUR displays.



1024-CHANNEL SPECTRA

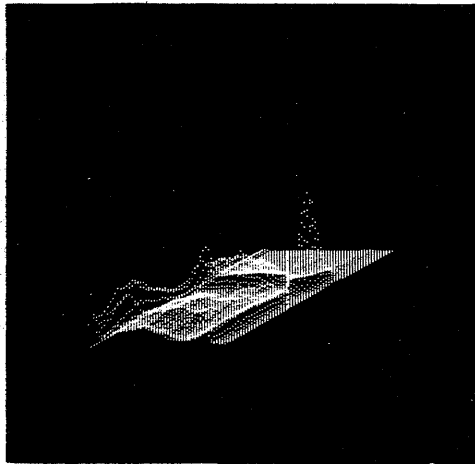
These views show markers and expansion capabilities. The small markers are at 64-channel intervals. The large markers can be set anywhere to single out regions for expansion. In the upper photos, the full-scale value is 2048 counts; below, 4096 counts.

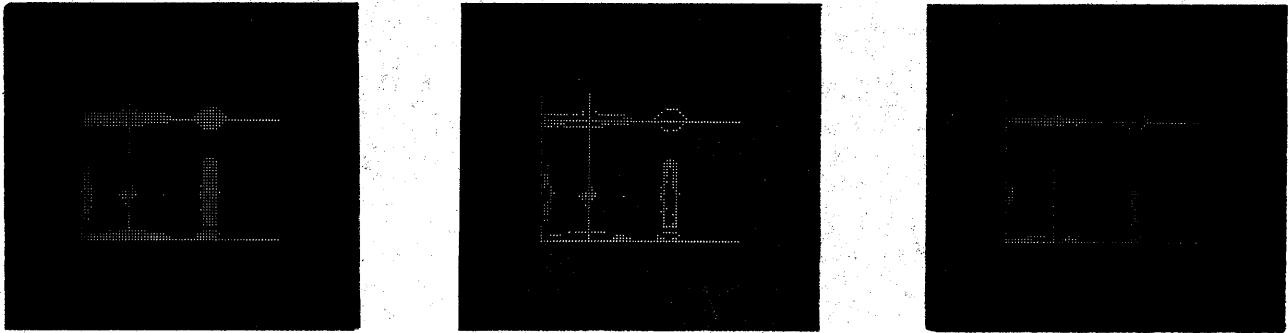




ISOMETRIC DISPLAYS

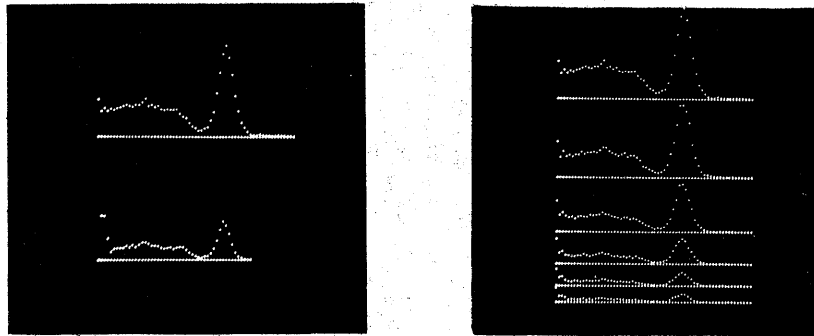
Above are two normal views of a 50 x 64 data matrix, a full-scale value of 2048 counts at the left and 4096 counts at the right. The third reverses the 4096-count view, making visible the areas masked behind ridges and peaks in the original.





CONTOURS AND SLICES

The three photos above show (from the left) an integral contour display of channels containing 320 or more counts, a differential contour display of channels containing 320 to 500 counts, and a differential contour display of channels containing 500 to 800 counts. Below are sectional views taken along the markers visible as straight lines in the contour displays. At the left are slices along the horizontal marker (top) and vertical marker. At the right are successive slices starting along the horizontal marker and moving toward the x-axis.



Page Zero Routines

- READ:** Read one character from the keyboard and print the same character. If the character is a rubout (377), return to BEGIN (0217). This rings the bell and starts the command search over again. If the character typed is not a rubout, the subroutine exists with C(AC) = 0 and with the character typed in register HOLD1 (0104). Both the keyboard flag and the teleprinter flag are cleared.
- INTER:** The same as for the One-Dimensional program with the exception that the address of the register to be incremented is determined by the contents of ORIGIN, the starting address of the data region. If the selected register is outside of the data matrix, it is ignored.
- OTHER:** Clears the teleprinter flag and tests for the keyboard flag. If found, the program jumps to CMMAND by means of RETURN. If not, control returns to the interrupted program by means of CONTIN.
- CONTIN:** Restores the accumulator from SAVEAC, turns on the program interrupt, and returns to the interrupted program.
- TTINTR:** Used when the data taking routine is disabled. This is accomplished by putting (JMP TTINTR) in location 0002. TTINTR starts by clearing the ADC flag and testing for the teleprinter flag. If found, sequence returns to the program which originated the type command. If the interrupt was not caused by the teleprinter, the keyboard flag is tested by OTHER+1.
- PRNTIR:** Is used for printing with the interrupt turned on. It initiates printing (TLS), clears the AC, and returns to the display program by means of CONTIN.
- TWSWIT:** The contents of this location determine whether or not the TWINKLE display is on. If C(TWSWIT) = NOP, control transfers to TWKLE. If C(TWSWIT) = SKP, control transfers to ADC.

Page One Routines

When the program is started at location 0200, the six constants, CONST1, CONST2, CONST3, CONST4, CONST5, CONST6, are initialized, a carriage return, line feed combination is printed, and the bell is rung. Control advanced to CMMAND which searches for and interprets legal commands.

- TABLE2: This is a table of the legal commands
- ENABLE: Accepts a carriage return, puts instruction (SNC = 6701) in location 0002, resets the clock, prints a line feed, and returns to CMMAND.
- CLCON1,
CLCON2: These are the low order and high order clock constants respectively (absolute addresses = 0276, 0277). These are set by the operator to the double precision 2's complement of the number of clock pulses to be accepted.
- SLICE: This is the slice routine. It accepts a character, (H or V) and displays the selected slice. The selected slice, i.e. horizontal or vertical, is with respect to the scope, not with respect to the current orientation of the matrix. Slice sets up some parameters which are used by WRITE. The contents of SSWIT determine whether the slice is horizontal or vertical. C(YORG) = starting address of slice location. C(DELTAY) = difference in addresses between successive slice elements.

Page Two Routines

- CONTUR:** This is the main contour display with markers.
- MARK:** The routine to set the horizontal and vertical markers. Calls the routine RDIN after setting it for a 2-digit input and conversion. Control returns to CMMAND.
- CLEAR:** Zeros the data region. Control returns to CMMAND.
- PRINT1:** Used for printing with interrupt turned off.

Page Three Routines

- ISO:** The entry point of the isometric display. Horizontal and vertical markers are displayed with increased intensity. For scopes without the intensity option, BRH, BRL = NOP.
- REVRSE:** Interchanges, by pairs, all constants controlling matrix orientation.
- TWINKL:** Entry point for the twinkle routine. Accepts a carriage return, types a line feed, sets TWSWIT for twinkle, turns interrupt on and waits for an ADC interrupt. When this occurs, control goes to TWKLE. Here, the word from the ADC is broken down into X and Y coordinates, and the corresponding point on the scope is intensified. If the word from the ADC is not too large, the appropriate core location is indexed.
- LNEND:** Types a carriage return and a line feed.

Page Four Routines

- WRITE:** The entry point for the printout routine. It looks at the parameters set up by SLICE, disables the data-taking routine, and proceeds to print the slice(s) selected. WRITE calls PNTDEC to convert the binary number in the AC to a decimal number.
- RDIN:** Converts the number of digits as specified by the C(AC) to a binary and exits with C(AC) = binary equivalent of decimal number typed in.
- TABLE6:** Is used for overflow printing.

Page Five Routines

- PNTDEC:** Converts the binary number in the AC to a 4-digit decimal number and prints it.
- FULL:** Is entered when a register has overflowed. It disables the data taking routine. When entered, C(HOLD1) = address of overflowed register. FULL converts this address to channel coordinates and prints this. It then prints the word "FULL."
- CKFULL:** Is entered when the clock has received the maximum number of counts. It disables the data taking routine, prints the word "CLOCK FULL," and returns to the display program that was on.
- PUNCH:** Punches the entire data matrix in BIN format. It calls LEADER to punch leader-trailer and BINPCH to punch a 12-bit word.
- BINPCH:** Punches a 12-bit word as two 6-bit words on tape and calls PCHECK.
- PCHECK:** Calculates the checksum which is the last word placed on the tape before the trailer code is punched.

Page Six Routines

- TABLE4: Used by PNTDEC
- TABLE1: Used by CMMAND. If routines are removed, this can be moved anyplace in core as can all of the routines on this page.
- VERT: Indexes vertical marker and returns to interrupted program.
- HORZ: Indexes horizontal marker and returns to interrupted program.
- DSABLE: Disables the data taking routine.

MULTIPARAMETER DISPLAY PROGRAM LISTING

```
ORGZ 2
INTER:  SNC                    ,OR JMP TTINTR
        JMP SETCLK
        ASF                    ,IS IT ADC?
        JMP OTHER
        DCA SAVEAC
        ARR
TWSWIT: NOP                    ,SWITCH TO DETERMINE IF TWINKLE
        JMP @TWNKLE            ,LOCATION INTER
        JMP ADC                ,IS SWITCH TO
CONTIN: TAD SAVEAC            ,DETERMINE STATE
        ION                    ,OF DATA HANDLING
        JMP @1
INDEX1: 0
INDEX2: 0
ADC:    CLL
        TAD ORIGIN
        DCA HOLD1
        TAD HOLD1
        TAD OVRCON
        SZL!CLA                ,INPUT TOO LARGE?
        JMP CONTIN            ,YES
        ISZ @HOLD1            ,NO OVERFLOW?
        JMP CONTIN            ,NO
        JMP @OVRFLO           ,YES
SETCLK: ACF                    ,CLEAR FLAG
        ISZ CLOCK1            ,INDEX CLOCK DOUBLE PRECISION
        JMP CONTIN+1
        ISZ CLOCK2                    ,CLOCK FULL?
        JMP CONTIN+1            ,NO
        JMP @CLKFUL            ,YES
TTINTR: ACF                    ,CLEAR FLAG
        TSF                    ,TELETYPE?
        JMP OTHER+1            ,NO
        DCA SAVEAC            ,YES
        TCF
        JMP @PRNTIR            ,RETURN TO PRINTING ROUTINE
OTHER:  TCF
        KSF                    ,KEYBOARD
        JMP CONTIN+1            ,NO
        JMP @RETURN            ,YES - GET COMMAND
PRNTIR: 0
        TLS
        CLA
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```

      JMP CONTIN
READ:  Ø           ,READ KEYBOARD
      CLA
      KSF
      JMP *-1
      KRB
      TLS
      TSF
      JMP *-1
      TCF
      DCA HOLD1
      TAD HOLD1
      TAD MRBOUT   ,IS CHARACTER A RUBOUT?
      SNA!CLA
      JMP @ERROR   ,YES
      JMP @READ    ,NO
MRBOUT: 74Ø1
RETURN: CMMAND
        CMMAND-2

HMARK:  Ø
VMARK:  Ø
MINUS4: 7774
MINUS2: 7776
HOLD1:  Ø
HOLD2:  Ø
COUNT1: Ø
COUNT2: Ø
COUNT3: Ø
COUNT4: Ø
COUNT5: Ø
XCOUNT:  Ø
YCOUNT: Ø
YORG:  Ø
DELTAY:  Ø
DELTAX:  Ø
CONST1: 7716
CONST2: 77ØØ
CONST3: ØØØ1
CONST4: Ø1ØØ
CONST5: Ø315
CONST6: Ø3Ø6
ORIGIN: 1432
XAXIS:  Ø
YAXIS:  Ø
OFF:    JMP TTINTR
ON:     SNC           ,SKIP IF NOT CLOCK 67Ø1
NOPERT: 7ØØØ
SKIP:   741Ø
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
ERROR: BEGIN
LNFFED: 0212
CARRTN: 0215
MINCR: 7563
CLOCK1: 0
CLOCK2: 0
MASK4: 0017
OVRFLO: FULL
CLKFUL: CKFULL
TWNKLE: TWKLE
OVRCON: 0146
MOVEX: 0
MOVEY: 0
CLCON: 1600
PRODC: 0
SAVEAC: 0
INPUT: RDIN
PRINT1: PRNT1
ADRSC1: TABLE1
ADRSC2: TABLE2-1
ADRSC4: HMARK-1
ADRSC6: TABLE6+4
ADRSC7: TABLE6-1
PRINT4: PNTDEC
PRM: 0315
PRF: 0306
XSIZE: 0100 ,X-DIMENSION
YSIZE: 0062 ,Y-DIMENSION
, (XSIZE)(YSIZE) MUST BE LESS THAN 6200 OCTAL
PLUS7: 7 , INCREASE IN X ON ISOMETRIC
PLUS4: 4 , INCREASE IN Y ON ISOMETRIC
, THESE DETERMINE SLOPE OF ISOMETRIC BASE
PLUSX: 0010 , SPACING CONSTANT FOR SLICE, CONTUR
MASK6: 0077 , MASKING CONSTANTS
MASK7: 7700 , FOR TWINKLE DISPLAY
LNEND: LINEND
PLUS12: 0012
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
ORG 200
      KCC                                ,INITIALIZE DISPLAY PARAMETERS
      TAD XSIZE
      DCA CONST4
      IAC
      DCA CONST3
      TAD XSIZE
      CMA:IAC
      DCA CONST2
      TAD YSIZE
      CMA:IAC
      DCA CONST1
      TAD PRF
      DCA CONST5
      TAD PRM
      DCA CONST6
BEGIN: CLA
      TAD BELL
      JMS @PRINT1
      TAD CARRTN
      JMS @PRINT1
      TAD LNFEED
      JMS @PRINT1
CMMAND: JMS READ                          ,DECODE COMMAND
      TAD ADRSC2
      DCA INDEX1
      TAD TABLE2+3
      DCA COUNT1
      TAD ADRSC1
      DCA HOLD2
      TAD HOLD1
      TAD @INDEX1
      SNA
      JMP .+5
      ISZ HOLD2
      ISZ COUNT1
      JMP .-5
      JMP BEGIN                            ,CHARACTER NOT ON LIST
      TAD @HOLD2
      DCA HOLD2
      JMP @HOLD2
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
TABLE2: 7452
        0016
        0005
        7763
        7766
        0021
        7766
        0006
        7777
        7766
        0001
        0003
        0002

BELL:   0207

ENABLE: JMS READ           ,ENABLE DATA TAKING ROUTINE
        TAD ON             ,INTERRUPT IS OFF
        DCA INTER         ,SET CLOCK
        TAD CLCON1
        DCA CLOCK1
        TAD CLCON2
        DCA CLOCK2
        JMP @RETURN+1

CLCON1: 0000             ,LOW ORDER CLOCK CONSTANT
CLCON2: 0000             ,HIGH ORDER CLOCK CONSTANT

SLICE:  JMS READ           ,THIS WHOLE ROUTINE MAY BE REMOVED
        TAD HOLD1         ,IF WRITE IS REMOVED. WRITE USES
        TAD TABLE2      ,SLICE TO DETERMINE REGION OF PRINT OUT

        SZA:CLA
        JMP .+3           ,REPLACE ROUTINE WITH SLICE: JMP @ERROR
        TAD NOPERT
        SKP
        TAD SKIP
        DCA SSWIT
        JMS READ
        TAD LNFEED
        JMS @PRINT1

SSWIT:  NOP
        JMP VSLICE       ,THE WRITE ROUTINE LOOKS AT THIS
        TAD HMARK       ,REGISTER TO SEE IF SLICE IS HORIZONTAL
        CMA
        DCA COUNT2      ,OR VERTICAL
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```

TAD CONST4
DCA YAXIS
TAD CONST2
DCA XCOUNT
TAD CONST3
DCA DELTAY
JMP SSTART
VSLICE: TAD VMARK
        CMA
        DCA COUNT2
        TAD CONST3
        DCA YAXIS
        TAD CONST1
        DCA XCOUNT
        TAD CONST4
        DCA DELTAY
SSTART: ISZ COUNT2
        SKP
        JMP .+3
        TAD YAXIS
        JMP .-4
        TAD ORIGIN
        DCA YAXIS
        TAD YAXIS
        DCA YORG
        DCA XAXIS
        ION
        BRL                                     ,FOR SCOPE WITH INTENSITY OPTION
PLOTS:  TAD XAXIS
        DXL
        TAD PLUSX
        DCA XAXIS
        DYS
        TAD @YAXIS
        CLL!RAR
        CLL!RTR
        DYS
        CLA
        TAD YAXIS
        TAD DELTAY
        DCA YAXIS
        ISZ XCOUNT
        JMP PLOTS
        JMP SSWIT                                     ,END OF SLICE ROUTINE
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
ORG 400
CONTUR: TAD ADRSC3           ,CONTUR CAN BE REMOVED
        DCA INDEX1         ,CHANGE TO CONTUR: JMP @ERROR
        JMS READ
        TAD HOLD1
        TAD MINCR
        SNA:CLA
        JMP .+5
        TAD MINUS4
        JMS @INPUT         ,GET LEVELS
        DCA @INDEX1
        JMP .-10
        TAD LNFEED
        JMS @PRINT1
        TAD HLEVEL
        CMA:IAC
        TAD LLEVEL
        DCA MHIGH
        TAD LLEVEL
        CMA
        DCA MLOW
        BRL                 ,FOR SCOPE WITH INTENSITY OPTION
CSTART: TAD VMARK
        CMA:IAC
        DCA COUNT4
        ION
        TAD ORIGIN
        DCA YORG
        TAD CONST2
        DCA XCOUNT
        DCA MOVEX
CG01:   TAD CONST1
        DCA YCOUNT
        TAD YORG
        DCA YAXIS
        TAD HMARK
        SZA
        CMA
        DCA COUNT3
        DCA MOVEY
        TAD MOVEX
        DXL
        SNA:CLA           ,X=0?
        JMP VMARKR       ,YES PUT IN AXIS
        ISZ COUNT4      ,NO-IS X AT MARKER?
        SKP              ,NO
        JMP VMARKR       ,YES
        DYS              ,PUT IN Y=0
        TAD NOPERT
        DCA CSWIT
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
CG02:  ISZ COUNT3      ,Y AT MARKER
        SKP
        JMP CG03      ,YES INTENSIFY
        CLL:IAC
        TAD @YAXIS
        TAD MLOW
        SNL
        JMP OUT
        TAD MHIGH
        SNL:CLA      ,INTENSIFY THIS POINT?
        JMP OUT      ,NO
CG03:  TAD MOVEY      ,YES
        DYS
OUT:    CLA
        ISZ YCOUNT
        SKP
        JMP CSTOP
        TAD MOVEY
        TAD PLUSX      ,PLUSX DETERMINES SPACING BETWEEN POINTS
        DCA MOVEY
        TAD YAXIS
        TAD CONST4
        DCA YAXIS
CSWIT:  NOP
        JMP CG02
        JMP CG03
VMARKR: TAD SKIP
        DCA CSWIT
        JMP CG03
CSTOP:  ISZ XCOUNT      ,SCANNED ALL X
        SKP
        JMP CSTART    ,YES
        TAD MOVEX      ,NO SET UP FOR NEXT X
        TAD PLUSX
        DCA MOVEX
        TAD YORG
        TAD CONST3
        DCA YORG
        JMP CG01      ,END OF CONTUR ROUTINE
MARK:  TAD ADRSC4      ,GET MARKERS
        DCA INDEX1
        JMS READ
        TAD HOLD1
        TAD MINCR
        SNA:CLA
        JMP .+5
        TAD MINUS2
        JMS @INPUT
        DCA @INDEX1
        JMP .-10
        JMP @RETURN+1
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

ADRSC3: LLEVEL-1
MLOW: 0
MHIGH: 0

CLEAR: JMS READ ,THIS ROUTINE CLEARS THE
TAD CLCON ,DATA REGION
DCA COUNT1
CMA
TAD ORIGIN
DCA INDEX1
DCA @INDEX1
ISZ COUNT1
JMP .-2
JMP @RETURN+1

PRNT1: 0
TLS
TSF
JMP .-1
TCF
CLA
JMP @PRNT1

LLEVEL: 0
HLEVEL: 0

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
ORG 600
ISO:   JMS READ                      ,SET SWITCH FOR ISO
      TAD SKIP                      ,THIS IS THE ONLY PLACE
      DCA TWSWIT                    ,WHERE THIS SWITCH IS SET
      TAD LNFEED
      JMS @PRINT1
ISOG01: TAD ORIGIN
      DCA YORG
      CLA:OSR                        ,SET UP SCALE
      SZA:CLA                        ,IF C(SR)=0,C(ISWIT2)=NOP
      TAD PLUS12                    ,IF C(SR)=0,C(ISWIT2)=RTR
      TAD NOPERT
      DCA ISWIT2
      DCA MOVEX
      DCA MOVEY
      DCA XAXIS
      TAD CONST1
      DCA YCOUNT
      ION
      TAD HMARK
      CMA
      DCA COUNT3
ISOG02: TAD YORG
      DCA YAXIS
      TAD CONST2
      DCA XCOUNT
      TAD VMARK
      CMA
      DCA COUNT4
      ISZ COUNT3
      JMP .+3
      TAD NOPERT
      SKP
      TAD SKIP3
      DCA ISWIT
ISOG03: ISZ COUNT4
ISWIT: JMP ISWIT+3
      BRH                            ,FOR SCOPE WITH INTENSITY OPTION
      SKP
      BRL
      TAD XAXIS
      DXL
      TAD PLUS7
      DCA XAXIS
      TAD @YAXIS
      CLL:RAR
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
ISWIT2: RTR
        TAD MOVEY
        DYS
        CLA
        TAD YAXIS
        TAD CONST3
        DCA YAXIS
        ISZ XCOUNT
        JMP ISOG03
        TAD MOVEY
        TAD PLUS4
        DCA MOVEY
        TAD MOVEX
        TAD PLUS7
        DCA MOVEX
        TAD MOVEX
        DCA XAXIS
        TAD YORG
        TAD CONST4
        DCA YORG
        ISZ YCOUNT
        JMP ISOG02
        JMP ISOG01
SKIP3:  JMP ISWIT+3
REVRSE: JMS READ
        TAD CONST1
        DCA HOLD1
        TAD CONST2
        DCA CONST1
        TAD HOLD1
        DCA CONST2
        TAD CONST3
        DCA HOLD1
        TAD CONST4
        DCA CONST3
        TAD HOLD1
        DCA CONST4
        TAD CONST5
        DCA HOLD1
        TAD CONST6
        DCA CONST5
        TAD HOLD1
        DCA CONST6
        JMP @RETURN+1
```

```
,REVRSE MAY BE REMOVED
,CHANGE TO REVRSE: JMP @ERROR
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
TWINKL: JMS READ
        TAD LNFEED
        JMS @PRINT1
        TAD NOPERT                ,SET TWINKLE SWITCH
        DCA TWSWIT                ,RESET ONLY BY ISOMETRIC ROUTINE
        ION
        NOP
        JMP .-1
TWKLE:  DCA HOLD1                ,GET X
        TAD MASK6
        AND HOLD1
        CLL:RAL
        RTL
        DXL
        CLA:CLL                  ,GET Y
        TAD MASK7
        AND HOLD1
        CLL:RAR
        RTR
        NOP                      ,SPACE FOR ADDITIONAL ROTATES WHEN
        DYS                      ,USING DIFFERENT EXPERIMENT CONFIGURATIONS
        CLA:CLL
        TAD HOLD1
        TAD ORIGIN
        DCA HOLD1
        TAD HOLD1
        TAD OVRCON
        SZL:CLA                  ,IS WORD TOO LARGE FOR ACCEPTANCE
        JMP CONTIN+1            ,YES
        ISZ @HOLD1              ,NO-OVERFLOW?
        JMP CONTIN+1            ,NO
        JMP @OVRFLO            ,YES
LINEND: 0
        TAD CARRTN
        JMS PRNTIR
        TAD LNFEED
        JMS PRNTIR
        JMP @LINEND
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
ORG 1000
WRITE:  JMS READ           ,THE ENTIRE WRITE ROUTINE
        TAD SCON          ,MAY BE REMOVED
        DCA INTER-1      ,CHANGE TO WRITE: JMP @ERROR
        TAD OFF
        DCA INTER
        TAD HOLD1        ,IF WRITE IS REMOVED SLICE
        AND MASK4        ,MAY BE REMOVED
        SNA
        TAD PLUS12
        CMA:IAC
        DCA HOLD1
        JMS @LNEND
        JMS SPACES
        JMS SPACES
        TAD SKIP
        CMA:IAC
        TAD @SCON
        SNA:CLA
        JMP WHORZ
        TAD VMARK
        DCA YCOUNT
        TAD CONST3
        DCA DELTAX
        TAD CONST5
        JMS PRNTIR
        TAD CONST6
        DCA MOVEY
        TAD CONST1
        DCA MOVEX
        JMP WSTART
WHORZ:  TAD HMARK
        DCA YCOUNT
        TAD CONST4
        DCA DELTAX
        TAD CONST6
        JMS PRNTIR
        TAD CONST5
        DCA MOVEY
        TAD CONST2
        DCA MOVEX
WSTART: TAD EQUAL
        JMS PRNTIR
        TAD HOLD1
        DCA COUNT1
        TAD YCOUNT
        JMS @PRINT4
        ISZ COUNT1
        SKP
        JMP WLAST
        TAD COMMA
        JMS PRNTIR
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
TAD SPACE
JMS PRNTIR
ISZ YCOUNT
JMP .-12
WLAST: JMS @LNEND
        TAD MOVEY
        JMS PRNTIR
        TAD EQUAL
        JMS PRNTIR
        JMS @LNEND
        TAD YORG
        DCA MOVEY
        DCA YCOUNT
AGAIN:  TAD YCOUNT
        JMS @PRINT4
        JMS SPACES
        TAD HOLD1
        DCA COUNT1
        TAD MOVEY
        DCA PRODCY
        TAD @PRODCY
        JMS @PRINT4
        ISZ COUNT1
        SKP
        JMP .+6
        TAD PRODCY
        TAD DELTAX
        DCA PRODCY
        JMS SPACES
        JMP .-11
        JMS @LNEND
        ISZ MOVEX
        SKP
        JMP CONTIN
        TAD MOVEY
        TAD DELTAY
        DCA MOVEY
        ISZ YCOUNT
        JMP AGAIN
SPACES: 0
        TAD SPACE
        JMS PRNTIR
        TAD SPACE
        JMS PRNTIR
        JMP @SPACES
SCON:   SSWIT      , POINTER TO SWITCH IN SLICE ROUTINE
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

COMMA: 0254
EQUAL: 0275
RDIN: 0
DCA COUNT1
DCA PRODC T
JMS READ
TAD HOLD1
AND MASK4
TAD PRODC T
ISZ COUNT1
SKP
JMP @RDIN
CLL:RAL
DCA PRODC T
TAD PRODC T
RTL
TAD PRODC T
DCA PRODC T
JMP RDIN+3

,CONVERT DECIMAL TO BINARY

TABLE6: 303
314
317
303
313
SPACE: 240
306
325
314
314
0

,TABLE FOR OVERFLOW PRINT OUTS

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
ORG 1200
PNTDEC: 0 ,CONVERT BINARY TO DECIMAL AND PRINT
      DCA HOLD2
      TAD MINUS4
      DCA COUNT5
      TAD ADRSC8
      DCA TEST
HERE:  TAD ZERO
      DCA CHAR
      TAD HOLD2
      CLL
      TAD @TEST
      SNL
      JMP .+3
      ISZ CHAR
      JMP .-5
      DCA HOLD2
      TAD @TEST
      CMA! IAC
      TAD HOLD2
      DCA HOLD2
      TAD CHAR
      JMS PRNTIR
      ISZ TEST
      ISZ COUNT5
      JMP HERE
      JMP @PNTDEC
ZERO:  0260
ADRSC8: TABLE4
TEST:  0

FULL:  TAD OFF ,CALCULATE LOCATION OF OVERFLOWED
      DCA INTER ,CHANNEL
      TAD ORIGIN ,USES THE ROUTINE PNTDEC
      CMA! IAC
      TAD HOLD1
      DCA HOLD1
      TAD PRM
      JMS PRNTIR
      TAD XSIZE
      CIA
      DCA TEST
      DCA CHAR
      TAD HOLD1
      TAD TEST
      SPA
      JMP .+3
      ISZ CHAR
      JMP .-4
      TAD XSIZE
      DCA HOLD1
      TAD CHAR
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
JMS PNTDEC
TAD PRF
JMS PRNTIR
TAD HOLD1
JMS PNTDEC
TAD ADRSC6           ,PRINT "FULL"
DCA INDEX1
JMP .+6

CKFULL: DCA SAVEAC           ,PRINT "CLOCK FULL"
TAD OFF
DCA INTER
TAD ADRSC7
DCA INDEX1
TAD @INDEX1
SNA
JMP .+3
JMS PRNTIR
JMP .-4
JMS @LNEND
JMP CONTIN

PUNCH: JMS READ           ,ENTIRE PUNCH ROUTINE MAY BE REMOVED
TAD OFF                ,ALONG WITH PCHECK,LEADER,AND BINPCH

DCA INTER
TAD LNFEED
JMS PRNTIR
JMS LEADER
CMA
TAD ORIGIN
DCA INDEX2
DCA CHAR
TAD CLCON
DCA COUNT5
TAD ORIGIN
CLL!CML
JMS BINPCH
TAD @INDEX2
CLL
JMS BINPCH
ISZ COUNT5
JMP .-4
TAD CHAR
CLL
JMS BINPCH
JMS LEADER
JMP CONTIN

BINPCH: Ø
DCA HOLD1
TAD HOLD1
RTR
RTR
RTR
```

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
      AND MASK9
      JMS PCHECK
      JMS PRNTIR
      TAD HOLD1
      AND MASK8
      JMS PRNTIR
      JMP @BINPCH
PCHECK: 0
      DCA HOLD2
      TAD HOLD2
      TAD CHAR
      DCA CHAR
      TAD HOLD2
      JMP @PCHECK
LEADER: 0
      TAD COD200
      CMA:IAC
      DCA HOLD2
      TAD COD200
      JMS PRNTIR
      ISZ HOLD2
      JMP .-3
      JMP @LEADER
```

```
COD200: 0200
MASK8:  0077
MASK9:  0177
CHAR:   0
ORG 1400
```

```
TABLE4: 6030
        7634
        7766
        7777
```

,CONVERSION TABLE USED BY PNTDEC

```
TABLE1: VERT
        HORZ
        CONTUR
        PUNCH
        CLEAR
        ISO
        SLICE
        MARK
        DSABLE
        ENABLE
        WRITE
        TWINKL
        REVRSE
```

,THIS TABLE MAY BE RELOCATED

MULTIPARAMETER DISPLAY PROGRAM LISTING (continued)

```
VERT:  ISZ VMARK           ,INDEX VERTICAL MARKER
        JMP CONTIN
HORZ:  ISZ HMARK           ,INDEX HORIZONTAL MARKER
        JMP CONTIN
DSABLE: JMS READ           ,DISABLE DATA TAKING ROUTINE
        TAD OFF           ,MAY BE MOVED
        DCA INTER
        JMP @RETURN+1
END
```

digital
EQUIPMENT
CORPORATION
MAYNARD, MASSACHUSETTS