

AD-A041 000

NAVAL RESEARCH LAB WASHINGTON D C  
A GENERAL PURPOSE MINI-COMPUTER BASED DIGITAL SIGNAL PROCESSING--ETC(U)  
MAY 77 T W HODGE, A P SKELTON

F/G 9/2

UNCLASSIFIED

NRL-MR-3502

NL

| OF |

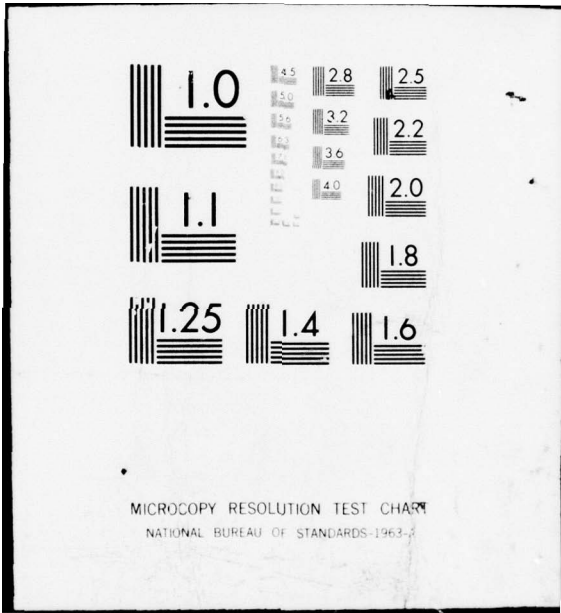
AD  
A041 000



END

DATE  
FILMED

7 - 77



12

NRL Memorandum Report 3502

ADA 041 000

# A General Purpose Mini-Computer Based Digital Signal Processing Laboratory

THEO W. HODGE

*Space Technology Branch  
Space Systems Division*

and

ANITA P. SKELTON

*Research Computation Center*

May 1977


DDDC  
JUN 29 1977  
MOSCOW



NAVAL RESEARCH LABORATORY  
Washington, D.C.

AD Nu. \_\_\_\_\_  
DDC FILE COPY

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 14 NRL Memorandum Report 3502	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) 6 A GENERAL PURPOSE MINI-COMPUTER BASED DIGITAL SIGNAL PROCESSING LABORATORY	5. TYPE OF REPORT & PERIOD COVERED Interim report on one phase of a continuing NRL problem.	
	6. PERFORMING ORG. REPORT NUMBER	
7. AUTHOR(s) 10 Theo W. Hodge and Anita P. Skelton	8. CONTRACT OR GRANT NUMBER(s) 16 17 XF-22-151-006	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, D.C. 20375	10. PROGRAM ELEMENT PROJECT, TASK AREA & WORK UNIT NUMBERS NRL Problem D03-17E 62712N XF12-151-006	
11. CONTROLLING OFFICE NAME AND ADDRESS Department of the Navy Naval Electronic Systems Command Washington, D.C. 20360	12. REPORT DATE 11 May 1977	13. NUMBER OF PAGES 31
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report) UNCLASSIFIED	
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE		
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) 251 950 Digital signal processing                      Varian Analog-to-digital controller                Versatec Triggerable digital delay line               Disk Computer programs                             Plotting Mini-computer		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The general design approach of hardware and software systems and the present performance capability of a digital signal processing laboratory is given. Block diagrams of the hardware and software systems are provided along with an example of a recent task. 		

DD FORM 1473  
1 JAN 73

EDITION OF 1 NOV 65 IS OBSOLETE  
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

hgg

CONTENTS

INTRODUCTION ..... 1  
HARDWARE SYSTEMS DESIGN ..... 1  
HARDWARE DESIGN ..... 4  
SOFTWARE AND PROGRAMMING CONSIDERATIONS ..... 9  
CONCLUSIONS ..... 13

ADDITIONAL TO

THIS  
DND  
UNCLASSIFIED  
IDENTIFICATION

White Section   
Red Section

BY DISTRIBUTION AVAILABILITY CODES

Dist. Avail. ORG/OF SPECIAL

A

Preceding Page BLANK - NOT FILMED

## A GENERAL PURPOSE MINI-COMPUTER BASED DIGITAL SIGNAL PROCESSING LABORATORY

### Introduction

A digital signal processing laboratory has recently begun operation in the Electronics Section, Space Technology Branch, Space Systems Division. This report details the general design approach of hardware and software systems and the present operating capability. Block diagrams of the hardware and software systems are provided along with an example of a recent task.

### Hardware Systems Design

The digital signal processing laboratory (Fig. 1, 2) is built around a Varian V-73 disk based mini-computer system.<sup>1</sup> This computer is a 16-bit machine with dual asynchronous Input/Output (I/O) busses and dual ported memory. It has I/O data transfer rates of up to 2.9 million words per second. Another feature of this machine is a 512 64-bit word Writable Control Store (WCS). The Disk has a 1.7 million 16-bit word capacity, expandable to 8 million words. The present computer memory size is 16K words.

Other key elements of the laboratory are: a 5 MHz 8-bit Computer Labs A-D/D-A converter sub-system, a phase-locked oscillator/synthesizer (PLO/SYN), a Tektronix 4012 graphics terminal, a Versatec 1600A high speed-high resolution electrostatic simultaneous printer/plotter, a Texas Instruments Silent 700 terminal, and a Remex high speed paper tape reader.

The original design goal for this digital signal processing system was the spectrum analysis on a continuous real-time basis of a 1.5 MHz receiver IF-BW with a Fast Fourier Transform (FFT) filter frequency resolution of 300 to 400 Hz. The first design therefore included a hardwired array processor with a 1.5 million word throughput rate. Sponsor induced program changes after the computer portion of this system was purchased necessitated system redesign to a multi-purpose signal processing laboratory.

The present system will digitize analog data of bandwidths of up to 100 KHz. The system has three phases, acquisition, pulse analysis and plotting. In the acquisition phase the system digitizes analog data, packs the data samples into computer words, reads the data into computer memory, unpacks the data and places the data on disk memory. During the

Note: Manuscript submitted April 22, 1977.

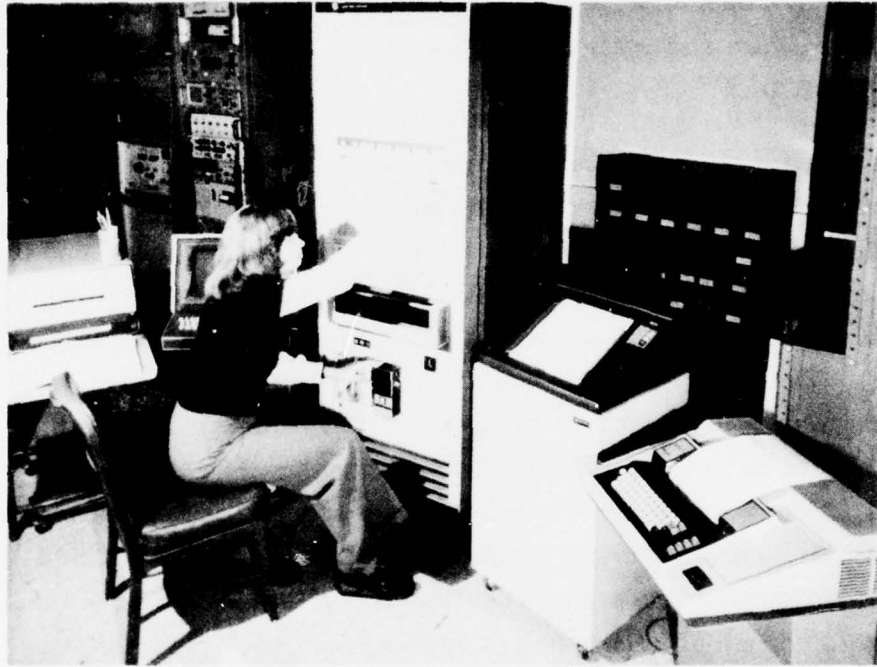


Fig. 1 — General purpose signal processing laboratory

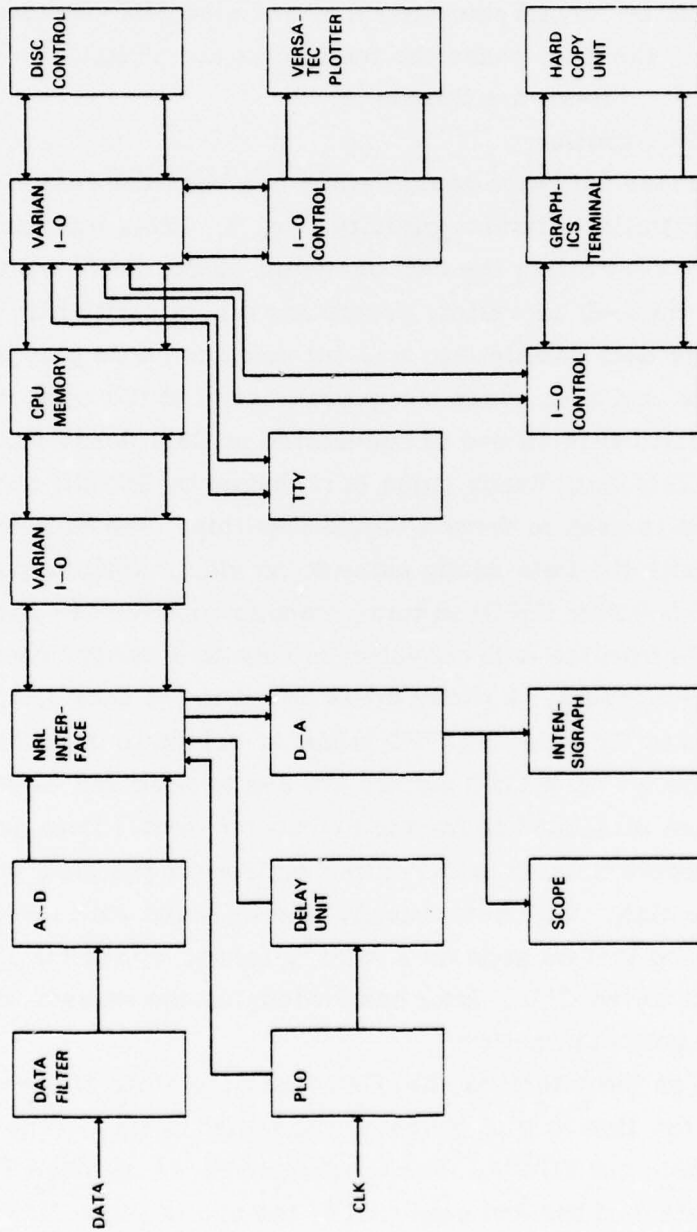


Fig. 2 — Block diagram of signal processing laboratory

pulse analysis phase the data is recalled from disk as needed and analyzed via software FFT algorithms. These transforms are displayed as power versus frequency plots on the graphics terminal and also stored in disk memory if desired. In the final phase the transforms are plotted.

#### Hardware Design

##### A. Analog-to-Digital Controller

The major in-house hardware design effort has been the design of the Analog-to-Digital controller interface circuits (Fig. 3). This interface unit contains the logic for controlling the A-D converter clock, buffering the 8-bit byte data from the A-D converter, generating a byte parity bit, and packing the 8-bit byte data sample into a 16-bit computer word plus byte parity. The interface unit then sends a request to send to the computer.

The process starts with an end of conversion or Data Ready pulse from the A-D converter. This Data Ready pulse is reshaped by Schmitt trigger and one-shot circuits and is used to drive a toggle flip-flop. The outputs of this flip-flop are Anded with the Data Ready pulse to provide a shift-in pulse to each First In-First Out Buffer (FIFO) in turn, starting with the low byte FIFO. The 8-bit data sample from the A-D converter is sent to a parity generator and to both FIFO buffer inputs. A parity bit is added to the data sample by the parity generator and the selected FIFO buffer is pulsed to gate the data sample and parity bits in. The FIFO buffers are 9-bits wide and 32 words deep. Two of these units are paralleled to form the computer word. Data gated into either FIFO buffer ripples through to the output register of the FIFO and sets the output ready line high. This output ready line is Anded with the output ready line of FIFO-2 so that as soon as a word is formed by the FIFO's a request to send is sent to the CPU. After acknowledging the request the CPU clocks the data into computer memory.

To output data to the interface, the CPU tests the state of the output Buffer ready line; if the line is true (meaning the buffer is empty) the data is clocked into the data out FIFO's. The input ready lines of these FIFO's are Anded and the output of the And gate tied to the above sense line. The data again ripples through both FIFO's and is clocked out to the D-A converter by an external clock. Data can thus be readout at any convenient rate.

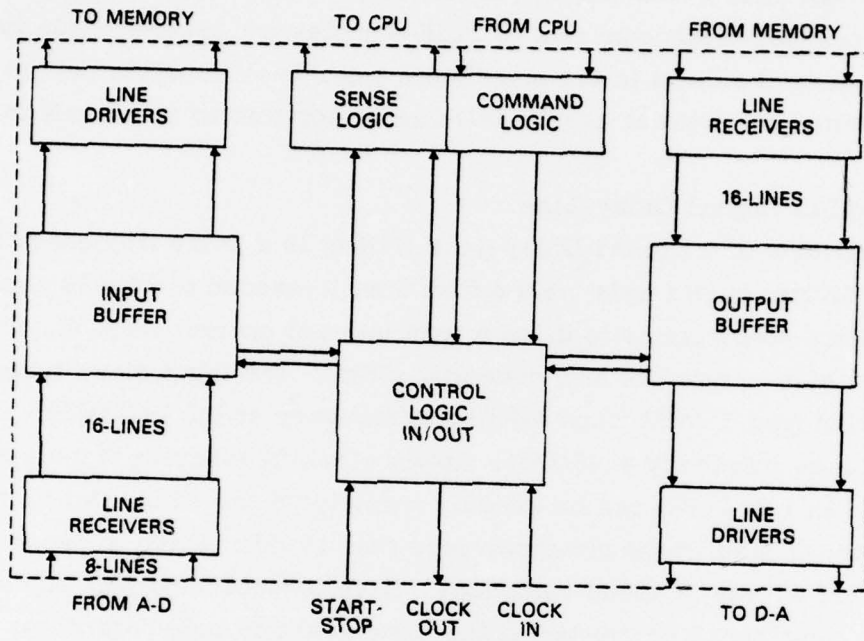


Fig. 3 - A-D converter/computer interface

FIFO buffers used at present have a 500K-word minimum thru-put rate, so that the maximum A-D conversion rate is 1 mega-samples per second. We plan to replace these FIFO's with 1-mega-word-rate units, and thereby increase the A-D conversion rate to 2-mega-samples per second. The present design uses a Non-Direct Memory Access (DMA) controller and is limited to the Non-DMA word rate of 225K-samples per second. However, line termination problems in the breadboard proto-type interface now in use has limited the present usable A-D conversion rate to about half this value.

#### B. Triggerable Digital Delay-Line

The Triggerable Digital Delay-Line (TDDL) is a pulse triggered clock frequency divider with a delay range from 1 milli-second to 99.999 seconds, and with start-stop outputs to drive a time interval counter (Fig. 4). The stop output also starts the A-D converter clock. The input clock frequency can be varied from 1 MHz (tape reference frequency at 120 IPS) to 125 KHz (tape reference frequency at 15 IPS). Lower clock frequencies (tape speeds from  $7\frac{1}{2}$  IPS to  $1\frac{3}{4}$  IPS) can be used by multiplying the TDDL dial settings by the factor of tape speed slowdown less than 15 IPS. Delay times are synchronized with tape speed variations. This is most important since it allows the digitizer to be started at the same event on tape regardless of tape speed.

#### TDDL Circuit Description

A trigger pulse from the 1 PPS output of a time-code reader is applied to the trigger input circuits where it is reshaped and sent to the set input of a trigger start-stop latching flip-flops. The Q output of this latch is Anded with the clock signal. The output this And gate is sent to the set input of the clock start latch whose function is to allow only the first clock pulse after triggering to be sent to the time interval start one-shot. Next, a binary counter receives the clock pulse. The counter's outputs are each sent to an And Gate. The other input of each Gate is tied to a front panel switch which allows one to select the proper counter output for a 125 KHz clock output signal. This clock signal is divided by 125 to give a 1000 pps signal. The 1000 pps signal is routed to the next divider

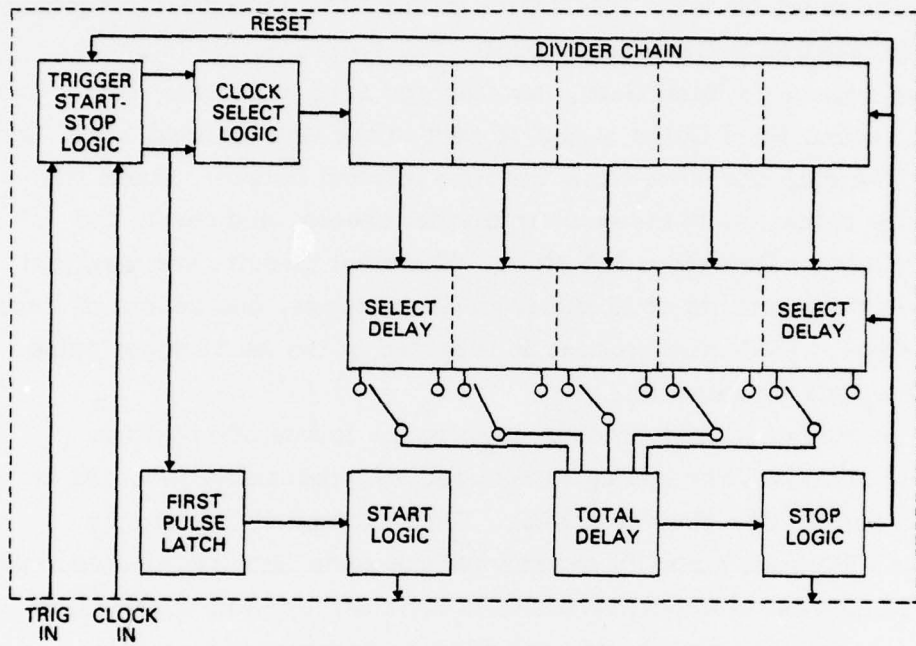


Fig. 4 - Triggerable delay-line (TDDL)

and to one input of an 8-input Nand Gate. Four of the other inputs are from later stages in the divider chain. The sixth input is the output of a SN7442 decimal decoder. These five inputs are normally high (at a logic one), so that the output of the Nand Gate is the 1000 pps signal, which is sent to a 7490 decade counter. The decade counter outputs are decoded and sent to a front panel switch. This switch selects the appropriate decoder output for the desired milli-second count. The switch output is sent to the above mentioned Nand Gate, inverted and sent to another Nand Gate. The second Nand Gate's output is sent to the time interval stop one-shot. The stop one-shot stops the time interval counter, starts the A-D converter clock, resets the counter/divider chains, and resets the trigger start latch which stops the clock. The other circuits are identical and give selectable counts of 10 and 100 milli-seconds, and counts of 1 and 10 second steps. The entire process is repeated at the next trigger pulse.

Normal Operations (acquisition)

This section describes procedures to scan, locate and digitize taped analog signals. The analog tapes used are presumably recorded at 120 ips with bandwidths of up to 2 MHz. Time codes and a reference frequency of 1 MHz are normally recorded at the same time as the data. The tape track of interest is up-converted to 10 MHz and applied to a bank of 100 analog filters. The outputs of each filter is detected and sampled, with the sampled output displayed on an oscilloscope. The times and types of spectral occurrences are then recorded. The exact times of spectral occurrences are found by recording the sampled output and the time code on separate channels of a Bill-Howell strip-recorder. A 100 Hz pulse train is recorded on another channel at the same time. Bandwidths and time duration of spectra observed is noted and a decision is made as to the bandwidth to be used for digitizing. Determining factors in the choice of the bandwidth to be digitized are: spectra bandwidth and time duration, initial SNR, translator bandwidth (40 KHz at present), and available computer memory.

If one assumes that the spectra of interest is centered in the translator passband, the output of the translator at the 40KHz BW setting is a signal with a bandwidth of from 4 to 44 KHz. This signal is low-passed at 44 KHz and applied to the A-D converter. At the same time, the time code signal from the tape recorder is FM-Demodulated and sent to a time code reader. The 1 PPS output of this reader is used to trigger the TDDL.

The reference signal from the tape is filtered and sent to a phase-locked oscillator synthesizer (PLO-SYN) and to the TDDL clock input. The output of the PLO-SYN is the A-D converter clock, while outputs from the TDDL start and stop a Hewlett-Packard Time-Interval counter and trigger the gated A-D converter clock circuit.

Once the appropriate delay is decided upon, the TDDL is set, the acquisition program is loaded into the computer, and the analog tape recorder is started. The tape recorder is normally operated at a 8:1 speed reduction (15 IPS) and one has time to arm the A-D converter interface so that the next pulse from the 1PPS time code reader output will trigger the TDDL which, in turn, will start the A-D converter at the proper time interval. The A-D converter interface packs the 8-bit data samples into 16-bit words, adds the byte parity bits and signals the computer to receive the data samples. After the computer memory block is filled, or the number of data samples desired taken, the computer will either stop the A-D converter clock or ignore further requests to send data from the A-D converter interface.

The tape recorder is then stopped and the graphics terminal can be observed to see if the spectra of interest was digitized. If desired the entire process can be repeated. Normally the TDDL settings would be adjusted so as to move the digitizer time window to an adjacent time slot. With repeated passes one can slide the digitizer time window through the time duration of the signal of interest.

Alternate means of data acquisition are putting the signal of interest on a video disk and using the disk sync pulse to trigger the TDDL or if the conversion rate is low enough, continuous digitizing with the computer transferring data from main memory to disc memory and to magnetic tape.

#### Software & Programming Considerations

The Varian V-73 mini-computer system (Fig. 5) utilizes a batch processing operating system (MOS-version K) with a Fortran compiler, a DAS assembler, a text editor, and a file system. We have expanded the operating system to include the Tektronix Plot-10 software package, and a U.S.I. modified version of Versatec's plotting software.<sup>2</sup>

Installation of vendor supplied software and the subsequent generation of routines particularly suited to our signal processing application

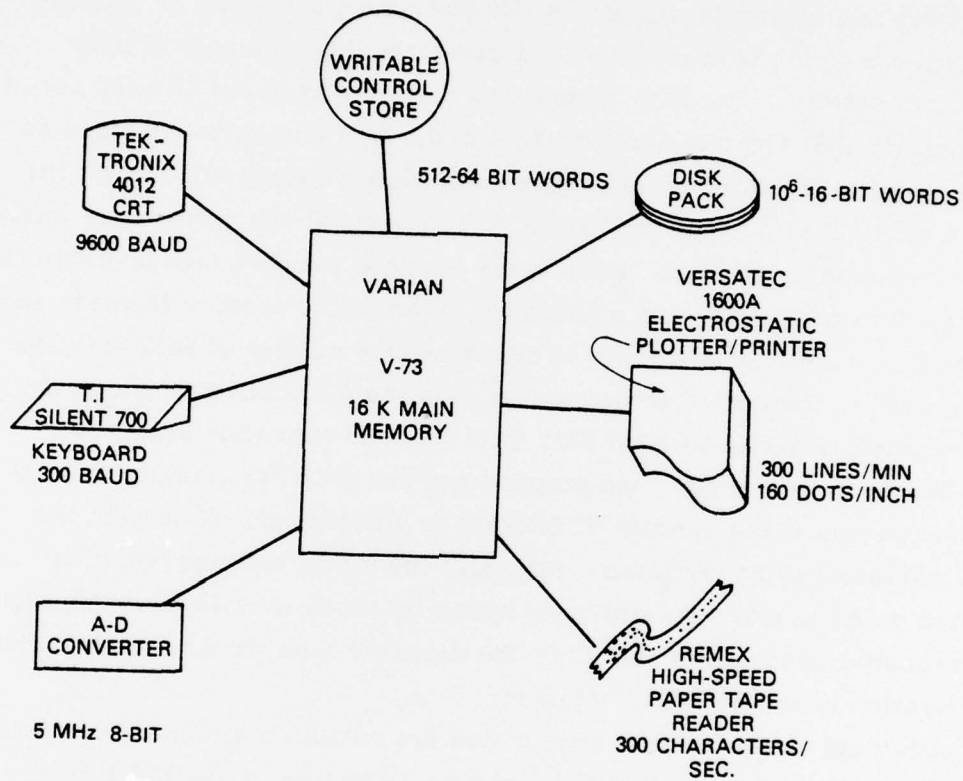


Fig. 5 - System components

was made unduly difficult by the lack of a single vendor for the entire system. Individual software packages had to be integrated into the Varian operating system. In particular the very sophisticated Versatec plotting software had to be hand tailored for the Varian, necessitating a time consuming process of code translation, a multitude of Fortran modifications & careful checking of test routines. Lengthy negotiations with Versatec resulted in the acquisition of a Varian compatible version of the Universal Versatec Software prepared by Underwater Systems, Inc.

At the initial stages of the operation it was necessary to generate assembly language code to select an input gate in the buffered I/Ø Controller and sense the state of the output control pulse. A ready state resulted in the input buffer of the controller being read into memory. The debugging of this buffered driver occurred simultaneously with the check out of the hardware A-D controller interface circuits. Signal transmission into the Varian via the A-D, and back out via the D-A was monitored with an oscilloscope to test the accuracy of this dual hardware/software effort.

All of the processing was completed under the restraint of limited central memory; the operating system generally left not more than 12 k of core for programs and data. Therefore, another time consuming phase of the signal processing effort was the efficient use of the disk for storage of intermediate results. Numerous utility programs were written yielding various methods of storing and retrieving the data, and displaying results alternately on the graphics terminal or the line printer.

A recent task designed to demonstrate the capabilities of this signal processing laboratory entailed the acceptance of data from an analog tape and subsequent analysis and display. The limited amount of central memory available, and the desire to process long strings of data, necessitated a three phase operation, with the data being stored on disk in the interim.

Phase I (TEKTRAN) consists of accepting the incoming digitized signals, unpacking the data and storing the raw data on disk. If desired, the data, at this point, can be biased and displayed on the graphics terminal. During this phase the Fourier Transform of the signal is computed and also stored on disk. The option of displaying the transform on the graphics terminal is also available. (Fig. 6)

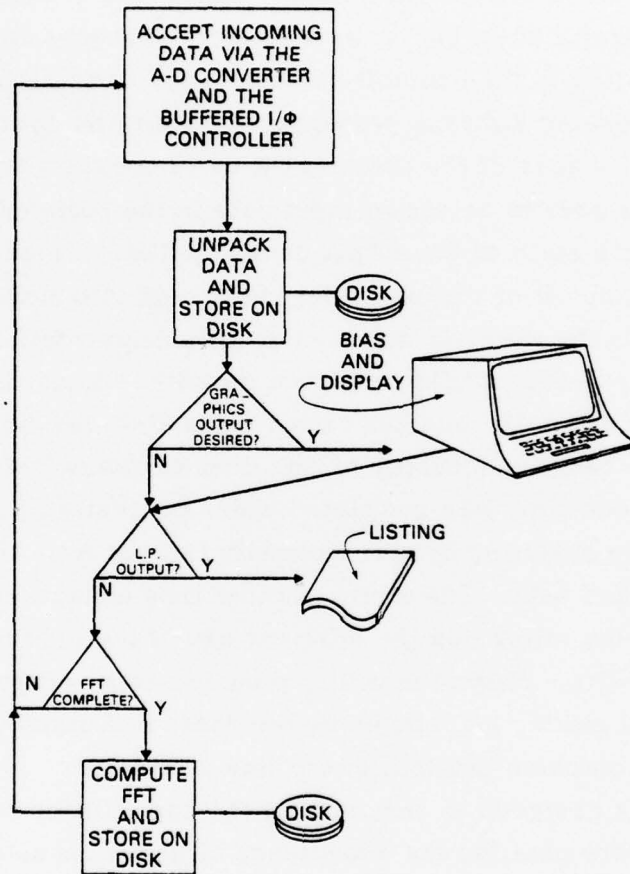


Fig. 6 - Phase I - Tektran

Phase II (EIGTRAN) consists of reading sequential files from disk (each data file being followed by its transform) and creating a plot file for each data file. Storage limitations require the successive readings of small amounts of data, until the entire data set is scanned, computing a scaling factor, and then repeating the entire process using the computed scaling factor to generate a plot file for each data file (Fig. 7).

The final phase involves the execution of VPLOT (a modified Versatec routine incorporated into the MOS operating system) to convert the pen movements delineated in the plot file to the ordered raster output which is necessary for the electrostatic plotter. The plots can be generated immediately or stored for output at a later time (Fig. 8).

### Conclusions

A mini-computer based digital signal processing laboratory was discussed in terms of hardware interface design. Pertinent hardware and software design and performance details were noted and a brief description of normal operations given. While the present signal processing capability is limited to signals of 100 KHz BW or less, the system can be easily expanded by the addition of DMI interface cards to process signals with bandwidths of up to 2 MHz.

- 
- 1 Varian V-73 System Handbook. Varian Data Machines, Irvine, CA.
  - 2 Versaplot Adaptation to VDM, Underwater System, Inc.

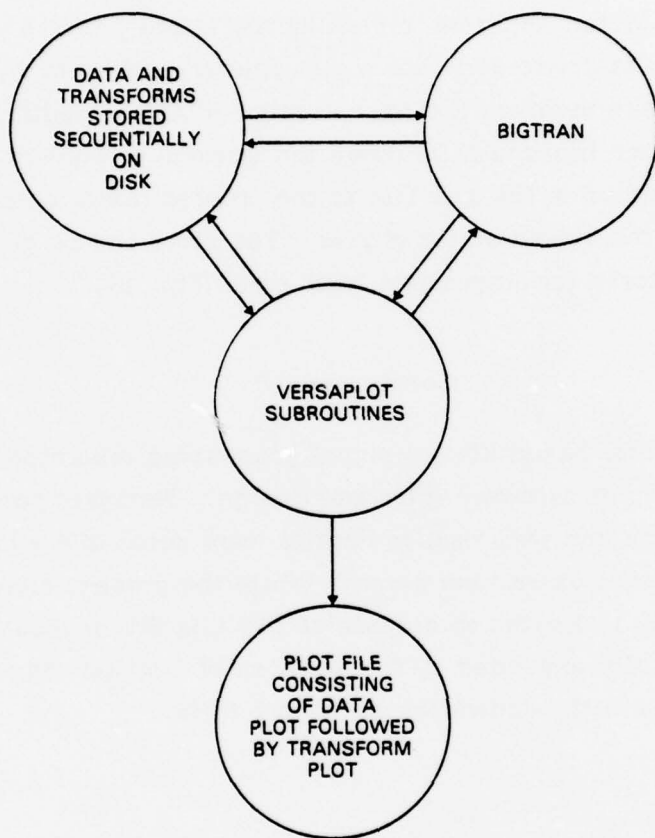


Fig. 7 - Phase II - Bigtran

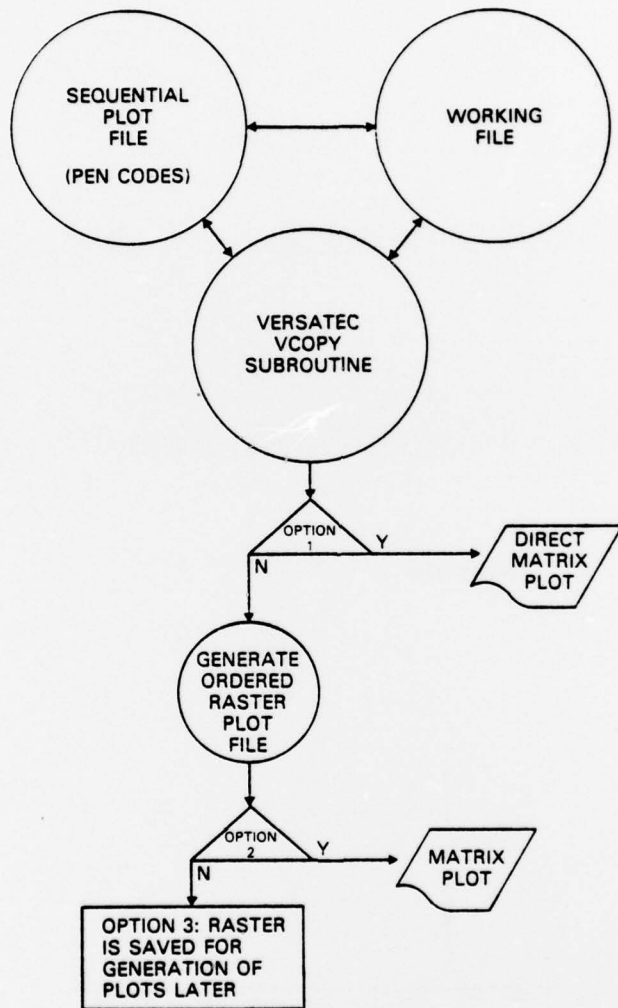


Fig. 8 - Phase III - Versatec Vcopy

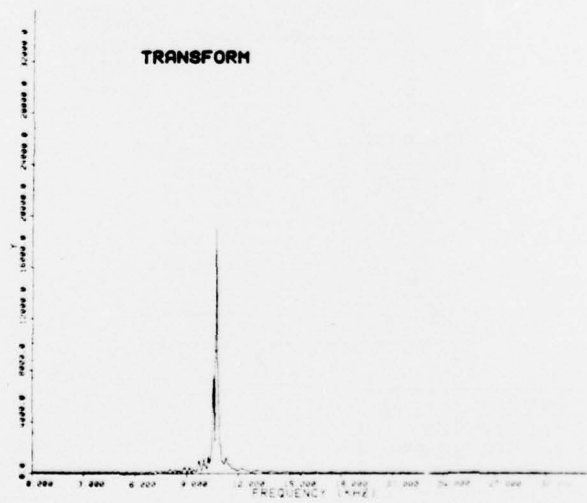
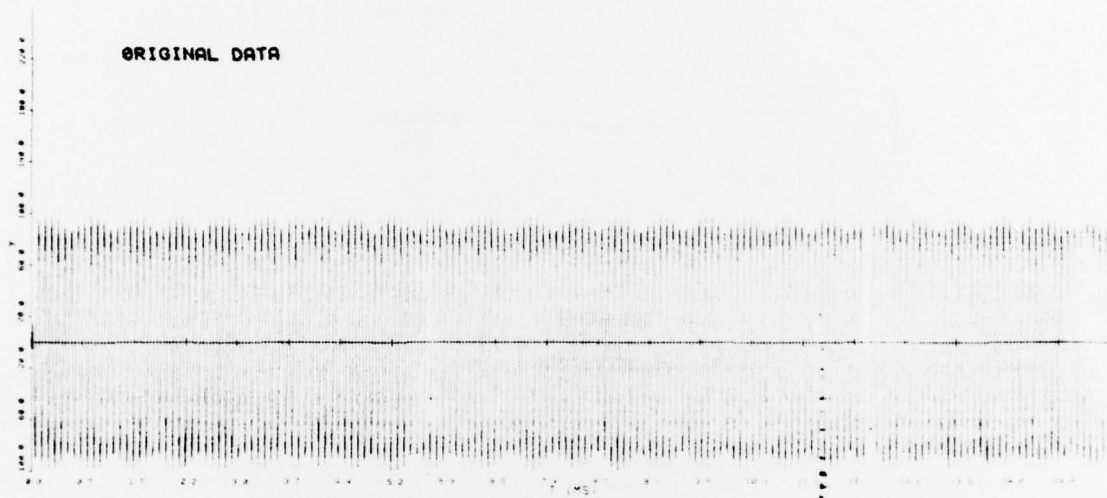


Fig. 9 - Calibration signal and power spectrum

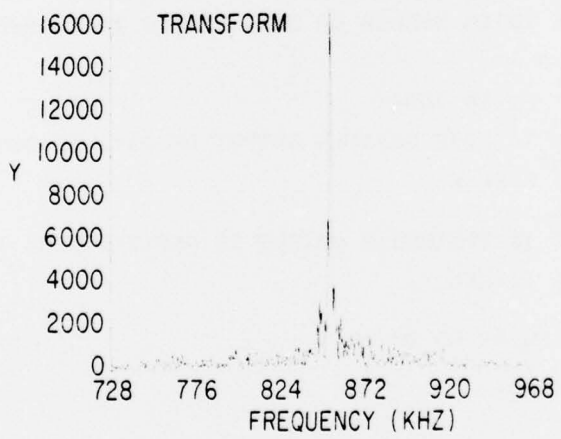
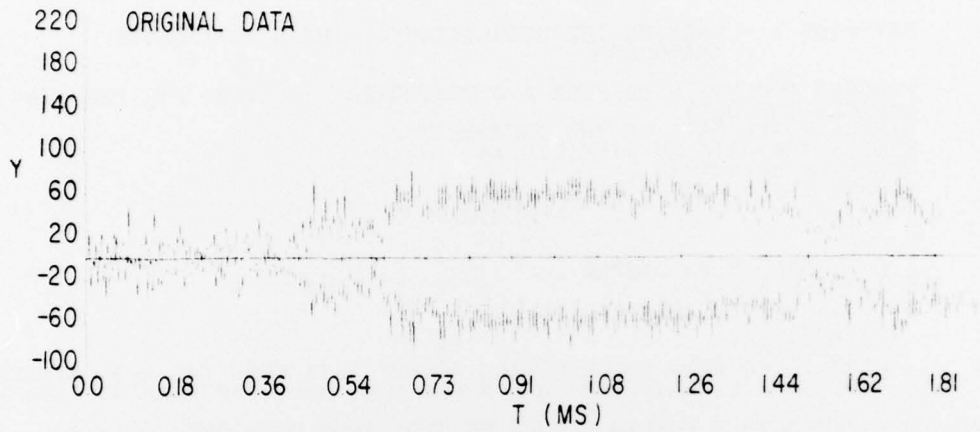


Fig. 10 — Radar echoes and power spectrum

Program Listings

PAGE 1

```

1 *   TEKTRAN
2 *
3 *   REVISION 1 - VARYING THE RESOLUTION OF THE DATA AND THE
4 *   TRANSFORM
5 *
6 *   PROGRAM ACCEPTS DATA FROM A-D CONVERTER -- 1024 PTS PER PASS
7 *   UNPACKS THE DATA
8 *   DISPLAYS THE DATA ON THE TEKTRONIX
9 *   STORES THE DATA ON DISK L.U.14
10 *  COMPUTES TRANSFORM
11 *  STORES TRANSFORM ON DISK L.U.14
12 *
13 *
14 *
15 *   EXTERNAL IN,DA,DISPLA
16 *   INTEGER DATA2
17 *   COMMON DATA2(1024)
18 *   REWIND 14
19 *   WRITE(3,88)
20 88  FORMAT(54H THIS PROGRAM WILL ACCEPT DATA FROM THE A-D CONVERTER,
21    1/,56H COMPUTE TRANSFORM, AND STORE DATA AND TRANSFORM ON DISK/)
22 *
23 *   *****EXTRACTS N POINTS (POWER OF TWO) FROM DATA SET*****
24 *
25 10  WRITE(3,87)
26 87  FORMAT(53H ENTER NUMBER OF DATA POINTS TO TRANSFORM (15 FORMAT)/
27    X)
28    READ(3,89) N
29 89  FORMAT(15)
30    IF(N.EQ.0) GO TO 1000
31    WRITE(3,84)
32 84  FORMAT(66H IF LINE PRINTER OUTPUT OF DATA IS DESIRED, TYPE 1, IF
33    X NOT, TYPE 0)
34    READ(3,85) LPFLAG
35 85  FORMAT(11)
36    WRITE(3,83)
37 83  FORMAT(54H IF TEKTRONIX OUTPUT IS DESIRED TYPE 1, IF NOT, TYPE 0
38    X)
39    READ(3,85) ITEKFL
40    CALL IN
41    CALL DA
42    IF(ITEKFL.EQ.0) GO TO 92

```

PAGE 2

```
43 92 IFAC=1024/N
44 DO 100 JJ=1,1024
45 JK=JJ*IFAC-IFAC+1
46 100 DATA2(JJ)=DATA2(JK)
47 CALL DISPLA(N)
48 DO 20 I=1,N
49 20 DATA2(I)=DATA2(I)*8
50 IF(LPFLAG.EQ.0) GO TO 101
51 WRITE(5,90) N
52 90 FORMAT(1H1.4H N= ,I5//)
53 WRITE(5,91) (DATA2(K),K=1,N)
54 91 FORMAT(10I8)
55 101 WRITE(14) (DATA2(K),K=1,N)
56 ENDFILE 14
57 CALL FTRA(N,DATA2)
58 IF(LPFLAG.EQ.0) GO TO 102
59 WRITE(5,90) N
60 102 WRITE(5,91) (DATA2(K),K=1,N)
61 WRITE(14) (DATA2(K),K=1,N)
62 ENDFILE 14
63 CALL SCAN(DATA2,DATA2,-N,550)
64 CALL MODE(-8,XMIN,DX,XORG)
65 CALL MODE(-9,YMIN,DY,YORG)
66 WRITE(3,78) XMIN,YMIN,DY,YORG
67 78 FORMAT(2F30.4,/,2F30.4)
68 IDY=DY
69 IF(IDY.EQ.0) GO TO 10
70 JMIN=YMIN
71 MAX=10*IDY+JMIN
72 IDY=(MAX-JMIN)/790
73 DO 30 I=1,N
74 30 DATA2(I)=(DATA2(I)-JMIN)/IDY
75 IF(LPFLAG.EQ.0) GO TO 103
76 WRITE(5,90) N
77 103 WRITE(5,91) (DATA2(K),K=1,N)
78 DO 31 I=1,N
79 31 DATA2(I)=DATA2(I)-390
80 IF(ITEKFL.EQ.0) GO TO 10
81 CALL DISPLA(N)
82 GO TO 10
83 1000 STOP
84 END
```

```

1 *      SUBROUTINES IN,DA,DISPLA
2 ***
3 ***      REVISION 1 - VARIABLE GRAPHIC OUTPUT
4 ***
5 ***
6 ***      'IN' ACCEPTS DATA
7 ***      'DA' UNPACKS DATA
8 ***      'DISPLA' YIELDS GRAPHIC OUTPUT ON TEKTRONIX
9 *
10      EXT      TPLO.CHOU,$SE
11      NAME      DA
12      NAME      IN
13      NAME      DISPLA
14      DATA2   COMN      1024
15      IN      ENTR
16      EXC      0060
17      LDXI     -512
18      LOOP    SEN      0060,*+5
19      NOP
20      JMP      *-3
21      IME      060,TEMP
22      LDBE     TEMP
23      STBE     DATA+512,1
24      IXR
25      JXNZ     LOOP
26      RETU*    IN
27      DA      ENTR
28      LDXI     -512
29      LDBI     -1024
30      SOF
31      BACK    LDAE     DATA+512,1
32      JOF      RIGHT

```

```

000000 000000 C
000001 000060 A
000002 006030 A
000003 177000 A
000004 101060 A
000005 000011 R
000006 005000 A
000007 001000 A
000010 000004 R
000011 102060 A
000012 000302 R
000013 006027 A
000014 000302 R
000015 006065 A
000016 001304 R
000017 005144 A
000020 001046 A
000021 000004 R
000022 001000 A
000023 100000 R
000024 000000 A
000025 006030 A
000026 177000 A
000027 006020 A
000030 176000 A
000031 007401 A
000032 006015 A
000033 001304 R
000034 001001 A

```

PAGE 2

000035	000051	R			
000036	006157	A	33	LEFT	ANAE MASKL
000037	001304	R			
000040	004350	A	34		LSRA 010
000041	006056	A	35		STAE DATA2+1024,2
000042	002000	C			
000043	005122	A	36		IBR
000044	005144	A	37		IXR
000045	001046	A	38		JXNZ DA+5
000046	000031	R			
000047	001000	A	39		JMP OUTPUT
000050	000060	R			
000051	006157	A	40	RIGHT	ANAE MASKR
000052	001305	R			
000053	006056	A	41		STAE DATA2+1024,2
000054	002000	C			
000055	005122	A	42		IBR
000056	001000	A	43		JMP BACK
000057	000032	R			
000060	006030	A	44	OUTPUT	LDXI ~1024
000061	176000	A			
			45	* DATA HAS BEEN UNPACKED THIS SECTION MASKS OUT SIGN BIT ANDSHIFTS IT.	
			46	* SETS UP IX AND IY ADDS BIAS TO IY THEN CALLS TPLO.	
000062	006015	A	47	LDAE	DATA2+1024,1
000063	002000	C			
000064	006150	A	48	ANAI	MASKS
000065	000200	A			
000066	004250	A	49	LRLA	3 MOVE SIGN BIT INTO PLACE
000067	001010	A	50	JAZ	*+8
000070	000077	R			
000071	006015	A	51	LDAE	DATA2+1024,1
000072	002000	C			
000073	006110	A	52	ORAI	0177400
000074	177400	A			
000075	006055	A	53	STAE	DATA2+1024,1
000076	002000	C			
000077	005144	A	54	IXR	
000100	001046	A	55	JXNZ	OUTPUT+2
000101	000062	R			
000102	001000	A	56	RETU*	DA
000103	100024	R			
000104	000000	A	57	DISPLA	ENTR
000105	002000	A	58	CALL	3SE

PAGE 3

000106	000000	E			
000107	000001	A	59	DATA	1
000110			60	NP	BSS
000111	006010	A	61		LDAI 1024
000112	002000	A			
000113	147000	I	62	SUB	(NP)*
000114	054166	A	63	STA	TEMP2
000115	006030	A	64	LDXI	-1024
000116	176000	A			
000117	006020	A	65	LOOP2	LDBI 0
000120	000000	A			
			66	*	CLEAR SCREEN. SET ITYPE=0
000121	006010	A	67		LDAI 27
000122	000033	A			
000123	054156	A	68	STA	TEMP
000124	002000	A	69	CALL	CHOU, TEMP
000125	000000	E			
000126	000302	R			
000127	006010	A	70	LDAI	12
000130	000014	A			
000131	054150	A	71	STA	TEMP
000132	002000	A	72	CALL	CHOU, TEMP
000133	000125	E			
000134	000302	R			
000135	006010	A	73	LDAI	4
000136	000004	A			
000137	054142	A	74	STA	TEMP
000140	002000	A	75	CALL	DELAY, TEMP
000141	000251	R			
000142	000302	R			
000143	005001	A	76	YZA	
000144	006057	A	77	STAE	ITYPE
000145	001306	R			
000146	005041	A	78	TXA	
000147	124133	A	79	ADD	TEMP2
000150	001002	A	80	JAP	R
000151	000215	R			
000152	006015	A	81	LOOP1	LDAE DATA2+1024, 1
000153	002000	C			
000154	006120	A	82	ADDI	BIAS
000155	000606	A			
000156	006057	A	83	STAE	IY
000157	001312	R			

PAGE 4

000160	006067	A	84	STBE	IX
000161	001311	R			
000162	005021	A	85	TBA	
000163	006140	A	86	SUB1	1023
000164	001777	A			
000165	001002	A	87	JAP	COPY
000166	000217	R			
000167	002000	A	88	PLOT CALL	TPLO. ITYPE. IX. IY
000170	000000	E			
000171	001306	R			
000172	001311	R			
000173	001312	R			
000174	005144	A	89	IXR	
000175	005041	A	90	TXA	
000176	124104	A	91	ADD	TEMP2
000177	001002	A	92	JAP	COPY
000200	000217	R			
000201	005021	A	93	TBA	
000202	006120	A	94	ADDI	1
000203	000001	A			
000204	005012	A	95	TAB	
000205	005001	A	96	TZA	
000206	005111	A	97	IAR	
000207	006057	A	98	STAE	ITYPE
000210	001306	R			
000211	001046	A	99	JXNZ	LOOP1
000212	000152	R			
000213	001000	A	100	JMP	COPY
000214	000217	R			
000215	001000	A	101	R RETU*	DISPLA
000216	100104	R			
000217	006010	A	102	COPY LDAI	105
000220	000151	A			
000221	054060	A	103	STA	TEMP
000222	002000	A	104	CALL	DELAY. TEMP
000223	000251	R			
000224	000302	R			
000225	006010	A	105	LDAI	27
000226	000033	A			
000227	054052	A	106	STA	TEMP
000230	002000	A	107	CALL	CHOU. TEMP
000231	000133	E			
000232	000302	R			

PAGE 5

000233	006010	A	108	LDAI	23
000234	000027	A			
000235	054044	A	109	STA	TEMP
000236	002000	A	110	CALL	CHOU, TEMP
000237	000231	E			
000240	000302	R			
000241	006010	A	111	LDAI	105
000242	009151	A			
000243	054036	A	112	STA	TEMP
000244	002000	A	113	CALL	DELAY, TEMP
000245	000251	R			
000246	000302	R			
000247	001000	A	114	JMP	LOOP2
000250	000117	R			
000251	000000	A	115	DELAY ENTR	
000252	002000	A	116	CALL	SSE, 1
000253	000106	E			
000254	000001	A			
000255			117	TIME	BSS 1
000256	077000	I	118	STX	SAVX
000257	067000	I	119	STB	SAVB
000260	066037	A	120	LDXE	(TIME)*
000261	100255	R			
000262	006010	A	121	LDAI	077777
000263	077777	A			
000264	005311	A	122	DAR	
000265	005000	A	123	NOP	
000266	001010	A	124	JAZ	COUNT
000267	000272	R			
000270	001000	A	125	JMP	*-4
000271	000264	R			
000272	005344	A	126	COUNT DXR	
000273	001046	A	127	JXNZ	*-9
000274	000262	R			
000275	037000	I	128	LDX	SAVX
000276	027000	I	129	LDB	SAVB
000277	001000	A	130	RETU*	DELAY
000300	100251	R			
000301	000777	A	131	X HLT	0777
	000606	A	132	BIAS EQU	390
	000200	A	133	MASKS EQU	128
	000177	A	134	MASKD EQU	127
000302		A	135	TEMP BSS	1

PAGE 6

000303		136 TEMP2	BSS	1
000304		137 DATA	BSS	512
001304	177400 A	138 MASKL	DATA	0177400
001305	000377 A	139 MASKR	DATA	0377
001306		140 ITYPE	BSS	1
001307		141 SAVX	BSS	1
001310		142 SAVB	BSS	1
001311		143 IX	BSS	1
001312		144 IY	BSS	1
		145	END	

ENTRY NAMES

000024 R DA 000104 R DISPLA 000000 R IN

EXTERNAL NAMES

000253 E SSE 000237 E CHOU 000170 E TPLO

SYMBOLS

000253 E SSE	000032 R BACK	000606 A BIAS	000237 E CHOU
000217 R COPY	000272 R COUNT	000024 R DA	000304 R DATA
000000 C DATA2	000251 R DELAY	000104 R DISPLA	000000 R IN
001306 R ITYPE	001311 R IX	001312 R IY	000036 R LEFT
000004 R LOOP	000152 R LGOP1	000117 R LOOP2	000177 A MASKD
001304 R MASKL	001305 R MASKR	000260 A MASKS	000110 R NP
000060 R OUTPUT	000167 R PLOT	000215 R R	000051 R RIGHT
001310 R SAVB	001307 R SAVX	000302 R TEMP	000303 R TEMP2
000255 R TIME	000170 E TPLO	000301 R X	

0 ERRORS ASSEMBLY COMPLETE

```

1 *      BIGTRAN
2 *
3 * *****
4 *      READ FROM DISK
5 *      AND PLOT FROM A FILE - USING SEGMENTS
6 *      FILES ARE OF VARYING LENGTHS, WITH THE FIRST FILE CONTAINING
7 *      A FUNCTION OR DATA, AND THE FILE IMMEDIATELY FOLLOWING CONTAINING
8 *      ITS FOURIER TRANSFORM
9 *
10 *
11 *
12 *      VARIABLE LENGTH PLOTS
13 *
14 *      REVISION 1:  FREQUENCY SCALE
15 *
16 * *****
17 *
18 *      INTEGER BUF
19 *      DIMENSION BUF(60),X(60),IALF1(8),IALF3(7),IALF4(5)
20 *      DATA SAME/9999./,IALF2/2HY /,IALF0/2HX /
21 *      DATA (IALF1(1),1=1,8)/2HFR,2HEQ,2HUE,2HNC,2HY ,2H(K,2HHZ,2H) /
22 *      DATA (IALF3(1),1=1,7)/2HOR,2HIG,2HIN,2HAL,2H D,2HAT,2HA /
23 *      DATA (IALF4(1),1=1,5)/2HTR,2HAN,2HSF,2HOR,2HM /
24 *      IPLOT=1
25 *      REWIND 14
26 *      WRITE(3,88)
27 *      88      FORMAT(55H PROGRAM PLOTS FUNCTION AND TRANSFORM STORED ON L.U. 1
28 *      X4/)
29 *      300     WRITE(3,86)
30 *      86      FORMAT(51H TYPE PTS. IN FILE, SAMPLING FREQUENCY (15,F10.0) )
31 *      READ(3,89)N,SAMFR
32 *      89      FORMAT(15,F10.0)
33 *      *      IF N=0, NO MORE PLOTS ARE DESIRED
34 *      IF(N.EQ.0) GO TO 700
35 *      WRITE(3,84)
36 *      84      FORMAT(65H IF LINE PRINTER OUTPUT OF DATA IS DESIRED TYPE 1, IF
37 *      XNOT, TYPE 0)
38 *      READ(3,85) LPFLAG
39 *      85      FORMAT(I)
40 *      *      WHEN THE DATA AND TRANSFORM ARE STORED ON DISK THE SYSTEM
41 *      *      CREATES 60 WORD RECORDS - THE FOLLOWING CODING DETERMINES
42 *      *      THE NUMBER OF RECORDS (NREC) CREATED

```

PAGE 2

```
43 NUM=N/60
44 NM=N-NUM*60
45 N=60-MM
46 IF(ND 100,101,100)
47 100 NUM=NUM+1
48 101 NREC=NUM
49 LEN=N/50+1
50 FLEN=LEN
51 IF(N.LE.350) FLEN=7.
52 CALL MODE(2,FLEN,SAME,SAME)
53 * FREQUENCY RANGE FOR TRANSFORM EQUALS ONE HALF THE SAMPLING
54 * FREQUENCY OR 'FOLDING FREQUENCY'
55 FOLDF=.5*SAMFR
56 FINPPT=FLEN/FLOAT(N-1)
57 * JK=1 IMPLIES PLOT OF FUNCTION
58 * JK=2 IMPLIES PLOT OF TRANSFORM
59 DO 600 JK=1,2
60 IF(JK.EQ.1) DELTAX=FLOAT(N-1)/FLEN
61 IF(JK.EQ.2) DELTAX=FOLDF/FLEN
62 * J=1, SCANNING OF DATA FOR SCALING
63 * J=2, PLOTTING OF SEGMENTS
64 DO 500 J=1,2
65 IFLAC=J-1
66 IF(LPFLAG.EQ.0) GO TO 179
67 WRITE(5,90) N,J,JK
68 90 FORMAT(1H1.4HN = .15,5H J = .12,6H JK = .12,/)
69 179 NFS=1PLOT-1
70 IF(NFS.EQ.0) GO TO 181
71 DO 180 JF=1,NFS
72 180 CALL SKIP
73 * LOCATES DATA FOR NEXT SEQUENTIAL PLOT
74 181 FK1=0.
75 KN=60
76 DO 200 I=1,NREC
77 IF(I.EQ.NREC) KN=-MM
78 READ(14)(BUF(K),K=1,60)
79 IF(LPFLAG.EQ.0) GO TO 194
80 WRITE(5,91) (BUF(K),K=1,60)
81 91 FORMAT(10I8)
82 194 DO 199 II=1,60
83 X(II)=FK1*FINPPT
84 FK1=FK1+1.
```

PAGE 3

```
85 199 CONTINUE
86 IF(IFLAG) 198,197,198
87 197 CALL MODE(7,FLEN,SAME,SAME)
88 CALL SCAN(X,BUF,KN,050)
89 * THIS CALL SEQUENTIALLY SCANS BLOCKS OF DATA --
90 * AFTER LOOKING AT LAST BLOCK. SCALING FACTORS ARE DETERMINED.
91 GO TO 200
92 198 CALL MODE(8,0.,DELTAX,SAME)
93 IF(I.EQ.NREC) GO TO 196
94 IF(I.EQ.1) GO TO 195
95 CALL DRAW(X,BUF,60,059)
96 GO TO 200
97 * IN ORDER TO ELIMINATE A GAP BETWEEN PLOT SEGMENTS
98 * SUBSEQUENT CALLS TO DRAW ARE WITH THE PEN DOWN
99 195 CALL DRAW(X,BUF,60,051)
100 * THIS CALL SEQUENTIALLY PLOTS BLOCKS OF DATA
101 GO TO 200
102 196 CALL DRAW(X,BUF,MM,059)
103 200 CONTINUE
104 REWIND 14
105 500 CONTINUE
106 IF(JK.EQ.1) CALL AXES(2.1,IALF0,2.1,IALF2)
107 IF(JK.EQ.2) CALL AXES(16.3,IALF1,2.1,IALF2)
108 CALL MODE(6.4,SAME,SAME)
109 CALL MODE(4.2,.15,SAME)
110 IF(JK.EQ.2) GO TO 502
111 CALL NOTE(1.75,8.0,IALF3,14)
112 GO TO 503
113 502 CALL NOTE(2.1,8.0,IALF4,10)
114 503 CALL DRAW(0.,0.,1,9000)
115 * THIS CALL ENDS A PARTICULAR PLOT
116 WRITE(3,87) IPLOT
117 87 FORMAT(10H PLOT NO. ,15.9H COMPLETE)
118 IPLOT=IPLOT+1
119 600 REWIND 14
120 GO TO 360
121 700 CONTINUE
122 CALL DRAW(0,0,0,999)
123 * THIS CALL INDICATES THAT ALL PLOTTING IS COMPLETED!
124 STOP
125 END
ENTRY/COMMON BLOCK NAMES
```