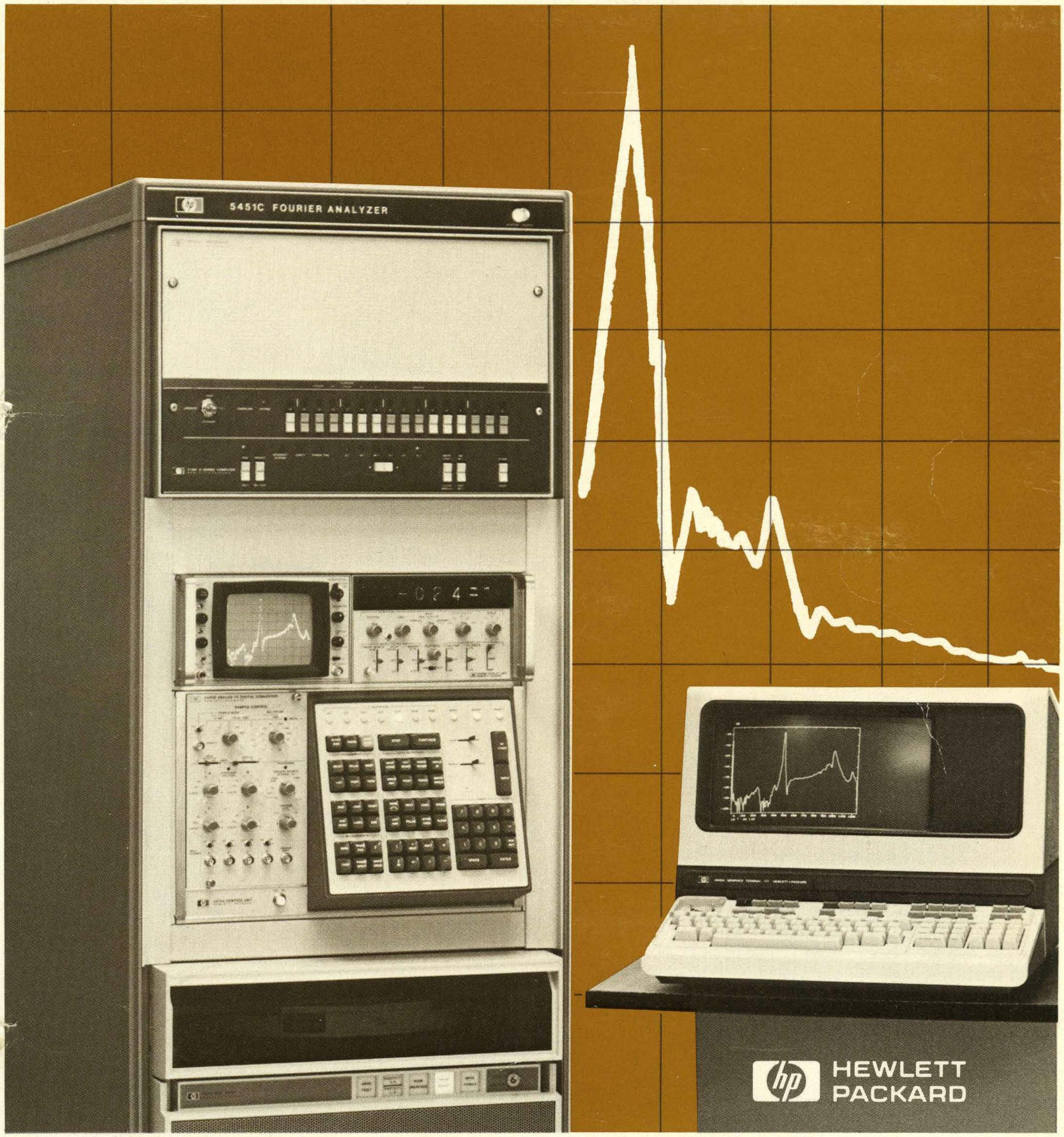
5451C FOURIER ANALYZER SYSTEM

SYSTEM OPERATING MANUAL



HEWLETT PACKARD

SAFETY

This product has been designed and tested according to International Safety Requirements. To ensure safe operation and to keep the product safe, the information, cautions, and warnings in this manual must be heeded.

CERTIFICATION

Hewlett-Packard Company certifies that this product met its published specifications at the time of shipment from the factory. Hewlett-Packard further certifies that its calibration measurements are traceable to the United States National Bureau of Standards, to the extent allowed by the Bureau's calibration facility, and to the calibration facilities of other International Standards Organization members.

WARRANTY

This Hewlett-Packard product is warranted against defects in materials and workmanship for a period of 90 days from date of shipment. During the warranty period, HP will, at its option, either repair or replace products which prove to be defective.

Warranty service of this product will be performed at Buyer's facility at no charge within HP service travel areas. Outside HP service travel areas, warranty service will be performed at Buyer's facility only upon HP's prior agreement and Buyer shall pay HP's round trip travel expenses. In all other cases, products must be returned to a service facility designated by HP.

For products returned to HP for warranty service, Buyer shall prepay shipping charges to HP and HP shall pay shipping charges to return the product to Buyer. However, Buyer shall pay all shipping charges, duties, and taxes for products returned to HP from another country.

LIMITATION OF WARRANTY

The foregoing warranty shall not apply to defects resulting from improper or inadequate maintenance by Buyer, Buyer-supplied software or interfacing, unauthorized modification or misuse, operation outside of the environmental specifications for the product, or improper site preparation or maintenance.

NO OTHER WARRANTY IS EXPRESSED OR IMPLIED. HP SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

EXCLUSIVE REMEDIES

THE REMEDIES PROVIDED HEREIN ARE BUYER'S SOLE AND EXCLUSIVE REMEDIES. HP SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER BASED ON CONTRACT, TORT, OR ANY OTHER LEGAL THEORY.

ASSISTANCE

Product maintenance agreements and other customer assistance agreements are available for Hewlett-Packard products.

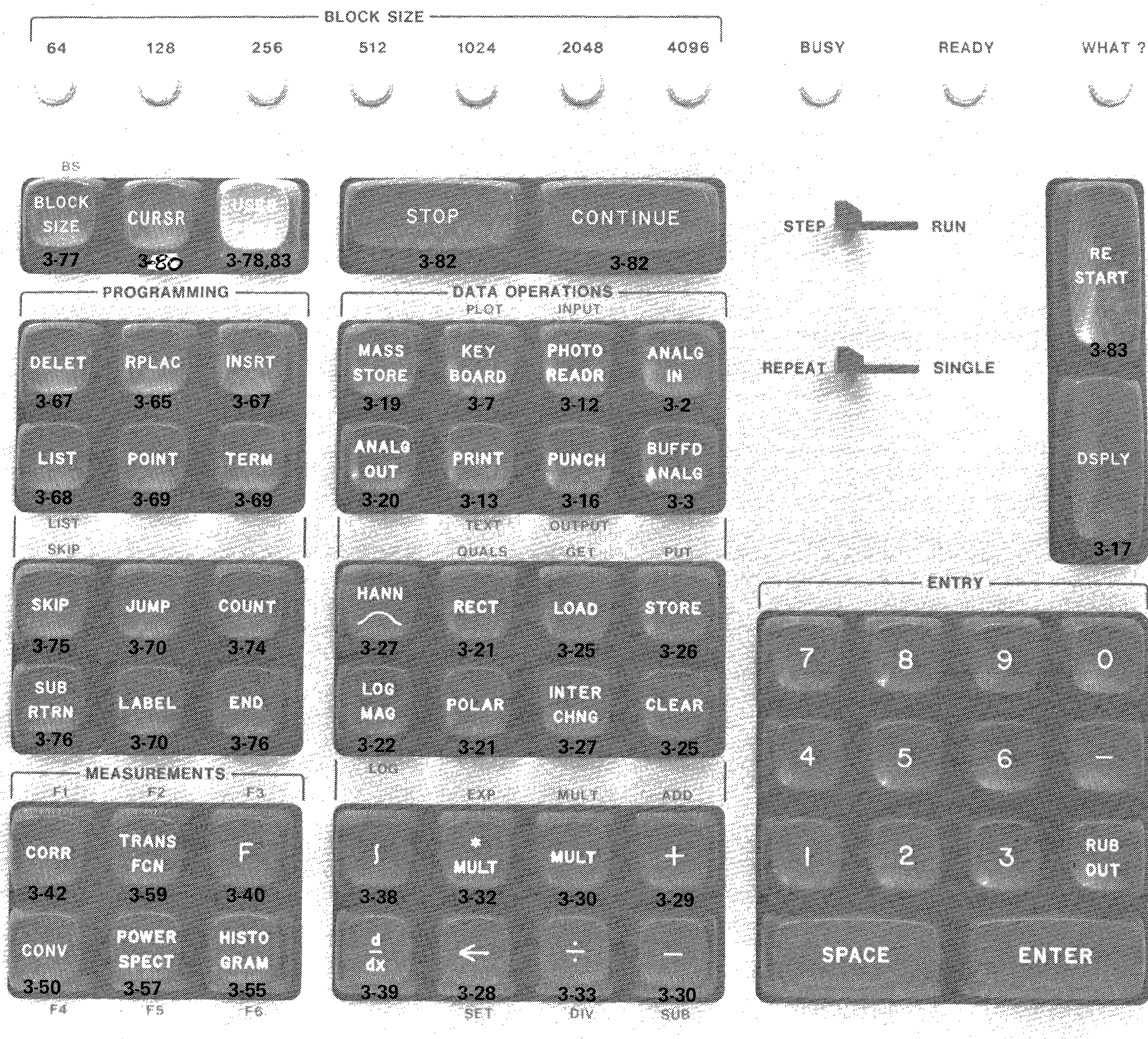
For any assistance, contact your nearest Hewlett-Packard Sales and Service Office. Addresses are provided at the back of this manual.

FEDERAL COMMUNICATIONS COMMISSION RADIO FREQUENCY INTERFERENCE STATEMENT

Warning: This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instructions manual, may cause interference to radio communications. As temporarily permitted by regulation it has not been tested for compliance with the limits for Class A computing devices pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference. Operation of this equipment in a residential area is likely to cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

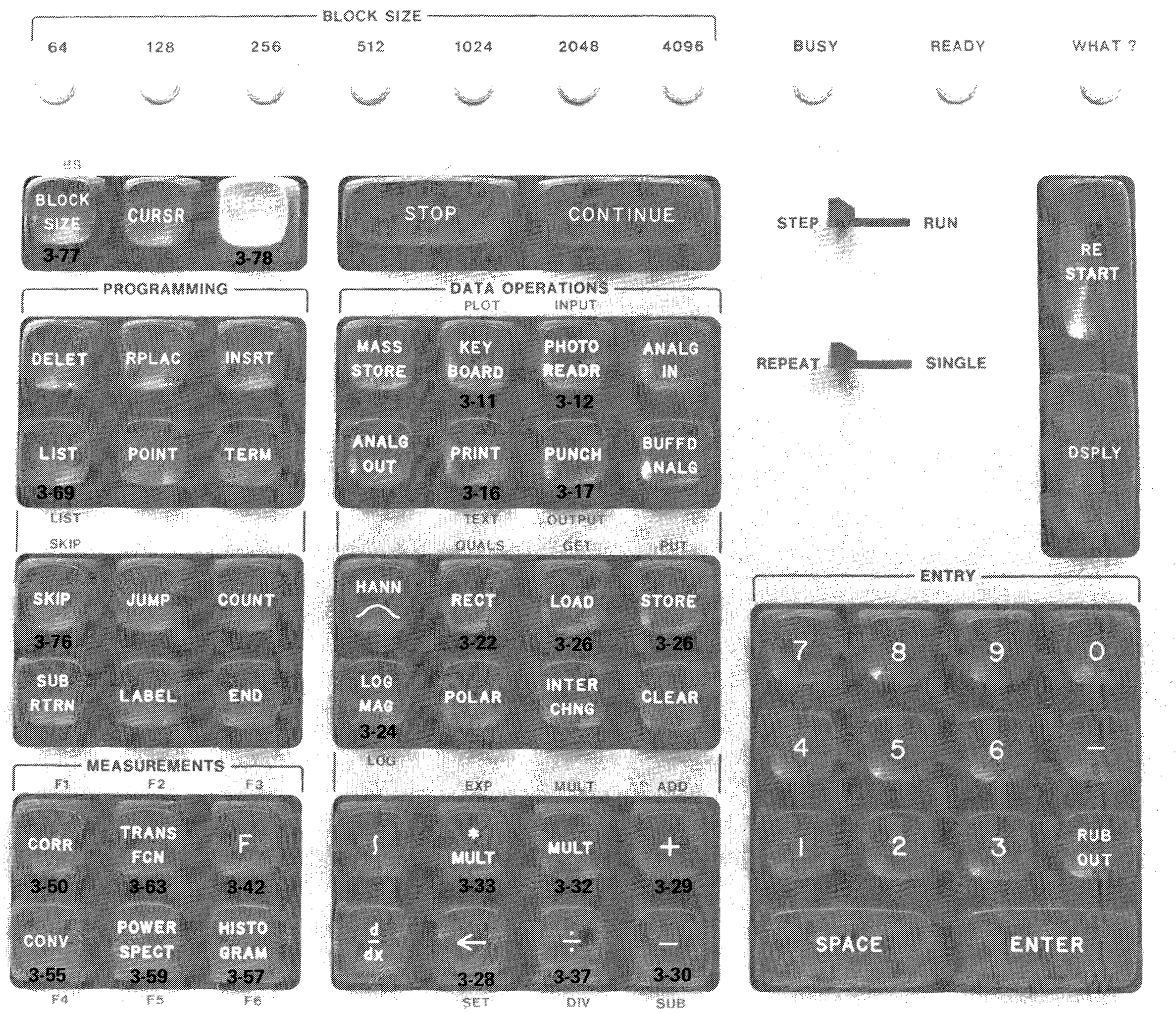
QUICK-KEY REFERENCE

STANDARD KEYS



QUICK-KEY REFERENCE

GOLD KEYS



HEWLETT PACKARD

SYSTEM OPERATING MANUAL

5451C

FOURIER ANALYZER SYSTEM

(This manual reflects information that is compatible with operating software date code 2101 & 2120

Printed: MARCH 1979

Manual Part No. 05451-90529
Microfiche No. 05451-90530

Printed in U.S.A.

HOW TO USE THIS MANUAL

If you have just received your system, note that Section 1 contains a physical description and system specifications.

If you want to know how to turn the system on, load the software, or use the Graphics Terminal, read the first part of Section 2.

If you have never operated the system before, read the introductory theory and perform the teaching demonstrations in Section 2. Examples of all the system features are presented.

If you need information on a specific key command, see Section 3.

If you need information on the measurement features of the system, refer to the remaining Sections and Appendices as necessary (see tab dividers or table of contents).

CAUTION

Interactive measurement programs and text are stored in unprotected records of the Disc in Files 3, 4, and 7. We have provided two programs, a power spectrum program and a transfer function program, which can be called using the Gold Keys F5 and F2 respectively. These programs use the information in these records.

Since you may also use these Gold Keys (or "soft keys") to call programs that you have written, care must be taken not to write over these records when saving your own keyboard programs and text messages.

Appendix D will show you which records are used. It also contains a listing of the two programs provided with the system.

CONTENTS

SECTION 2 — SYSTEM OPERATION *(continued)*

SECTION 5 — BAND SELECTABLE FOURIER ANALYSIS *(continued)*

TABLES

FIGURES

SECTION 1
GENERAL INFORMATION

SYSTEM INFORMATION

The HP 5451C Fourier Analyzer System performs analyses of time and frequency data containing frequencies from dc to 50 kHz. The system analyzes time-series data such as mechanical vibrations, sonar echoes, tidal action, biomedical phenomena such as brain waves and nerve impulses, voltages and currents in electronic systems, and acoustic phenomena such as speech sounds. These analyses may detect signals hidden in noise, as in sonar, or may locate critical functions in complex systems. Both continuous and transient data may be processed.

Keyboard programming allows you to perform the following operations automatically, without special software:

- Forward and inverse Fourier transform
- Magnitude and phase spectrum
- Power and cross power spectrum
- Transfer function
- Coherence function
- Convolution
- Auto and cross correlation
- Hanning and other weighting functions
- Histogram
- Scaling
- Ensemble averaging (time and frequency)

Six editing keys operate an on-line resident editor so that a sequence of steps configured into an automatic measurement procedure may be changed on-line without the need to do off-line editing, compiling and testing. In fact the series of steps or program used to perform a particular operation can be stored on the Disc for easy re-entry into the Fourier Analyzer.

Data input and output is likewise controlled from the Keyboard. Data can be entered in analog form through the 2-channel or 4-channel Analog-to-Digital Converter, or in digital form on magnetic tape or punched paper tape, or manually from the Keyboard.

Results of all operations are displayed on the oscilloscope. In addition, results can be printed out in decimal numbers or plotted on the Graphics Terminal. With optional equipment, data can be punched on paper tape, plotted on an external X-Y plotter, or digital plotter.

The Fourier Analyzer is a completely calibrated system; all displays and data outputs are accompanied by a scale factor relating them to physical units. This calibration results from digital techniques being used in all computations. Computer programming knowledge is not required for operation of the Fourier Analyzer; all operations are controlled through the Keyboard. However, provisions are made to incorporate user-written subroutines written in Assembly language or Fortran into the Fourier system.

SYSTEM CONFIGURATION

A Fourier Analyzer System consists of a basic system plus a number of customer-chosen options. Since the resulting configuration will often be unique, the list of component units is not provided in this manual, but in a separate System SUMMARY REPORT, supplied with the system. This document also tells what Processor I/O slots are used by the various units (ADC, Terminal, etc).

IDENTIFICATION NUMBERS

MODEL NUMBER AND NAME

Each unit in the standard Fourier Analyzer System is identified by model or specification number and name as a separate instrument; these are:

Model 180D Oscilloscope (with Option H51)

Model 54451B Processor

Model 2648A Graphics Terminal

Model 5460A Display Plug-in Unit

Model 5466B ADC Plug-in Unit

Model 5475A Control Unit

Model 7909A Disc Drive

SERIAL NUMBERS

Each Fourier Analyzer System is identified by a 2-section system serial number (5451C-000). The number is on a stick-on plate mounted on the rear of each unit in the system. The 3-digit number is a serial number unique to each Fourier Analyzer System.

In addition, each plug-in unit in the Fourier Analyzer System, including the Processor, is identified by a serial number (0000A00000). The first section is a serial prefix number, used to document changes to the unit; the second portion of the serial number is a number unique to each instrument (of that model number).

Include complete model name, model number, and serial number of any unit or units in all correspondence about your system.

STORAGE AND SHIPMENT

PACKAGING

To protect valuable electronic equipment during storage or shipment, always use the best packaging methods available. Your Hewlett-Packard Sales and Service Office can provide packaging material such as that used for original factory packaging. Contract packaging companies in many cities can provide dependable packaging on short notice.

SCOPE OF THIS MANUAL

OPERATING INFORMATION

This manual contains only the information required to operate the Fourier Analyzer. For service or troubleshooting information, see the 5451C System Service Manual. For information on software programming, see the 5451C System Software Manual.

OTHER MANUALS

To locate specific manuals in the documentation shipped with the system, refer to the System Summary Report for the contents of each binder.

SYSTEM SPECIFICATIONS & CHARACTERISTICS

The specifications in Table 1-1 describe the system's warranted performance. Those items under the heading of "Characteristics" go beyond the guaranteed specifications and give typical performance for some additional parameters and operations. These are included only to give you information which may be useful in applying the system.

Table 1-1. System Specifications and Characteristics

SPECIFICATIONS
(Specifications describe the standard system's warranted performance.)

ANALOG-TO-DIGITAL CONVERTER

Input Voltage Range: ±0.125V to ±8V peak in steps of 2.
Input Coupling: dc or ac.
Input Channels: 2 channels wired for 4 standard. 4 channels optional with plug-in cards.
Resolution: 12 bits including sign.
Input Frequency Range: dc to 50 kHz, 5 Hz to 50 kHz, ac coupled (100 kHz optional).
Sample Rate:
Internal: 100 kHz max. (1, 2, 3, or 4 channels simultaneously). (200 kHz optional on 1, 2, 3, or 4 channels.) [50 kHz max. (3 or 4 channels simultaneously).†]
External: An external time base may be used to allow external control of the sampling rate up to 100 kHz (200 kHz optional). One sample can be taken for each clock pulse (TTL level).
Internal Clock Accuracy: ±0.01%.

DISPLAY UNIT

Vertical Scale Calibration: Data in memory is automatically scaled to give a maximum on-screen calibrated display. The scale factor is given in volts/division, volts^2/division, or in dB offset.
Log Display Range: 80 dB with a scale factor ranging from 0 to +998 dB. Offset selectable in 4 dB steps.
Linear Display Range: ±4 divisions with scale factor ranging from 1×10^{-512} to 5×10^{512} in steps of 1, 2, and 5.
Digital UP/DOWN Scale: Allows 8 up-scale and 2 down-scale steps (calibrated continuous scale factor).
Horizontal Scale Calibration:
Linear Sweep Length: 10, 10.24 or 12.8 divisions.
Log Horizontal: 0.5 decades/division.
Markers: Intensity markers every 8th or every 32nd point.

BASE SOFTWARE

Transform Accuracy: The expected rms value of computational error introduced in either the forward or inverse FFT will not exceed 0.1% of the rms value of the transform result.
Dynamic Range: ≥75 dB for a minimum detectable spectral component in the presence of one full scale spectral component after twenty ensemble averages for a block size of 1024.

EXECUTION TIMES*

Fourier Transform: <55 ms
Stable Power Spectrum Average: <80 ms
Stable Tri-Spectrum Average: <220 ms

REAL TIME BANDWIDTHS*

Fourier Transform: >7.5 kHz
Stable Power Spectrum Average: 5.4 kHz
Stable Tri-Spectrum Average: 1.9 kHz

MASS STORAGE SOFTWARE

MAXIMUM REAL TIME DATA ACQUISITION RATE (Single Channel):
BS 256: 17K
BS 1024: 71K (25 kHz†)
BS 2048: 144 kHz
BS 4096: 132 kHz (30 kHz†)

OFF-LINE BSFA SOFTWARE

Center Frequency Range: dc to one-half the Real Time Data Acquisition Rate.
Center Frequency Resolution: Continuous resolution to the limit of the frequency accuracy for center frequencies ≥0.02% of the sampling frequency.
Frequency Accuracy: ±0.01%
Bandwidth Selection: In steps of f/5n where n = 2, 3, 4, etc.
Max. Resolution Enhancement: >400
Dynamic Range:** 90 dB from peak out-of-band spectral component to the peak level of the passband noise.
80 dB from peak in-band spectral component to the peak level of the passband noise.
Out-of-Band Rejection: >90 dB
Passband Flatness of the Digital Filter: ±0.01 dB

ENVIRONMENTAL CONDITIONS

Temperature Range: 10°C to 40°C (104°F).

**For band limited random noise type signals at block size 1024, no display, no Hanning.*
***After eight ensemble averages of a power spectrum at block size 1024. Reduced by 10 dB at the exact center of the band.*
†These rates apply to systems with modules 5466B and 54451A/B having a serial prefix lower than 1842.

NOTE — Specifications subject to change without notification.

OTHER MANUALS

To locate specific manuals in the documentation shipped with the system, refer to the System Summary Report for the contents of each binder.

SYSTEM SPECIFICATIONS & CHARACTERISTICS

The specifications in Table 1-1 describe the system's warranted performance. Those items under the heading of "Characteristics" go beyond the guaranteed specifications and give typical performance for some additional parameters and operations. These are included only to give you information which may be useful in applying the system.

Table 1-1. System Specifications and Characteristics

SPECIFICATIONS
(Specifications describe the standard system's warranted performance.)

ANALOG-TO-DIGITAL CONVERTER

Input Voltage Range: ±0.125V to ±8V peak in steps of 2.
Input Coupling: dc or ac.
Input Channels: 2 channels wired for 4 standard. 4 channels optional with plug-in cards.
Resolution: 12 bits including sign.
Input Frequency Range: dc to 50 kHz, 5 Hz to 50 kHz, ac coupled (100 kHz optional).
Sample Rate:
Internal: 100 kHz max. (1, 2, 3, or 4 channels simultaneously). (200 kHz optional on 1, 2, 3, or 4 channels.) [50 kHz max. (3 or 4 channels simultaneously).†]
External: An external time base may be used to allow external control of the sampling rate up to 100 kHz (200 kHz optional). One sample can be taken for each clock pulse (TTL level).
Internal Clock Accuracy: ±0.01%.

DISPLAY UNIT

Vertical Scale Calibration: Data in memory is automatically scaled to give a maximum on-screen calibrated display. The scale factor is given in volts/division, volts^2/division, or in dB offset.
Log Display Range: 80 dB with a scale factor ranging from 0 to +998 dB. Offset selectable in 4 dB steps.
Linear Display Range: ±4 divisions with scale factor ranging from 1×10^{-512} to 5×10^{512} in steps of 1, 2, and 5.
Digital UP/DOWN Scale: Allows 8 up-scale and 2 down-scale steps (calibrated continuous scale factor).
Horizontal Scale Calibration:
Linear Sweep Length: 10, 10.24 or 12.8 divisions.
Log Horizontal: 0.5 decades/division.
Markers: Intensity markers every 8th or every 32nd point.

BASE SOFTWARE

Transform Accuracy: The expected rms value of computational error introduced in either the forward or inverse FFT will not exceed 0.1% of the rms value of the transform result.
Dynamic Range: >75 dB for a minimum detectable spectral component in the presence of one full scale spectral component after twenty ensemble averages for a block size of 1024.

EXECUTION TIMES*

Fourier Transform: <55 ms
Stable Power Spectrum Average: <80 ms
Stable Tri-Spectrum Average: <220 ms

REAL TIME BANDWIDTHS*

Fourier Transform: >7.5 kHz
Stable Power Spectrum Average: 5.4 kHz
Stable Tri-Spectrum Average: 1.9 kHz

MASS STORAGE SOFTWARE

MAXIMUM REAL TIME DATA ACQUISITION RATE (Single Channel):
BS 256: 10 kHz
BS 1024: 39 kHz (25 kHz†)
BS 4096: 80 kHz (30 kHz†)

OFF-LINE BSFA SOFTWARE

Center Frequency Range: dc to one-half the Real Time Data Acquisition Rate.
Center Frequency Resolution: Continuous resolution to the limit of the frequency accuracy for center frequencies >0.02% of the sampling frequency.
Frequency Accuracy: ±0.01%
Bandwidth Selection: In steps of f/5n where n = 2, 3, 4, etc.
Max. Resolution Enhancement: >400
Dynamic Range:** 90 dB from peak out-of-band spectral component to the peak level of the passband noise.
80 dB from peak in-band spectral component to the peak level of the passband noise.
Out-of-Band Rejection: >90 dB
Passband Flatness of the Digital Filter: ±0.01 dB

ENVIRONMENTAL CONDITIONS

Temperature Range: 0°C to 40°C (104°F).

For band limited random noise type signals at block size 1024, no display, no Hanning.
***After eight ensemble averages of a power spectrum at block size 1024. Reduced by 10 dB at the exact center of the band.*
†These rates apply to systems with modules 5466B and 54451A/B having a serial prefix lower than 1842.

SUPPLEMENTAL CHARACTERISTICS

(Supplement Characteristics are intended to provide useful information for system applications by giving typical, but not warranted, performance parameters.)

ANALOG-TO-DIGITAL CONVERTER

Input Impedance: 1 MΩ in parallel with <75 pf.
Sample Rate Control:
Maximum Frequency Mode: Maximum frequency selectable from 0.1 Hz to 50 kHz (100 kHz optional) in steps of 1, 2.5, 5. This mode automatically sets maximum frequency independent of block size.
Frequency Resolution Mode: Frequency resolution selectable from 0.2 MHz to 100 Hz in steps of 1, 2, 5. This mode automatically sets frequency resolution and sample record length independent of block size.

DISPLAY UNIT

Data may be displayed in single sweeps or refreshed continuously in the following forms:

Y AXIS	X AXIS
Amplitude	Time (Linear or Log)
Real Part	Frequency (Linear or Log)
Imaginary Part	Frequency (Linear or Log)
Magnitude (Linear or Log)	Frequency (Linear or Log)
Phase	Frequency (Linear or Log)
Imaginary Part (NYQUIST PLOT)	Real Part

BASE SOFTWARE

System Accuracy and Range: The Fourier Transform is implemented using conditional scaling for maximum accuracy with no data overflows allowed. All calculations use floating point arithmetic on a block basis with full 16- and 32-bit arithmetic where applicable.
Maximum Block Size: 4096 time domain points.
Minimum Block Size: 64 time domain points.
Data Space: 28K words (16K words standard in systems with serial prefix below 1842).
Program Space: 32K words.

BSFA SOFTWARE

Maximum BSFA Blocksize: 1024 time domain points. (2048 with option 670.) See also Table 5-1 in BSFA section of manual.

MASS STORAGE

Disc Unit:
Capacity: 9.8 megawords
Data Transfer: 7.5 million bits/second
Discs: 2 (1 fixed, 1 removable)
% of Real Time at 100 kHz ADC Sampling Rate (Single Channel):
BS 256: 17% **BS 1024:** 71% (25%†) **BS 4096:** 100% (30%†)
Number of Records Per File:
Data Block: 1171 records (4096 blocksize maximum/record).
ADC Throughput: 770 records (4096 blocksize max./record).
Program Stack: 340 records (470 steps/record).
ASCII Text: 3700 records (128 words/record).
Index: 160 records (10 pointers/record).
System Coreload: 4 records (32K words/record).
Common: 650 records (256 words/record).
Overlay: 30 overlays (8K words maximum, 7K words maximum with 265).

OPTIONS 265

The magnetic tape option is used for ADC Throughput only.

Maximum Real Time Data Acquisition Rate (Single Channel):
Opt. 265
BS 256: 9 kHz
BS 1024: 21 kHz
BS 4096: 30 kHz

Number of Tracks: 9
Read/Write Speed: 45 ips
Density: 1600 bpi.
Data Transfer Rate: 72K cps max.
Rack Height: 610 mm (24 in.)

POWER REQUIREMENTS, SIZE, WEIGHT

Power Source: 115/230 volts ±10%, 50/60 Hz. 2000 watts typical for base system.
Size:
Panel Height: 1422 mm (56 in.)
Overall: 1631 mm (64¼ in.), height; 533 mm (21 in.), width; 762 mm (30 in.), depth.
Weight: Net weight for a base system (excluding terminal) — 197.5 kg (433 lbs.).
Price and Ordering Information: Consult the 5451C Fourier Analyzer System's Ordering Information Guide.

SYSTEM INSTALLATION

Included in the 5451C System is on-site instalation. On installation, a trained Hewlett-Packard representative will perform an operational demonstration to ensure that the system is functioning normally.

TRAINING

A course on Fourier analysis and system operation is optionally available at HP's Santa Clara, California facility. On-site training can also be provided, if desired.

†These percentages apply to systems with 5466B and 54451A/B modules having a serial prefix lower than 1842.

SUPPLEMENTAL CHARACTERISTICS

(Supplemental Characteristics are intended to provide useful information for system applications by giving typical, but not warranted, performance parameters.)

ANALOG-TO-DIGITAL CONVERTER

Input Impedance: 1 MΩ in parallel with <75 pf.

Sample Rate Control:

Maximum Frequency Mode: Maximum frequency selectable from 0.1 Hz to 50 kHz (100 kHz optional) in steps of 1, 2.5, 5. This mode automatically sets maximum frequency independent of block size.

Frequency Resolution Mode: Frequency resolution selectable from 0.2 mHz to 1000 Hz in steps of 1, 2, 5. This mode automatically sets frequency resolution and sample record length independent of block size.

Input Mode: There is a buffered and non-buffered analog mode. In the buffered mode, other operations can be performed on previously collected data while the ADC collects current input data into a buffer.

DISPLAY UNIT

Data may be displayed in single sweeps or refreshed continuously in the following forms:

Y Axis	X Axis
Amplitude	Time (Linear or Log)
Real Part	Frequency (Linear or Log)
Imaginary Part	Frequency (Linear or Log)
Magnitude (Linear or Log)	Frequency (Linear or Log)
Phase	Frequency (Linear or Log)
Imaginary Part (NYQUIST PLOT)	Real Part

CRT Positioning: Three markers to aid in adjusting trace position as well as vertical and horizontal controls are provided for display positioning.

Origin: Left edge of display, zero amplitude.

+FS: Positive full scale, center of display.

−FS: Negative full scale, center of display.

Analog Plotter Output: Any displayed data can be output to a plotter or remote oscilloscope. HP 10640B interface is required for operation.

Amplitude: 0.5V per oscilloscope display division.

Linearity: 0.1% full scale.

Interpolation: Linear interpolation in 0.05% steps.

Type of Display: Points, bars, or continuous (interpolation).

BASE SOFTWARE

System Accuracy and Range: The Fourier Transform is implemented using conditional scaling for maximum accuracy with no data overflows allowed. All calculations use floating point arithmetic on a block basis with full 16- and 32-bit arithmetic where applicable.

Data Word Size: 16-bit imaginary with 32 bits preserved for double precision functions. Division, addition, or subtraction operations performed in 16 or 32 bits depending on data.

Maximum Block Size: 4096 time domain points.

Minimum Block Size: 64 time domain points.

Permanent Data Space: 28K words (16K words standard in systems with serial prefix below 1842, optionally expandable to 28K with option 011).

Permanent Program Space: 32K words.

BSFA SOFTWARE

Maximum BSFA Blocksize: 1024 time domain points. (2048 with option 670.) See also Table 5-1 in BSFA section of manual.

MASS STORAGE

Disc Unit:

Capacity: 2.45 megawords

Data Transfer: 2.5 million bits/second

Discs: 2 (1 fixed, 1 removable)

% of Real Time at 100 kHz ADC Sampling Rate (Single Channel):
BS 256: 10% **BS 1024:** 39% (25%†) **BS 4096:** 80% (30%†)

Number of Records Per File:

Data Block: 214 records (4096 blocksize maximum/record).

ADC Throughput: 199 records (4096 blocksize max./record).

Program Stack: 138 records (470 steps/record).

ASCII Text: 690 records (128 words/record).

Index: 69 records (10 pointers/record).

System Coreload: 4 records (32K words/record).

Common: 286 records (256 words/record).

Overlay: 20 overlays (8K words maximum, 7K words maximum with option 261 or 265).

OPTIONS 261 & 265

The magnetic tape options are used for ADC Throughput only.

Maximum Real Time Data Acquisition Rate (Single Channel):

Opt. 261		Opt. 265	
BS 256:	6 kHz	**BS 256:**	9 kHz
BS 1024:	12 kHz	**BS 1024:**	21 kHz
BS 4096:	15 kHz	**BS 4096:**	30 kHz

Number of Tracks: 9

Read/Write Speed: 45 ips

Density: Option 261, 800 bpi; Option 265, 1600 bpi.

Data Transfer Rate: Option 261, 36K cps max.; Option 265, 72K cps max.

Rack Height: 610 mm (24 in.)

POWER REQUIREMENTS, SIZE, WEIGHT

Power Source: 115/230 volts ±10%, 50/60 Hz. 1800 watts typical for base system.

Size: Dimensions are for a typical system (excluding cabinet and terminal).

Height: 771 mm (28 in.); **Width:** 425 mm (16¾ in.);

Depth: 616 mm (24¼ in.).

Cabinet:

Panel Height: 1422 mm (56 in.)

Overall: 1631 mm (64¼ in.), height; 533 mm (21 in.), width; 762 mm (30 in.), depth.

Weight: Net weight for a base system (excluding terminal) — 163.3 kg (358 lbs.).

Price and Ordering Information: Consult the 5451C Fourier Analyzer System's *Ordering Information Guide.*

SYSTEM INSTALLATION

Included in the 5451C System is on-site installation. On installation, a trained Hewlett-Packard representative will perform an operational demonstration to ensure that the system is functioning normally.

TRAINING

A course on Fourier analysis and system operation is optionally available at HP's Santa Clara, California facility. On-site training can also be provided, if desired.

†These percentages apply to systems with modules 5466B and 54451A/B having a serial prefix lower than 1842.

SECTION 2
SYSTEM OPERATION

This section describes how to operate the Fourier Analyzer System and should be read before attempting to use the rest of this manual.

The contents of this section are presented in three major parts: turning the system on and off, basic theory of operation, and demonstrations of use.

After reading this section you should be adequately trained to begin making measurements, and also should have learned where to refer in the rest of the manual for detailed descriptions.

TURN-ON/TURN-OFF PROCEDURE

These paragraphs provide instructions for initial and routine turn-on/turn-off of the Fourier Analyzer System. Included are instructions for loading the system software from the Disc.

PREPARATION

The following assumes that the system has been installed and checked out. To verify proper system cable interconnection, refer to the diagram in Section 1 of the 5451 System Service Manual. The I/O cables must be connected in proper sequence from the rear of the Processor to all internal and peripheral units; also, the Terminal data cable must be connected at the rear of the system.

FIRST TIME TURN-ON

The following can be considered a first-time turn-on procedure. Thereafter leave all subsystem power switches on and simply use the cabinet power switch to turn the system on and off.

1. Turn on cabinet power by pressing the SYSTEM ON/OFF button in upper right corner of cabinet. (Check first that the RUN|STOP switch on the Disc has been left in the STOP position.)

2. Open the front panel of the Processor with the key provided. Slide the LOCK/OPERATE switch to the OPERATE position (next to POWER switch).

3. Press the Processor rocker POWER switch to the ON position. Before locking the front panel, verify the following condition:

   Each time power is applied, the Processor performs a brief self-test (about 1 second). When this is complete, note the state of the A B M T P S register lights on the front panel. Only the T light should be on — if other (or all) lights are on, switch the Processor to OFF and back to ON and only the T light should appear.

4. Turn on power to the Display Unit by placing the LINE toggle switch in the up position (located in lower left corner of CRT).

5. Turn on power to the other Units by pressing white OFF/ON pushbutton below the ADC (lighted when on).

6. Turn on the Terminal by pressing the rear LINE rocker switch to ON.

NORMAL TURN-ON

The following procedure should be used after the system has been installed and checked out.

1. Check that the RUN/STOP switch on the Disc is in the STOP position.

2. Turn on system power by pressing to light the SYSTEM ON/OFF button in upper right corner of cabinet.

3. Pull open Disc door and insert System Software Disc Pack. Close door firmly and set switch to . Wait for DRIVE READY light.

4. On the Processor, check that the RUN light is off and the T register light (only) is on.

5. Load the software as directed in the next paragraph. (An abbreviated command sequence is given below:)

   a. Disc RUN/STOP switch to RUN, DRIVE READY light on.

   b. S register to 111700octal. Press STORE, PRESET, IBL/TEST, and RUN.

      Look for halt 102076, press RUN.

      Look for halt 102077, press RUN.

   c. Look for message on Terminal.

LOADING SYSTEM SOFTWARE FROM THE DISC

1. Check that power is applied to all units in the system.

2. Pull open the Disc door and insert the System Software Disc Pack. (The DOOR UNLOCKED light should be on and the RUN/STOP switch to the STOP position).

3. Place the RUN/STOP switch on the Disc to the RUN position.

4. Wait until the DRIVE READY light comes on before proceeding.

5. On the Processor front panel, press the ◄▭▭► rocker switch repeatedly (in either direction) until only the S register light is on.

6. Press CLEAR DISPLAY and enter the octal number 111700in the display register (bits 15, 12, 9, 8, 7, and 6 to on, all others off).

7. Press STORE, PRESET, IBL/TEST, and RUN.

   The Processor should halt with 102076_8 in the display register (bits 15, 10, 5, 4, 3, 2, 1 on). If not, repeat from step 5 above.

8. Press RUN. The Processor should halt with 102077_8 in the display register (bits 15, 10, 5, 4, 3, 2, 1, 0 on).

9. Press RUN. The Terminal should print the following message:

   BLOCKS # / SIZE / SPACE

   N XXXX ZZZZZ

   where:

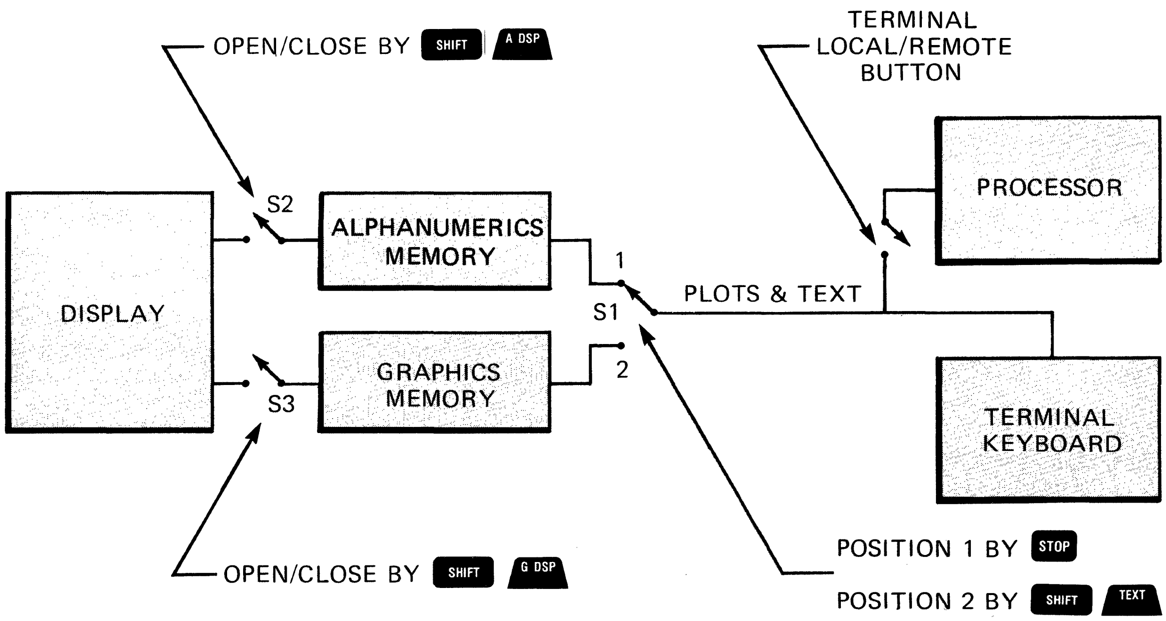
   N = Number of data blocks currently in system
   XXXX = Size of the N data blocks
   ZZZZZ = Total data space available

The READY light on the Keyboard should now be on, the corresponding block size light should be on, and a block of data will be presented on the Display Unit. The system is now ready for use.

SETTING UP YOUR 2648A GRAPHICS TERMINAL

The Graphics Terminal is designed to display both printed text and graphic plots. You can display the text or graphic plots individually or at the same time. If you are already familiar with this Terminal, proceed to the paragraph on Turn-Off. Because many different terminal modes are available, you need to know how to set up your Terminal in the modes the system requires.

Within the Terminal are two independent memories — the ALPHANUMERICS memory and the GRAPHICS memory. As the operator you have control via the Terminal keyboard over which memory receives data from the Processor and which memory is displayed on the Terminal screen. Below is a simple block diagram of the Terminal, showing three figurative switches that are set by pressing Terminal keys.



Switch S1 selects the memory into which text and plot data are entered. The Alphanumerics Memory is used for text only and the Graphics Memory is used for text and plot data. To set switch S1 to the Alphanumerics Memory, press STOP on the Terminal. This now sends all the information (text and plots) to the Alphanumerics Memory. To set switch S1 to the Graphics Memory, press both the SHIFT and TEXT keys. (Note that holding the SHIFT key while pressing one of the Graphics Control Group keys [on near right] performs the function written on the front of the key.)

Figurative switches S2 and S3 in the block diagram control which memory is displayed. To activate switch S2 to enable/disable displaying the Alphanumerics Memory, press the SHIFT and A DSP (Alphanumerics Display) keys together. These keys alternately *enable or disable* the memory to the display depending on the previous state. The "flashing bar" cursor is displayed when the Alphanumerics Memory is enabled.

To activate switch S3 to enable/disable displaying the Graphics Memory, press the SHIFT and G DSP (Graphics Display) keys at the same time. Pressing the G CURSOR key will turn the Graphics Cursor on and off so you can tell if the Graphics Display is enabled.

To clear the Alphanumerics Display press ↖ followed by CLEAR DSPLY (in the Display Control Group on far right). If you wish to clear the Graphics Display, press SHIFT CLEAR . These keys clear the appropriate memory regardless of the condition of any of the switches in the diagram above.

Practical Exercise

Let's go through a short practice session to get better acquainted with your Terminal. First we will set up the Terminal switches and buttons as follows to allow it to run with the Fourier System:

DUPLEX to FULL
PARITY to NONE
BAUD RATE to 2400
REMOTE depressed
CAPS LOCK depressed
AUTO LF depressed
All others UP (No lights should be ON)

We put the Terminal in its initial power-on state by pressing the **RESET TERMINAL** button twice very quickly (within 0.5 seconds). For this exercise only we will operate the Terminal in the "LOCAL" mode (**REMOTE** button in the UP position). Press the following keys on the Terminal:

STOP This directs all text from the Keyboard to the Alphanumerics Memory. (The initial power-on state directs text to the Graphics Memory).

SHIFT **G DSP** This turns off the Graphics Display (The initial power-on state displays both the Graphics and Alphanumerics Memories).

Now type "ALPHANUMERICS MEMORY" on the Terminal. In this sequence of commands we have directed the text to the Alphanumerics Memory (by pressing **STOP**) and have disabled the Graphics Memory (by pressing **SHIFT** **G DSP**). Next we will disable the Alphanumerics Memory and enable the Graphics Memory to receive and display text. We do this by pressing the following keys:

SHIFT **A DSP** This turns off the Alphanumerics Display.

SHIFT **G DSP** This enables the Graphics Display.

NOTE

The TERMINAL READY message is in the Graphics Memory.

SHIFT **TEXT** This has now enabled all text to go to the Graphics Memory and turned on the Graphics Cursor (+).

SHIFT **CLEAR** This clears the Graphics Memory of the "TERMINAL READY" message and places the Graphics Cursor in the upper left-hand corner of the display.

Press **▼** and hold momentarily to move the cursor to the left center of the screen — exact position is not critical.

Type "GRAPHICS MEMORY" on the Terminal.

We have now shown how we can direct our information to the Alphanumerics or Graphics Memory with the **STOP** and **SHIFT** **TEXT** keys respectively. At this point we can also control which memory we are displaying on the screen with the **SHIFT** **A DSP** and **SHIFT** **G DSP** keys. Now let's show that we can also display both memories at the same time:

Press **SHIFT** **A DSP**

Notice the difference in character, shape, and size. This is because the Alphanumerics Memory has a "standard" character set. However, the characters displayed from the Graphics Memory are "plotted" or drawn on the screen (in capital letters only).

This exercise should be sufficient to get you started on your Terminal. There are a number of other functions available to you on the Terminal and these can be found in your 2648A Graphics Terminal User's Manual and Reference Manual. (See the System Configuration Notice for binder number.)

Summary

For most operations within the Fourier System, information is directed from the Processor to the Alphanumerics Memory and the Alphanumerics Memory is displayed. This condition is initially set up by entering:

RESET TERMINAL (twice) Command sequence for any state → ALPHANUMERICS (POWER UP)

STOP

SHIFT G DSP

When subsequent Graphics plots are to be made on the Terminal, the Terminal may be placed in the Graphics mode by entering:

SHIFT TEXT Command sequence for ALPHANUMERICS → GRAPHICS.

G CURSOR

SHIFT A DSP

SHIFT G DSP

To return to the normal Alphanumerics mode, the command sequence is:

STOP Command sequence for GRAPHICS → ALPHANUMERICS.

SHIFT G DSP

SHIFT A DSP

The Terminal may be switched between the Alphanumerics and Graphics modes by using the above two command sequences.

As you begin the demonstrations that follow, keep in mind that if your Terminal is not set up correctly you may not see the proper result. Therefore, before continuing be sure you have a good understanding of your Terminal and how to enable the Alphanumerics and Graphics displays.

NOTES

The handshake operation for the Graphics memory is limited in speed. Therefore, the maximum Graphics Memory BAUD RATE is 2400. Fourier systems shipped after mid 1979 include 2648A Graphics Terminals that have been modified to allow the BAUD RATE to be set at 9600. These products may be identified by the "E55" designation on the product tag on the inside of the rear terminal cover.

Remember that the REMOTE button must be in the down position when operating with the Fourier System.

TURN OFF

The Processor has a volatile memory so all data and programs which you wish to save must be stored on the Disc. (This is true anytime since momentary power interruptions may also erase Processor memory.) Therefore, before performing the steps below, use the procedures described in Section 4 to store needed information on the Disc.

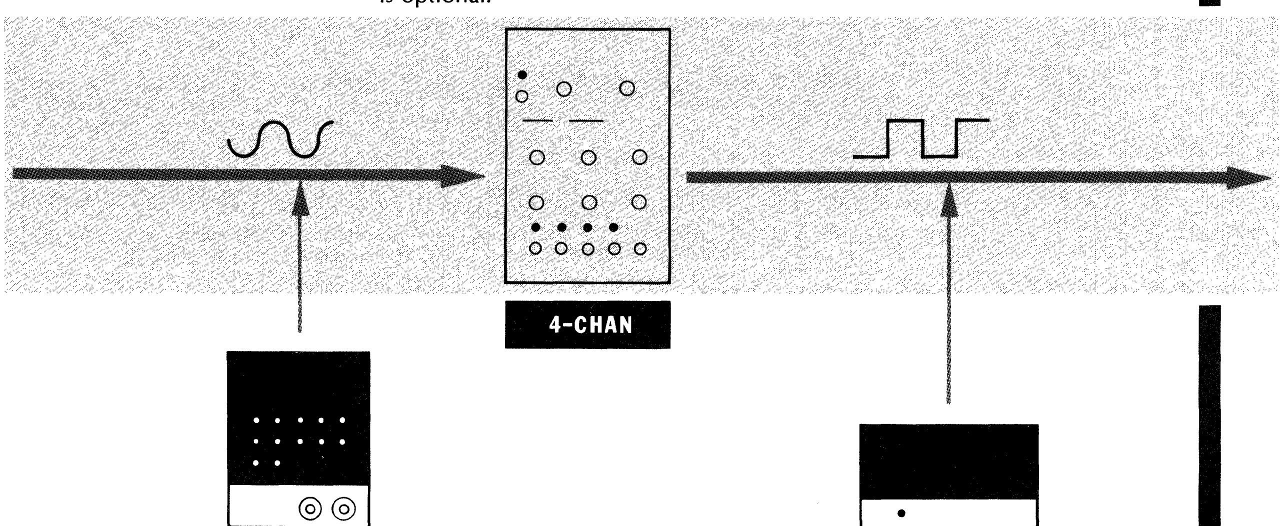
1. Place the RUN|STOP switch on the Disc to the STOP position.
2. When the DOOR UNLOCKED light comes on (about 1 minute) pull down to open the door on the front of the Disc and remove the disc pack. Close the door and store the disc pack on its side in a clean place.
3. Turn off main cabinet power by pressing the SYSTEM ON/OFF switch in the upper right corner of the cabinet.
4. Check that power is off to all peripherals not plugged into cabinet power sockets.

INTRODUCTION TO SYSTEM OPERATION

The 5451C FOURIER ANALYZER provides dual-channel frequency domain analysis of any time signal from dc to 50 kHz. The system is self-contained with an ADC to digitize the input signal, a Processor to analyze the signal, and a Display to show the result. Further, a Keyboard is included to allow operator control over the program, a Graphics Terminal displays plots and listings of the data, and a Disc Drive provides data and program storage. The relationship of these units is illustrated below. Optional peripherals that have been added to enhance operation are shown outside of the main data flow.

INPUT

The 5466B ADC samples the analog input signal at a selectable rate and converts each sample into a 12-bit digital word. The standard ADC accepts two inputs for cross measuring, and a 4-channel input is optional.



The 54440A Filter prevents aliasing effects in the measurement signal by providing ten programmable ranges of low-pass filtering.

The 54470A Pre-Processor adds on-line Band-Selectable Fourier Analysis (zooming effect). This BSFA technique increases resolution and dynamic range in any narrow band selected between dc and 50 kHz.

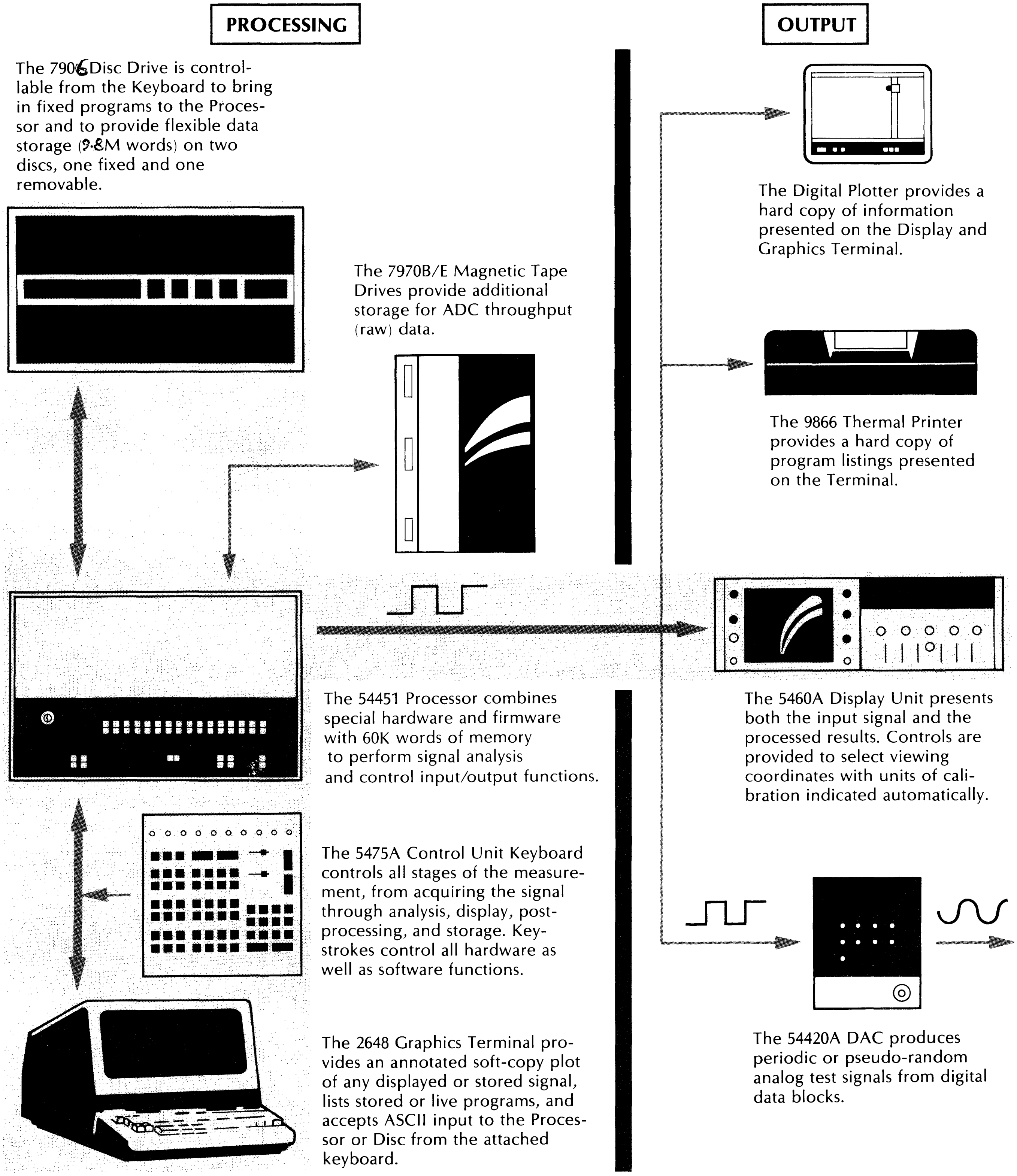


Figure 2-1. System Data Flow Diagram

LEARNING TO OPERATE THE FOURIER ANALYZER

This section is organized to teach the operator the full capabilities of the Fourier Analyzer System. First some theory of signal sampling and data structure is discussed followed by demonstrations of Keyboard commands. All of the features of the system are presented, along with uses of all the Keyboard groups.

The following demonstrations proceed from basic functions to the more complex operations, concluding with Keyboard Programming. This latter feature makes the fullest use of the power of the system, allowing you to automate repetitive measurements, or tailor the system for specific needs. It is important, then, that the demonstrations be followed in the sequence presented.

After performing the demonstrations, you should be ready to make your own measurements, referring as necessary to later sections for specific information. Section 3 contains complete descriptions of every key arranged by functional order (see Quick-Key Reference or Table of Contents for page numbers). Sections 4 through 8 contain detailed reference material on the major features of the Fourier Analyzer.

ADC SAMPLING PARAMETERS

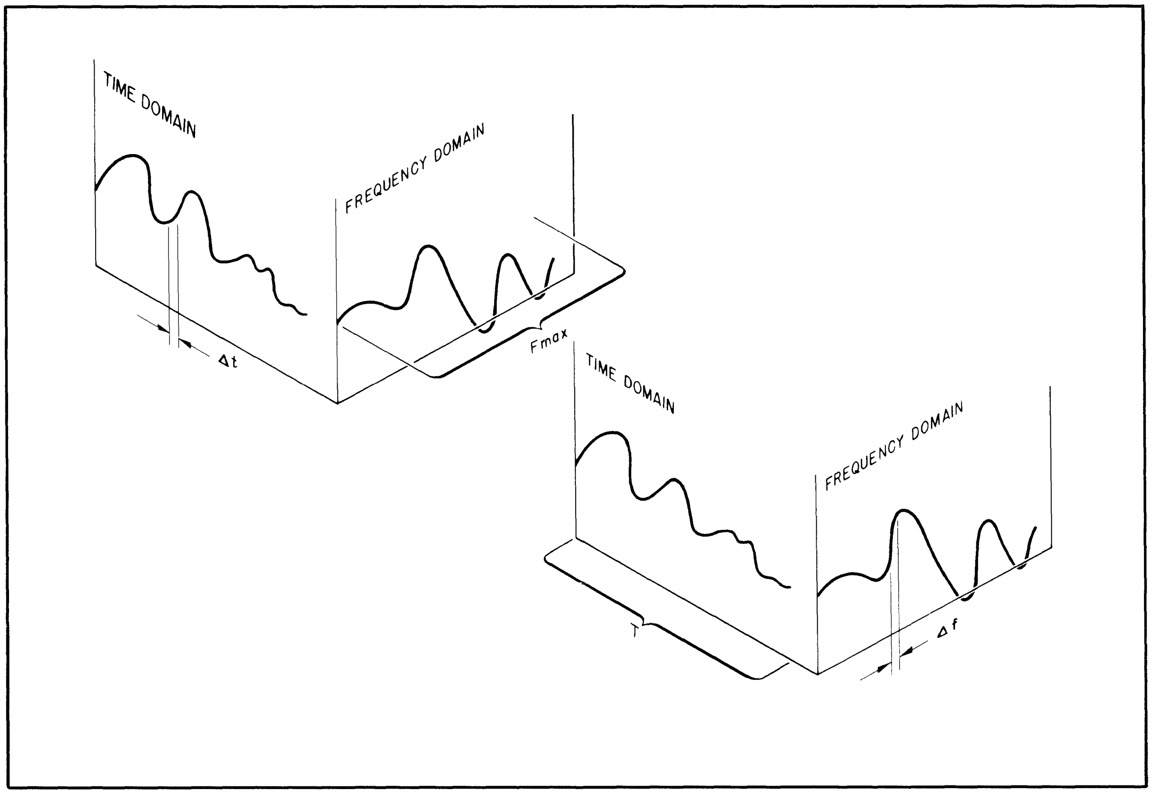
The input signal to be measured is sampled at discrete time intervals by the ADC. Each sample is converted to a digital data word which can then be processed within the system. The following paragraphs explain the sampling technique employed in the Analyzer and how the switches on the ADC control the rate and number of samples. Unfold Figure 2-6, ADC Front Panel Controls, at the end of this subsection for reference as you read these paragraphs.

Figure 2-2 shows a sampled time record and its corresponding spectrum. Terms which define the scale of the time and frequency records are Δt, T, F_{max}, F_s, Δf. These terms are related to each other as follows:

The sample frequency (F_s) and the sample interval (Δt) are reciprocals of each other:

$$F_s = 1/\Delta t$$

Figure 2-2. ADC Sampling Parameters



Shannon's sampling theorem requires that there be 2 samples for the highest frequency in the record, so that:

$$F_{max} = \frac{F_s}{2} = \frac{1}{2\Delta t}$$

If N data points are taken (i.e., a data block size of N), with samples spaced Δt apart, a time sample record T will be recorded in memory whose length is:

$$T = N\Delta t$$

Since each spectrum has two values associated with each frequency (i.e., real part and imaginary part, or magnitude and phase), the display in the frequency domain will have only N/2 data points. Thus, the frequency resolution (Δf) will be determined by:

$$\Delta f = \frac{F_{max}}{N/2} = \frac{F_s}{N}$$

and since

$$F_s = \frac{1}{\Delta t}$$

and

$$T = N\Delta t$$

we can see that

$$\Delta f = \frac{1}{T}$$

i.e., the frequency resolution (Δf) is determined by the sample record length (T). The four quantities Δf, F_{max}, Δt, and T completely determine the time and frequency scales of the Fourier Analyzer.

For further clarification, the parameters are broken down as follows according to their domain.

Time Domain Parameters

Δt The time between samples, called the "sample interval", (Δ TIME on the ADC panel).

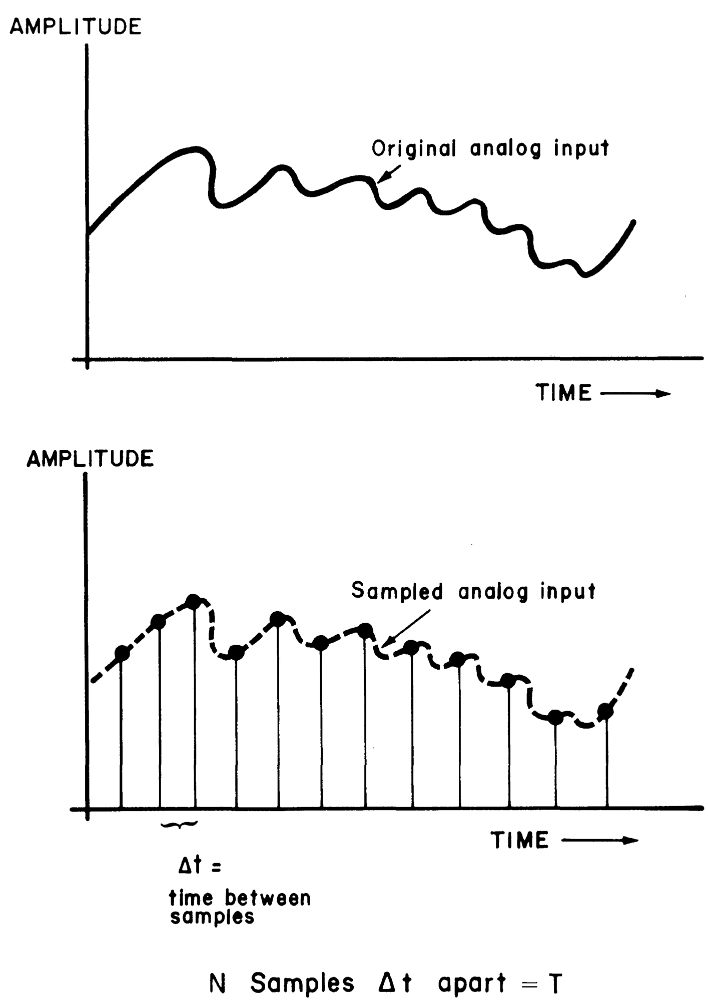
N The number of samples taken: This is the data block size (BLOCK SIZE on the Keyboard).

T The total time of the sample record, called "total record length". (TOTAL TIME on the ADC panel). From Figure 2-3 it can be seen that:

$$\text{total record length} = \text{no. of samples} \times \text{sample interval}$$

$$T = N\Delta t$$

Figure 2-3. Time Domain Parameters



Frequency Domain Parameters

The following parameters pertain to a standard, or baseband, measurement. The operator may also choose to increase the resolution of any frequency measurement, or portion of a spectrum, through a built-in digital filtering technique known as Band Selectable Fourier Analysis. The BSFA technique results in different relationships between parameters as compared to the standard measurement, and is introduced later in this section (see Demonstration of Measurement Functions).

Δf The number of Hz between frequency points, or, the frequency resolution. Origin of display is 0 Δf (dc component); next point is 1 Δf; next point is 2 Δf; next 3 Δf, etc. No finer resolution than Δf may be assigned to any frequency value.

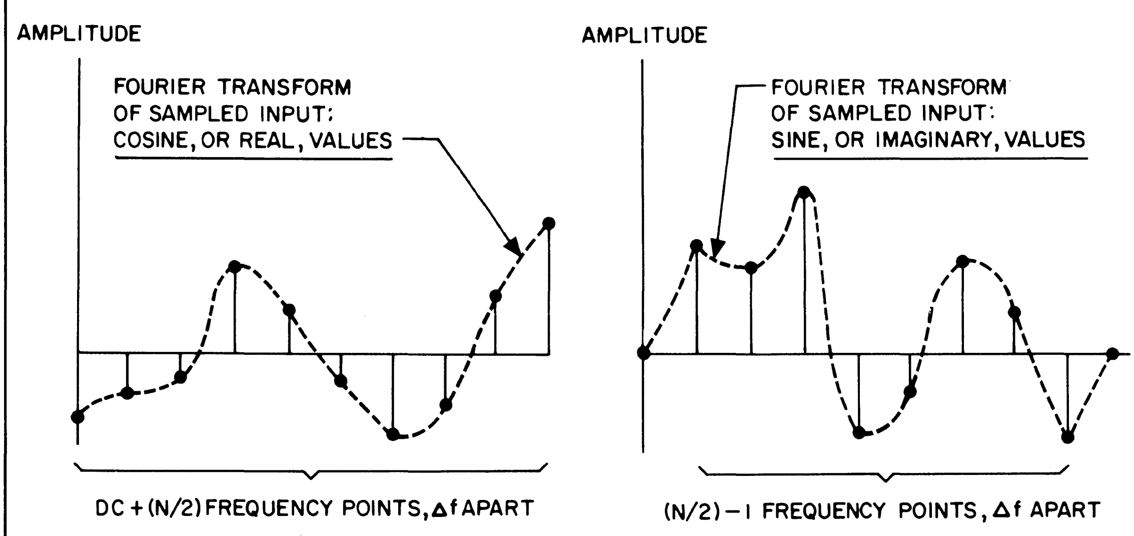
N/2 The number of frequency points: the frequency domain occupies a block of N points, but this block is broken into two displays: real or imaginary (depending on MODE switch setting). The real (cosine) display consists of (N/2)+1 points and the imaginary (sine) display consists of (N/2)−1 points, for a total of N points. It should be understood that these N points cover N/2 frequencies, since there is a real and imaginary value for each frequency (except for dc and highest frequency, which only have real values).

F_{max} The maximum frequency of the display, or in other words, the bandwidth. (MAX FREQ on the ADC panel). From Figure 2-4 it can be seen that:

maximum frequency = no. of frequency points × frequency resolution

$$F_{max} = N/2 \times \Delta f$$

Figure 2-4. Frequency Domain Parameters



Choosing Sample Parameters

Table 2-1 summarizes the time and frequency domain parameters (Δt, F_{max}, Δf, T). The table permits the user to obtain the best trade-off on the parameters he is interested in. The SAMPLE MODE switch and the MULTIPLIER switch on the ADC panel enable the user to select a convenient value for one of the two values on each side of the SAMPLE MODE switch. The other value is then automatically fixed, as shown in Table 2-1. Thus if you choose a frequency resolution (Δf) then the total record length (TOTAL TIME) is automatically fixed. The remaining two points are determined by the block size. The following example presents a typical situation.

Suppose you must have a 1 Hz frequency resolution and at the same time a 5 kHz maximum frequency. Go into Table 2-1 at line 3. In this case,

$$\Delta f = 1$$

In the last column, you see that the equation relating frequency resolution and maximum frequency is:

$$F_{max} = N/2 \times \Delta f$$

so

$$5000 = N/2 \times 1$$

$$N = 10{,}000$$

but the largest block size is 4,096, so an N of 10,000 is impossible. Suppose you agree to settle for a lower maximum frequency and substitute for N a block size of 4096. This will give the largest possible F_{max}:

$$F_{max} = 4096/2 \times 1$$

$$F_{max} = 2048 \text{ Hz}$$

So if you want 1 Hz resolution you must settle for an F_{max} of 2048 Hz.

This is the kind of manipulation of ADC parameters which you must be able to do. The parameters are set with the SAMPLE MODE and MULTIPLIER switches on the ADC, plus the BLOCK SIZE key on the Keyboard.

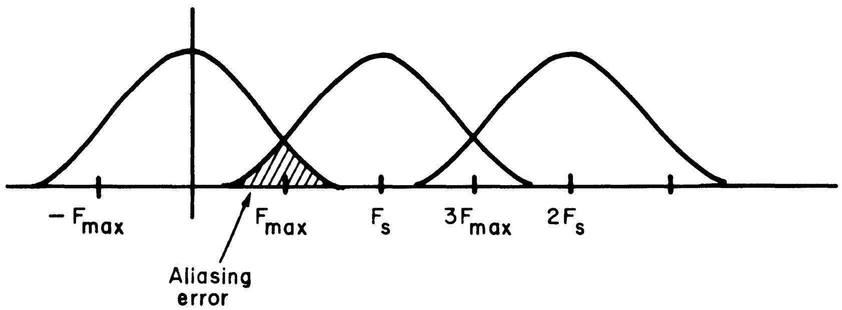
Table 2-1. Selecting Values for Data Sampling Parameters

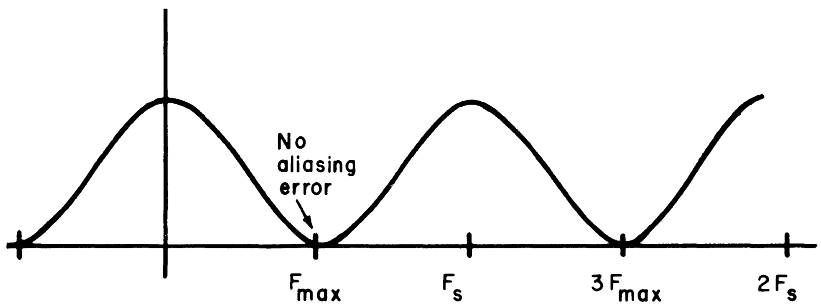
Choose convenient round number for parameter shown.	Chosen parameter automatically fixes the value of parameter below, because of relationship in parentheses.	Then make either of the remaining two parameters (can't be both) as close as possible to the desired value by choosing N* in the relationships shown.
1. Δt	$F_{max} \left(F_{max} = \frac{1}{2\Delta t}\right)$	T $(T = N\Delta t)$ $\Delta f \left(\Delta f = \frac{1}{N\Delta t}\right)$
2. F_{max}	$\Delta t \left(\Delta t = \frac{1}{2F_{max}}\right)$	T $(T = N\Delta t)$ $\Delta f \left(\Delta f = \frac{1}{N\Delta t}\right)$
3. Δf	$T \left(T = \frac{1}{\Delta f}\right)$	$\Delta t \left(\Delta t = \frac{T}{N}\right)$ $F_{max} \left(F_{max} = \frac{N}{2} \cdot \Delta f\right)$
4. T	$\Delta f \left(\Delta f = \frac{1}{T}\right)$	$\Delta t \left(\Delta t = \frac{T}{N}\right)$ $F_{max} \left(F_{max} = \frac{N}{2} \cdot \Delta f\right)$
*N, the data block size, is always a power of 2.		

Aliasing

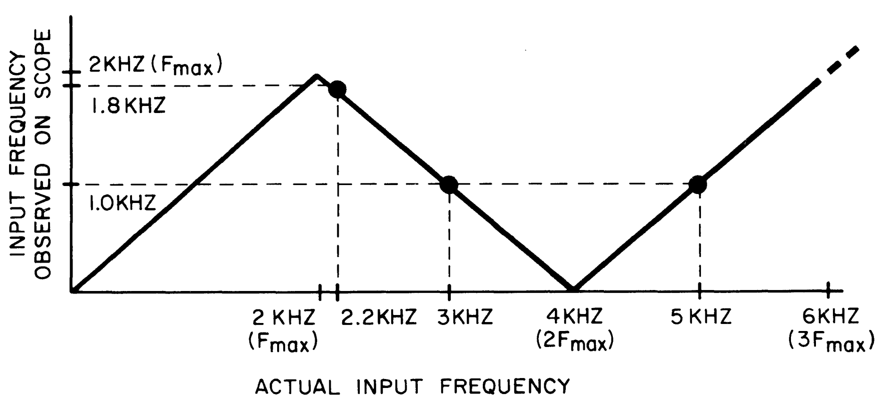
Aliasing is a phenomenon that results from sampling analog inputs, and must be kept in mind to avoid possible erroneous results. It comes about from the fact that when an analog input is sampled, the spectrum replicates around multiples of the sample frequency F_s, as shown in the following diagram. Now since F_{max} is half of F_s, it follows that if any frequencies greater than F_{max} are present they will fold back as frequencies less than F_{max}.

In the figure below, F_{max} is 2 kHz, F_s is 4 kHz. A frequency of 2.2 kHz will therefore be seen as 1.8 kHz; 3 kHz as 1; 5 kHz again as 1, etc.





To avoid the problem, you must make sure that the F_{max} you set is higher than the highest frequency in the data. (Or use a low pass filter, such as the optional 54440 Low Pass Filter, to attenuate unwanted higher frequencies.)

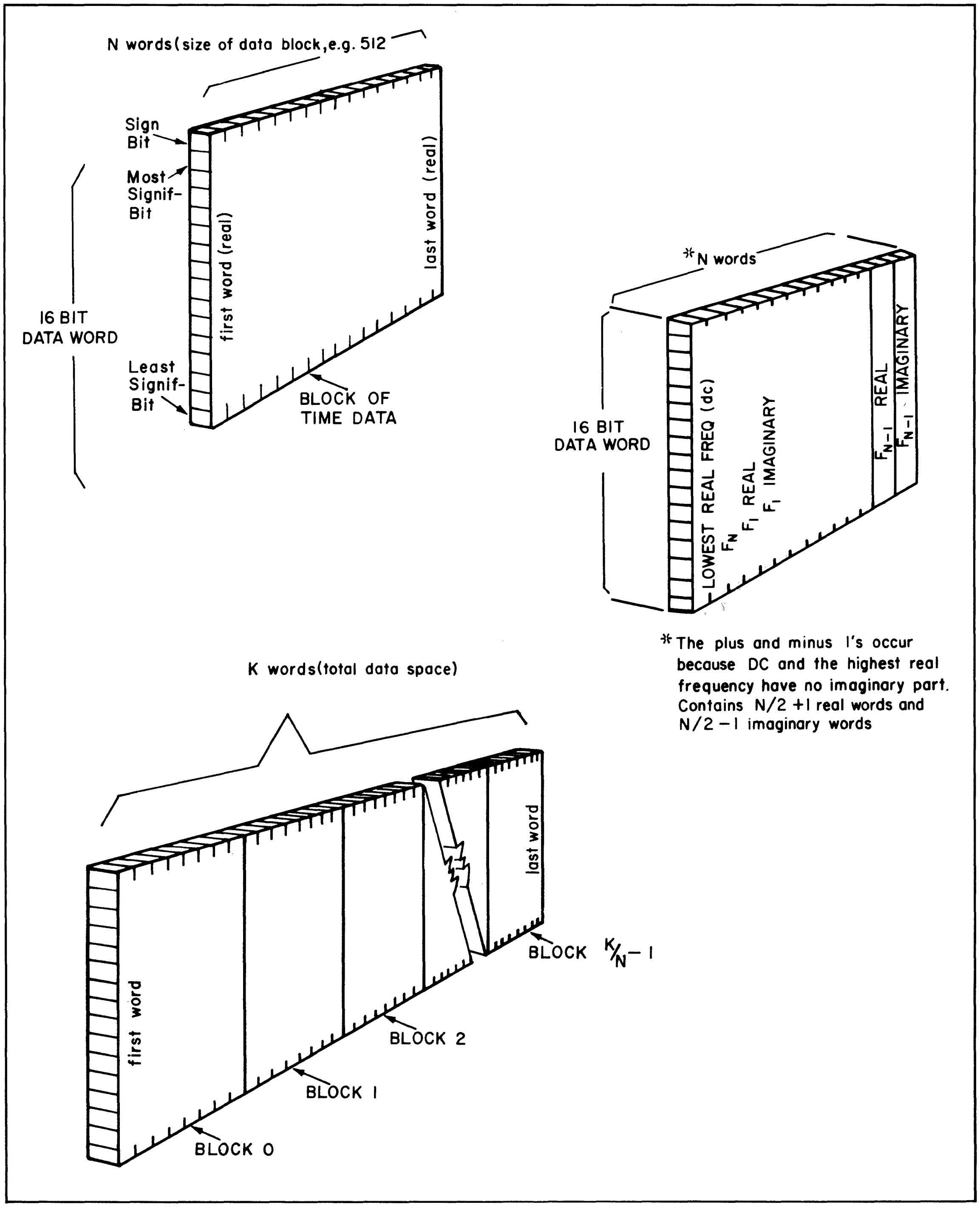


HOW DATA IS PROCESSED

THE DATA WORD

The smallest element of data that may be entered or taken out of the Fourier Analyzer memory for an input/output processing operation is the data word. Each data word represents a value at a given point in time of a time series, or of a correlation function, or a value of a spectrum at a given frequency. The data word is a 16-bit binary number representing an integer from −32,768 to +32,767, a range of greater than 96 dB. Groups of these data words are collected together to form the next element of data storage, the data block, as shown in Figure 2-5.

Figure 2-5. Data Memory Storage



THE DATA BLOCK

The data block may contain a quantity of data words equal to any power of 2 from $2^6 = 64$ to $2^{12} = 4096$ words. A data block will contain a set of time samples of a time series or a set of frequency components of a spectrum. This time record or spectrum may represent one element in an ensemble average or it may be the result of some complex operation such as a power spectrum or correlation function.

Data Qualifiers

Associated with each data block are "data qualifiers", consisting of an amplitude scale factor, a block calibrator, and a frequency code. The amplitude scale factor is a power of 10 which multiplies all N data words in the block. The block calibrator is a 16-bit binary number which multiplies all N data words in the block. Its magnitude is always greater than 0.1 but never more than 1. The data word, block calibrator, and scale factor may thus represent any number from -1×10^{-512} to $+1 \times 10^{+511}$. This numerical system is called "floating point on a block basis", and allows full use of all 16 bits during calculations without overflow. When an overflow occurs in an operation, all words in the data block are divided by 2 (shifted right one bit) and the block calibrator and scale factor are compensated accordingly. In this way, full accuracy of the data word is retained and no loss of calibration occurs during calculations. The frequency code defines the sampling parameters used by the Analog-to-Digital Converter if data was taken in via the ADC mode. A table of the frequency codes used is provided under the PRINT key in Section 3.)

Time vs. Frequency Blocks

When the data in a data block represents the values associated with a spectrum or other function of frequency, the data is stored differently than when the data block contains a time series. In the Fourier Analyzer, from an N-point real time series, we compute and display N/2 positive frequencies, plus dc. Each frequency except the highest has 2 independent numerical values: a real (cosine) and an imaginary (sine). The highest frequency has only a real, but no imaginary value. The actual arithmetic is as follows: N/2 positive real frequency values plus dc value = (N/2)+1 real points. The imaginary side has no dc value and no value for the highest frequency, hence it has (N/2)–1 points. Adding the real and imaginary points together we get (N/2)+1 plus (N/2)–1, or a total of N points in the frequency domain from N points in the time domain (see Figure 2-5). The negative frequencies that are generated by the double-sided transform are not computed or displayed because the negative half of the frequency (of a real-time series) is symmetrical with the positive half of the spectrum.

When the data block contains a time record, the first sample point of the time record is contained in the first word (numbered 0) of the data block, and the last point of the time record is found in the last word (numbered N–1) of the data block. However, when the data block contains values in the frequency domain, the real and imaginary values are interlaced (see Figure 2-5). The only exceptions to this are the lowest real frequency (dc) and the highest real frequency (F_n). The dc value is stored in the first word (0) of the data block and the F_n value is stored in the adjacent word (1).

In this manual, the term channel will be used for the 1 data word defining a point in the time domain or the 2 data words defining a point in the frequency domain. Thus, there will be N channels in the time domain and N/2 channels in the frequency domain. Details on how to direct the data word into and out of the real/magnitude or imaginary/phase part of memory are included with the related keyboard commands in Section 3.

Double Precision Data Blocks

Under certain circumstances the result of a data block operation will be a double precision data block. These blocks are composed of data words 32 bits long (normally 16). The most significant bits are stored in one data block word, while the least significant bits are stored in the following data block word. In some cases, a double precision block has no imaginary values (e.g., auto power spectrum) so that no extra storage is needed for the block. In cases where the data does have imaginary values (e.g., cross power spectrum) two data blocks are then needed to store the data.

Data Block Relationships

At any particular time all data blocks must be of the same size. The number of blocks that are available for a set of operations can then be determined by dividing the total number of memory words by the block size. For example, if a block size is set to 512 words in a system with 16,384 (16K) data words available, 32 data blocks numbered from 0 to 31 are available. As shown in Figure 2-5 the data blocks lie in consecutive order in the memory with the first word (word 0) of data block 0 in the first word (the 0 word) of data memory and the last word of the block in the N–1 word of the data block. If data is contained in memory and the block size is changed, the location of a particular data word is not changed, but the boundaries of the data block are.

The following example will clarify this point.

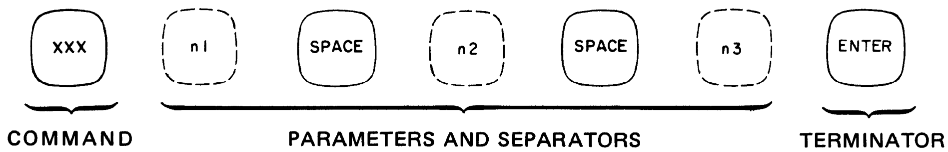
Example. If the block size is 512, and a time series is represented in memory, there will be 512 numbered data positions representing an equal number of time points. Consider the 260th word of block 0 (i.e., channel 259). If the block size is changed to 1024, the new 0 block will include the 260th word of memory and this data point will still be channel 259 of block 0. However, if the block size had been changed from 512 to 256, the 260th word of memory would now be in the 2nd data block and would be the 4th word of data block 1. This word would be numbered channel 3 of data block 1 since the first channel of the data block is number 0.

Important: If the block size is expanded with a given set of data in memory, there may be an error in the scale factor display. The reason is as follows: suppose the block size is 128, and the block 0 scale factor is $1(10^1)$ and the block 1 scale factor is $2(10^2)$. Now the block size is doubled. Then block 0 contains data having two different size factors. The first will dominate and be read out on the scale factor display meaning that the display will be wrong for the second half of the block. This must be kept in mind whenever block sizes are changed while there is data in more than 1 block.

However, when an original block is broken up into smaller blocks, the new blocks will take on the scale factor of the original block (unless the CODED BLOCK SIZE command is given). For instance, if block 0 is halved, the scale factor for the new block 1 will be the same scale factor as was the original block 0. (See BLOCK SIZE command, Section 3.)

THE KEYBOARD COMMAND

All operations on the Fourier Analyzer are initiated from the Keyboard. (Unfold Figure 2-7 at the end of this section). The general form of the Keyboard command is:



Parameters may be any integer up to 32,767 and must be separated from each other by the SPACE key; the number of parameters vary for each command. Command formats for all keys are given in detail in Section 3 and abbreviated in Appendix B.

For information on a particular key, see the Quick-Key reference on the inside front cover for specific page numbers.

No command is executed or entered until the ENTER key is pressed. If a mistake is made during the entry of a command, it can be erased by pressing the RUB OUT key, then the ENTER key. An incorrect command will turn the WHAT? light on and print an error message on the Terminal.

EXAMPLE OF TYPICAL COMMAND

Suppose you want to print out the contents of channels 22 to 31 of data block 2. Look under the PRINT key in Section 3 and you will find the following format for the PRINT command:

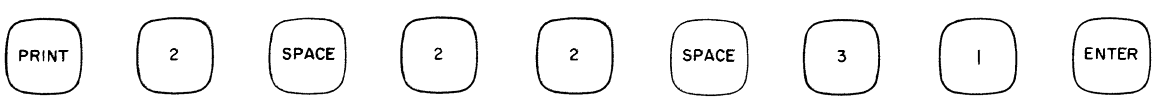
where:

n1 is the number of the data block to be printed out.

n2 is the first channel to be printed out.

n3 is the last channel to be printed out.

In which case, to complete our example you would enter this command as follows:



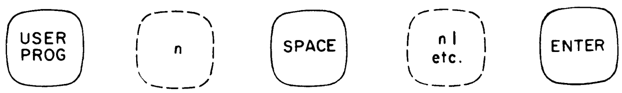
NOTE

The term *variable parameter* as encountered later in this manual means that, rather than entering discrete values for n1, n2, n3 and so on, to indicate channels, data blocks, etc., an address is entered instead that contains a variable value for a channel, data block, etc. Variable Parameters are introduced in the Demonstration of Variable Parameters later in this section and defined fully in Section 3.

EXAMPLE OF GOLD KEY COMMAND

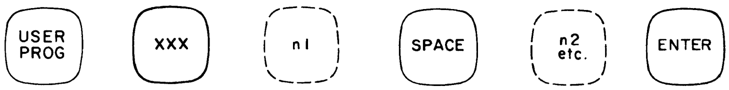
The gold USER PROGram key is a dual function key. Pressed by itself it initiates a User Program command (also called Y-command). When followed by one of the other Keyboard keys that have a gold function printed above or below the key, the gold function is initiated. In this latter case, the USER PROGram key acts as a shift key similar to that on a typewriter or calculator.

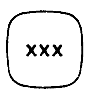
To use the USER PROGram key to initiate User Programs only, follow the format below:



where n is the 4-digit user program command and n1 through n6 are optional program parameters (see Section 7 for complete listing).

To use the USER PROGram key as a gold shift key, follow the format below:



where 

is one of the gold-labeled keys (e.g., press KEY BOARD key to get gold PLOT function) and n1 through n5 are program parameters. Note the absence of the SPACE key between the gold-labeled key and n1.

Most gold functions are actually represented by an existing User Program command. For example, gold LIST is User Program 1806. The gold function then is simply an abbreviated method of entering the User Program number.

User Programs are introduced in the demonstrations to follow and are fully defined in Section 7. The gold function keys are defined in Section 3, together with the associated standard key (e.g., gold PLOT, immediately after KEY BOARD key.

MEANING OF DEFAULT

A default system is built into the Keyboard program to save keystrokes in many situations. For instance, in the example of the PRINT command given previously, if n2 and n3 are defaulted (i.e., not entered) all of the channels in the data block will be printed out. If n1 is defaulted as well, all of data block 0 will be printed out — defaulting of n1, the block, always indicates block 0. Default values for each command are given with the key descriptions in Section 3.

DEMONSTRATION OF SOME BASIC KEYBOARD FUNCTIONS

The demonstrations that follow are intended to be a familarization procedure for the new owner of the Fourier Analyzer. These demonstrations also explain the use of the rest of this manual, which is set up in reference form so that you can quickly find complete information as you need it. (A fast way to locate any subject is by using the index at the very end of this manual.)

In order not to interrupt the sequence once it has begun, background information on three important subjects must be mastered first. (1) ADC sampling parameters, (2) how data is processed, and (3) the Keyboard command structure. These previous paragraphs should be read before beginning the demonstrations.

In the following demonstration, we will enter a pulse through the Analog-to-Digital Converter (ADC) and exercise some of the ADC controls. Then we will enter a pulse manually from the Keyboard and proceed through the Fourier transform to a series of other commands which show all of the display coordinates. At the conclusion, we will demonstrate a simple Gold Key operation.

INSTRUMENT PREPARATION REQUIRED

At this point, it is assumed that the Fourier Analyzer has been installed and checked out according to the Operational Check provided in the Fourier Analyzer System Service Manual. Power should be on, and the READY light should be lit. For instructions on turning on the Fourier Analyzer, see the Turn-On Procedure at the beginning of this section.

HOW TO SET SAMPLING RATES

This paragraph is a summary of the previous discussion on ADC Sampling Parameters and is provided as a review before we get into the demonstration. Again, unfold Figure 2-6 at the end of this section for full information on ADC controls.

The rate at which the ADC samples the analog input is set by the SAMPLE MODE and MULTIPLIER switches. There are four sampling parameters: The two in the time domain are the interval between samples (Δt) and the total record length (T); in the frequency domain there is frequency resolution (Δf) and maximum frequency (F_{max}). The following tells how to set the parameter values on the ADC switches.

The left side of the SAMPLE MODE switch permits the user to select either a desired maximum frequency (F_{max}) or a desired sample interval (Δt). Choosing one automatically fixes the other because of the relationship:

$$F_{max} = \frac{1}{2\Delta t}$$

The two positions of the switch enable you to select two different ranges of magnitude, which differ by a factor of 1000. Values are set with the MULTIPLIER switch: black numbers for Δt, blue for F_{max}. The other two parameters, frequency resolution (Δf) and sample record length (T), are found by the equations:

$$\Delta f = F_{max}/(N/2)$$

$$T = N\Delta t$$

where N is the block size.

Alternatively, the right side of the SAMPLE MODE switch permits the user to select either a desired frequency resolution (Δf), or a desired sample record length (T). One automatically fixes the other because of the relationship:

$$\Delta f = 1/T$$

The remaining two parameters, F_{max} and Δt can be computed from the equations:

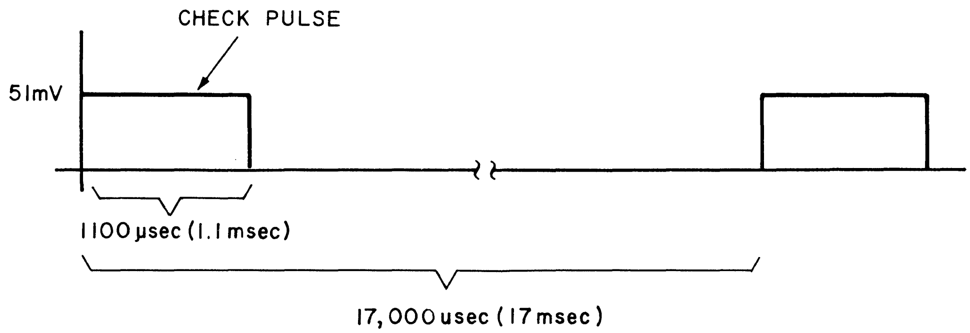
$$F_{max} = N/2\Delta f$$

$$\Delta t = \frac{T}{N}$$

ENTERING AN ANALOG PULSE

1. **Set ADC controls** (Refer to foldout Figure 2-6)

   In order to eliminate the need for an external signal generator, we will use the internally-generated CHECK pulse as an analog input. This signal is a rectangular pulse of about 51 mV amplitude and 1100 μs (1.1 ms) length, as shown below. The frequency of the pulse is the line frequency. This means that the period T is the reciprocal of the line frequency, or about 17,000 μs (17 ms) for a 60 Hz line. Thus, the pulse is present in only 1/17th of the time T.



   Since, as much as possible, we want to look at the pulse and not the space in between, we will begin by choosing a Δt that will give us an adequate number of samples of the pulse only. Eleven (11) samples would be adequate: the pulse is 1100 μs long. 1100/11 gives a Δt of 100 μs. Therefore, on the ADC we set:

   SAMPLE MODE to kHz/μs
   MULTIPLIER to black 100
   EXT/INT to INT

   The CHECK pulse can be accessed at any time by putting the OVERLOAD VOLTAGE switch in the CHECK position. We will be using channel A only, so set:

   OVERLOAD VOLTAGE switch A to CHECK

   The INPUT SELECTOR switch determines the number of input channels used for a given measurement starting with channel A. Because we are going to look at one channel only, set:

   INPUT SELECTOR to A

   Notice under the TRIGGER SOURCE switch we have five positions available, two of which include an AC or DC coupled EXTernal source (note BNC labeled TRIGGER input). The other three positions control internal triggering as follows when an analog input has been given: FREE RUN which starts the ADC sampling immediately; INTERNAL (A) which triggers on the input signal to INPUT CHANNEL A (note BNC connectors); and LINE which triggers on the power line frequency.

   To sample the CHECK pulse we will set:

   TRIGGER SOURCE to FREE RUN

2. **Set Display Unit Controls** (Refer to foldout Figure 2-8)

The system makes measurements by digitizing the input signals and storing the digital data in the Processor. Then the information derived can be manipulated and displayed a number of ways on the Display Unit. The following paragraphs are concerned with familiarizing the user first with the ADC controls then the Display Unit controls.

For now, set the Display Unit controls as follows:

On the CRT portion of the Display Unit, set the following:

MAGNIFIER to X1
DISPLAY to INT
INTENSITY as necessary to show beam or line
FOCUS & SCALE as desired

On the control portion of the Display Unit, set the following:

GAIN to CAL
MODE to REAL/MAGNITUDE
POLAR ANG/DIV to any position
SCALE to straight up

All lever switches to center except:

DISPLAY TYPE to BAR
DISPLAY FUNCTION to CAL

Adjust VERTICAL/POSITION and HORIZONTAL/POSITION (on CRT) to center the beam on left center of CRT. Then set:

DISPLAY/FUNCTION to DISPLAY

3. **Enter Keyboard Commands** (Refer to foldout Figure 2-7)

The REPEAT/SINGLE switch on the Keyboard has to do with continuous or single displays of data blocks. In the SINGLE position, one block only of each selected input channel is displayed. In the REPEAT position, blocks are displayed repeatedly, and the BUSY light stays on, disabling further Keyboard entry.

To allow us to continuously sample and display the CHECK pulse, set:

REPEAT/SINGLE to REPEAT

We are now ready to enter our first keyboard commands. As stated before, all keyboard keys are explained in Section 3. Whenever you want to know more about a key than is explained here (purpose, parameters, etc.), refer to that section.

A mistake in a command entry can be corrected, provided the ENTER key has not been pushed, by pressing RUBOUT ENTER.

The sample parameters for a CHECK pulse are not critical, so enter an arbitrary block size of 256 as follows:

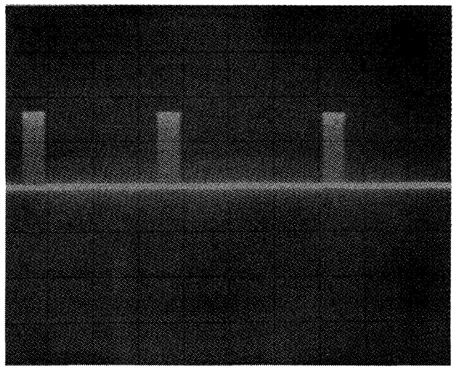
[BLOCK SIZE] [2] [5] [6] [ENTER]

On the Keyboard, the BLOCK SIZE 256 light should be on and the line on the CRT will comprise 256 points.

The ANALOG IN command initiates the measurement, taking in data to the Processor through the ADC. We can enter the CHECK pulse into block 0 and also display this block by defaulting all parameters and entering simply:

ANALG IN   ENTER

CHECK pulses will now appear on the CRT similar to that shown below:



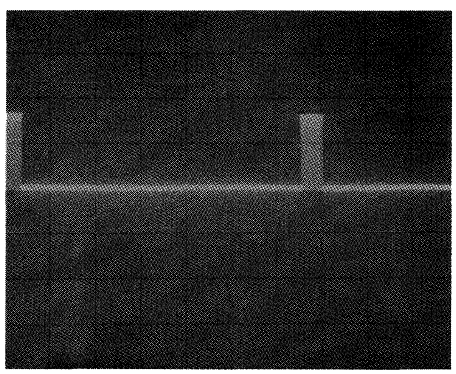
The CHECK pulses appear to run across the screen because in the REPEAT mode with a free running trigger, we are displaying data as fast as it can be sampled. The next step describes ways to best view the data.

4. **Vary Trigger and Sample Controls**

We can sync the CHECK pulses by triggering on the positive edge of the pulse as follows:

TRIGGER SOURCE to INTERNAL (A)
TRIGGER SLOPE to POS (outer knob CW)
TRIGGER LEVEL to 2 o'clock (or CW until pulse stabilizes)

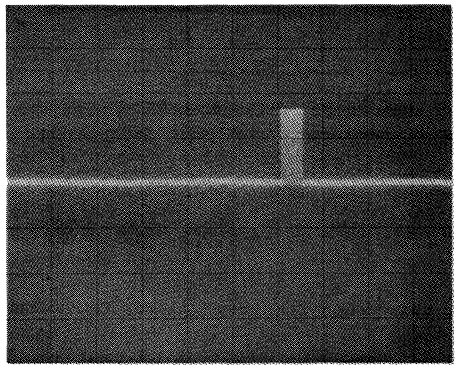
Note that the TRIGGERING light comes on and that a stable display appears as shown below since now we are triggering solely on the leading edge of the CHECK pulse.



Now set:

TRIGGER SLOPE to NEG

The display will change to that shown below since we are now triggering on the negative edge of the CHECK pulse. Note that the left pulse is no longer visible and the right pulse has shifted left one pulse width.



Return:

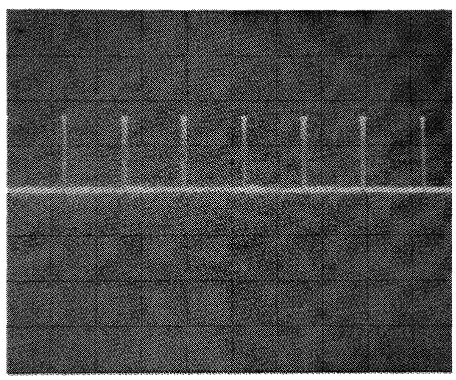
TRIGGER SLOPE to POS

Now let's see how varying the sample interval (Δt) affects the display. Note that the SAMPLE MODE switch points to kHz/μs (under MAX FREQ/Δ TIME) and the MULTIPLIER points to 100/10/5. We use the black lettering to determine that we are sampling with a Δt of 100 μs (and the blue lettering to determine that our F_{max} is 5 kHz.

To change our sample rate, set:

MULTIPLIER to black 500

Now our Δt is 500 μs and the display appears as follows:

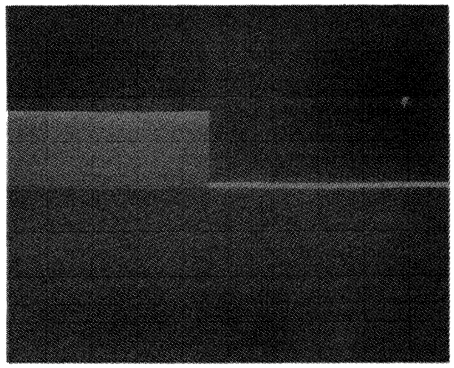


The reason we see more CHECK pulses of a narrower width is because the Δt has increased by 5, meaning we are taking fewer samples of the CHECK pulse.

If our area of interest is in the CHECK pulse, then we will want to reduce Δt to where we take as many samples as possible. To show this, set:

MULTIPLIER to black 10

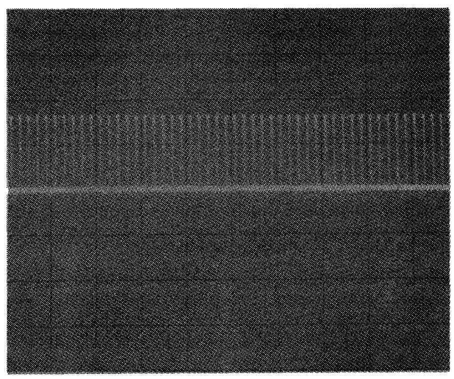
With the time between samples now at 10 μs, we can see that many more samples are taken of the pulse as shown below:



We can also change Δt with the SAMPLE MODE switch; for example, set:

SAMPLE MODE to blue Hz/ms (Δ TIME)

Again, reading the black lettering only, we see that Δt now equals 10 ms. The display appears as follows:



At this sample rate the pulses may disappear and reappear over a minute or so interval. This is normal since with a Δt of 10 ms for a 1 ms pulse, it is as likely to sample the 0 volts between pulses as the pulse itself. After viewing, set:

SAMPLE MODE to kHz/μs
MULTIPLIER to black 100

5. **Hold Block of Data**

   Up to this point we have been continuously inputting and displaying new data. We can capture a block of data by setting the Keyboard switch:

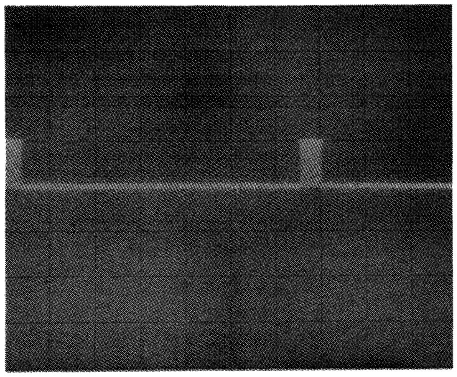
   REPEAT/SINGLE to SINGLE

   The Keyboard READY light comes on indicating that a data block has been stored in the Processor and the Keyboard is enabled.

   Note that the scale factor display, which was blank in the REPEAT position, is now lighted on the Display Unit. Also, our CHECK pulse has been re-scaled.

6. **Interpret Scale Factor Display**

   The scale factor display gives the vertical scale of the scope. The factor is expressed as 1, 2, or 5 times 10 to some exponent. For example, the scale factor for the above CHECK pulse might be 5 × 10 -002, meaning 5×10^{-2}, so that each vertical division is 0.05, or 50 millivolts. This is about the amplitude of the CHECK pulse (51 mV) as shown below.



   This scale factor is displayed automatically at all times except for special cases such as REPEAT/SINGLE switch in REPEAT position, as we saw earlier. Turn the SCALE switch left and right, and note the increase and decrease in the size of the pulse, and how the scale factor changes accordingly. The display always remains calibrated regardless of switch position.

REVIEW

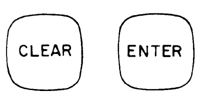
So far, we have taken an analog pulse into the ADC, seen how some of the sampling controls work, looked at triggering, and explained the scale factor display. Next we want to put in a pulse manually, then do a Fourier transform, and see the coordinate systems available on the Fourier Analyzer. We will also exercise some of the Display Unit controls.

ENTERING A PULSE MANUALLY

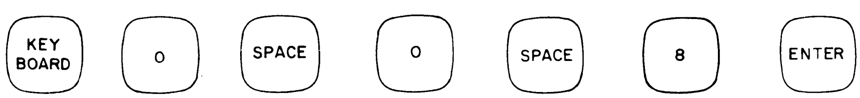
Another means of entering data is the manual Keyboard method. Using the KEYBOARD key we can enter data points in memory to simulate a signal. As explained under this key in Section 3, there are two methods available to do this: the block-fill method, and the point-by-point method. In this example we will use the simpler block-fill method, which fills a specified number of channels with a uniform amplitude.

1. **Enter Block Fill Command**

   First, clear data block 0 by pressing:



   Then, build a rectangular pulse in block 0, from channel 0 thru channel 8, by pressing:



   Check that the BUSY light turns on, meaning that the Keyboard is waiting for the next two commands.

2. **Enter Scale Factors**

The scale factors required to keep the data calibrated are: the block multiplier, the coordinate code, and the frequency code. These are explained under the KEYBOARD key. For now, let the block multiplier be 10^{-4}, and the coordinate code can be 0 since we want to be in time domain, and let the frequency code be 42, which gives us an F_{max} of 5 kHz (see frequency code table under PRINT key). Therefore, enter: (Use minus key in the ENTRY group.)

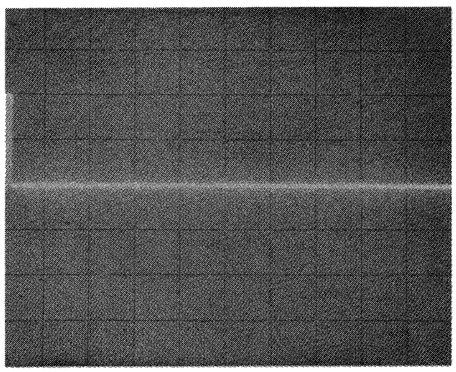
KEY BOARD – 4 SPACE 0 SPACE 4 2 ENTER

3. **Enter Data**

To create a 1-volt amplitude on the pulse, with a multiplier of 10^{-4}, enter:

1 0 0 0 0 ENTER

Note that the KEYBOARD key was not required for this last entry. The READY light now turns on and the pulse is displayed as shown.



To save having to re-enter the pulse later, should the need arise, we can store it in block 1 by pressing:

STORE 1 ENTER

Now it is in both block 0 and block 1, with block 1 being displayed. To view block 0 again, press:

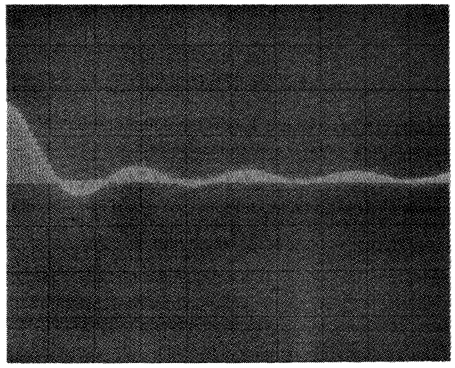
DSPLY ENTER

4. **Do Fourier transform**

We are now ready to do the Fourier transform. Since we are operating on block 0 only, we can default the parameters and enter simply:

F ENTER

The Fourier transform consists of two spectral series: a real or cosine series, and an imaginary or sine series. The familiar curve, sin X/X, will appear on the display as follows:



5. **Determining horizontal scale**

From the previous discussion we know how to read the vertical scale. The pulse we input was given an F_{max} of 5 kHz. This means we are now displaying dc on the left of the CRT and F_{max} on the right of the CRT. To determine the frequency of a particular channel, we can calculate either of two ways: (1) a rough scale is available by considering each horizontal division on the scope to be $F_{max}/10$ (there are 10 horizontal divisions), (2) a more accurate scale is determined by simple calculation. If the SAMPLE MODE switch is in one of the two left-hand positions, then the formula to use is:

$$\frac{F_{max}}{N/2} = \Delta f$$

F_{max} is read directly from the SAMPLE MODE and MULTIPLIER switches; N is block size. Δf is the frequency resolution. To read the frequency of a particular channel on the CRT, begin with dc on the left, then the next point to the right is $1\Delta f$, next $2\Delta f$, etc. In our case, therefore, the calculation is:

$$\frac{5000}{256/2} = 39.06 \text{ Hz}$$

There is a 39.06 Hz between each point on the scope, beginning with 0 Hz on the left.

If the SAMPLE MODE switch is in one of the two right-hand positions, then the formula is:

$$\Delta f(N/2) = F_{max}$$

where Δf is read directly from the switch settings, N is the block size.

The full explanation of how these various data sampling parameters are related was given earlier in this section under ADC Sampling Parameters. In many cases, of course, you will be interested in setting frequency parameters, and then will have to use other relationships to determine the values in the time domain.

NOTE

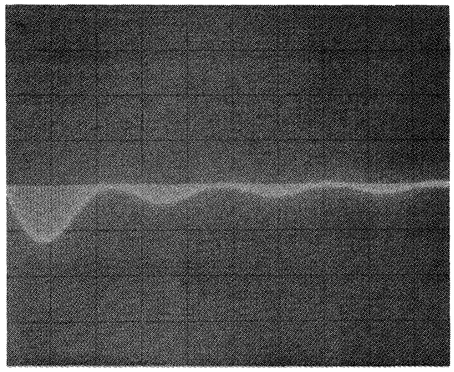
A third and more precise method of determining the value of displayed points is by using the X cursor (demonstrated later in this section).

6. **The real (cosine) and imaginary (sine) series**

   With the MODE switch in the REAL/MAGNITUDE position, the real (cosine) series is displayed. Now set:

   MODE switch to IMAGINARY/PHASE

   The resulting display of the imaginary portion of the spectrum should appear as follows:



   For more information on Display Unit controls, refer to foldout Figure 2-8 at the end of this section.

7. **Polar magnitude and phase display**

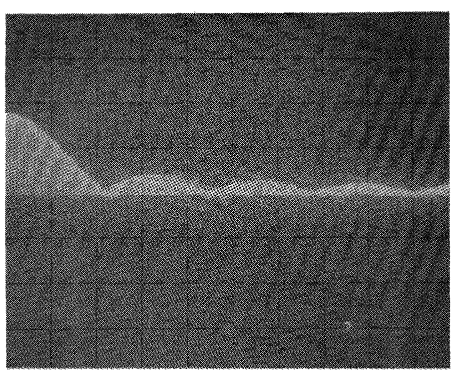
   Another Keyboard command will convert the above display to polar coordinates, i.e., magnitude and phase. First set:

   MODE switch to REAL/MAGNITUDE

   then press:

   POLAR   ENTER

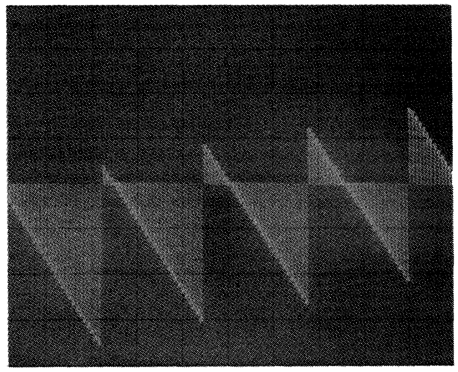
   The display presents the magnitude portion of the spectrum as follows:



To display the phase portion of the spectrum, set:

MODE switch to IMAGINARY/PHASE

Now the CRT shows the phase display, with the scale reading 0 to +180° on the top half of the screen, and 0 to –180° on the bottom half (each division of CRT = 45° with POLAR ANG/DIV switch in straight up "45" position).



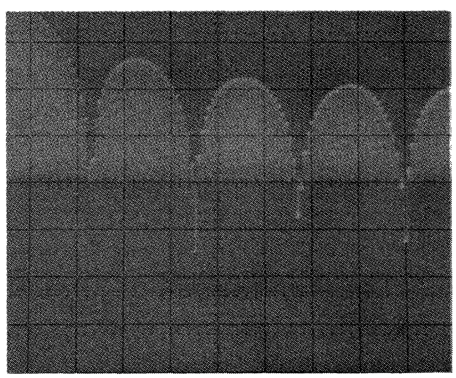
Frequency scale, of course, is as before. Note that the scale factor display is blank. For phase displays, the scale factor is given (and set) on the POLAR ANG/DIV switch. Vary the phase scale by turning the switch left and right. This changes the degrees-per-cm on the CRT to that indicated on the switch with 0° always remaining at the center of the CRT. Set:

MODE switch to REAL/MAGNITUDE

and press:

LOG MAG   ENTER

This converts the vertical scale to logarithmic, giving a display as follows with the SCALE switch 2 positions CW from straight up:



Display readout indicates dB level at top line of screen, each division is then 10 dB down.

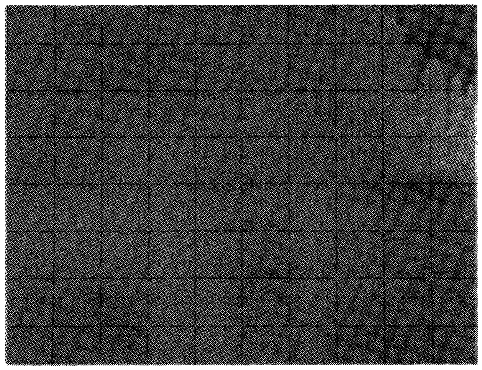
Markers are available for every 8th and 32nd point in the display, as an aid in identifying points of interest. Set the MARKER switch on the Display Unit to 8 then 32. Some adjustment of the CRT intensity may be necessary to make the markers stand out. Return MARKER switch to OFF.

Set the TYPE switch on the Display Unit to POINT and then to CONT positions and observe the different types of display. Reset switch to BAR.

To make a Bode plot (log vertical and horizontal scales), we set:

HORIZONTAL/ORIGIN switch to LOG

which sets the horizontal axis to logarithamic (the LOG MAG command sets the vertical axis to log) with the resulting display:



8. **Plot the Results**

For a final exercise in this demonstration, we will plot the display on the Terminal using the Gold Key PLOT function. Set:

SCALE back to straight up

Notice that on the previous display most of our information is on the right side of the screen. We can modify this display to more easily view the data of interest by calling up one of the resident User Programs. (These programs are provided as a user convenience for a variety of applications and are introduced throughout the manual; see Section 7 for a full listing.)

Enter the following command and note that the number of horizontal decades is reduced on the display:

USER PROG 3 0 2 4 SPACE 2 ENTER

NOTE

The USER PROGram key doubles as a 'gold shift' key to initiate the functions lettered in gold above or below a number of Keyboard keys. This is done by first pressing the gold USER PROGram key, and then the associated command key.

We will demonstrate the gold PLOT function (above KEY BOARD key) as follows:

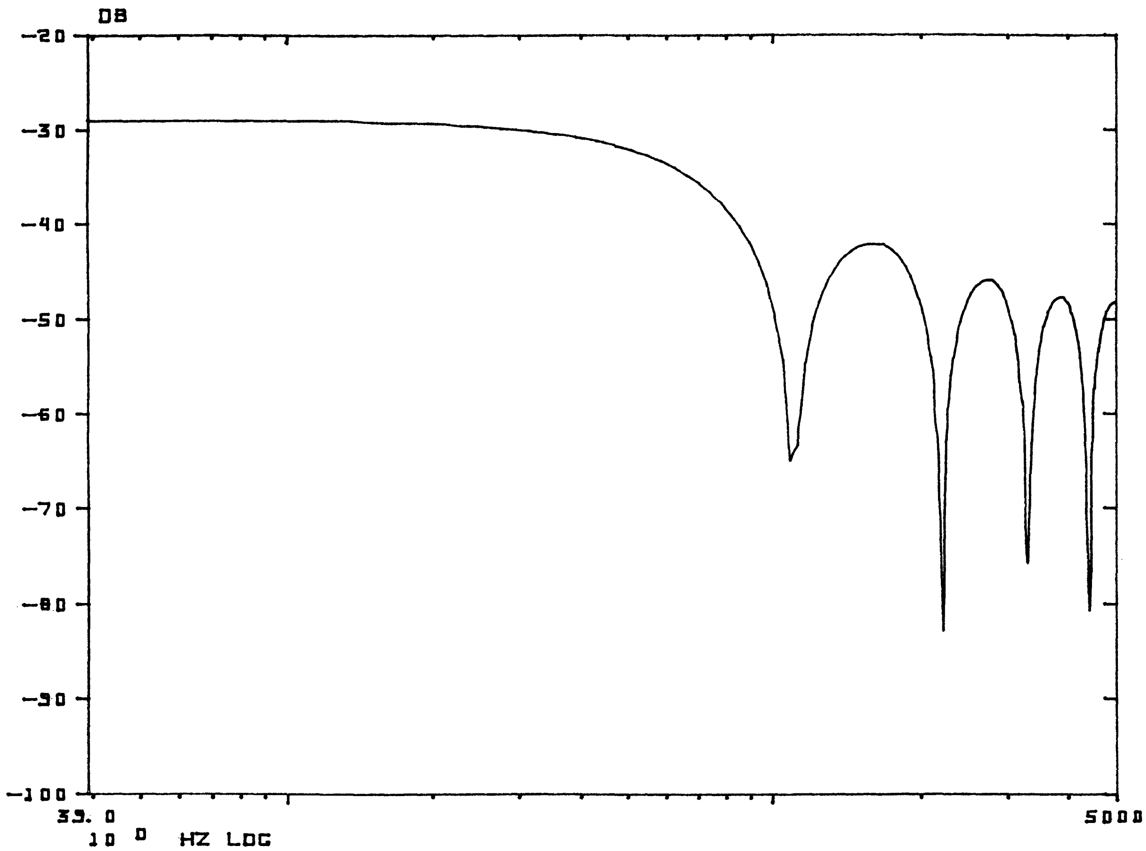
1. Set up the Terminal by pressing the gray RESET TERMINAL key twice in rapid succession (<0.5 second). This clears the Terminal except for the 'TERMINAL READY' message in the upper left screen area and leaves both Alphanumerics and Graphics memories on.

2. To disable the Alphanumerics display, hold down the SHIFT key and press the A DSP key once to remove the flashing cursor. Check that the REMOTE key and CAPS LOCK key are depressed.

Plot the display by pressing:

USER PROG KEY BOARD ENTER

The Terminal should display an annotated plot as follows:



To return the Terminal to display Alphanumerics mode, press the following:

STOP directs text to Alpha memory

SHIFT A DSP enables Alpha display

SHIFT G DSP disables Graphics display

More information on plotting with the Graphics Terminal is given in the upcoming Demonstration of Display Functions, and in Section 6.

BASEBAND vs. BSFA MEASUREMENT

In the following demonstration we will make both a baseband measurement and a BSFA measurement on a sinusoidal input signal. Then we will display both types of measurements to demonstrate the increased frequency resolution of the BSFA measurement. The Disc in your system has two files which are used for raw data storage and retrieval. File 1 is used to store data blocks from the Processor memory. File 2 is used for storage of raw ADC throughput data only. This throughput data is taken directly from the ADC and stored on the Disc. In this case the Processor is only being used as a buffer between the ADC and Disc. During this demonstration we will also use the Disc for storage of a data block as well as real-time ADC throughput data.

For this demonstration an oscillator will be needed. Set the oscillator for 2 kHz and an amplitude of 200 mV p-p. Connect the oscillator output to the INPUT CHANNEL A on the ADC.

NOTE

This setup will be required again in the final demonstration in this section.

Set up the ADC as follows:

SAMPLE MODE to INT, White Hz/ms (ΔFREQ)
MULTIPLIER to Black 100 (10 Hz)
OVERLOAD VOLTAGE A to .25V
INPUT SELECTOR to A
TRIGGER SOURCE to FREE RUN

Set up the Display Unit as follows:

MODE to REAL/MAGNITUDE
SCALE to Straight up (12 o'clock)
Lever Switches to Center, except DISPLAY TYPE to CONT
VERTICAL GAIN to CAL
VERTICAL POSITION to Situate X axis on center horizontal line of CRT

Now press the following keys in the order given (check that REPEAT/SINGLE switch is set to SINGLE):

CLEAR ENTER

BLOCK SIZE 1 0 2 4 ENTER

This sets the number of data points to be input. The 1024 BLOCK SIZE lamp should light and 1024 points will be on the display.

ANALG IN | ENTER

This reads in one block of data into block 0.

HANN | ENTER

This applies the Hanning window to the measured data.

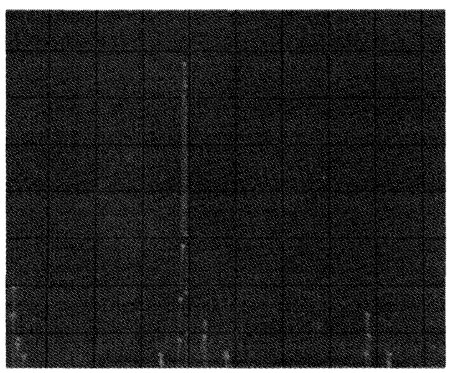
F | ENTER

This converts our time measurement to a frequency spectrum in rectangular coordinates. Note: The FREQ and RECT light in the upper right corner of the Display Unit are now on.

Notice that in this baseband measurement we have a frequency resolution (ΔF) of 10 Hz as set by our ADC SAMPLE MODE switch setting.

LOG MAG | ENTER

The magnitude is now being displayed on the CRT in log magnitude and polar coordinates (see lights on display). The CRT should look similar to the picture below:



MASS STORE | 3 | 1 | ENTER

Position to record 0 of the Data File on the Disc. (For full information on this key, see Section 4.)

MASS STORE | 2 | 1 | ENTER

Write the contents of block 0 on the Disc. (We will read it back later.)

We shall now read in a number of data records to the ADC Throughput File on the Disc. After storing on the Disc we will use BSFA on this raw data to show increased resolution on the same input signal.

[MASS STORE] [3] [2] [ ] [ ] [ ] [ ] [ENTER]

Position and write throughput data on the Disc starting at record 0.

[MASS STORE] [2] [2] [SPACE] [1] [SPACE] [5] [0] [ENTER]

Write 50 records of raw data on the Disc.

[USER PROG] [4] [1] [SPACE] [2] [0] [0] [0] [SPACE]

[1] [0] [0] [SPACE] [ ] [ ] [0] [ENTER]

Initialize the BSFA operation for 2 kHz center frequency and 100 Hz bandwidth starting at record 100 on the disc. Note that the Terminal prints out the center frequency of the display (2000 Hz) and the frequency resolution (ΔF) we will have after the next step. Note the resolution enhancement on the Terminal printout:

CNTR FREQ: 2000
HZ/DIV: 12.80
DF: 0.2500
BLOCKS LEFT: 14
ZOOM POWER: 40

[USER PROG] [4] [5] [SPACE] [1] [SPACE] [1] [SPACE] [1]

[ENTER]

The Processor now enters the BUSY mode to access the raw data previously stored on the throughput file, then performs BSFA, and displays the results.

[LOG MAG] [1] [ENTER]

Display the BSFA results in log magnitude.

[MASS STORE] [2] [1] [SPACE] [1] [ENTER]

Store BSFA measurement on the Disc in Data File.

MASS STORE | 3 | 1 | ENTER

Position the Data File back to record 0.

MASS STORE | 1 | 1 | ENTER

Read back from the Disc the baseband measurement we made in the beginning of this demonstration and store in block 0.

MASS STORE | 1 | 1 | SPACE | 1 | ENTER

Read back the BSFA measurement stored earlier on the Disc and store in block 1.

We now have both the baseband and BSFA measurement in the Processor in data blocks 0 and 1 respectively. Now compare these measurements by pressing:

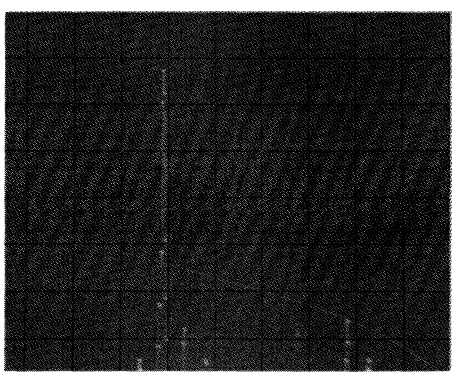
DSPLY | 0 | ENTER *(baseband block)*

and

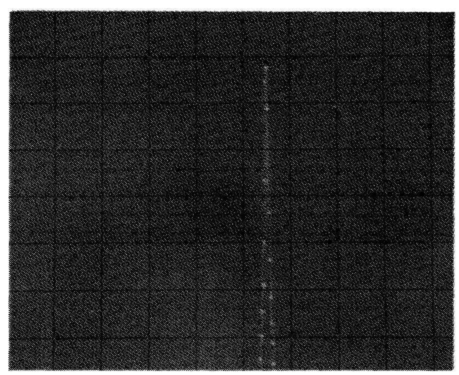
DSPLY | 1 | ENTER *(BSFA block)*

By alternating the display, the increased resolution (to 0.25 Hz) of BSFA over the original measurement (10 Hz) can be seen.

Baseband Display



BSFA Display



NOTE

Keep the data accumulated in this demonstration for use in the demonstration that follows (i.e., do not press any additional keys unless directed).

DEMONSTRATION OF DISPLAY FUNCTIONS

This demonstration explains the uses of the CURSOR key on displayed functions, and how to make annotated plots on the Terminal.

DEMONSTRATION OF CURSOR

The cursor is a vertical line on the display that is used to quickly identify vertical and horizontal coordinates of a point by printing their values on the Terminal. The cursor is controlled from the Processor S-register buttons, bits 9 thru 13. (For complete information, see CURSoR key in Section 3.) To demonstrate the cursor let's use the baseband measurement loaded from the Disc in the previous demonstration.

To display block 0, press:

DSPLY ENTER

To turn the cursor on, press:

CURSR ENTER

The cursor will now be on the left side of the display. Bits 9 and 10 on the Processor control the positioning of the cursor. Hold bit 9 ON and watch the cursor sweep from left to right across the display. (Bit 10 causes sweep left). Stop the cursor near the center of the display. To step only one data point at a time press bit 9 (or 10) momentarily. Once you have practiced positioning the cursor with bits 9 and 10, place the cursor over the peak amplitude on the display.

Press bit 11 on the Processor and observe the Terminal printout. The printout will show:

CHANNEL XXX

This is the channel number of the data point identified by the cursor.

FREQUENCY XXX

This is the frequency value of the data point above.

AMPLITUDE XXX

This is the value of the data point at the above channel number.

To expand about the cursor, press bit 13 on the Processor once. Again press bit 13 to expand even further. Use bits 9 and 10 to position the cursor to another data point near the peak amplitude. Press bit 11 to get a new printout on the Terminal. Now, to return to a full block display, press bit 13 OFF (lower half of toggle switch). Here we see we have returned to the baseband measurement originally displayed.

Our cursor is still on the display but we turn it off temporarily by pressing bit 12 OFF (lower half of toggle switch). To put the cursor display back on, press bit 12 ON. To disable the cursor, press:

CURSR – I ENTER

DEMONSTRATION OF GRAPHICS

Your Fourier Analyzer contains software which allows you to produce an annotated plot of displayed data. (We've already seen a brief demonstration of this capability using the Gold Key PLOT function.) This plot may be traced on the Terminal or on an optional digital plotter. Control of the plot size, type, and annotation is achieved by User Programs. Once all plot parameters have been fixed, they may be retained to allow repetitive plotting with only a few Keyboard strokes.

To demonstrate this graphics capability let's first construct some data in the system. The following sequence of commands will produce data for plotting.

BLOCK SIZE   6   4   ENTER

Sets data block size to 64 points.

CLEAR   ENTER

Clears out data block 0.

KEY BOARD   0   SPACE   0   SPACE   5   ENTER

KEY BOARD   1   0   SPACE   9   ENTER

Press   1   ENTER

These last three commands (above) build a pulse in data block 0.

F   ENTER

This command converts data block 0 to a frequency spectrum.

USER PROG   5   8   2   1   SPACE   6   ENTER

This command tells the Processor that the Terminal will be our plot device.

NOTE

On the Terminal press the RESET TERMINAL button twice very quickly and the message "TERMINAL READY" should appear. Next while holding the SHIFT key down press the Terminal A DSP key once. This will cause the flashing cursor on the Terminal to be shut off.

To plot the signal on the Display Unit press the following sequence of keys:

USER PROG | 5 | 8 | 0 | 4 | ENTER

Sets plot size for full screen.

USER PROG | 5 | 8 | 0 | 5 | ENTER

Sets plot origin for lower left hand corner.

USER PROG | 5 | 8 | 0 | 9 | ENTER

Read the Display Unit switches.

USER PROG | 5 | 8 | 1 | 0 | ENTER

Set horizontal range of plot.

USER PROG | 5 | 8 | 1 | 3 | SPACE | 1 | ENTER

This command shuts off the echo to the Terminal. This means that all commands which follow will not be printed on the Terminal as they have up to this point.

USER PROG | 5 | 8 | 1 | 4 | ENTER

Erases the Terminal screen.

USER PROG | 5 | 8 | 1 | 5 | ENTER

Plots the display on the Terminal.

USER PROG | 5 | 8 | 1 | 6 | ENTER

Draws axis on the Terminal.

USER PROG | 5 | 8 | 1 | 8 | SPACE | 1

Sets graphics software to draw all grid lines.

USER PROG | 5 | 8 | 1 | 7 | ENTER

Annotates plots and draws all grid lines.

To enable the keyboard echo to the Terminal, again press RESTART. We now have a fully annotated plot of our display on the Terminal.

Now that we've entered the values above, we can call up User Program 5800 to do repetitive plotting without re-entering all the values above. See Section 6 for details.

DEMONSTRATION OF VARIABLE PARAMETERS AND TEXT BUFFERS

The following paragraphs demonstrate two powerful capabilities of the Fourier Analyzer: Variable Parameters and Text Buffers.

VARIABLE PARAMETERS

Thus far we have considered entering Keyboard commands with constant parameters. For example, the command to store data block 0 into data block 1:

STORE   1   ENTER

has a constant first parameter equal to 1. However, the system will also allow this or any other parameter to be a "variable", which allows for far greater command flexibility, especially when writing Keyboard Programs.

Variable Parameters serve the same functions as the Variables "X" and "Y" in the algebraic expression:

$Y = X + 1$

This expression says "Take the value of variable X, add one to that value, and let the result be the value of variable Y." In this expression, the names "X" and "Y" really refer to locations where the actual values of "X" and "Y" are stored. In a similar manner, Variable Parameters in the Fourier System are used to denote memory locations where values are stored.

In the Fourier system, Variable Parameters are referred to by assigned *numbers* (rather than by letters such as "X" and "Y" above), so that there is in the system a Variable Parameter #1, Variable Parameter #2, and so forth. The value of a Variable Parameter, once assigned by the operator, does not change except by program (or operator) direction. The number of Variable Parameters in the system is fixed — for a complete list of the available Variable Parameters, see Section 3. There is a one-to-one correspondence between a Variable Parameter number and a storage location in memory, which means that a specific Variable Parameter can have only one value at any time.

Because Variable Parameters are specified by number, we need a convention to distinguish a Variable Parameter from a constant Parameter in Keyboard commands. Therefore, Variable Parameters are distinguished from fixed integers in Keyboard commands by adding the character "D" after the Variable Parameter number — for example: "0D", "1D", "2000D", etc. When entering a Variable Parameter from the Keyboard, the "DSPLY" Key (with mnemonic "D ") is used to add the character "D" (or the character "D" may be typed on the Terminal). For example, to enter Variable Parameter number 1 in a STORE command instead of a fixed value of 1, the entry would be:

STORE   1   DSPLY   ENTER

This will print "X>1D" on the Terminal, where "1D" is the name referring to Variable Parameter number 1. This command says to the system: 'Find the value of Variable Parameter 1, then store data block 0 into the data block specified by that value.'

Note that, in the discussion above, the numeric Parameter "1" by itself was assumed to be a fixed integer (rather than a Variable Parameter) when used with the "STORE" command. *In general,* all integer parameters are interpreted by the system as simple integer values. *However,* some of the system Gold Key functions and User Programs that operate specifically on or with Variable Parameters interpret some of their parameters as Variable Parameter numbers *which means that the "D" character suffix after the*

Variable Parameter number is "implied" and as such need not be given. An example of this implicit Variable Parameter specification is the gold SET command:

[USER PROG] [←] [0] [SPACE] [1] [DSPLY] [ENTER]
SET

Note that the Terminal echoes "Y__0 1D". This command says 'assign the value of VP 1 to VP 0.' Note that we specified VP 1 explicitly (using the D suffix) and VP 0 implicitly. The implicit specification may only be used to specify a parameter as a Variable Parameter if that parameter is interpreted by the program as being a Variable Parameter number, as is the case with the first parameter of the gold SET command.

All Gold Key commands and User Programs interpret their parameters in a predetermined manner, as explained under each key and in the Variable Parameter paragraph in Section 3.

In the following discussion we shall see examples of both explicit and implicit Variable Parameter specification.

The real benefit of Variable Parameters arises from the fact that in a Keyboard Program the parameters of a Keyboard command may be dynamically changed without having to change the command itself. For example, the command:

[CLEAR] [1] [DSPLY] [ENTER]

(printed on the Terminal as "CL 1D") would have different meanings depending on the value of Variable Parameter number 1. If the value of Variable Parameter 1 was "2", this instruction would be interpreted by the system to mean "clear block 2"; however, if the value of parameter 1 was "10", the instruction would then be interpreted as 'clear block 10". Without editing the command we have changed its meaning in the program.

There are many operations which may be performed on or using Variable Parameters. Some of these operations are arithmetic functions, while other operations set, list, input, or output values. Also, some functions allow access to and manipulation of data in data blocks. The next example will demonstrate some of these functions.

To begin, we set the system block size to 64 by entering:

[BLOCK SIZE] [6] [4] [ENTER]

Note that the "64" BLOCK SIZE light is lighted on the Keyboard.

We now clear block 0 and then create a data block with a constant value of 1 (similar to the block fill method demonstrated earlier, or see KEY BOARD key in Section 3):

[CLEAR] [ENTER]

[KEY BOARD] [0] [SPACE] [0] [SPACE] [6] [4] [ENTER]

[KEY BOARD] [–] [4] [SPACE] [0] [SPACE] [0] [ENTER]

[1] [0] [0] [0] [0] [ENTER]

Now let's input some Variable Parameter values to the system. We wish to enter values for Variable Parameters 0,1, and 2, which we do by entering:

INPUT

[USER PROG] [PHOTO READR] [0] [SPACE] [2] [ENTER]

Note that, with the gold INPUT command, the Variable Parameter numbers were specified implicitly to be 0 and 2.

Also note that after pressing ENTER, the system is in the BUSY mode — this is because the system is now waiting for the values of parameters 0, 1, and 2 to be entered. We will now enter the values of 0, 10, and 20 for Variable Parameters 0, 1, and 2, respectively:

[0] [SPACE] [1] [0] [SPACE] [2] [0] [ENTER]

(The SPACE key serves as a delimiter for this entry.) Having received the required number of values, the system now returns to the READY mode.

Let's list the values we just entered for parameters 0 through 2. We can do this by entering (make sure the Processor S-register is cleared before entering this command):

[USER PROG] [LIST] [0] [SPACE] [2] [ENTER]

LIST

(Again, note the implicit Variable Parameter specification.)
The following listing should occur:

0D = 0 (value of parameter 0 = 0)
1D = 10 (value of parameter 1 = 10)
2D = 20 (value of parameter 2 = 20)

Now lets get the value of one of the data points on the data block just created into Variable Parameter 2000. Enter the following:

GET

[USER PROG] [LOAD] [2] [0] [0] [0] [SPACE] [0]

[SPACE] [1] [5] [ENTER]

This loads Variable Parameter 2000 with the amplitude value of the data in block 0, channel 15. (Again, note the "implicit" specification of VP 2000).

To list the value just obtained from the data block, we can enter the following command, which will list the value of Variable Parameter 2000:

[USER PROG] [LIST] [2] [0] [0] [0] [ENTER]

LIST

The following printout should now occur:

2000D = .9999*

**The value is not identically equal due to the conversion from data block format to floating point format. See Section 3, Variable Parameter User Programs, Y 1821.*

This is the value we originally put into the block. Notice that the value of parameter 2000 is a floating point value (as compared to previously entered and printed integer values) since this Variable Parameter has been defined to be of floating point type. Complex type (floating point real and imaginary values) Variable Parameters are also defined in the system, as discussed in Section 3.

Now let's divide the data value just obtained by 2 and then put it back into the block. To do the division, we enter:

USER PROG ÷ 2 0 0 0 ENTER

DIV

Default command for division by 2.

We then list the Variable Parameter to verify the division:

USER PROG LIST 2 0 0 0 ENTER

LIST

The Terminal will display the following:

2000D = .4999

The Variable Parameter is now half of its original value.

Now before we put the value back into the block, let's clear out a portion of the block around the channel of interest to make the inserted data point easier to see. To do this we could enter the command:

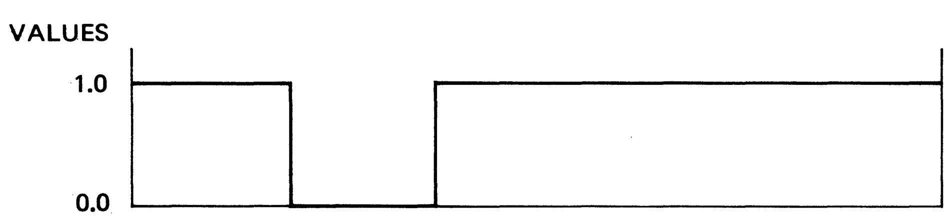
CLEAR 0 SPACE 1 0 SPACE 2 0 ENTER

This clears block 0 between channels 10 and 20 (recall that we obtained the data value from channel 15). However, we can use the Variable Parameters we previously entered (0,1,2) to do this as follows (recall that parameter 0 = 0, parameter 1 = 10, parameter 2 = 20):

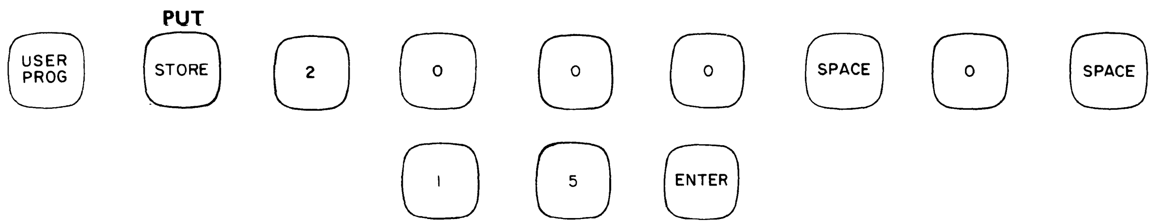
CLEAR 0 DSPLY 1 DSPLY 2 DSPLY ENTER

(Note that "spaces" are not necessary since the echoed characters for the display command are "D", the space is already supplied by that command). This is an example of a basic command which requires the DSPLY mnemonic to be used after the variable number if a Variable Parameter is intended (explicit Variable Parameter specification).

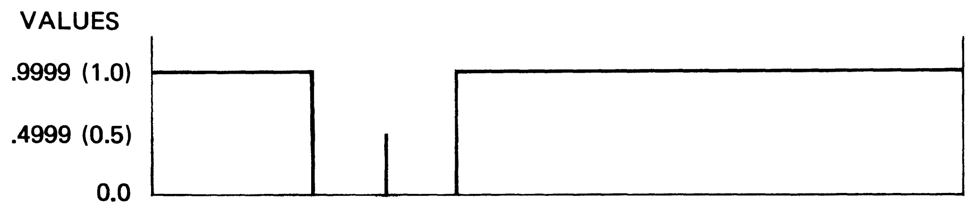
Now put the Display into the BAR mode (DISPLAY/TYPE switch). The resulting display should appear as follows:



We now insert (or "put") the new data value back into block 0 at channel 15:



This command puts the value of parameter 2000 into data block 0 at channel 15. The display should now appear as follows:



The previous sequence of steps illustrates how Variable Parameters may be used in the Keyboard command sequence, and how they are useful in accessing and operating on data in the system. There are also several significant benefits to be derived from using Variable Parameters within Keyboard Programs which are illustrated both in the remaining portions of this section and in the Variable Parameter discussion of Section 3.

TEXT BUFFERS (See also User Programs 5803, 5819, 5838, 5839 in Section 6)

We will now discuss and demonstrate the capability of the Fourier Analyzer to print and save ASCII text. Actually, two separate and distinct capabilities are available, each having its own advantages.

The first capability for text saving and printing is that of the MASS STORE key (see Section 4). File 4, ASCII Text, of the Disc is used for saving lines of text entered from the Keyboard. It is especially useful for saving messages and then recalling them via the MASS STORE, 'Search for Key' command, which will locate and print on the Terminal any message saved in File 4 that begins with the characters you supply.

An alternate text handling capability is also resident in the Fourier Analyzer. Although the Search Key capability is not available, we can edit stored text via an interactive editor that greatly simplifies the text entry process.

This text capability utilizes modules called 'text buffers' which may contain anywhere from 1 to 99 separate messages of varying lengths. There are a number of text buffers stored on the Disc (for the exact number and their location, refer to Section 4) and there is one core-resident text buffer. When it is desired to print a message from a text buffer, that buffer must first be read from Disc to core, then the message may be printed. The text editor can either edit text buffers on the Disc or in core.

Cértain text buffers are reserved for system use and should therefore not be used by the operator. Section 6, Y 5803, details which buffers are affected.

In the following demonstration we will see how messages are entered into text buffers, and we will use some of the text buffer editing commands. The message we enter will be used later when we learn how to write Keyboard Programs.

Let's assume we'd like to insert a message into text buffer number 20 on the Disc. We accomplish this by first invoking the Text Buffer Editor. Press:

USER PROG | 5 | 8 | 0 | 3 | SPACE | 2 | 0 | ENTER

When we make this entry, the system will enter the BUSY mode and the following messages will appear on the Terminal:

WARNING — CLEAR NEW DISC BUFFERS
ILLEGAL MESSAGE
???

The "ILLEGAL MESSAGE" warning is printed because this text buffer is completely uninitialized (it will not appear again once the buffer is initialized). We are now in the Text Buffer Editor, and can use some of the editor commands to enter our message. We will enter message number 1 into this buffer. To do this, we enter (on the system Terminal, with the AUTO LF key depressed):

CL (RETURN) to initialize the text buffer
/I (RETURN) to insert a message
01 (RETURN) the message number
ENTER DESIRED NUMBER OF AVERAGES (RETURN) the message itself
/* (RETURN) message terminator

Now list out the contents of the text buffer to verify the entry, by entering on the Terminal:

/L (RETURN)

The contents of the text buffer are then listed:

01
ENTER DESIRED NUMBER OF AVERAGES
/*

If you have made an error in the entry of the message and the above message is not printed out, you may correct it by first entering from the Terminal:

/D01 (RETURN) deletes message 1 if it exists

and then inserting the line over again as above.

When the correct message is listed on the Terminal, we may exit the Text Buffer Editor by entering "/ (RETURN)" from the Terminal (or TERM ENTER from the Keyboard. Note that many of the previously used 2-character commands are also Keyboard key mnemonics, see Appendix D for complete list). Text buffer 20 is now saved back on the Disc. (A copy of it also resides in core.) The Fourier Analyzer is now in the READY state.

We can now use another Keyboard command to print out message number 1 in the current core text buffer (a copy of Disc text buffer 20). We do this by entering the command:

USER PROG | PRINT (TEXT) | 1 | SPACE | 1 | ENTER

The previously entered message:

"ENTER DESIRED NUMBER OF AVERAGES"

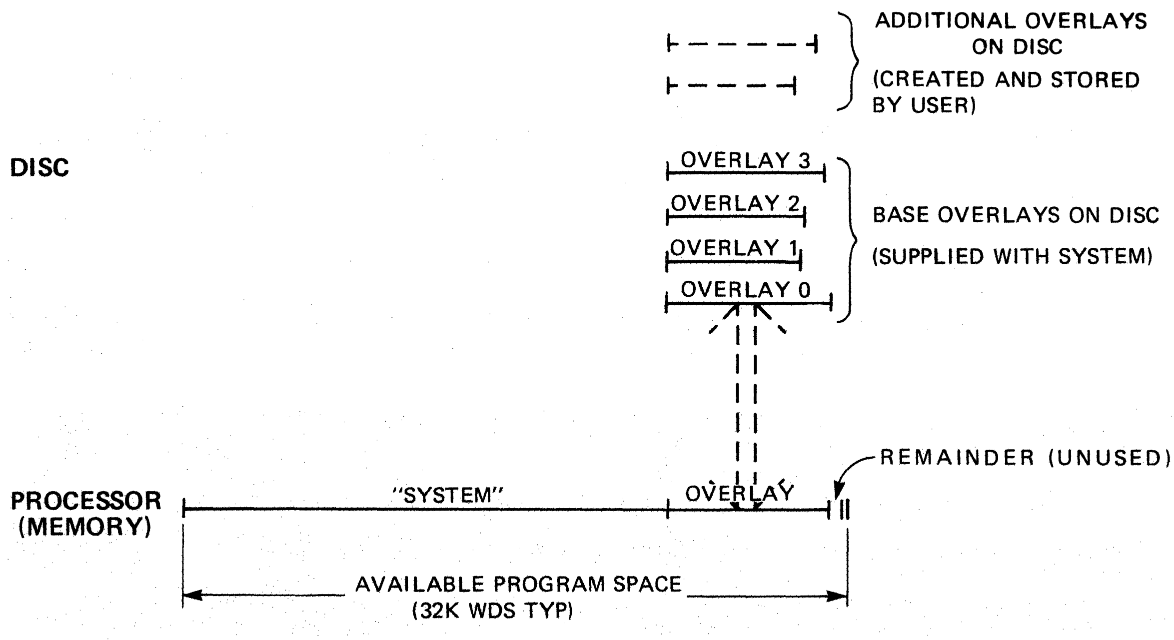
will be printed on the Terminal.

We have just seen how to use the interactive Text Buffer Editor to enter a message of text into a Disc text buffer, and how to use the gold TEXT function to print a message from a core text buffer. As we shall see shortly, this capability is extremely useful in providing interactive text within Keyboard Programs. For more information on text buffers, see Section 6.

OVERLAY SWAPPING

Let's briefly take a look at the concept of 'overlay swapping', and how it is used in the Fourier Analyzer. Thus far we have been executing some basic commands and User Programs without much concern for exactly *how* they were being executed. However, once you start operating your Fourier System in earnest you will want to know how the system software is structured and what impact this structure has on system operation. A few of the basic ideas are presented here; see Sections 4 and 7 for more information.

Basically, the system software is composed of many pieces, one 'system', and multiple 'overlays'. The system and one of the overlays are always in operation within the Processor ("in core"). Pictorially, this is shown as follows:



The "system" contains software for most of the basic Keyboard commands and important User Programs. Each "overlay" contains software for additional commands and specific User Programs.

As we shall see, overlays allow much more capability to be added to the Fourier Analyzer than would be possible if the total program space was limited to the space available in the Processor. The reason for this is that the *Disc* is used as storage for overlays (programs) not currently in use — then, when a program is required from a specific overlay, the system automatically reads that overlay from the Disc into the Processor and uses it as depicted in the above diagram. In this manner, the program space in the Fourier Analyzer is effectively that of the Processor *plus* the overlay storage space on the Disc.

The Fourier Analyzer includes several overlays already stored on the Disc. In addition, the capability exists for you to add your own overlays to the system — this subject is completely described in the System Software Manual.

Let's find out some more about what is specifically contained in each of the current system overlays. From the Keyboard, enter the command (you may wish to decrease the BAUD RATE setting of your Terminal for this operation to, say, 300, so that you may more carefully observe the printout that will occur).

MASS STORE | 5 | 8 | SPACE | 9 | 9 | ENTER

When this command is entered, a listing of the User Programs resident in the "system" and each of the "overlays" will be printed on the Terminal:

SYSTEM

1800
1801
.
.
.
.

OVERLAY 00

0005
0006
.
.
.
.

OVERLAY 01

0005
0006
0041
0044
0045
.
.
.
.

OVERLAY 02

0005
0006
0040
0043
0044
0045
.
.
.

OVERLAY 03

5800
5803
5804

We now recognize some of the User Programs we have used in preceding discussions: Y (User Program) 3024 (from the system), Y 41 and Y 45 (from overlay 1), and Y 5803 (from overlay 3) to name a few.

Even though some of these User Programs were in different overlays, there was nothing special we had to do to access them. If a User Program is not in the overlay currently being used, the Fourier system will automatically search the Disc overlays starting with overlay 0 until it finds an overlay containing the requested User Program, and then read that overlay into core and execute the User Program. This is why we didn't have to be concerned about the specific locations of the User Programs we were using when we executed the commands.

Although this automatic overlay swapping makes the system quite easy to use, some considerations must be taken into account when writing Keyboard Programs that use more than one overlay. These considerations are explained in detail in Section 7.

DEMONSTRATION OF KEYBOARD PROGRAMMING

A Keyboard Program is a sequence of Keyboard-entered commands which the Fourier Analyzer will perform automatically. When you begin a program, the memory locations may be empty, but most likely they will be partially filled with commands from a previous program. Therefore, setting up a new program can be considered to be editing the contents of the program memory. Hence, every program entry must begin with an editing command.

In general, you will begin with a REPLACE command, naming line 0 as the line to be replaced by one or more new commands. These new commands will then automatically push previous commands down in the memory and off the end. The BUSY light should come on after the REPLACE command and remain on while the program is being entered.

After the program steps are entered, the END command is given to indicate the end of the program. Then the TERMINATE command is used to terminate the programming mode i.e., return to the READY mode.

To check that the steps have been entered correctly, the LIST command is used, producing a listing on the Terminal. In the listing, line numbers are automatically assigned to each commmand, indicating the sum of the previous command elements up to that point. If any corrections or changes are required, one again returns to the programming mode by the RePLACe, DELETe, or INSeRT keys, referencing the line numbers to be changed or corrected. After the changes have been made, the TERMINATE command again terminates the programming mode.

There are two ways to start the program running: one is from a given label, the other is from a given line. To start from a given line, the POINT command is used to set the internal pointer to the desired starting line. Then the CONTINUE key is pressed. The pointer always points to the line being processed during the running of a program, or to the line being edited during an editing operation.

SUBROUTINES

Any group of Keyboard commands may be begun with a label statement and ended with the SUB ReTuRN key. This group of commands can be executed from any point in the overall program by jumping to the label number at the beginning of the subroutine. After jumping to the subroutine the system will return to the next step after the JUMP command. This allows programs that are used over and over again to be written only once.

PROGRAM STACK JUMPING

An extension of the subroutine capability is the ability of the system to jump from the body of a core-resident, running program to a disc-resident program and back again. This can be thought of as an extension of the subroutine capability. A second parameter on the JUMP key determines which disc-resident program is desired. The system will allow nesting of subroutine jumps (either within the present stack or to disc-resident stacks) up to 10 levels deep.

PARAMETER SELECTION

Since the use of Variable Parameters in a Keyboard Program greatly enhances the flexibility of a routine, we will want to include these in our demonstration. As described previously, any parameter in any Keyboard command may be replaced by a Variable Parameter. The contents of this location rather than the number of the location determines the value of the command parameter. The contents of a Variable Parameter location may be set manually via the Keyboard, or more importantly, changed automatically while the program is running. They can be changed via arithmetic operations between variables, by means of responses to questions on the Terminal, or by the outcome of some test conditions.

INSERTING TEXT

Text statements may be incorporated into Keyboard Programs to annotate programs, to prompt the user toward some action, or to ask for an input from the operator. We will use an example from the previous demonstration to incorporate a text buffer.

A PROGRAM EXAMPLE

As an introduction to programming the Fourier Analyzer, we will do a simple power spectrum averaging program, again employing the CHECK signal used in our earlier discussion of the analog input. We will take the CHECK signal in randomly, i.e., so that the position of the signal in the data window will be random, then average the power spectrum of a number of inputs to arrive at an average power value of the signal.

The sequence of our program is as follows:

Clear block 1, as this is the block where the successive sums will be accumulated. Here, any residual data could cause erroneous results.

Analog input to block 0. This takes in a block of data which consists of the CHECK signal at some random location in the block.

Fourier transform block 0. This converts the time waveform into its Fourier spectral components.

Conjugate multiply block 0. This multiplies block 0 by its conjugate, thus forming what is known as the power spectrum of the signal.

Add block 1 to block 0. This is the summing step in computing the average. The sum resides in block 0.

Store block 0 in block 1. This shifts the new sum into block 1 so that a new block of data can be entered into block 0.

Count, (i.e., repeat) the steps from the analog input command the number of times desired to achieve the final sum.

Divide block 1 by the number of count times, above, to compute the average.

End Program and terminate program mode.

PROCEDURE

The CHECK signal has an amplitude of about 51 mV, and a length of 1100 μs. Since its repetition rate is the power line frequency (60 Hz), the period is about 17 ms (or 20 ms for 50 Hz line frequency). We will be sure to get at least 1 pulse in every input if we choose a sample interval (Δt) of 100 μs, and a block size of 256. This then gives a total record length of 0.1×256 or 25.6 ms, which is more than the 17 ms period. Therefore, on the ADC, set:

INPUT SELECTOR to A
SAMPLE MODE to kHz/μs
MULTIPLIER to black 100
TRIGGER SOURCE to FREE RUN
OVERLOAD VOLTAGE A to CHECK

Next, set the block size by pressing:

[BLOCK SIZE] [2] [5] [6] [ENTER]

Look at the Terminal for a printout of each Keyboard command. A complete list of the Terminal symbols, in alphabetical order, is given in Appendix B.

Now to begin programming at line zero, we use the REPLACE command as follows:

RPLAC   0   ENTER

Note the BUSY light turns on. Now label the starting point of the program:

LABEL   1   ENTER

Then we clear out any residual data in block 1 by pressing:

CLEAR   1   ENTER

Next, put in another label. This is because the remaining portion of the program will be a loop, i.e., a repeated sequence of steps, and this is the starting point for the loop. Press:

LABEL   2   ENTER

Now the analog input is entered. We will take the data into block 0, and display block 1. The command is:

ANALG IN   0   SPACE   1   ENTER

Next we must Fourier transform the time domain data into the frequency domain. The command is:

F   ENTER

This results in a complex frequency spectrum (real and imaginary components) from dc to the selected maximum frequency.

To obtain the power spectrum of the waveform (in terms of volts^2), we conjugate multiply the data in block 0 by pressing:

* MULT   ENTER

Since this program is intended to produce an average, we have to sum up the individual data records. The data comes into block 0 and if we add it to block 1, then the sum will reside in block 0. To shift the summation into block 1, we enter:

+   1   ENTER

and

STORE   1   ENTER

To obtain the average, we will first sum 250 inputs and then divide by 250. This means, first, repeating the process from label 2 through the above step 250 times, which is entered by the command:

COUNT 2 SPACE 2 5 0 ENTER

Then press, for the division:

÷ 1 SPACE 2 5 0 ENTER

To see the function more clearly, change the coordinates from rectangular to log magnitude by pressing:

LOG MAG 1 ENTER

This is the end of the program so press:

END ENTER

And finally, to get out of the programming mode, we must press:

TERM ENTER

The BUSY light is now off and the READY light is on, indicating we are out of the programming mode. To list the program, press:

LIST ENTER

The correct listing will appear on the Terminal as follows (note that the Terminal prints, from left to right, the line number, the command symbol, and the parameters):

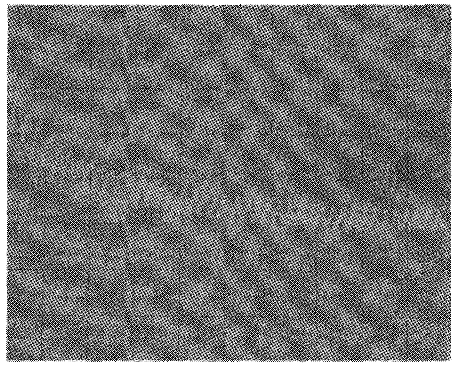
TERMINAL					MEANING
1	L	1			Label 1
5	CL	1			Clear block 1
9	L	2			Label 2
13	RA	0	1		Analog input to block 0, display block 1
18	F				Fourier transform block 0
21	* -				Conjugate multiply block 0
24	A+	1			Add block 0 to block 1
28	X>	1			Store contents of block 0 in block 1
32	#	2	250	0	Count (repeat steps) from label "2", 250 times. 0 means that no repetitions had occurred at time of LIST command. If the program is stopped in the midst of running, and then the COUNT line is listed, the number of repetitions up to the STOP will be given.
38	:	1	250		Divide block 1 by 250
43	TL	1			Take Log of Block 1
46	.				End of program

RUNNING THE PROGRAM

There are two ways of starting a program. One is by setting the pointer to the starting line number, and then pressing the CONTINUE key; the other is by using a JUMP command. (Note: a third way is to use one of the gold "soft" keys F1 through F6 on the Keyboard as explained in Section 3; however, for our present purpose we will restrict ourselves to two.) First, we'll use the JUMP command to jump to label 1 as follows:

JUMP　1　ENTER

The program should now run through and stop, displaying the final average. A typical display might be as follows:



To start the program using the pointer, press the following keys, remembering that, with the POINT command, a line rather than a label is named.

POINT 1 ENTER

Then press:

CONT

The program should again run through.

CORRECTIONS AND CHANGES

To illustrate how program corrections and changes can be made, we will modify the program to allow us to enter the number of averages desired. This will also demonstrate the Variable Parameter capability of the system.

To change a line we can use either the two commands DELETE and INSERT, or the single command REPLACE.

In the REPLACE command, the lines to be replaced are named. In our example, we want to make the second parameter of the COUNT command a variable (line 32 by our previous listing) and make the second parameter of the DIVIDE command a variable (line 38 of the program). Therefore, first press:

RPLAC 3 2 SPACE 3 8 ENTER

The system will go into the BUSY mode, meaning that it is waiting for more commands. Next press:

COUNT 2 SPACE 1 DSPLY ENTER

The Terminal will echo "# 2 1D" where 1D denotes that Variable Parameter number 1 will be used to store the number of averages desired. Now press:

÷ | I | SPACE | I | DSPLY | ENTER

Again, Variable Parameter 1D replaces the fixed number 250 with which we divide the final result to obtain the average power.

To exit the programming mode, press:

TERM | ENTER

Since we have set up Variable Parameter 1 to be the number of averages, our next change is to insert a command which will allow us to enter the desired number. To do this we will use the INSeRT key. We would like to set the number of averages at the start of the program, say after line 5, so we enter:

INSRT | 5 | ENTER

The command to input a value into a Variable Parameter involves the gold INPUT function (see PHOTO READeR key). Prior to this command though, we will first print a message using the text buffer that will prompt us to enter the number of averages. This message has been set up in the previous Demonstration for Text Buffers. Press:

USER PROG | 5 | 8 | 3 | 8 | SPACE | 2 | 0 | ENTER

This recalls the relevant text buffer from Disc to core. Then press:

USER PROG | PRINT (TEXT) | I | SPACE | I | ENTER

This will print the message on the Terminal when the program is run. Then press:

USER PROG | PHOTO READR (INPUT) | I | ENTER

When the program is run, this will cause the system to wait for the value of Variable Parameter 1 (the number of averages) to be entered before continuing through the program.

Finally, press:

TERM | ENTER

Our program has now been edited. Note that any number of sequential commands can be entered with one INSERT command.

To check the program, press:

LIST ENTER

The program can again be listed to check the proper entry. it should appear as follows:

1	L	1	
5	CL	1	
9	Y	5838	20
14	Y W	1	1
30	Y R	1	
25	L	2	
29	R A	0	1
34	F		
37	* -		
40	A+	1	
44	X>	1	
48	#	2	1D
54	:	1	1D
59	T L	1	
63	.		

Note also that we changed the program from the bottom up since line numbers change automatically below the line that was changed. This procedure keeps us from having to check for new line numbers each time we make a change.

Now run the program again by pressing:

JUMP I ENTER

When the program asks for the number of averages, enter the number via the ENTRY keys on the Keyboard, for example, 25 ENTER.

Subroutining and program stack jumping are additional programming techniques that are covered under the JUMP key in Section 3.

REVIEW

After performing these demonstrations, you should now be familiar with the analytic features of the system, such as baseband vs. BSFA measurements, display cursor operation, and graphics plotting capability, as well as such programming features as gold key functions, variable parameters, text buffers, overlay swapping, and keyboard programs.

You should then be ready to begin your own measurements, referring as necessary to the remaining sections in this manual alluded to throughout these demonstrations. The following final demonstration may be useful to you in applying two types of measurements, power spectrum and transfer function, which we have pre-written and stored as examples of combining the major features of the system into applied measurement programs.

DEMONSTRATION OF AUTOMATIC MEASUREMENTS (F1-F6 Keys)

The Gold Keys F1 through F6 (or "soft keys") can be used to invoke entire measurement programs that you have previously written and stored on the Disc.

We have provided two such programs, a power spectrum program and a transfer function program on F5 and F2 respectively, to serve as examples. The listings of these programs are provided in Appendix D. **Since the programs and the interactive text are stored in unprotected areas of the Disc (Files 3, 4, and 7), care must be taken not to write over these records when saving your own Keyboard programs and text messages (see Appendix D for allocation of file records).**

F5 POWER SPECTRUM

First we will demonstrate the power spectrum measurement program associated with soft key F5, using the ADC CHECK pulse as the input signal. Place the Terminal in Alphanumerics mode (see Turn-On Procedure) and enter the following command on the Keyboard:

USER PROG | POWER SPECT | ENTER

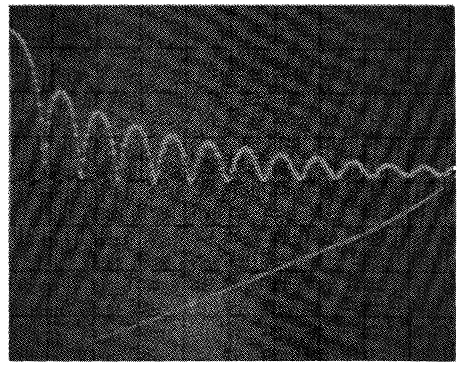
This initiates the power spectrum Keyboard Program. Now perform the following actions in response to the text messages that appear on the Terminal.

TEXT MESSAGES ON TERMINAL	ACTIONS TO TAKE
HP POWER SPECTRUM PROGRAM SET ADC FREQUENCY RANGE AS DESIRED (SAMPLE MODE & MULTIPLIER) SET ADC TRIGGER TO "FREE RUN" CHANGE BLOCK SIZE IF DESIRED PRESS "CONTINUE" WHEN READY	Set ADC controls: SAMPLE CONTROLS for 10 kHzF_{max} TRIGGER SOURCE to FREE RUN INPUT SELECTOR to A OVERLOAD VOLTAGE A to CHECK Enter on Keyboard: BLOCK SIZE 512 ENTER CONTINUE
ARE HP FILTERS INSTALLED? 0=NO, 1=YES	Filters will not be used for this measurement. Enter: 0 ENTER
SET KEYBOARD REPEAT/SINGLE SWITCH TO "REPEAT" PRESS "CONTINUE" WHEN READY	Place REPEAT/SINGLE switch to REPEAT. Press: CONTINUE
SET OVERLOAD VOLTAGES AND TRIGGER LEVELS FOR SIGNAL AMPLITUDES MOVE REPEAT/SINGLE SWITCH TO "SINGLE" WHEN READY. IF SOURCE IS NOT IN FREE RUN, TRIGGER THE SYSTEM AGAIN TO CONTINUE.	Note CHECK pulses across the screen. Place REPEAT/SINGLE switch to SINGLE.
ENTER THE NUMBER OF AVERAGES DESIRED	For the number of averages, press: 2 0 ENTER
ENTER MEASUREMENT TYPE 1=BASEBAND 2=ZOOM	We will make a baseband measurement first, and a BSFA measurement in the next example. Press: 1 ENTER
PRESS "CONTINUE" FOR MEASUREMENT	To begin the measurement, press: CONTINUE

The system now enters the BUSY mode to make the measurement using the parameters we have entered. When the measurement is completed, the following message will appear on the Terminal and the display will be similar to the photograph shown. Refer to Section 3, POWER SPECTrum Key for explanation.

```
MEASUREMENT COMPLETE

TO COPY DISPLAY ON TERMINAL:
   PUT TERMINAL IN GRAPHICS MODE
   PRESS "GOLD KEY" "PLOT" (enter)
TO MAKE ANOTHER MEASUREMENT:
   PUT TERMINAL IN ASCII MODE (enter)
   PRESS "CONTINUE"
```



F5 POWER SPECTRUM (BSFA)

This Soft Key program is the same as the previous example except we will use an oscillator for a signal and select a zoom (BSFA) measurement after the message ENTER MEASUREMENT TYPE.

Enter the following command (as before) to initiate the automatic power spectrum program:

[USER PROG] [POWER SPECT] [ENTER]
F5

This restarts the power spectrum Keyboard Program. Perform the actions noted in response to the Terminal messages:

TEXT MESSAGES	ACTION
HP POWER SPECTRUM PROGRAM SET ADC FREQUENCY RANGE AS DESIRED (SAMPLE MODE AND MULTIPLIER) SET ADC TRIGGER TO "FREE RUN" CHANGE BLOCK SIZE IF DESIRED PRESS "CONTINUE" WHEN READY	Set ADC as follows: SAMPLE MODE to kHz/μs MULTIPLIER to black 100 INPUT SELECTOR to A OVERLOAD VOLTAGE A to .25 TRIGGER to FREE RUN With a block size of 512, press: CONTINUE
ARE HP FILTERS INSTALLED? 0=NO, 1=YES	With a clean sinewave we will not need filters. Press: 0 ENTER
SET KEYBOARD REPEAT/SINGLE SWITCH TO "REPEAT" PRESS "CONTINUE" WHEN READY	Place switch to REPEAT; press: CONTINUE
SET OVERLOAD VOLTAGES AND TRIGGER LEVELS FOR SIGNAL AMPLITUDES MOVE REPEAT/SINGLE SWITCH TO "SINGLE" WHEN READY. IF SOURCE IS NOT IN FREE RUN, TRIGGER THE SYSTEM AGAIN TO CONTINUE	Set output frequency of oscillator to 2 kHz and connect to channel A. Vary amplitude for sinewave of 3 cm p-p on the display (about 200 mV p-p) Place REPEAT/SINGLE switch to SINGLE

ENTER THE NUMBER OF AVERAGES DESIRED

To take 5 averages, press:

5 ENTER

ENTER MEASUREMENT TYPE
1=BASEBAND
2=ZOOM

For this example we will zoom on the signal. Press:

2 ENTER

ENTER ZOOM MEASUREMENT MODE
1=ON LINE, PREPROCESSOR
2=OFF LINE, PREPROCESSOR
3=OFF LINE, SOFTWARE

We will use off-line software zoom. Press:

3 ENTER

HOW WILL YOU SPECIFY ZOOM BANDWIDTH?
1=NUMERIC ENTRY -- CTR FREQ & BANDWIDTH
2=CURSOR -- ON PRIOR MEASUREMENT

We will use the numeric entry to zoom. Press:

1 ENTER

ENTER CENTER FREQUENCY

We will zoom around the 2000 Hz output of the oscillator. Press:

2000 ENTER

ENTER BANDWIDTH

We will use a bandwidth of 200 Hz around our center frequency of 2000 Hz. Press:

200 ENTER

ANALYZE OLD OR NEW DATA?
1=OLD (FROM THROUGHPUT FILE)
2=NEW

We will sample our oscillator signal (new data). Press:

2 ENTER

THROUGHPUT WILL USE TRACKS 135 THROUGH 198 ON THE LOWER (FSDS) DISC. IS THIS OK?
1=NO -- ABORT
2=YES -- PROCEED
3=NO -- ASK ME FOR TRACK #

We will use the tracks specified. Press:

2 ENTER

PRESS "CONTINUE" FOR MEASUREMENT

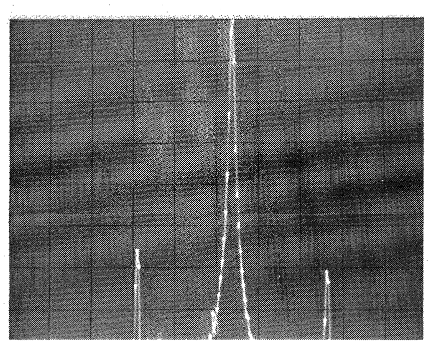
Now we will make our throughput measurement followed by zooming (BSFA) on the data. Press:

CONTINUE

THROUGHPUT COMPLETED
CNTR FREQ: 2000 HZ/DIV: 25.00
DF 0000.976525 BLOCKS LEFT: 47
ZOOM POWER: 20
BASEBAND: HZ/DIV: 500
DF: 0019.52125 BLOCKS LEFT: 48
ZOOM POWER: 1
MEASUREMENT COMPLETE

TO COPY DISPLAY ON TERMINAL:
PUT TERMINAL IN GRAPHICS MODE
PRESS "GOLD KEY" "PLOT" (enter)
TO MAKE ANOTHER MEASUREMENT:
PUT TERMINAL IN ASCII MODE
PRESS "CONTINUE"

Our measurement will take about 30 seconds to store (TRIGGER light on) and another 30 seconds to compute before the zoomed measurement is displayed. See example below.



The message above informs us of the amount of zooming we have used as well as the Δf of a baseband measurement at the present switch settings of the ADC. The display you get may not be identical to the one shown since the amount of harmonics will depend on the distortion of the oscillator used. The amount of offset from the center of the display (2000 Hz) is dependent on the exact frequency output of the oscillator.

F2 TRANSFER FUNCTION

Next we will demonstrate the transfer function measurement program associated with soft key F2. Clear the display and enter the following Keyboard command:

[USER PROG] [TRANS FCN (F2)] [ENTER]

This initiates the transfer function Keyboard Program and causes the following message to appear on the Terminal. Perform the responses indicated at right to set up the measurement parameters.

TEXT MESSAGES	ACTION
HP TRANSFER FUNCTION PROGRAM SELECT EXCITATION TYPE 1=RANDOM - BASEBAND OR ZOOM 2=HP DAC - BASEBAND ONLY 3=TRANSIENT - BASEBAND ONLY	Since we will use the CHECK pulse as our A and B inputs to simulate the transfer function of a bare wire, enter a signal type of random as follows: 1 ENTER
SET ADC FREQUENCY RANGE AS DESIRED (SAMPLE MODE & MULTIPLIER) SET ADC TRIGGER TO "FREE RUN" CHANGE BLOCK SIZE IF DESIRED PRESS "CONTINUE" WHEN READY	Set ADC controls: SAMPLE CONTROLS for 25 kHz F_{max} TRIGGER SOURCE to FREE RUN OVERLOAD VOLTAGE A & B to CHECK INPUT SELECTOR to AB With a block size of 512, press: CONTINUE
ARE HP FILTERS INSTALLED? 0=NO, 1=YES	Filtering will not be used so press: 0 ENTER
SET KEYBOARD REPEAT/SINGLE SWITCH TO "REPEAT" PRESS "CONTINUE" WHEN READY	Place REPEAT/SINGLE switch to REPEAT and press: CONTINUE
TURN ON RANDOM EXCITATION SOURCE SET OVERLOAD VOLTAGES AND TRIGGER LEVELS FOR SIGNAL AMPLITUDES MOVE REPEAT/SINGLE SWITCH TO "SINGLE" WHEN READY. IF SOURCE IS NOT IN FREE RUN, TRIGGER THE SYSTEM AGAIN TO CONTINUE.	Note CHECK pulses across the screen. Place REPEAT/SINGLE switch to SINGLE.
ENTER NUMBER OF AVERAGES DESIRED	For number of averages, press: 10 ENTER
ENTER MEASUREMENT TYPE 1=BASEBAND 2=ZOOM	For type of measurement, press: 1 ENTER
PRESS "CONTINUE" FOR MEASUREMENT	To begin the measurement, press: CONTINUE

The system now enters the BUSY mode to make the measurement. When the system returns to the READY mode, the following message will appear on the Terminal. Enter the commands shown to view the various displays.

```
MEASUREMENT COMPLETE
TO DISPLAY RESULTS, PRESS:

  "DISPLAY" "0" LOG TRANSFER FCN
  "DISPLAY" "1" COHERENCE
  "DISPLAY" "2" INPUT POWER SPECT *
  "DISPLAY" "3" OUTPUT POWER SPECT
  "DISPLAY" "4" CROSS POWER SPECT
TO COPY DISPLAY ON TERMINAL:
  PUT TERMINAL IN GRAPHICS MODE
  PRESS "GOLD KEY" "PLOT" (enter)
TO MAKE ANOTHER MEASUREMENT:
  PUT TERMINAL IN ASCII MODE
  PRESS "CONTINUE"
```

This display should be similar to the previous power spectrum display.
**Log magnitude, real part.*

1. **EXT/INT:** determines whether sample rate (F_s) is controlled *internally* by the SAMPLE MODE and MULTIPLIER switches, or *externally* through the BNC connector. External control allows for a one-to-one correspondence between the external clock and the sample mode. Requires a TTL input (low = <0.8V, high = >2.4V). The UNCAL light remains on in the external mode.

2. **SAMPLE MODE:** in left half of range, selects maximum frequency (F_{max}), called MAX FREQ, and sample interval (Δt), called Δ TIME, or one can select Processor control of the sampling parameters in the REMOTE position.

   In right half of the range, selects frequency resolution (Δf) called Δ FREQ, and total record length (T), called TOTAL TIME.

   Switching between the two positions on either side is equivalent to multiplying or dividing MULTIPLIER switch value by 1000.

   See Table 2-1 for instructions on choosing parameters.

   The REMOTE position disables all front panel switches except TRIGGERING and INPUT SELECTOR. Used only with the Modal & Signature Analysis Options, this position allows Processor control of the ADC front panel.

3. **MULTIPLIER:** selects values for parameters chosen by SAMPLE MODE switch.

4. **UNCAL:** lights when input sampling conditions set up by SAMPLE MODE and MULTIPLIER switches are not valid.

5. **TRIGGERING Light:** indicates that ADC is sending data to Processor. When data transfer is completed, light goes off. Operations with low duty cycles, such as the input of short blocks of data at high frequencies, will only light this indicator dimly.

6. **TRIGGER SOURCE**

   LINE: Trigger operates at power line frequency. The TRIGGER LEVEL switches are not active in this mode.

   INTERNAL (A): trigger operates on signal applied to INPUT A, as long as this signal has 1 division peak-to-peak amplitude on the display. The triggering point on the signal is controlled by the TRIGGERING LEVEL and slope controls.

   FREE RUN: trigger operates whenever an encode command is received from Processor. In this mode, blocks of data will be collected as fast as they can be accepted by the Fourier Analyzer, but not in synchronization with any external trigger signal. This mode is useful when data to to be used in computing power spectra, autocorrelation, or other functions that do not contain phase or time information between two signals.

   EXT: trigger operates from trigger signal applied to TRIGGER INPUT jack. In this mode, the trigger will work from any signal with a peak-to-peak amplitude greater than 100 mV. For simplicity of operation, no input attenuator for this mode is provided. However, a logarithmic limiter in the ADC makes triggering possible over a wide dynamic range. This results in the TRIGGER LEVEL control (for this mode only) being more sensitive for small trigger signal amplitudes than large — that is, the trigger level resolution is a constant percentage of the trigger signal amplitude. The SLOPE control works as it does in the INTERNAL (A) mode.

   AC: places a capacitor in series with the input jack to block dc.

   DC: no capacitor, signal is coupled directly through.

7. **TRIGGER LEVEL:** establishes voltage level on input waveform at which triggering occurs: + for positive amplitude, – for negative amplitude. SLOPE switch setting determines whether triggering occurs on positive slope (increasing side) or negative slope (decreasing side) of input waveform.

8. **SLOPE**

   POS: triggering occurs on positive-going slope of input waveform. Voltage level at which triggering occurs is set by TRIGGER LEVEL switch.

   NEG: triggering occurs on negative-going slope of input waveform. Voltage level is set by TRIGGER LEVEL switch.

9. **TRIGGER INPUT:** input for external trigger signal. TRIGGER SOURCE control must be set on EXT to use this input.

10. **INPUT CHANNEL A, B, etc:** inputs for analog signals; BNC-type connectors. For single channel analysis, the time-varying signal is applied to channel A. For dual channel measurements such as transfer function, the inputs are applied to A and B. C and D channels are not active unless optionally entered.

11. **AC-DC:** the AC position puts blocking capacitor between the corresponding input channel connector and corresponding input attenuator (OVERLOAD VOLTAGE switch). This capacitor passes frequencies down to 5 Hz (3 dB down). AC switch position should only be used when the dc in the signal to be analyzed is greater than about 50 times the rms amplitude of the signal itself. Input signal amplitude for switch in AC position must never exceed 200 Vdc.

    The DC position couples input signal straight from input channel connector to input attenuator (OVERLOAD VOLTAGE switch). Peak signal amplitude must never exceed ±17V or the protective fuse will open (spare fuse on A1 board in ADC).

12. **OVERLOAD VOLTAGE (input attenuators A, B, C, D):** select input voltage ranges corresponding to the INPUT CHANNEL analog signals. The positions represent peak voltages (±.125V through ±8V) allowed in each range. Total range over the switch settings is 36 dB. An input voltage which exceeds the peak voltage set will light the OVERLOAD VOLTAGE light.

    The Fourier Analyzer takes into account the OVERLOAD VOLTAGE switch setting whenever the Processor receives data from the ADC. Thus, all further data operations are on a calibrated basis. Therefore, it is not necessary to record the OVERLOAD VOLTAGE switch setting or use a calibration signal to establish the absolute value of a frequency or time record.

13. **OVERLOAD VOLTAGE:** lights or flashes if any single sample in a record exceeds the peak input voltage.

    CHECK position: enters a 51 mV pulse of 1100-μs approximate length into the ADC. This pulse has the frequency of the power line. When using this test signal, the ADC may be triggered in the LINE, INTERNAL, or FREE RUN mode (or in the EXT mode if an external trigger signal is provided).

14. **INPUT SELECTOR:** selects from 1 to 4 channels (A, AB, ABC, ABCD) for input to the Processor. Channel A data stored in block n1, channel B in block n1 + 1, etc.

15. **DISPLAY:** works with the REPEAT/SINGLE switch on the Keyboard to determine which analog input is to be displayed when the REPEAT/SINGLE switch is in the REPEAT position.

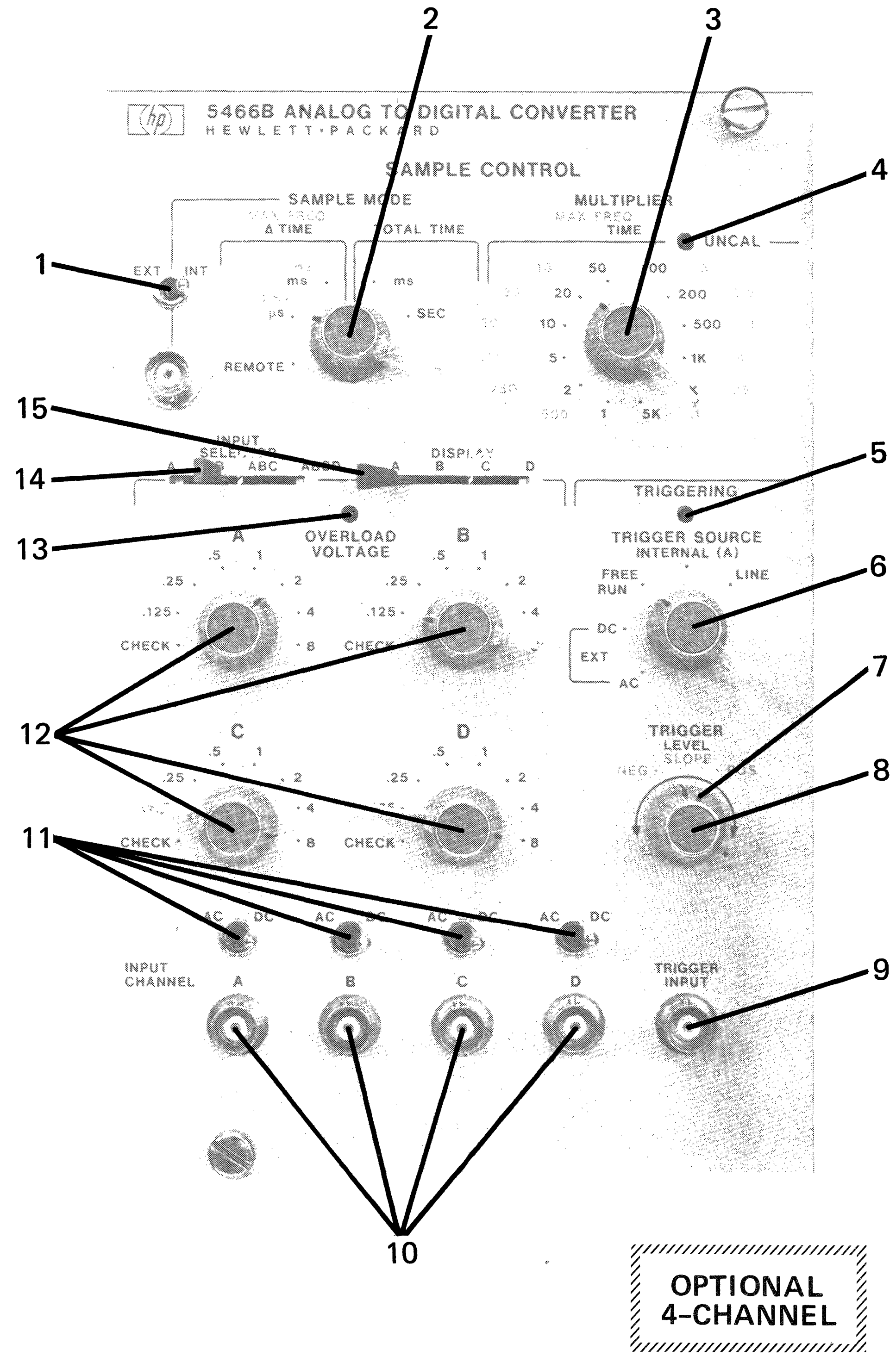


Figure 2-6. 5466A Analog-to-Digital Converter (ADC)

Figure 2-6. 5466A Analog-to-Digital Converter (ADC)

LIGHTS

BLOCK SIZE Lights

One of these will always be lighted to indicate the maximum number of channels selected per data block. See BLOCK SIZE key.

BUSY Light

This light normally comes on to indicate that the system is making a measurement, and further Keyboard entries are disabled. The STOP and RESTART keys, however, are still enabled. Certain commands that are the first of a series, e.g., editing commands, cause the BUSY light to come on in anticipation of further entries.

READY Light

This light is on whenever the system is not making a measurement and is ready to accept commands.

WHAT? Light

This light comes on to idnicate an illegal command entry from the Keyboard. See Appendix A for definitions.

SWITCHES

STEP-RUN Switch (set to RUN for normal operation)

In the STEP position, this switch causes the program to proceed one step at a time. When the step is completed, the appropriate display of results is given, and then the program stops. To make the program continue, the CONTINUE key must be pressed. The next step is then performed, results are displayed, and the program again stops. Thus, the STEP position is useful in debugging a program. In the RUN position, the switch causes the program to continue through all its steps automatically.

REPEAT/SINGLE Switch (set to SINGLE for normal operation)

The REPEAT/SINGLE switch on the Keyboard is used in conjunction with the DISPLAY switch on the ADC. With the REPEAT/SINGLE switch in the REPEAT position, whatever channel has been set on the DISPLAY switch will then be continuously displayed.

When the switch is set to the SINGLE position, only one sample record of the input is taken. Note that, in the REPEAT position, the BUSY light is on, meaning that no other commands can be executed. The switch *must* be in the SINGLE position to execute other commands. The starting time for each sweep when the switch is in the REPEAT position, is determined by the setting of the TRIGGERING controls. In the SINGLE position the second parameter of the ANALOG IN command (n2) determines the data block to be displayed.

In programming, if the switch is in the REPEAT position, then when the program reaches the analog-in step, it will pause and the analog input will be continuously displayed. As soon as the is in the SINGLE position when the program reaches that step, it will simply display the single input and then proceed.

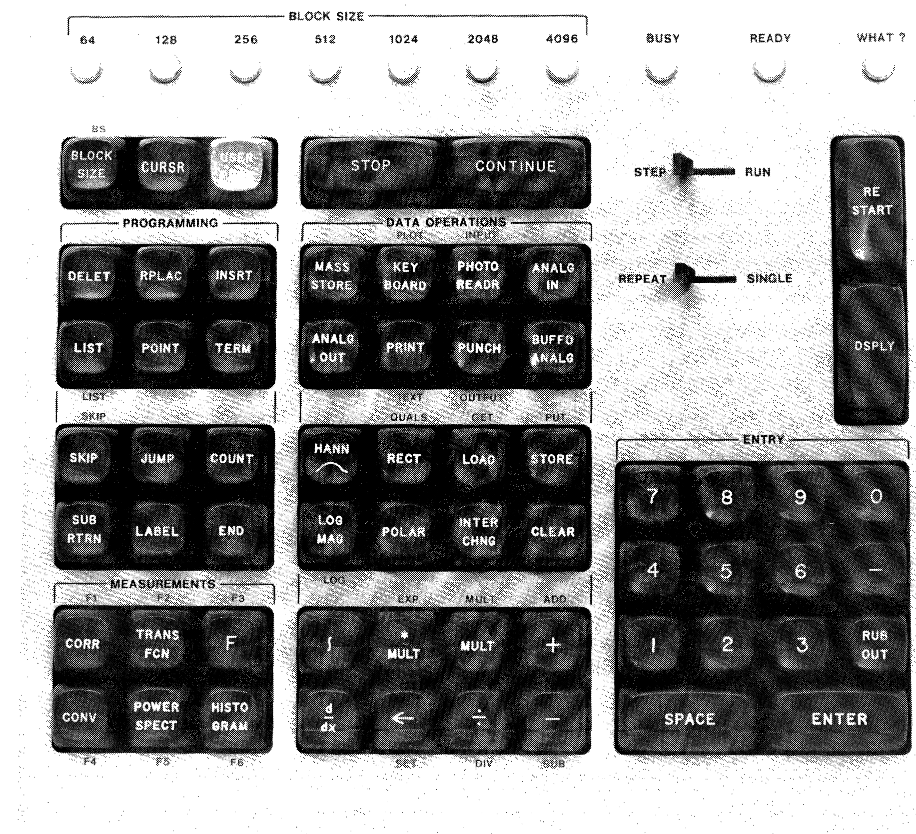


Figure 2-7. 5475A Keyboard Switches

Figure 2-7. 5475A Keyboard Switches

1. **SCALE FACTOR:**

for linear scales — the value is each vertical division on the scope. For example, a scale factor of 5 × 10 -005 means that each vertical division is 5×10^{-5} or 0.00005 volts.

The scale factor for a linear amplitude scale will always be correct if the GAIN control is in the CAL position. The Fourier Analyzer will automatically adjust the digital output to give the best possible resolution with no display overflow. When the scale factor is not correct during certain special situations (such as the analog-in REPEAT mode), the scale factor display will be blanked out.

for log scales — the value is the top line, in dB referenced to 1 volt, of the scope display; in other words, the offset.

domain and coordinate readout — on the right of the scale factor display, the domain of the data displayed (time or frequency) and the coordinates are given:

FREQ: means that the horizontal axis has units of frequency, with dc on the left edge (the left edge is channel zero), and maximum frequency (F-submax) on the right. Each point is then Δf apart, the second point from the left being 1Δf, the third 2Δf, etc. *When the FREQ light is out, it means that the horizontal axis has units of time.*

RECT: indicates that vertical axis is real (cosine) or imaginary (sine), depending on setting of the MODE switch.

POLAR: indicates that vertical axis is magnitude or phase, depending on setting of MODE switch.

LOG: indicates that vertical axis is logarithmic. The scale factor display in this case gives the value of the top line of the display in dB with reference to 1 volt. (For example, a scale factor display of -020 in LOG means that the top line of the display is -20 dB). Each vertical division is equal to 10 dB.

2. **SCALE:** This is a digital expander of the vertical scale. In the upright position, makes scale factor display automatic, so that Fourier Analyzer sets the scale factor for best resolution without overflow. When switch is turned clockwise, scale expands. When counterclockwise, scale compresses.

This switch has two principal uses: first, it expands the display vertically over a 500:1 (2.3 decade) range, for examination of small details near zero, when large values are also present. Second, it overrides the automatic scale factoring of the Fourier Analyzer when taken off the vertical position so that two different displays, whose scale would be otherwise automatically changed, can be photographed or plotted on the same scale. When display is logarithmic, each position off the vertical adds +4 dB to the display. Thus, the switch can be used to set up a non-fractional decade scale (i.e., a top line value of -16 dB can be switched to -20 dB, etc.). At the same time, the display is shifted up 4 dB to maintain calibration. This enables the user to look at the data more than 80 dB below maximum value.

The SCALE switch does not function during a phase display. It does affect the display on external plotters and scopes. Keep in mind that the external device plots or displays whatever is on the scope screen.

3. **POSITION:** adjust vertical position of display. Used as part of calibration procedures.

4. **GAIN:** expands display vertically. When in CAL position, scale factor will always be kept correct by Fourier Analyzer. This control allows the vertical scale to be expanded up to 2 times. It functions for all types of displays, and is a continuous expansion of the scope screen display only (i.e., does not change scale factors and has no effect on external scope or plotter). This control should be generally left in the CAL position so the display will always be calibrated in accordance with the scale factor display.

5. **MODE:**

COMPLEX: displays a complex, or Nyquist, plot of a spectrum, in which the horizontal axis is the real part, and the vertical axis the imaginary part of the spectrum. Each point then represents a frequency. The distance to each point from the origin is the magnitude, and the angle of the magnitude from the positive horizontal is the phase. Since this display requires an equal scale along each axis, the 8 × 10 cm scope display is automatically changed to an 8 × 8 cm display when the MODE switch is put in the COMPLEX position. The center is then 1 cm to the left of the center vertical axis on the scope screen.

REAL/MAGNITUDE: displays real part of spectrum if scale factor display reads RECT. Displays magnitude if scale factor display reads POLAR. Displays log magnitude if scale factor reads LOG. For time functions, this position will display the real part of the time series.

IMAGINARY/PHASE: displays imaginary part of spectrum if scale factor display reads FREQ and RECT. Displays phase if scale factor display reads POLAR. In this case, the vertical scale is determined by the setting of the POLAR ANG/DIV switch, with the numbers representing degrees per division. The numeric part of the scale factor display will be blanked out when the display reads POLAR. For time functions, this position will display the real part of the time series, since the imaginary part does not exist.

6. **POLAR ANG/DIV:** sets number of degrees (angle) per vertical division when scale factor display reads POLAR and MODE switch is in PHASE position.

7. **SWEEP LENGTH:**

10: sweep is completed in 10 horizontal divisions of scope screen. This will, however, produce a fractional number of channels per division, which the 10.24 and 12.8 positions avoid (e.g., with data block size of 512, number of channels per division = 512/10 = 51.2).

10.24: expands sweep to 10.24 horizontal divisions. Part of the sweep will now be off screen, but number of channels per division on screen can be made non-fractional in this position, or failing this, the 12.8 position (e.g., with data block size of 512, number of channels per division = 512/10.24 = 50).

12.8: expands sweep to 12.8 horizontal divisions. Part of sweep will now be off screen, but number of channels per division in this, or failing this, the 10.24 position, can be made non-fractional (e.g., with data block size of 512, number of channels per division = 512/12.8 = 40).

Note: sweep expansion has the same effect on an external plotter as it does on the display.

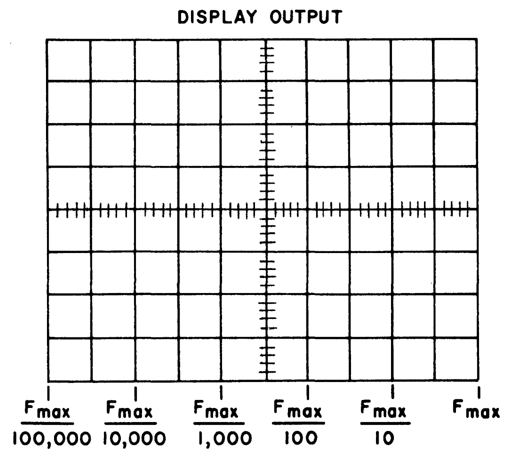
8. **ORIGIN:**

LEFT: puts horizontal origin on the left edge of the scope screen with channel 0 at the left edge, and channel N–1 at the right edge. Sweep moves from channel 0 to channel N–1.

CENTER: channel 0 is in center of screen, channel N/2 at the right edge, channel N/2+1 at left edge, channel N–1 is immediately left of channel 0. Sweep moves from channel 0 to the right edge, then in from the left edge to channel N–1. This mode of sweep is useful in certain cases where display of correlation or convolution is desired. This mode does not function in the frequency domain, where the origin is always on the left edge.

LOG: sets a logarithmic horizontal scale. The conversion for this scale is done digitally, so there are no conversion errors. The number of decades contained in a display is determined by the block size. For example, a block size of 128 has 64 frequency points and thus has less than 1 decade of log horizontal display. A block size of 512 has 256 frequency points and thus has 2-1/2 decades of log horizontal display.

The horizontal log scale is illustrated below.



9. **MARKER:**

OFF: no markers.

8 PT: gives a marker every 8 channels on the horizontal axis, counting origin as channel 0.

32 PT: gives an intensity marker every 32 channels on the horizontal axis, counting origin as channel 0. (See demonstration on back of this sheet.)

10. **PLOT RATE:**

EXT: allows "seek command" from Fourier Analyzer to tell external X-Y plotter to plot a point. Will not give next point until "completed plot" signal for previous point received from plotter.

"internal" (i.e., off EXT position): external X-Y plotter plots points at rate set by Fourier Analyzer. This control varies that rate from 5 points/sec (counterclockwise position) to 300 points/sec (clockwise position). Fourier Analyzer does not wait for "plot completed" signal from external plotter before sending next point.

11. **ARM-PLOT:**

ARM: outputs first channel to X and Y outputs of external X-Y plotter, and, if the SERVO signal is being used to control the plotter servo enable, it turns on the servo. If the plotter servo drive is manually controlled, it should be turned on when switch is set to ARM. Plotter moves to first point to be plotted.

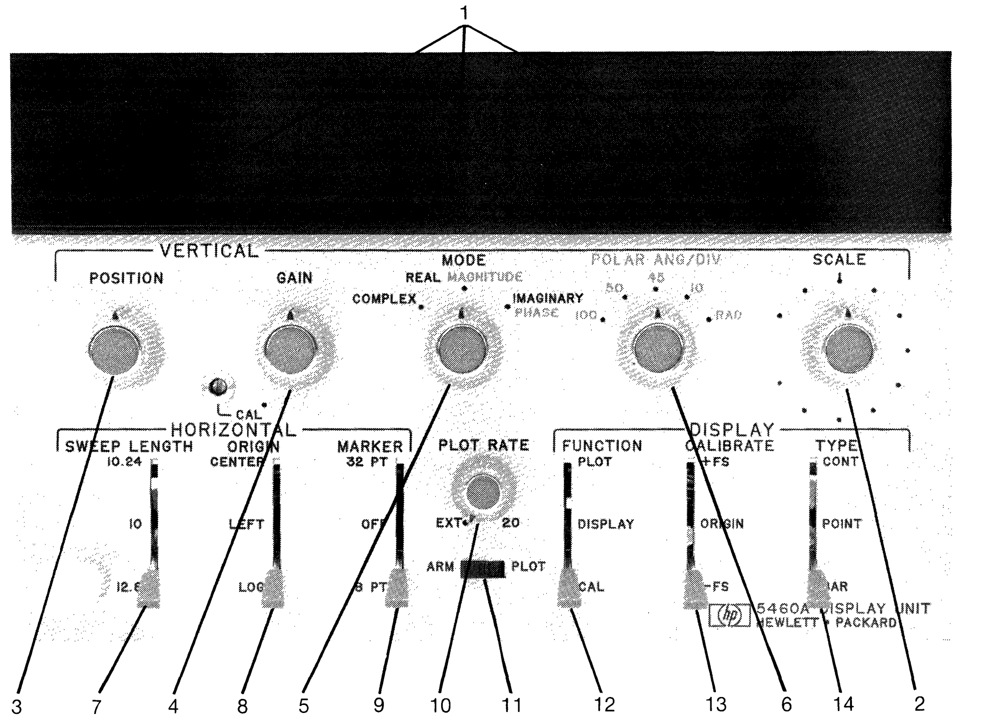
PLOT: outputs X and Y points to be plotted by external plotter. Note: ARM position *must* precede PLOT position.

12. **FUNCTION:**

PLOT: makes display output from Fourier Analyzer available to external X-Y plotter, via connectors at rear of Display Unit. Shuts off scope display. Requires output card purchased through 10640B.

DISPLAY: makes display output from Fourier Analyzer available at scope screen. Turns off servos in external X-Y plotter.

CAL: puts calibration dot at one of three positions on scope screen, as set by CALIBRATE switch.



13. **CALIBRATE:**

ORIGIN: allows calibration dot to be centered on horizontal axis, left edge of screen, using VERTICAL POSITION control on Display Unit control panel, and HORIZONTAL POSITION control on scope panel.

+FS: allows calibration dot to be centered at mid-top line of screen, using screwdriver adjustment on vertical GAIN CAL control.

–FS; allows calibration dot position to be checked (should be at mid-bottom line of screen) after ORIGIN and +FS adjustments. This position can be used for GAIN control screwdriver adjustment, and the +FS used for check, if desired.

No horizontal gain calibration is provided. The basic stability of the display will not allow the horizontal gain to drift an amount greater than the resolution of the oscilloscope. In addition, the horizontal markers eliminate the need for precise horizontal gain (to identify a frequency point, one simply has to count the dots and multiply by Δf).

14. **TYPE:**

CONT: display will be a continuous line. Greatest usefulness is in time domain and log displays.

POINT: display will be a series of points. Greatest usefulness is for linear displays in frequency domain.

BAR: display will be a series of vertical bars. Useful for linear displays in frequency domain.

(see over)

Figure 2-8. 5460A Display Unit

Example of Calibrating the Scope for Normal Use

To calibrate the scope for normal plus and minus readings (as opposed to full scale positive-only readings, covered in the next example), make the following settings:

FUNCTION to CAL
CALIBRATE to ORIGIN

Now use the VERTICAL POSITION control (on Display Unit panel) to situate the calibration dot exactly midway on the left vertical line (i.e., on the 0 point on the vertical axis). Then use HORIZONTAL POSITION control (on scope panel) to situate calibration dot exactly on vertical axis. Then set:

DISPLAY CALIBRATE to +FS

Use the screwdriver adjustment on the GAIN control CAL position to set the dot exactly on the top line of the screen. The dot should be on the center vertical axis. Then set:

DISPLAY CALIBRATE to –FS

The dot should be on the bottom line of the screen, on the center vertical axis.

Examples of Using Markers to Determine a Channel Number

It is desired to determine in what channel a particular point lies. First set following display conditions:

SWEEP LENGTH to 10
ORIGIN to LEFT
MARKER to OFF
FUNCTION to DISPLAY
TYPE to POINT
MODE to REAL MAGNITUDE
SCALE to straight up position

We will manually enter data into the 0 data block, using a data block size of 128. To do this, press the following keys on the Keyboard:

[BLOCK SIZE] [1] [2] [8] [ENTER]

[CLEAR] [ENTER]

[KEY BOARD] [0] [SPACE] [8] [ENTER]

[KEY BOARD] [–] [4] [SPACE] [0] [ENTER]

[3] [0] [0] [0] [0] [ENTER]

[TERM] [ENTER]

Note that this data — which would be a single point — is in the time domain (FREQ light is off), and has RECT coordinates.

To determine what channel the point is in, set the MARKER switch to 8 PT. Note that the data point is more intensified (as is every 8th point thereafter, across the screen). Therefore, we know that the data point is in channel 8.

It may be desired to move the data point a certain number of channels. Assume we want to move it 96 channels to the left. Press:

[←] [0] [SPACE] [9] [6] [ENTER]

Then set MARKER switch to 32 PT position. Now the data point is slightly beyond the first 32-point marker. We can count up from that marker, or switch to 8 PT, and see that the data point is in channel 40.

Figure 2-8. 5460A Display Unit

SECTION 3
KEY DESCRIPTIONS

INTRODUCTION

This section explains the use of each Keyboard key. The keys that have a gold function printed above or below are described briefly following the normal function of the key, with references to other sections that explain the gold key in detail.

The keys are divided into subsections as follows: *Page*

INPUT MODES

These paragraphs provide operating instructions for the following input modes; analog (ADC) and manual (Keyboard). In addition, a brief description of the keys that control an optional Photoreader and Punch are covered. Complete information on using these units is provided in Appendix E.

ANALOG INPUTS (ANALoG IN & BUFFereD ANALoG KEYS)

The analog input commands initiate the input of data from the Analog-to-Digital Converter (ADC). The analog input may be from one or both channels of the standard ADC, or from one to four channels of the optional 4-channel ADC. There are two types of analog input commands: ANALOG IN and BUFFERED ANALOG. Both commands can be used for 1-, 2-, 3-, or 4-channel input and are unique in that they permit you to take in data to one data block while simultaneously displaying another. This is primarily useful in programs where you may obtain an automatic display as the program proceeds. Before using an ANALOG IN command, be sure you understand the use of the INPUT SELECTOR and DISPLAY switches on the ADC, the REPEAT/SINGLE switch on the Keyboard, and the TRIGGERING controls, as explained in Section 2.

ANALG IN

Analog In

Before giving an ANALOG IN command for single channel operation, the INPUT SELECTOR switch on the ADC has to be in the channel A position, meaning channel A is the input channel. The command is as follows:

ANALG IN | n1 | SPACE | n2 | SPACE | n3 | SPACE | n4 | SPACE | n5 | ENTER

where the parameters may be used to specify the analog input operation as follows:

no parameters —

read data into block 0, no display during data input.

n1 given —

read data into block n1, no display during data input.

n1, n2 given —

read data into block n1, display block n2.

n1, n2, n3 given —

read n3 channels of multiplexed data into block n1, display block n2 (for use only with multiplexer).

n1, n2, n3, n4 given —

read data into sequential blocks starting at block n1, display block n2, channels n3 to n4 (partial block display). Number of sequential blocks determined by setting of INPUT SELECTOR switch.

n1 through n5 given —

read data into n5 sequential blocks starting at block n1, display block n2, channels n3 to n4, (partial block display). For use only with multiplexer.

For multiple-channel operation, the ANALOG IN command is identical to the single-channel operation, except that the INPUT SELECTOR switch must be in an appropriate position. Channel A will be read into block n1 and channel B into block n1+1.

For more than two-channel operation, the third and fourth channel analog inputs require an additional data block used as a buffer to speed up data handling in the Processor. A three-channel input therefore requires four data blocks, and a four-channel input requires five data blocks.

Data Block Requirements—Analog Input Command

DISPLAY/INPUT Switch (or INPUT SELECTOR)	**Data Blocks Used**			
	Ch. A	**Ch. B**	**Ch. C**	**Ch. D**
A	n1			
AB	n1	n1 + 1		
ABC (4-ch. ADC)	n1	n1 + 1	n1 + 2, & nL	
ABCD (4-ch. ADC)	n1	n1 + 1	n1 + 2	n1 + 3, & nL

where n1 is the data block specified in the Analog In command and nL is the last data block defined by:

$$nL = \frac{\text{available data memory}}{\text{block size}} - 1$$

BUFFD ANALG Buffered Analog

The BUFFERED ANALOG command uses two data blocks as buffer blocks for each data input channel, block n1 named in the command plus the last available data block. This permits the input of data from one to four channels of the ADC into buffer blocks while simultaneously doing an operation on another block. If the latter operation requires less time than the time to input a block of data, then no data will be lost in successive inputting of records and a real-time analysis will be performed.

The command form for single channel buffered analog is as follows (INPUT SELECTOR switch on ADC to A):

BUFFD ANALG [n1] SPACE [n2] SPACE [n3] SPACE [n4] SPACE [n5] ENTER

where the parameters may be used to specify the analog input operation as follows:
(Note: Display parameters are ignored if Overlap Processing is selected. Refer to Overlap Processing section.)

no parameters —

read data into block 0, no display during data input.

n1 given —

read data into block n1, no display during data input.

n1, n2 given —

read data into block n1, display block n2.

n1, n2, n3 given —

read n3 channels of multiplexed data into block n1, display block n2 (for use only with multiplexer).

n1, n2, n3, n4 given —

read data into sequential blocks starting at block n1, display block n2, channels n3 to n4 (partial block display). Number of sequential blocks determined by setting of INPUT SELECTOR switch.

n1 through n5 given —

read data into n5 sequential blocks starting at block n1, display block n2, channels n3 to n4, (partial block display). For use only with multiplexer.

A 2-, 3-, or 4-channel input operation can be performed with the BUFFERED ANALOG command by placing the INPUT SELECTOR switch to AB, ABC, or ABCD,

Data Block Requirements—Buffered Analog Command

INPUT SELECTOR	Data Blocks Used			
Switch	**Ch. A**	**Ch. B**	**Ch. C**	**Ch. D**
A	n1 & nL			
AB	n1 & nL	n1 + 1 & nL - 1		
ABC	n1 & nL	n1 + 1 & nL - 1	n1 + 2 & nL - 2	
ABCD	n1 & nL	n1 + 1 & nL - 1	n1 + 2 & nL - 2	n1 + 3 & nL - 3

where nL is the last data block $\left(= \frac{\text{available data memory}}{\text{block size}} - 1\right)$ and n1 is the data block specified in the command.

Sample Program

The primary application of the BUFFERED ANALOG command is in a Keyboard Program, where real time analysis is desired. The program must contain a COUNT command (described later in this section), as real time input will continue until runout of the loop counter. A listing of a sample program using buffering is shown as follows:

AUTO SPECTRUM SUMMATION AVERAGE

L	1		Label 1
CL	1		Clear Block 1
L	2		Label 2
RB	0	1	Buffered Analog Command (data to "0", display "1")
F	0		Fourier transform block 0
A+	1		Add block 0 to block 1
X>	1		Store block 0 into block 1
#	2	100	Repeat steps from label 2, 100 times
:	1	100	Divide block 1 by 100
.			End program

If the BUFFERED ANALOG command comes from the Keyboard there is only one input sweep. But if the command comes from the program stack, real time input continues until runout of the next COUNT command.

The only restriction on the buffering process is that several data blocks must be sacrificed to serve as blocks for real time input (one for single channel input and two for dual channel).

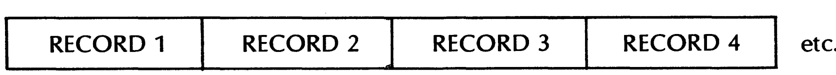
Real Time Display

A calibrated fast display can be included in a program, as part of an ANALOG IN or BUFFERED ANALOG command. When included as part of a command, the timing difference between the input rate and the display rate will cause an occasional small loss of data from the real-time input. If real-time input is essential, a single calibrated display sweep can be included in the loop by a separate command instead of using the display option of either of the analog input commands. The programs below provide an example of how this can be done.

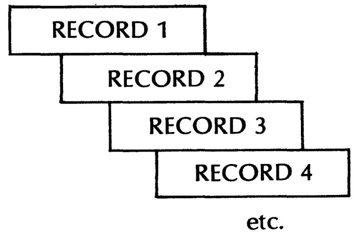
The data loss that may be caused by this program . . .	**. . . can be eliminated by using this program**	
L 0	L 0	
CL 1	CL 1	
L 1	L 1	
RB 0 1	RB	This (or similar) replacement allows throughput of a real-time spectrum to 2.5 kHz.
H1	D 1 4	
F	H1	
SP	F	
#1 100	SP	
	#1 100	

Overlap Processing

When the processing time for a particular operation is less than the time required to input a new analog record, it is possible to process all of the values sampled by the ADC. This is called "real time processing". Real time processing is made possible by use of the buffered input mode on the ADC. In this mode additional buffer input blocks are used in the following way. The first data record (T seconds long) is read into the input buffer and transferred into the true input block. The next record is then started into the buffer block. While the buffer block is being filled, the processing operation (i.e., power spectrum average or tri-power spectrum average) is performed on the data in the true input block. If a whole new record has not been recorded into the buffer block when the processing loop is ready for a new record, processing will stop until a whole new record can be transferred to the true input block. If a display term is used with the BUFFERED ANALOG (RB) command, at least one (more if there is time) display sweep will be executed while waiting for the buffer to completely fill. In non-overlap processing, using the RB command, data utilization is as shown in the diagram below.



In overlap processing, when the processing loop finishes before the buffer is filled, the last T seconds of data are read out of the buffer and transferred to the true input block. When overlap processing is used, data utilization as shown in the diagram below results.



When a display term is used with the RB command and overlap processing is enabled, the display term is ignored (no display given). A display may be obtained by using a separate DISPLAY command, as described in the paragraph on Real Time Display on the previous page. Whether display is used or not, the amount of overlap will be as much as is allowed by the processing loop time. Using the last T seconds of available data results in two benefits. When analyzing random data, a lower variance in the result is achieved in a given time. When analyzing changing phenomena with high resolution, more records per second may be observed.

To implement overlap processing, an RB command is used in any program stack. When the processing time is shorter than the input record length, the system will be operating in real time. To put the system into an overlap processing mode, set bit 15 of the Processor S-register to 1.

The program below is an example of a power spectrum average program that will go into the overlap mode when S-register bit 15 is set to 1.

```
LABEL  0  ENTER
CLEAR  1  ENTER
LABEL  1  ENTER
BUFFD ANALG  ENTER
DSPLY  1  SPACE  4  ENTER
F  ENTER
POWER SPECT  ENTER
COUNT  1  SPACE  1000  ENTER
END  ENTER
```

Using the program above with a blocksize of 1024, overlap processing will occur for maximum frequencies at or below approximately 2.5 kHz (or 5 kHz if the display step is deleted).

Overlap processing should not be enabled at sample rates above those where real-time processing is possible. At these high rates, the software will attempt to keep a continuous input to the buffer, and will reduce or prevent processing.

KEY BOARD

Key Board

The KEY BOARD command is used to manually enter data into the Fourier Analyzer from the Keyboard. The following paragraphs describe the methods available.

MANUALLY ENTERING DATA FROM THE KEYBOARD

This mode of input, and the Terminal output, are a direct data interface between man and machine because they use a decimal number system. The optional punched tape mode (Photoreader input and Punch output), on the other hand, is a data interface between the Fourier Analyzer and itself or other computers, and uses a binary number system (refer to Appendix E for additional information on option interfacing).

NOTE

> For greatest accuracy, the data block to be used should be cleared before data is entered via the Keyboard. If the block is not cleared before data is entered, the entered data will be adjusted (if required) to be expressed using the scale factor and calibrator values for the block; this may result in a reduction of the accuracy of the data.

Command to Manually Enter Data

The manual data input command is in four parts as explained in the following paragraphs. Having read these, refer thereafter to Figure 3-1 for a quick reference on how to use the KEY BOARD command.

1. **Enter block-fill or point-by-point fill command.** Block fill means that a single data value will be entered into the channel range specified, as for example, to form a rectangular pulse. Point-by-point fill means that a sequence of different values will be entered into the channel range, as in the case of a triangular waveform. Note that there are no defaults allowed in the block-fill command. (The block-fill command with n3 defaulted of course becomes the point-by-point fill.)

2. **Enter scale factors.** These are:

   n1 — the block multiplier exponent, i.e., n1 in the expression "10^{n1}" which multiplies every data word to be entered in step 3. Note: the data word to be entered should be as close to (but less than) the absolute value of 32767 as possible (n1 should be adjusted accordingly). Thus, for example, to enter the data word 1, it would be best to use an n1 of -4 (10^{-4}) and then use the data word 10000. If n1 had been 0 ($10^0 = 1$), and the data word punched in had been 1, the word would exist in memory as 00001, and the loss of the least significant digit would mean the loss of the data word. Keep in mind that the data word entered must be between -32,768 and +32,767. Thus, 1 could not be entered as 100,000 ($\times 10^{-5}$). And, 4 would have to be entered as 4000 ($\times 10^{-3}$) not as 40,000 ($\times 10^{-4}$). The range of n1 is -512 to +511.

   n2 — the coordinate code, from the table below, which tells the system the type of data entry you will be making. (For more information, see Appendix E, Data Representation.)

Coordinate Code	Time	Frequency	Rectangular	Polar	Log*	Linear	Single Precision	Double † Precision
0	x		x			x	x	
2	x		x		x		x	
4		x	x			x	x	
5		x		x		x	x	
7		x		x	x		x	
12		x	x			x		x
14		x	x		x			x

*Log data cannot be entered manually from the keyboard.
†Entries via keyboard must be between +32767 and -32768.

n3 — the frequency code. This can be any number from the table below. The term "frequency-code" comes from the fact that, when analog data is printed out, this position in the data format is reserved for a number which represents the SAMPLE MODE and MULTIPLIER switch settings on the ADC. Note that the system will accept any number from 0 through 16,383 for the parameter. However, if any number other than those listed in the table are used, the horizontal axis of the data will not be calibrated. The system uses this code to compute the horizontal axis scale (frequency). If n3 is defaulted, the machine assumes the previous value.

Freq. Code	F_{max} (MAX FREQ)	Δt (Δ TIME)	Freq. Code	Δf (Δ FREQ)	T (TOTAL TIME)
47	100 kHz	5 μsec	56	1000 Hz	1 msec
43	50 kHz	10 μsec	52	500 Hz	2 msec
39	25 kHz	20 μsec	63	200 Hz	5 msec
46	10 kHz	50 μsec	59	100 Hz	10 msec
42	5 kHz	100 μsec	55	50 Hz	20 msec
38	2.5 kHz	200 μsec	62	20 Hz	50 msec
45	1 kHz	500 μsec	58	10 Hz	100 msec
41,8	.5 kHz	1000 μsec	54	5 Hz	200 msec
37,4	.25 kHz	2000 μsec	61	2 Hz	500 msec
33,15	0.1 kHz	5000 μsec	57,24	1 Hz	1000 msec
			53,20	.5 Hz	2000 msec
			49,31	.2 Hz	5000 msec
11	50 Hz	10 msec	27	100 mHz	10 sec
7	25 Hz	20 msec	23	50 mHz	20 sec
14	10 Hz	50 msec	30	20 mHz	50 sec
10	5 Hz	100 msec	26	10 mHz	100 sec
6	2.5 Hz	200 msec	22	5 mHz	200 sec
13	1 Hz	500 msec	29	2 mHz	500 sec
9	0.5 Hz	1000 msec	25	1 mHz	1000 sec
5	0.25 Hz	2000 msec	21	0.5 mHz	2000 sec
1	0.1 Hz	5000 msec	17	0.2 mHz	5000 sec

Ø = general code (will plot general shape of curve)
99 = special BSFA data (see section 5 for additional information)

If data has already been entered in memory, and only the scale factors are being changed, the Keyboard input cycle can be terminated at this point by pressing TERM and ENTER. The Fourier Analyzer then goes from BUSY to READY.

3. **Enter data**

The data word is entered by simply pressing the required integer keys on the Keyboard, then ENTER. For time domain data, there will be one word, followed by ENTER, for each point. Example: 4000 ENTER. For frequency domain data there will be two words for each point, and these are entered as follows:

In the case of rectangular coordinates:

real data word [SPACE] imaginary data word [ENTER]

In the case of polar coordinates:

magnitude data word [SPACE] phase data word [ENTER]

NOTES

The imaginary data word or phase data word must not be defaulted when you are entering frequency domain data.

Zero must always be entered for the imaginary data word or phase data word for dc or the highest frequency (Fn) whenever either of these points is part of the data. The reason for this is explained later in this section under the discussion of the Fourier Transform used in the Fourier Analyzer.

4. **Press TERM ENTER**

This is to terminate the manual input cycle. It is not required for a block fill command. The system now goes from BUSY to READY, and is ready for further operation.

Important note: The first of the four Keyboard entry steps can be included in a program (i.e., determination of block or point-by-point fill). The program will halt when it reaches this step until the remainder of the steps have been entered. Then the program will automatically continue.

Figure 3-1. Example KEY BOARD Commands

1. **Enter Command For:** No defaults allowed, except as shown

Block Fill

[KEY BOARD] [BLOCK n1] [SPACE] [CHANNEL n2] [SPACE] [CHANNEL n3] [ENTER]

where:

n1 is the data block.
n2 is the first channel.
n3 is the last channel.

Point-by-Point Fill

or [KEY BOARD] [BLOCK n1] [SPACE] [CHANNEL n2] [ENTER]

where:

n1 is the data block (default value = block 0).
n2 is the first channel (default value = channel 0).

2. **Enter Scale Factors:**

[KEY BOARD] [EXP n1] [SPACE] [COORD CODE n2] [SPACE] [FREQ CODE n3] [ENTER]

where:

n1 is the block multiplier exponent (default value = current value for the selected data block).

n2 is the coordinate code (default value = current value for the data block).

n3 is the frequency code (default value = current value for the selected data block).

The "EXP" term ("n1") should not be defaulted in this entry. Default causes the Fourier Analyzer to assume that the data values you enter will already be expressed in terms of the current scale factor and block calibrator for the selected block, rather than as decimal numbers to be converted.

3. **Enter Data:**

[REAL WORD] [SPACE] [IMAG. WORD] [ENTER] or [MAG. WORD] [SPACE] [PHASE WORD] [ENTER]

(IN .01 DEGREE)

4. **Press:**

[TERM] [ENTER]

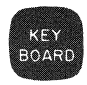
(if point-by-point fill)

ALTERNATE INPUTS

The KEY BOARD command is intended to enable entry from the Terminal keyboard, however, an optional Photoreader can also be used as the input device. This requires that a manual data tape be prepared (on a teleprinter with punched output or other punch device) in accordance with Appendix E, and that bit 3 of the Processor be on before entering step 1 of the manual data entry command. The PHOTO READER and REPLACE/INSERT commands are also affected by the state of bit 3. The following table shows the effect of bit 3 on the input devices for these commands. Refer to Appendix E for additional information.

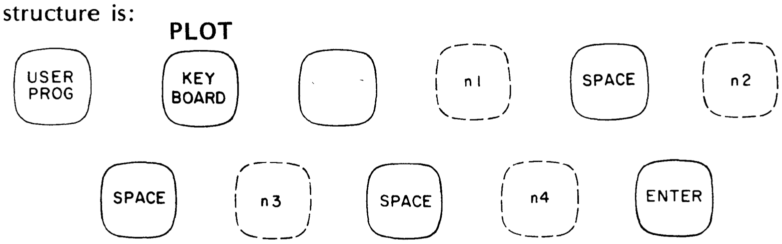
Command	Bit 3 = 0	Bit 3 = 1
KEY BOARD	Terminal	Photoreader
PHOTO READER	Photoreader	Terminal
RPLAC or INSRT	Terminal	Photoreader

PLOT



Plot

The gold PLOT command initiates "automatic" plotting, that is, calls up a User Program that incorporates the most-recently-used plot parameters such as plot size, location, annotation, etc. The command structure is:



where:

n1 = device type: (6 for 2648 Terminals, 10 or 35 for optional Plotters)

= 6: default

= -1: execute programs with current parameters except 5815, 5816, and 5817 retain default values.

n2 = block number to plot
default = block on display

n3 = starting channel of plot
default = first channel in block

n4 = ending channel of plot
default = last channel in block

For additional information, refer to Section 6, 5800 Automatic Plotting.

PUNCHED TAPE INPUT/OUTPUT

A Photoreader and High Speed Punch, if present in a system, make an effective input/output pair as data can be punched out and re-entered rapidly and accurately. Data that is punched out for hard copy storage can be re-entered at any time by using the Photoreader input command as described in the following paragraphs. For complete information, see Appendix E.

PHOTO READR

Photoreader

To make the tape run through an optional Photoreader and enter data into the Fourier Analyzer, use the following command:

[PHOTO READR] [BLOCK n1] [SPACE] [CHANNEL n2] [SPACE] [CHANNEL n3] [ENTER]

where:

n1 is the data block into which the data is to be entered.

n2 is the first channel in that data block to receive data.

n3 is the last channel to receive the data.

The table below gives default values for this command:

Element	Meaning of Element	Default Value of Element
n1	data block into which data is to be entered	data block 0
n2*	first channel to receive data	channel 0
n3*	last channel to receive data	successive data words on the tape are entered channel-by-channel beginning with n2, i.e., a point-by-point command results.
*NOTE: if n2 and n3 are both given, then the first data word only on the tape is entered in all the channels, n2-n3, i.e., a block fill results.		

For additional information regarding punched tape input or output, refer to Appendix E.

INPUT
PHOTO READR

Input (Variable Parameter)

The gold INPUT command inputs Variable Parameter values from the Keyboard or Terminal. The command structure is:

[USER PROG] [INPUT / PHOTO READR] [ ] [n1] [SPACE] [n2] [ENTER]

where:

Variable Parameters n1 through n2 are set to the values typed on the Terminal or Keyboard.

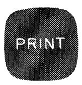
For additional information refer to the Variable Parameter paragraphs (specifically Y 1808) in this section.

OUTPUT MODES

These paragraphs provide operating instructions for system output modes such as the scope and Terminal displays and printouts, and optional output to a printer, punch, or plotter.

DATA OUTPUT VIA TERMINAL

The Terminal may be used to obtain a printout of the data in any block, or in any part of a data block. The data is printed in the decimal number system. If the system contains a High Speed Punch, this data can be punched on paper tape as well. For additional information on the High Speed Punch output, refer to the PUNCH command in this section and in Appendix E.



Print

NOTE

If bit 6 of the Processor register is set, the data is output to an optional Printer.

The form of the command for displaying data is:

PRINT BLOCK n1 SPACE CHANNEL n2 SPACE CHANNEL n3 ENTER

where:

n1 is the data block to be printed out. (See table below for default values.)

n2 is the first channel to be printed out.

n3 is the final channel to be printed out.

Default Values for PRINT Command

Element	Meaning of Element	Default Value of Element
n1	data block to be printed out	data block 0
n2*	starting channel of printout	whole data block is printed out
n3*	end channel of printout	n2
*If n2 and n3 are defaulted whole data block is printed.		

Format of Printed Data

Each printout begins with a line giving all the calibration factors associated with the data. The first item in the line is SF (meaning scale factors) and the form of the line is shown below:

SF n1 n2 n3

where:

n1 is the amplitude scale factor for linear data (each word is multiplied by 10^{n1}); or the value in dB to be added to each data word, for logarithmic data.

n2 is the coordinate code from the table below. (See also Appendix E, Data Representation.)

Coordinate Code	Time	Frequency	Rectangular	Polar	Log*	Linear	Single Precision	Double † Precision
0	x		x			x	x	
2	x		x		x		x	
4		x	x			x	x	
5		x		x		x	x	
7		x		x	x		x	
12		x	x			x		x
14		x	x		x			x

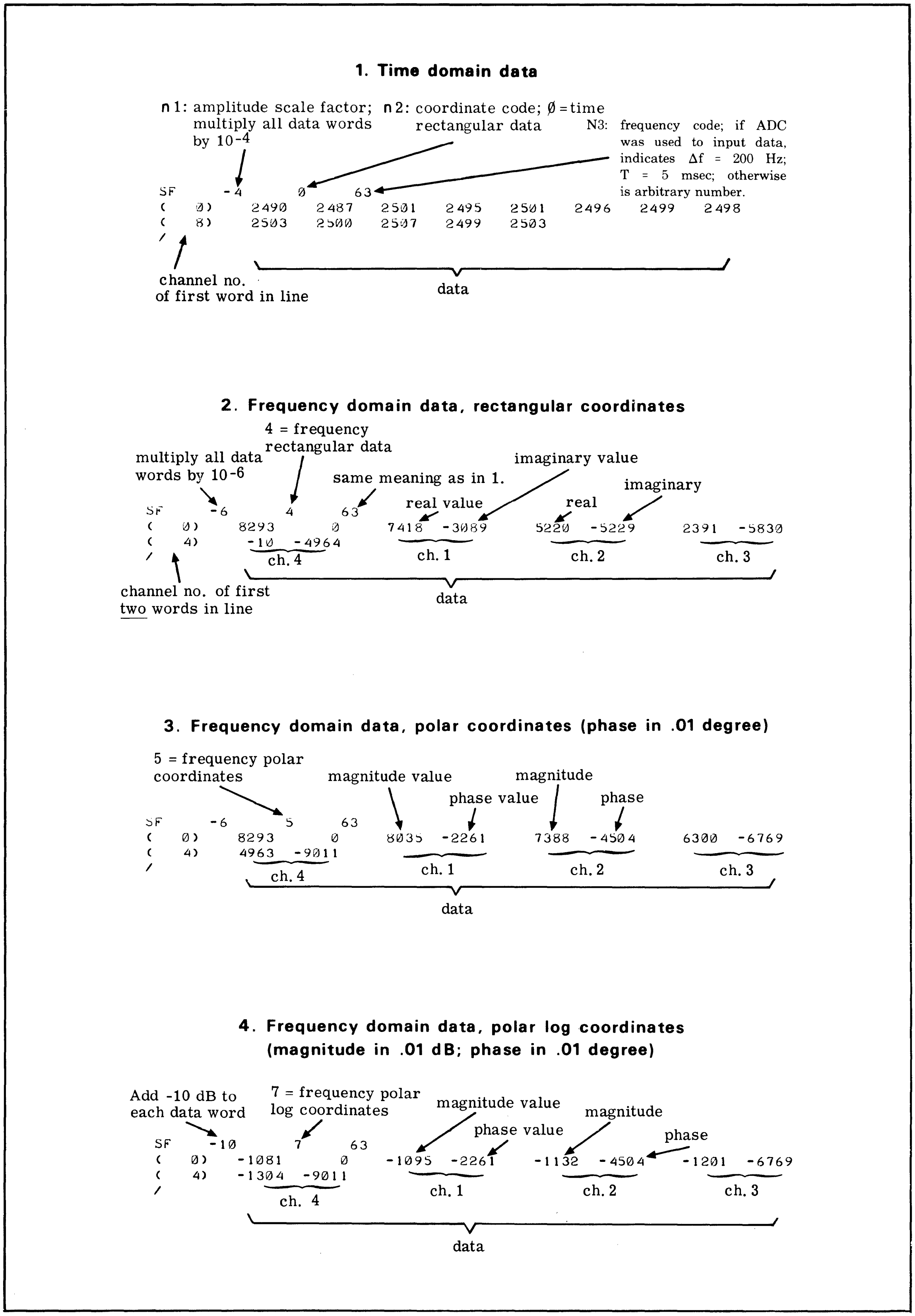
*Log data cannot be entered manually from the keyboard.
†Entries via keyboard must be between +32767 and -32768.

n3 is the frequency code from the following table.

Freq. Code	F_{max} (MAX FREQ)	Δt (Δ TIME)		Freq. Code	Δf (Δ FREQ)	T (TOTAL TIME)
47	100 kHz	5 μsec		56	1000 Hz	1 msec
43	50 kHz	10 μsec		52	500 Hz	2 msec
39	25 kHz	20 μsec		63	200 Hz	5 msec
46	10 kHz	50 μsec		59	100 Hz	10 msec
42	5 kHz	100 μsec		55	50 Hz	20 msec
38	2.5 kHz	200 μsec		62	20 Hz	50 msec
45	1 kHz	500 μsec		58	10 Hz	100 msec
41,8	.5 kHz	1000 μsec		54	5 Hz	200 msec
37,4	.25 kHz	2000 μsec		61	2 Hz	500 msec
33,15	0.1 kHz	5000 μsec		57,24	1 Hz	1000 msec
				53,20	.5 Hz	2000 msec
				49,31	.2 Hz	5000 msec
11	50 Hz	10 msec		27	100 mHz	10 sec
7	25 Hz	20 msec		23	50 mHz	20 sec
14	10 Hz	50 msec		30	20 mHz	50 sec
10	5 Hz	100 msec		26	10 mHz	100 sec
6	2.5 Hz	200 msec		22	5 mHz	200 sec
13	1 Hz	500 msec		29	2 mHz	500 sec
9	0.5 Hz	1000 msec		25	1 mHz	1000 sec
5	0.25 Hz	2000 msec		21	0.5 mHz	2000 sec
1	0.1 Hz	5000 msec		17	0.2 mHz	5000 sec
Ø = general code (will plot general shape of curve) 99 = special BSFA data (see section 5 for additional information)						

Examples of typical displays are given in Figure 3-2.

Figure 3-2. Examples of Printouts



At the start of each printout line, the channel number of the first word in the line is given in parentheses. If the printout is of time data, eight channels of data per line are printed. If the printout is of frequency data, only four channels per line are printed. For each frequency channel, first the real or magnitude word is printed (depending on coordinates), then the imaginary or phase word. If the printout is of double precision data, first the most significant bits of the data channel are printed, then 0 is printed for the least significant bits. The channel number at the start of each line will change by eight for each line of time data, and four for each line of frequency data. Log printouts are in hundredths of a dB. The scale factor (n1) must be added to each dB value. Phase is printed in hundredths of a degree.

To stop a printout while it is in process, simply press the STOP key on the Keyboard. The Terminal will finish printing its current line and stop.

PRINT Text
TEXT

The gold TEXT command writes a line or lines of text on the Terminal or on the optional Plotter. The command structure is:

USER PROG | PRINT (TEXT) | n1 | SPACE | n2 | ENTER

where:

n1 is the message ID number (must be core-resident).

If n1 is zero or defaulted, the same message as the previous call is written.

If n2 is defaulted, the text is written on the plot device.

If n2 is 1, the text is written on the Terminal.

For additional information on the use of the TEXT command, refer to User Program 5819 in Section 6.

PUNCHED TAPE OUTPUT

Paper tapes containing data from system data blocks can be punched with an optional High Speed Punch. Also, the Punch output may be printed on the Terminal (refer to Appendix E for details), which is useful when it is desired to view data block information in its original form, i.e., without the block calibrator having been applied to the data (as it is in the PRINT command format).

PUNCH Punch

The command to punch a paper tape is given below. Before executing this command, place power switch to ON.

The command form is:

PUNCH | BLOCK n1 | SPACE | CHANNEL n2 | SPACE | CHANNEL n3 | ENTER

where:

n1 is the data block to be punched out.

n2 is the first channei to be punched out.

n3 is the last channel to be punched out.

The following table has default values.

Element	Meaning of Element	Default Value of Element
n1	data block to be punched out	data block 0
n2*	first channel to be punched out	whole data block is punched out
n3*	last channel to be punched out	n2
*If n2 and n3 are defaulted, entire data block will be punched out.		

For additional information on the use of the PUNCH command, refer to Appendix E.

PUNCH
OUTPUT

Output (Variable Parameter)

The gold OUTPUT command prints the values of Variable Parameters n1 through n2. The command structure is:

USER PROG | PUNCH (OUTPUT) | [ ] | n1 | SPACE | n2 | ENTER

where:

n1 is the starting Variable Parameter.

n2 is the ending Variable Parameter. The default value of n2 is n1.

For additional information, refer to the Variable Parameter paragraphs (specifica!ly Y 1809) in this section.

SCOPE DISPLAY OUTPUT

The Display Plug-in and the Oscilloscope form the Display Unit of the Fourier Analyzer. The purpose of the Display Plug-in is to convert digital data in the Processor memory to analog signals that may be displayed on the 8 × 10 cm screen of the scope, plotted on the Graphics Terminal, or plotted on a standard X-Y plotter. Control of system operations connected with the display — such as digital expansion and log horizontal scale — are provided on the Display Plug-in.

It is important to note that any operation called for by the Display Unit does not change the form or coordinates of the data in the memory, but only modifies the data before it is transmitted to the Oscilloscope. Thus, display operations do not cause computational errors to build up. Only operations initiated by the Keyboard cause changes in data form in memory.

DSPLY

Display

There are three types of display commands in the Fourier Analyzer: block display, a repeat calibrated display in which the entire block is displayed; partial block display, a repeat calibrated display in which only a given number of channels are displayed across the entire face of the scope; and a coded display, where combinations of single/repeat, calibrated/uncalibrated, and full/partial block displays are possible.

If block n1 is in double precision, the least significant bits will not be shown, i.e., the imaginary part of block n1 will appear equal to zero.

The command for block display is:

DSPLY BLOCK n1 ENTER

where:

n1 is the data block. Default value = block 0.

The command for partial block display is:

DSPLY n1 SPACE n2 SPACE n3 ENTER

where:

n1 is the data block. No default allowed.

n2 is the first channel to be displayed. No default allowed.

n3 is the last channel to be displayed. No default allowed.

The commands for coded display are:

DSPLY n1 SPACE n2 ENTER

where:

n1 is the data block. No default allowed.

n2 is display code. Default value = 0 (calibrated repeat, full block display).

or,

DSPLY n1 SPACE n2 SPACE n3 SPACE n4 ENTER

where:

n1 is the data block. No default allowed.

n2 is the starting channel of a partial block display. No default allowed.

n3 is the ending channel of a partial block display. No default allowed.

n4 is display code. Default is 0 (calibrated repeat, partial block display).

Display Code (n2 Value)	Meaning of Code
0	Calibrated repeat sweep display
4	Calibrated single sweep display
8	Uncalibrated repeat sweep display
12	Uncalibrated single sweep display

The Fourier Analyzer operates on an automatic display concept, that is, the result of an operation is displayed. There is one exception: the ANALOG IN command, where the block displayed depends on the block named in n2 of the command, or in the REPEAT mode on the setting of the DISPLAY/INPUT switch of the ADC.

If a repeat sweep display command is executed from the program memory, the program will display and stop until ordered to continue by pressing the CONTINUE key or by another command. However, if a single sweep display command is used the display will take one sweep of the block specified and then continue on with the program.

STORAGE DEVICE OUTPUT

MASS STORE MASS STORE

The MASS STORE command is used to control digital mass storage devices such as Disc or Mag Tape units. The command form is:

MASS STORE (CODE) SPACE (PARAMETERS) ENTER

where (code) represents a 2-digit number from the Mass Store Command Matrix indicating operation and file.

NOTE

Full information on the terms used in the code portion of the command, plus operating procedures are provided in Section 4.

FILE → / ↓ OP	1 DATA BLOCK	2 ADC THROUGHPUT	3 PROGRAM STACK	4 ASCII TEXT	5 INDEX BLOCK	6 SYSTEM CORELOAD	7 COMMON	8 OVERLAYS
1 READ NEXT RECORD	11 Block n1	12 Channel n1 Sample n2	13	14	15	16	17	18 (See Note 1)
2 WRITE NEXT RECORD	21 Block n1	22 n1 Channels n2 Records	23	24	25		27	
3 POSITION TO RECORD (OR OVERLAY)	31 Record n1 n2 if n1 Relative	32 Record n1 n1 if n1 Relative	33 Record n1 n2 if n1 Relative	34 Record n1 n2 if n1 Relative	35 Record n1 n2 if n1 Relative	36 Record n1 n2 if n1 Relative	37	38 (See Note 2)
5 SEARCH FOR KEY	51	52	53	54				58 (See Note 3)
6 STORE KEY	61	62	63					
7 TRANS CRIBE ADC		72						

Note 1:
n1 Defaulted: Force Load Overlay
n1 Given: Load Overlay If It's Not Already in Core

Note 2:
Overlay n1
n2 If n1 Relative

Note 3:
List User Programs in System and/or Overlays

ANALG OUT **Analog Output**

The ANALG OUT command controls the optional Digital-to-Analog Converter (DAC).

The following command format activates the DAC output from the system Keyboard:

ANALG OUT n1 SPACE n2 SPACE n3 ENTER

where:

n1 specifies the data block used by the ANALOG OUT command. It cannot be the same data block used by the ANALOG IN command.

n2 determines the nature of the DAC command. If n2 = 1, the time-domain data is read out once (after the ADC triggering conditions are met). If n2 = 0, or defaulted, a continuous analog output is selected and the analog signal is read out repeatedly until an ANALOG OUT stop command or a RESTART is executed. If n2 ≥ 2 then the data block n1 is filled with random data with a peak value equal to n2.

n3 sets the DAC filters to one of nine cutoff frequencies. If n3 = 0, the DAC filter is bypassed, that is, set to the throughput state. If n3 is defaulted, the DAC software automatically sets the filter cutoff to the optimum filter range. ($.38\ F_{max} \leq F_c \leq .8\ F_{max}$ where F_c = the cutoff frequency of the filter.)

The following command turns off continuous DAC output (analog stop command):

ANALG OUT ENTER

The analog stop command establishes a zero output value in the DAC and stops the output. When the DAC is started, the ON and RUN lights turn on. When the analog stop command is entered, the ON and RUN lights turn off and the amplifier output is set to ground. The filter is set to the throughput state (filters are bypassed) and the DAC is cleared and ready for the next command.

For additional information on the ANALG OUT key, refer to Appendix E.

NOTE

The 54420A DAC cannot operate simultaneously with the Cursor, 54470A OR DURING ADC THROUGHPUT.

PROCESSING OPERATIONS

These paragraphs describe Keyboard functions concerned with the processing of data. The functions are broken down according to the three processing groups on the Keyboard: DATA OPERATION keys are used to transfer data between data blocks or to change coordinates of a given data block. They do not change the domain of the data. These keys also perform basic arithmetic operations on data blocks. MEASUREMENTS keys involve complex operations, most of which change the basic units of the data. Fourier Transform (F), for example, changes the data domain from time to frequency, or vice versa. PROGRAMMING keys are used to add your own Keyboard Programs (or subroutines) to the program memory.

POLAR Polar Coordinates

This command converts data in a frequency domain data block from rectangular coordinate series to linear polar coordinates. If the data block is in time domain or in double precision, the command is not executed. The command form is:

POLAR BLOCK n1 ENTER

where:

n1 is the data block. Default value = data block 0.

In the magnitude display, the vertical axis is amplitude and the horizontal axis is frequency. In the phase display, the vertical axis is degrees, and the horizontal axis is again frequency. Switching between the two displays is accomplished by the MODE switch on the Display Unit. Magnitude is always a positive value. The phase display runs 0 to 180° on the top half of the scope screen, and 0 to –180° on the bottom half. Thus, a phase of 190° would be shown as –170°.

The magnitude scale can be expanded using the SCALE switch on the Display Unit. The phase scale is expanded by using the POLAR ANG/DIV switch. The SCALE switch has no effect on the phase display; note also that the scale factor display is dark when phase is being shown, since the POLAR ANG/DIV switch defines the scale factor.

RECT Rectangular Coordinates

This command converts data in a selected data block from polar, log or log polar coordinates to linear rectangular coordinates real (cosine) series and imaginary (sine) series. The command structure is:

RECT BLOCK n1 ENTER

where:

n1 is the data block. Default value = data block 0.

In rectangular coordinates, the vertical axis on the scope is amplitude and the horizontal is frequency. By using the MODE switch on the Display Unit, one can display either the real (cosine) components or the imaginary (sine) components of a spectrum.

QUALS

RECT

Qualifiers (Variable Parameter)

The gold QUALS command gets or puts data block qualifiers into five Variable Parameters starting at n1. The command structure is:

QUALS

[USER PROG] [RECT] [ ] [n1] [SPACE] [n2] [SPACE] [n3]

where:

n1 is the starting Variable Parameter.

n2 is the block to which the qualifiers position.

n3 determines whether the qualifiers are put into the Variable Parameters or read from the Variable Parameters. (IF n3 = Ø, THEN GET THE QUALIFIERS FROM BLOCK n2 and PUT INTO THE VARIABLE PARAMETER BEGINNING AT n1. IF n3 ≠ Ø THEN PUT QUALIFIERS BEGINNING at n1 in DATA BLOCK n2.

For additional information, refer to the Variable Parameter paragraphs (specifically Y1823) in this section.

LOG MAG

Log Magnitude

This command takes the log magnitude of the data block, point by point, and stores result back into the same data block. The command structure is:

[LOG MAG] [BLOCK n1] [ENTER]

where:

n1 is the data block. Default value = block 0.

If the data block is in double precision, the result will be in double precision. If the data block is in frequency domain, but not in double precision, it will first be converted into polar coordinates (magnitude and phase), and then the log of the magnitude will be taken.

LOG SCALE FACTOR DISPLAY

The scale factor display specifies the offset, i.e., the value of the top line of the scope display in dB, referenced to 1 volt (peak). The standard scope calibration is 10 dB/division. The SCALE switch on the Display Unit permits you to adjust the offset to decade values such as 0, 20, 40, etc. For each position turned to the right, –4 dB is added to the offset.

LOG DISPLAYS

In a logarithmic printout on the Terminal (accomplished by the PRINT command), the dB values are printed in hundredths. The scale factor n1 at the start of the print, i.e.,

SF n1 n2 n3

is the number of db offset which must be added to all dB values in the data. This offset will be some multiple of 10, regardless of what the scale factor display was before the PRINT command. In other words, the system automatically makes sure that the offset is converted to a multiple-of-ten offset (unless the gold LOG MAG function is used to change the calibration). The parameters n2 and n3 are the coordinate code and frequency code; see PRINT key.

COORDINATE CODE 12 USAGE

The software assumes that any data having coordinate code "12" (rectangular, linear, double-precision) was obtained by performing a conjugate multiply operation on data that had coordinate code "4" (rectangular, linear, single-precision), and assumes that this data is calibrated in terms of "V^2" (instead of "V") — regardless of how this coordinate code "12" data was obtained. This can cause unexpected results when data with coordinate code "12" is entered manually and converted to log magnitude.

An example comparing conversion of manually-entered single-precision data and manually-entered double-precision to log magnitude is presented below.

If the data values 1,10,100,1000, and 10000 are entered with coordinate code "4", and a log magnitude operation is then performed, the coordinate code will be changed to "7" (polar, log, single-precision), and the vertical spacing between the dots (in a MAGNITUDE display) will be 2 cm or 20 dB/div.

If the same data values are entered with coordinate code "12", and a log magnitude operation is then performed, the coordinate code will be changed to "14" (rectangular, log, double-precision), and the vertical spacing between the dots (in a REAL display) will be 1 cm or 10 dB/div.

In reality, these two data displays represent the same difference (linearly) between points, since the coordinate code "7" vertical calibration is:

$$dB_7 = 20 \log \frac{V_n}{V_{ref}}$$

where V_n is the original data value (in volts)
V_{ref} is 1 volt peak

and the coordinate code "14" vertical calibration is:

$$dB_{14} = 10 \log \frac{(V_n)^2}{(V_{ref})^2}$$

where these terms have the same meaning as given above

Consider the examples below:

Using the data values "1" and "100", entered with coordinate code "4", LOG MAG gives the following deflections (and coordinate code "7"):

$$\text{for "1": } \frac{(20 \log \frac{1}{1})}{10} = 0 \text{ cm}$$

$$\text{for "100": } \frac{(20 \log \frac{100}{1})}{10} = 4 \text{ cm}$$

Using the data values "1" and "100", entered with coordinate code "4", conjugate multiplication gives the values "1" and "10000" (and coordinate code "12"); a LOG MAG operation on this data gives the following deflections (and coordinate code "14"):

$$\text{for "1": } \frac{(10 \log \frac{1}{1})}{10} = 0 \text{ cm}$$

$$\text{for "}100^2\text{": } \frac{(10 \log \frac{10000}{1})}{10} = 4 \text{ cm}$$

Using the data values "1" and "100", entered with coordinate code "12", LOG MAG gives the following deflections (and coordinate code "14"):

$$\text{for "1":} \quad \frac{10\,(\log \frac{1}{1})}{10} = 0 \text{ cm}$$

$$\text{for "100":} \quad \frac{10\,(\log \frac{100}{1})}{10} = 2 \text{ cm}$$

NOTE

The vertical scale calibration of LOG MAG display is 10 dB/cm. Refer to the "gold-key" capability of LOG MAG for information on how to change the vertical scale calibration.

SUMMARY OF COORDINATE TRANSFORMATION COMMANDS

Coordinates of Data Block After Execution of Command

Domain of Data	[LOG MAG]	[POLAR]	[RECT]
Time	Log	Not Executed	Linear
Freq	Log Polar	Linear Polar	Linear Rectangular

[LOG MAG] Log

LOG

The gold LOG command changes the vertical scale calibration from 10 dB/cm up to 80 dB/cm in steps of 10. The command structure is:

[USER PROG] [LOG MAG] [ ] [n1] [ENTER]
LOG

where:

n1 is the number of dB to be displayed full scale (10 to 80 in steps of 10). Default = 80 dB.

For additional information, refer to Section 7, paragraph Y 3022.

CLEAR Clear

There are two types of CLEAR functions: block and partial block. The command to clear an entire data block changes domain of data block to time linear rectangular coordinates and sets the frequency code to "0". The command form is:

CLEAR BLOCK n1 ENTER

where:

n1 is the data block. Default value = block 0.

The command to partially clear a block, i.e., clear a given range of channels, does not change the coordinate code or frequency code. The format is as follows:

CLEAR n1 SPACE n2 SPACE n3 ENTER

where:

n1 is the data block.

n2 is the first channel to be cleared.

n3 is the last channel to be cleared.

Following is table of default values:

Element	Meaning of Element	Default Value of Element
n1	Data block to be cleared	Block 0
n2	First channel to be cleared	Channel 0
n3*	Last channel to be cleared	Channel n2
*if n3 is defaulted, then channel n2 will be the only channel cleared.		

LOAD Load

This function transfers data from block n1 to block 0. Any previous data in block 0 is written over. Block 0 is displayed. Data in block n1 is unaffected. The command structure is:

LOAD n1 ENTER

where:

n1 is the data block. Default value is 0.

NOTE

See STORE command to transfer data *out of* block 0.

GET LOAD

Get (Variable parameter)

The gold GET command loads data block information. The command structure is:

USER PROG | GET LOAD | | n1 | SPACE | n2 | SPACE | n3 | ENTER

where:

n1 is the Variable Parameter that is set to the value of block n2, channel n3.

n2 is the block number.

n3 is the channel number.

For additional information, refer to the Variable Parameters paragraphs (specifically Y 1821) in this section.

STORE

Store

This function transfers data from block 0 to block n1. Block n1 is displayed and the data in block 0 is left untouched. Any previous data in block n1 is written over. The command structure is:

STORE | BLOCK n1 | ENTER

where:

n1 is the data block. Default value is 0.

PUT STORE

Put (Variable Parameter)

The gold PUT command stores data block information. The command structure is:

USER PROG | PUT STORE | | n1 | SPACE | n2 | SPACE | n3 | ENTER

where:

n1 is the Variable Parameter that is placed into block n2, channel n3.

n2 is the block number.

n3 is the channel number.

For additional information refer to the Variable Parameter paragraphs (specifically Y 1822) in this section.

INTER CHNG Interchange

This function interchanges data in blocks 0 and n1. Block 0 is displayed. No data in any block is lost as a result of this operation. The command is:

INTER CHNG   BLOCK n1   ENTER

where:

n1 is the data block. Default value is 0.

In order to interchange two data blocks neither of which is block 0, and do so without loss of data in any other block, a procedure such as that shown below must be used. Three different sets of data, A, B, and C reside initially in the three blocks 0, 1, and 2. The commands shown will interchange B and C without loss of set A.

Initial conditions:

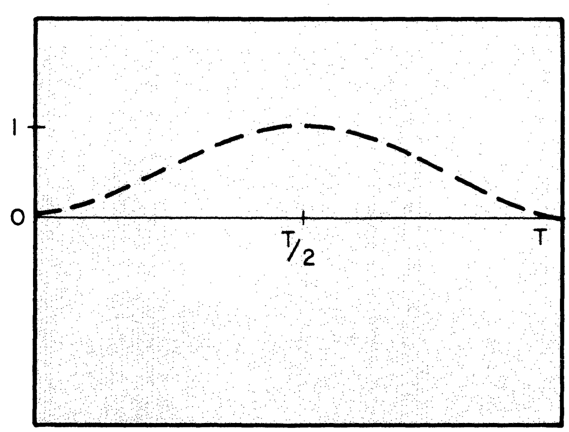
Data Block	0	1	2
Set of Data in Block	A	B	C

The following commands interchange blocks 1 and 2 without loss of block 0:

Data block	0	1	2
The command INTERCHNG 1 ENTER yields . . .	B	A	C
The command INTERCHNG 2 ENTER yields . . .	C	A	B
The command INTERCHNG 1 ENTER yields . . .	A	C	B

HANN Interval-centered Hanning

This command multiplies a data block by the window function, $1/2 - 1/2\cos(2\pi t)/T$. The data block must be in the time domain for the command to be executed.



To use this command to suppress leakage lobes in the frequency domain, Hann data block n1 and then execute a Fourier transform.

The command for Hann is:

[HANN] [BLOCK n1] [ENTER]

where:

n1 is the data block. Default value = block 0. Repeated Hanns are permitted but the block must be in the time domain.

[←] Block Shift or Rotate

This command shifts or rotates the entire block a given number of channels to the left. If in frequency domain, real will be rotated into real, and imaginary into imaginary. An exception is F_n which has only a real component and is stored immediately after the dc real component. When a Block Shift or Rotate command is given, the dc real component is rotated into the location formerly occupied by the real component of F_n–1, and the F_n real component is rotated into the location formerly occupied by the imaginary component F_n–1. Similary, the real component of F_1 moves to the location formerly occupied by the real component of dc, and the imaginary component of F_1 moves to the location formerly occupied by the real component of F_n.) The data storage format illustration in Section 2, Figure 2-5, of this manual may help you visualize this operation. The command form is:

[←] [n1] [SPACE] [n2] [ENTER]

where:

n1 is the data block to be shifted. No default allowed.

n2 is the number of channels the block is to be shifted. No default allowed.

[←] Set (Variable Parameter)

SET

The gold SET command sets a Variable Parameter to the specified value. The command structure is:

[USER PROG] [←] [ ] [n1] [SPACE] [n2] [SPACE] [n3] [ENTER]

SET

where:

n1 is the Variable Parameter number.

n2 is the number to be multiplied by a power of 10.

n3 is the power to which the base 10 is raised.

For additional information, refer to the Variable Parameters paragraphs (specifically Y 1800) in this section.

+ Block Addition

This command linearly adds the data in any block, point by point, to the data in block 0, and stores and displays the results in block 0. The data in the other block is left unchanged. If only one block is double precision, then a single precision add will be performed; if both blocks are in double precision, a double precision add will be performed.* The command structure is:

+ | BLOCK n1 | ENTER

where:

n1 is any data block. Default value = 0 (in which case, data block 0 is added to itself.)

Since block addition is a linear operation, the data blocks should be in linear rectangular coordinates for more meaningful results. However, the magnitudes in polar coordinates may be added (for example, to obtain an ensemble average), but the resulting phase sum may be meaningless. If the two data blocks are of different domain or coordinates, the result will be in the domain or coordinate of block 0.

ADD + Add (Variable Parameter)

The gold ADD command adds the specified values. The command structure is:

USER PROG | ADD + | | n1 | SPACE | n2 | SPACE | n3 | ENTER

where:

n1 is a Variable Parameter that is set to the sum of n2 and n3. The default value of n3 is 1.

For additional information refer to the Variable Parameter paragraphs (specifically Y1801) in this section.

− Block Subtraction

This command subtracts the data in any block, point by point, from the data in block 0, and stores and displays the result in block 0. The data in the other block is left unchanged. If only one block is double precision, then a single precision subtract will be performed; if both blocks are in double precision, a double precision subtraction will be performed.* The command structure is:

− | BLOCK n1 | ENTER

where:

n1 is any data block. Default value = data block 0 (in which case, the result would be zero as data is being subtracted from itself.)

*Note: The result will be truncated to single precision if the data blocks differ by more than 4 decades.

Since block subtraction is a linear operation, the data blocks should be in linear rectangular coordinates for more meaningful results. However, the magnitudes in polar coordinates may be subtracted (for example, to obtain an ensemble average), but the resulting phase sum may be meaningless. If the two data blocks are of different domain or coordinates, the result will be in the domain or coordinate of block 0.

− Subtract (Variable Parameter)

SUB

The gold SUB command subtracts the specified values. The command structure is:

USER PROG	− (SUB)	n1	SPACE	n2	SPACE	n3	ENTER

where: (n1 = n2 - n3)

n1 is a Variable Parameter that is set to the difference between n2 and n3. The default value of n3 is 1.

For additional information refer to the Variable Parameter paragraphs (specifically Y1802) in this section.

MULT Loop Counter Multiply

KEYBOARD)

This command is used solely inside a count loop in a program; its purpose is to multiply any block by the loop counter integer. The command is executed only inside a loop, in a user-entered program, while the program is running and the BUSY light is on. If you attempt to execute the command directly from the Keyboard, a WHAT? signal will result. The command is:

MULT	BLOCK n1	SPACE	0	ENTER

where:

n1 is the block to be multiplied. No default allowed.

0 is a code number for this command.

MULT Block Multiply

This command performs a linear multiplication of block 0, point by point by any other block, n1, with the results being stored and displayed in block 0. The data in the other block remains unchanged. If only one block is in double precision, a single precision multiply will result; if both blocks are in double precision, a double precision multiply will result. The command form is:

MULT	BLOCK n1	ENTER

where:

n1 is any data block. Default value = block 0 (i.e., block 0 is then squared).

The domain of block 0 determines the type of point by point multiplication. Since multiplication is a linear operation, the data blocks involved should both be in the same domain and in linear rectangular coordinates for meaningful results.

MULT Integer Multiply

This function multiplies block n1 by the integer n2, and stores and displays the result in block n1. The command form is:

MULT n1 SPACE n2 ENTER

where:

n1 is the data block. No default allowed.

n2 is the integer, which can be any value from –32,768 through +32,767 (excluding zero which is loop counter multiply). No default allowed.

To multiply a data block by a fraction or a decimal, integer multiplication and division are used. For example, to multiply a data block by $\sqrt{2}$ (1.414), integer multiply the block by 1414, then integer divide by 1000.

MULT Complex Multiply

This function multiplies block n1 by a complex number (n2 + jn3). The command format is:

MULT n1 SPACE n2 SPACE n3 ENTER

where:

n1 is the data block. No default allowed.

n2 is the real part and n3 is the imaginary part of a complex number, which can be any value from –32,767 to +32,767. No default allowed.

If n1 is in the time domain, n3 is disregarded and an integer multiplication is executed (n1 $\times$ n2).

MULT

Multiply (Variable Parameter)

The gold MULT command multiplies specified values. The command structure is:

MULT

[USER PROG] [MULT] [ ] [n1] [SPACE] [n2] [SPACE] [n3] [ENTER]

where:

n1 is a Variable Parameter that is set to the product of n2 and n3. The default of n3 is 2.

For additional information refer to the Variable Parameter paragraph, (specifically Y 1803) in this section.

Block Conjugate Multiply

This function multiplies block 0 point-by-point by the complex conjugate of any other block, n1, and displays and stores the results in block 0. The data in the other block remains unchanged. If block 0 is in the time domain, a non-conjugate multiply will be performed. If only one block is in double precision, a single precision multiply will result; if both blocks are in double precision, a double precision multiply will result.

[* MULT] [BLOCK n1] [ENTER]

where:

n1 is any data block. Default value = block 0 (in that case, block 0 would be multiplied by the complex conjugate of itself. This is the operation used to obtain the power spectrum from the Fourier transform.)

Since block conjugate multiply is a linear operation the data blocks involved should be in linear rectangular coordinates for meaningful results.

CREATING DOUBLE PRECISION BLOCKS

When a self-conjugate multiply is performed (block 0 multiplied by its complex conjugate) then the result will be a double precision data block. A frequency domain data block which is completely real and where each data word (excluding dc and F_n) is 32 bits, is known as a double precision block. A self-conjugate multiply can only be executed by the following command:

[* MULT] [0] [ENTER]

where:

Block 0 is in the frequency domain.

Since the results of a self-conjugate multiply are purely real, the least significant bits are stored where the imaginary parts were formerly; thus each data word is now double precision (32 bits). The self-conjugate multiply and power spectrum (described later in this section) commands are the only processing commands that can create a double precision block. A cross-conjugate multiply whose result can have both real and imaginary parts is always single precision (16 bits).

A resident User Program is available (Y 3011) which allows you to disable/enable double precision block creation for self-conjugate multiply operations.

[* MULT] Conjugate

This command conjugates block n1, that is, changes the imaginary part of each channel. The command is:

[* MULT] [BLOCK n1] [SPACE] [0] [ENTER]

This command will be executed only if block n1 is in the frequency domain, single precision.

EXP [* MULT] Exponentiate (Variable Parameter)

The gold EXP command raises a Variable Parameter to a specified power. The command structure is:

[USER PROG] [EXP * MULT] [ ] [n1] [SPACE] [n2] [SPACE] [n3] [ENTER]

where:

n1 is the Variable Parameter that is set to the value of n2 to the power of n3 ($n2^{n3}$).

For additional information refer to the Variable Parameter paragraphs (specifically Y 1827) in this section.

[÷] Loop Counter Divide

This command is used solely inside a count loop in a KEYBOARD program; its purpose is to divide any block by the loop counter integer. The command is executed only inside a loop, in a user entered program, while the program is running and the BUSY light is on. If you attempt to execute the command directly from the Keyboard, a WHAT? signal will result. The command is:

[÷] [BLOCK n1] [SPACE] [0] [ENTER]

where:

n1 is the block to be divided. No default allowed.

0 is the code number for this command.

÷ Integer Divide

This function divides block n1 by the integer n2. The result is stored and displayed in block n1. The command structure is:

[÷] [BLOCK n1] [SPACE] [INTEGER n2] [ENTER]

where:

n1 is the data block to be divided. No default allowed.

n2 is the integer, which can be any positive value from 1 through +32,767. No default allowed.

÷ Block Division

This command divides the data in block 0, point by point, by the data of any other block, n1, and displays and stores the result in block 0. The data in the other block remains unchanged. If both blocks involved are double precision the result will be double precision. If block 0 is in double precision but block n1 is not, then block 0 will be truncated before the command is executed. If block n1 is in double precision but block 0 is not, the result will be single precision. The command form is:

[÷] [BLOCK n1] [ENTER]

where:

n1 is the data block by which block 0 is to be divided. No default allowed.

Since block division is a linear operation, both data blocks should be in linear rectangular coordinates for meaningful results. The domain of block 0 determines the type of point by point division.

The number system used in the Fourier Analyzer is floating point on a block basis. This means that all the data words stored in a block represent numbers between -1 and +1 and that the block scale factor applies to all words in the data block and scales them to the correct value. When two data blocks are divided, the scale factors of the two are first subtracted, and the resulting scale factor determined. Then the division proceeds on a channel-by-channel basis using a complex division algorithm. Since the largest number that can be stored in memory is equivalent to 1.0 (multiplied by the block scale factor), the following procedure is used to assure that no number larger than 1.0 need be stored.

When the quotient overflows 1.0 in a channel, the quotients in all prior channels plus the overflow channel are divided by 2 and the scale factor is adjusted, leaving the result with the correct numerical value. The unprocessed channels of the dividend are also divided by 2, assuring the proper result in those channels. This procedure is repeated until the quotient between the two numbers of a given channel is less than 1.0. If this process were allowed to proceed unchecked, as in the case where the divisor block has a very low value for one channel, and the dividend block a value near unity for that channel, the result would be that the high ratio for the channel would be computed with accuracy, while all other values in the data block would be scaled down (divided by 2) so many times as to lose any significant resolution or accuracy. To limit this problem, a maximum of 8 scaledowns are allowed. This means that the quotient and dividend will be divided by 256 in the worst case. A resident User Program is available which allows you to select the number of scaledowns (see Section 7).* Scaledown of the data block is only performed when the division is successful. If a division is not successful, the quotient is set equal to zero and block division continues.

*(SECTION 7, Y3012)

Example: The following example illustrates the use of a number of processing operation keys and specifically illustrates the effects of the dividend overflow. Set a block size of 128 and enter a constant value of 3.2 into all channels of data block 0 with the following set of commands:

KEY BOARD   0   SPACE   0   SPACE   1   2   7 (or 8)   ENTER

KEY BOARD   –   4   SPACE   0   ENTER   3   2   0

0   0   ENTER

A constant value or 3.2 as shown in Figure 3-3(a) is now in data block 0. Store this function in data block 1 with the following command:

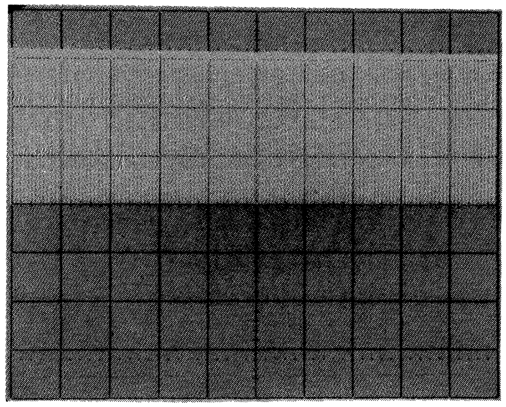
STORE   1   ENTER

Note that the function stored in block 1 is identical to that which is left in block 0. This can be checked with the following commands:

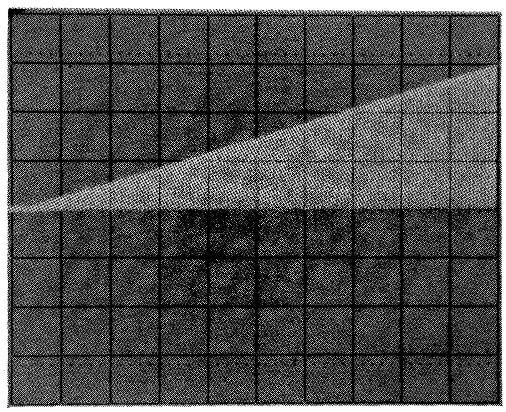
DSPLY   1   ENTER

DSPLY   ENTER

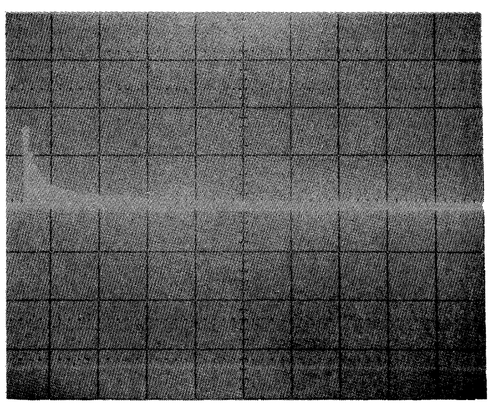
Figure 3-3. Example of Block Division, Showing Overflow



(a) Constant of 3.2 in all channels



(b) After integration of channels 5 through 127



(c) After block division is set to zero

Divide the value in data block 1 by the integer value 64 with the following integer divide command:

DIV 1 SPACE 6 4 ENTER

Note that the constant in block 1 is now 3.2/64 or 0.05. If we integrate block 1 from channel 5 to 127, the integral of the constant value will be a ramp which starts with channel 5 and increases to the end of the data block. This command will execute the integration (format explained in a following paragraph):

INTEG 1 SPACE 5 SPACE 1 2 7 ENTER

The result of this integration is shown in Figure 3-3(b). If the constant value in data block 0 is divided by data block 1, the low values in the first few channels will cause the divide to overflow. Do this division with the following command:

DIV 1 ENTER

The result of the block division is shown in Figure 3-3(c). Note that the first six channels are set to zero due to overflow.

÷ DIV Division (Variable Parameter)

The gold DIV command sets Variable Parameter n1 to the value of n2 divided by n3. The command structure is:

USER PROG | ÷ (DIV) | n1 | SPACE | n2 | SPACE | n3 | ENTER

where:

n1 is the Variable Parameter in which the result of the division is stored. $(n1 = n2/n3)$

n2 is the dividend.

n3 is the divisor. Default value of n3 is n2.

For additional information refer to the Variable Parameter paragraphs (specifically Y 1804) in this section.

∫ Integration

There are two types of integration commands; block integration, in which the entire data block is integrated, and partial block integration, in which only a given number of channels in the block are integrated. The command for block integration is:

∫ BLOCK n1 ENTER

where:

n1 is the data block. Default value = block 0.

The command for partial block integration is:

∫ BLOCK n1 SPACE CHANNEL n2 SPACE BLOCK n1 ENTER

where:

n1 is the data block.

n2 is the first channel of the range to be integrated.

n3 is the last channel of the range to be integrated.

Default table is as follows:

Element	Meaning of Element	Default Value of Element
n1	data block	block 0
n2	first channel to be integrated	channel 0
n3*	last channel to be integrated	channel n2

*If n3 is defaulted, then channel n2 will be the start and end of the range of integration.

This function was designed principally for integrating spectra. In a normal spectrum, the amplitude at each frequency is taken as the total area in the channel width Δf. Integrating the spectrum is then simply the sum of the amplitudes channel by channel. The individual area approximations are taken to be rectangular, not trapezoidal. In the integration process, the value of the first integrated channel remains unchanged, the value of the second integrated channel becomes the sum of the first and second channel amplitudes, the value of the third integrated channel becomes the sum of the second (sum) plus the third, etc. In the Fourier Analyzer, integration and differentiation are exactly reciprocal processes.

Differentiation

There are two types of differentiation commands: block differentiate, in which an entire data block is differentiated, and partial block differentiate, in which a given range of channels is differentiated. The command structure for block differentiation is:

[$\frac{d}{dx}$] (BLOCK n1) [ENTER]

where:

n1 is the data block. Default value = block 0.

The command for partial differentiation is:

[$\frac{d}{dx}$] (BLOCK n1) [SPACE] (CHANNEL n2) [SPACE] (CHANNEL n3) [ENTER]

where:

n1 is the data block.

n2 is the first channel to be differentiated.

n3 is the last channel to be differentiated.

Just as integration in the Fourier Analyzer is merely the summing of amplitudes, so differentiation is successive subtraction of amplitudes (the so-called first difference approach to differentiation). The value of the first channel is left unchanged. The value of the second channel is then taken as the value of the second in the original function minus the value of the first. The value of the third is the original third minus the second (difference), etc.

It will be noticed that, after repeated differentiations of functions which have slopes that are relatively low, repeated integrations will not bring back the exact original function. This is because the difference between small differences (low slope) is a small number, and repeated differentiations eventually yield numbers too small for the resolution of the Fourier Analyzer. Repeated integrations cannot bring these lost values back, hence, the slight discrepancy between the reintegrated function and the original.

Element	Meaning of Element	Default Value of Element
n1	data block	block 0
n2	first channel to be differentiated	channel 0
n3*	last channel to be differentiated	channel n2
*If n3 is defaulted, then channel n2 will be the start and end of the range of differentiation.		

MEASUREMENT FUNCTIONS

These paragraphs describe the MEASUREMENTS keys which perform operations related to analysis of the sampled signal.

F Fourier Transform

This command takes the forward or inverse Fourier transform of a data block depending on domain of block. The Fourier transform is taken on single precision blocks; if the data block is double precision it will be truncated before command is executed. The Fourier transform can be taken on one data block, or on two data blocks in succession.

The command structure for the Fourier transform is:

[F] [BLOCK n1] [SPACE] [BLOCK n2] [ENTER]

where:

n1 is a data block. Default value = block 0.

n2 is another data block. Default value = transform taken only on block n1.

A Fourier transform always takes data blocks n1 and n2 from one domain to the other. The result of a Fourier transform is always in rectangular coordinates. The value of the imaginery part of the dc and the imaginery part of the highest frequency channels is always 0, as explained below. Both time and frequency domain data must be in linear rectangular coordinates prior to being forward or inverse transformed.

FOURIER TRANSFORM USED IN FOURIER ANALYZER

The following is the definition of the Discrete Finite Transform (DFT) used in the Fourier Analyzer:

(forward) $$F\,(m\Delta f) = \frac{1}{N} \sum_{n=0}^{n=N-1} f\,(n\Delta t)\, e^{-\frac{i2\pi nm}{N}}$$

(inverse) $$f\,(n\Delta t) = \sum_{m=0}^{m=N-1} F\,(m\Delta f)\, e^{+\frac{i2\pi nm}{N}}$$

where:

Δt = time increment

Δf = frequency resolution

N = the total number of points in the time domain, i.e., the block size

N time points theoretically suggest there should be N frequency points. Since the data computed is always a pure real time series, the negative frequencies that are normally present in a double-sided transform are perfectly symmetrical with the positive frequencies. These negative frequencies therefore are redundant data and are not stored. In the Fourier Analyzer, from an N-point real time series, we compute and display N/2 positive frequencies, plus a real (cosine) and an imaginary (sine). The highest frequency has only a real, but no imaginary value. The actual arithmetic is as follows: N/2 positive real frequency values plus dc value = N/2 + 1 real points. The imaginary side has no dc value and no value for the highest frequency. Therefore it has N/2 + 1 - 2, or N/2 - 1 points. Adding the real and imaginary points together we get N/2 + 1 plus N/2 - 1, or a total of N points in the frequency domain from N points in the time domain.

The display of the imaginary components of the spectrum always shows a 0 in the dc and the highest frequency channels. Likewise 0 phase shows in these channels.

When taking an inverse transform of a frequency spectrum, the Fourier Analyzer assumes that the spectrum is symmetrical and thus computes a pure real time series.

Important Note: When entering frequency domain values from the Keyboard, zero must be entered for the imaginary or phase value of dc and the highest frequency, when these two points are part of the data.

PHASE CONSIDERATIONS

Each frequency in the spectrum after a Fourier Transform can be written as:

$$F(m\Delta f) = A(m\Delta f) + iB(m\Delta f)$$

The spectrum values of the real part, A, and the imaginary part, B, depend on the starting point in time (i.e., the relative phase of the signal). Thus, the linear Fourier spectrum shape will vary both in amplitude and polarity from sample record to sample record, depending on the starting point of the signal. To get a constant spectrum shape, independent of the starting point of the signal, it is necessary to convert the real and imaginary components into magnitude and phase or into the appropriate power spectrum. This can be done using the POLAR key to obtain a magnitude spectrum, or in the case of an autopower spectrum, by doing a self complex conjugate multiply (see *MULT key).

In the case of polar coordinates, the magnitude can be written:

$$|F(m\Delta f)| = \sqrt{A^2(m\Delta f) + B^2(m\Delta f)}$$

and the phase can be written:

$$\angle F(m\Delta f) = \arctan \frac{B(m\Delta f)}{A(m\Delta f)}$$

The power spectrum is:

$$F(m\Delta f)^2 = A^2(m\Delta f) + B^2(m\Delta f)$$

While the magnitude and power spectrum do not depend on the starting point or phase of the input time series, the phase angle, of course, does.

AMPLITUDES IN SPECTRA

Since the Fourier transform algorithmn used only stores the positive half of the spectrum, a sine wave of peak amplitude A will yield a spectral line of amplitude A/2. The rms value for the sinewave is $A/\sqrt{2}$, so that to find the rms value in the spectrum, it is necessary to multiply the displayed amplitude by $\sqrt{2}$, (i.e., 1.414). See the Integer Multiply command in this section.

A complete tabulation of the values given by the Fourier transform and the constants needed to correct these are shown in the table below.

Amplitude Values For an A sin ωt Input

Function	Fourier Analyzer Gives	Mult By	To Get
Linear or Magnitude Spectrum	A/2	$\sqrt{2}$	RMS value $(A/\sqrt{2})$
Linear or Magnitude Spectrum	A/2	2	Peak value A
Power Spectrum	$A^2/4$	2	Power $(A^2/2)$

NORMALIZING THE SPECTRUM

The transform is taken without reference to the effective bandwidth set by the ADC sampling controls (for information on these controls, see Section 2). Since the effective noise power bandwidth of the Fourier Analyzer is 1/T, where T is the sample record length, multiplication of the spectral values by T will normalize the above power spectrum to per unit bandwidth. To normalize a linear or magnitude spectrum to per $\sqrt{Hz}$, simply multiply the spectrum by $\sqrt{T}$ In this way, measurements of random noise and other non-coherent signals that can be defined as having a constant spectrum density, can be normalized to give equivalent readings independent of sample rate and bandwidth of the Fourier Analyzer.

However, if sinusoids and other coherent signals are normalized in this way, they will have amplitudes that depend on system bandwidth. Thus, if one is seeking equivalent measurements for coherent signals, it is best not to normalize.

F3

F

F3

The gold F3 command calls a prewritten program if one has been placed under control of this key. The command structure is:

USER PROG | F3 F | ENTER

For information on how to write programs and how to place them under control of these keys, refer to Soft Keys paragraph in this section.

CORR

Correlation

Correlation is an operation done between two blocks of data, therefore by the rule discussed under the general form of the Keyboard command in this section, one of the blocks must be block 0, and only the second block is named in the command. Results are stored and displayed in block 0. Correlation is performed on single precision blocks so that if any of the data blocks involved are double precision, they will be truncated before execution of the command.

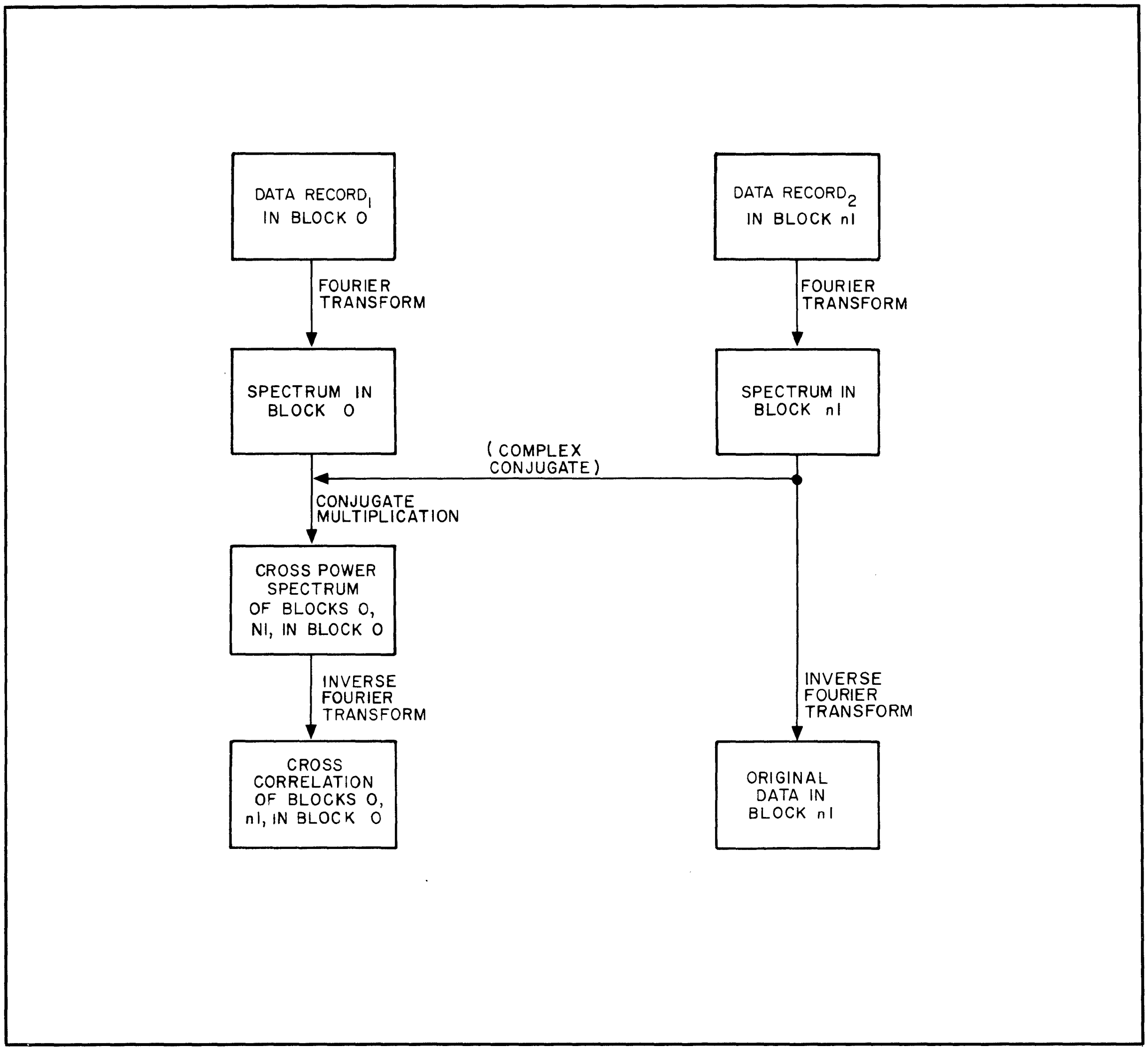
If the correlation is between block 0 and another block, then the result is a cross correlation as shown in Figure 3-4. If between block 0 and itself, the result is an autocorrelation, as shown in Figure 3-5. The command structure is:

CORR   BLOCK nI   ENTER

where:

n1 is any data block. Default value = block 0 (in that case, an autocorrelation will result).

Figure 3-4. Flow Graph of Cross Correlation Computational Procedure

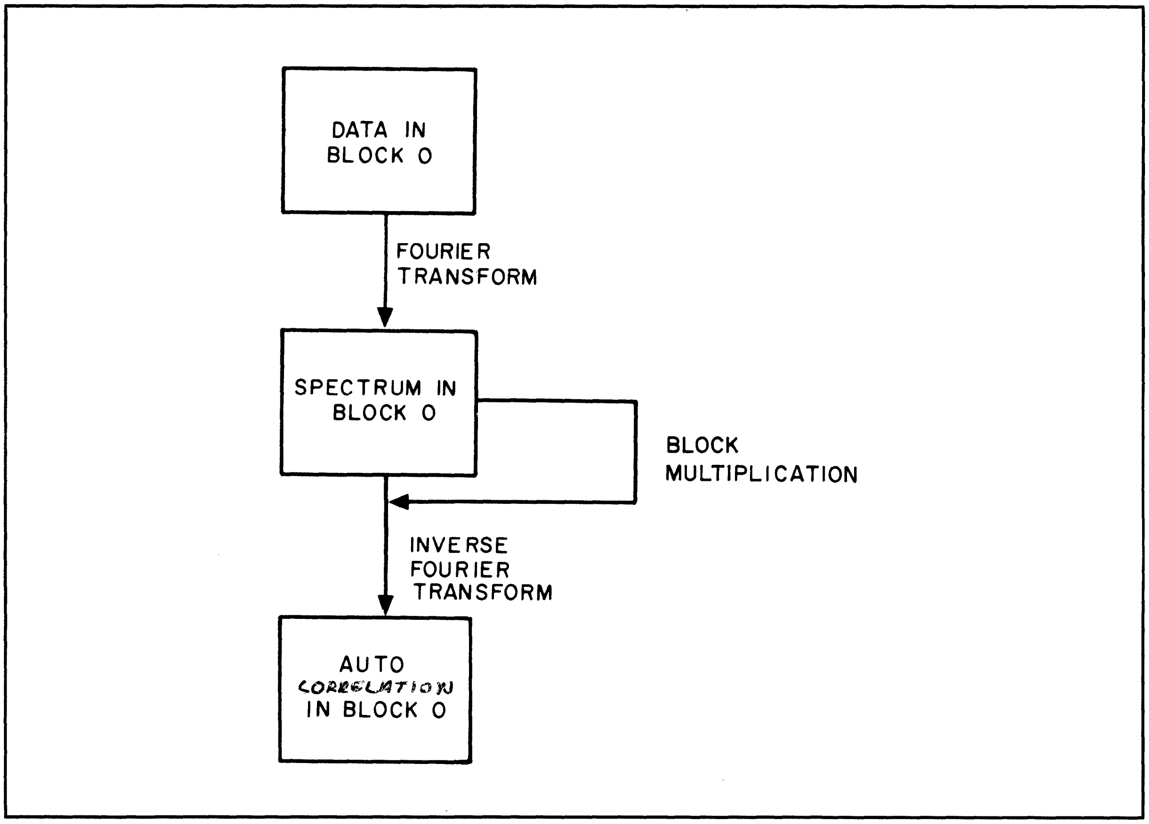


The means by which the Fourier Analyzer obtains the correlation function is as follows: first the system takes the Fourier transform of both blocks (cross correlation) or of block 0 (autocorrelation). The Fourier Analyzer then does a conjugate multiply on both blocks or block 0 with itself. This yields the power spectrum. Finally, the inverse Fourier transform is taken on the power spectrum, to produce the correlation function.

NOTE

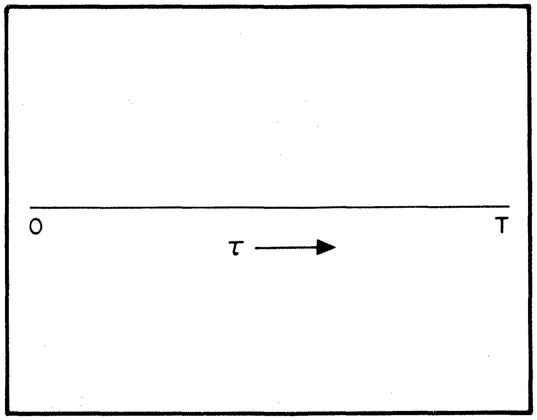
When beginning a correlation with data in the time domain, it is necessary to multiply the result by the block size to achieve the proper scaling and calibration.

Figure 3-5. Flow Graph of Autocorrelation Computational Procedure

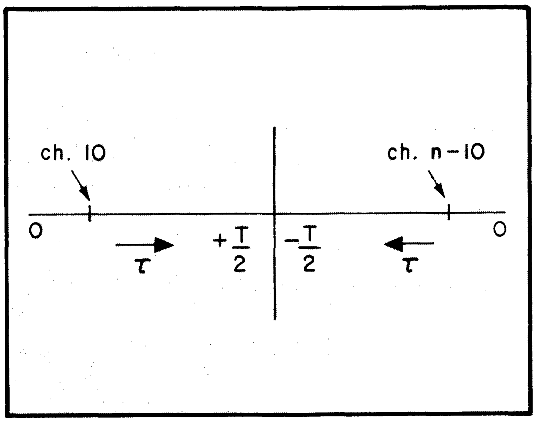


The final correlation function is in the time domain, but with the horizontal axis representing values of τ (lag) not t (time) as originally, over a range of $\pm T/2$. T is the length of the data window. This is shown in the three sets of axes in Figure 3-6. Note that the position of the origin can be switched with the ORIGIN switch on the Display Unit panel. The switch of origins is a display function, not a change in channel locations in the data block. With the ORIGIN switch on LEFT, the channels run from 0 on the left to N–1 on the right. When the origin is set to CENTER, these channel numbers retain their same relative positions.

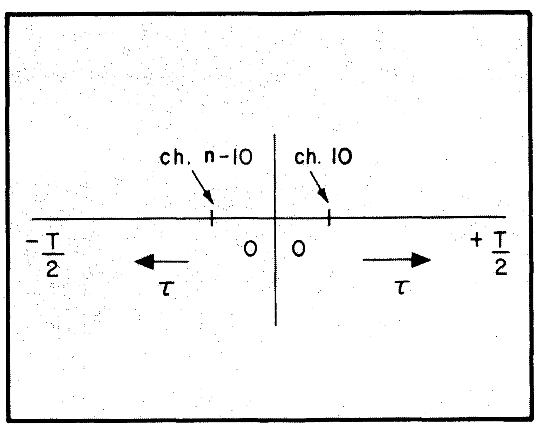
Figure 3-6. Correlation Function Examples



NORMAL TIME
DOMAIN DISPLAY



AFTER CORRELATION,
<u>ORIGIN</u> SWITCH TO <u>LEFT</u>



AFTER CORRELATION,
<u>ORIGIN</u> SWITCH TO <u>CENTER</u>

Example 1: Set a block size of 128, then manually create a rectangular pulse in block 0, 11 channels wide, via the following commands:

CLEAR ENTER
KEY BOARD 0 SPACE 0 SPACE 10 ENTER .

KEY BOARD -4 SPACE 0 ENTER
30000 ENTER

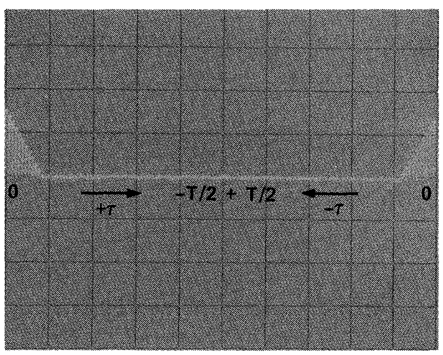
Take the autocorrelation of the pulse by pressing:

CORR ENTER

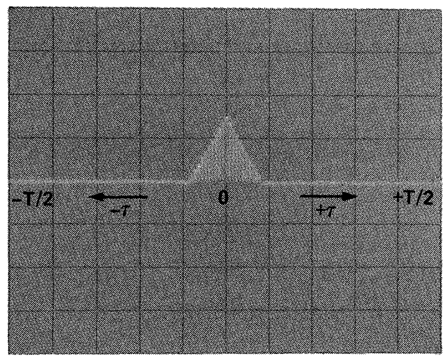
MULT 0 SPACE 128

Correct the calibration by multiplying by the blocksize. The result should be as shown in Figure 3-7(a). Switching the ORIGIN switch to CENTER should give the result shown in Figure 3-7(b).

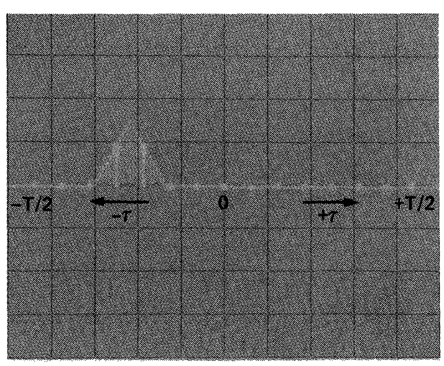
Figure 3-7. Correlation Function Displays



(a) ORIGIN switch to LEFT



(b) ORIGIN switch to CENTER



(c) After original pulse was shifted 100 channels (8 point markers)

Example 2: Set the same block size and create the same rectangular pulse as in Example 1.

Transfer the pulse to block 1 by pressing:

STORE 1 ENTER

Shift the pulse in block 1, 100 channels to the left:

← 1 SPACE 1 0 0 ENTER

Take the cross correlation of block 0 and 1 by:

CORR 1 ENTER

Correct the calibration by multiplying by the blocksize.

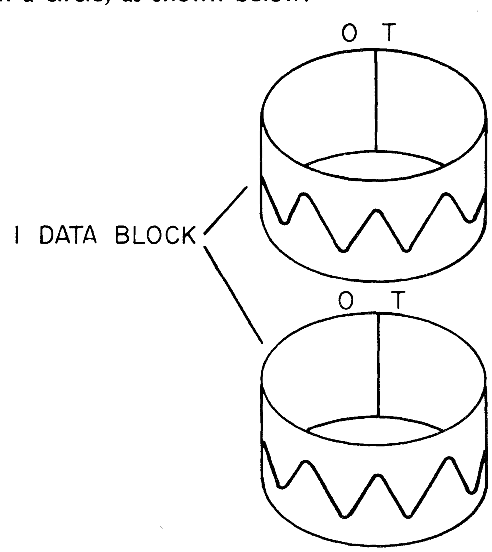
MULT 0 SPACE 128

The result should be as shown in Figure 3-7(c), and indicates a lag between the two signals of 28 channels.

WRAP-AROUND ERROR

In any correlation or convolution operation using the Fast Fourier Transform, the effect known as wrap-around error must be taken into account. It occurs whenever one of the signals to be correlated or convolved is 0 for less than half of the record length T.

To understand the cause of wrap-around error, we must realize that correlation or convolution in the Fourier Analyzer is done by going through the frequency domain. This means that whatever portion of the original signal the Fourier Analyzer takes in, it assumes the data block to be periodic, with the period having the record length T, as shown in Figure 3-8, line 3. Now when correlation or convolution is done, an error results because the original signal was in fact not periodic in this way. The very first increment of shift, as shown in Figure 3-8, line 4, introduces an error, because, for example, point (a) is being multiplied by point (b), rather than (a') by (b'), which should be the case, as shown in Figure 3-8, lines 4 and 5. The term "wrap-around error" comes from the fact that the repetitive records can be conceived as a single record, bent or wrapped around in a circle, as shown below:



The way to eliminate the problem is shown in Figure 3-8, lines 6 through 9. First we clear out T/4 channels on each end of one data block to BE shifted. Now, when this record is to be shifted, it does not immediately run into the next repetition in the other record. In fact, it can be shifted a total of T/4 increments before this happens, as shown in Figure 3-8, line 7, and the correlation is thus valid over this range. But shifts from T/4 to T/2 of course do run into the next repetition, and hence are no good, just as, in the first case (Figure 3-8, line 4) the first shift was no good. (The same applies to shifts in the opposite direction, since correlation involves a +T/2 and a −T/2 shift.) We now have a good correlation from 0 to |T/4|, bad from |T/4| to |T/2|. Therefore, in the final display, we simply clear out, or ignore, the outer 1/4 of the correlation or convolution function.

The actual procedure is as follows: (See also sample correlation subroutine in Appendix C.)

1. In the case of autocorrelation, take the data into block 0, and STORE in block n1. (By the nature of the STORE command, this will now leave the data in both block 0 and block 1.) In the case of the cross correlation, simply introduce the data into blocks 0 and n1.

2. Next CLEAR out the first and last quarter of either of the blocks.

3. Correlate the blocks, using the CORRELATION command.

4. When the resulting correlation function is displayed, the last quarters will also be in error. Either ignore these quarters, i.e., the channels running from +T/4 to +T/2 shifts, and from −T/4 to −T/2 shifts, or clear them out also. The half of the correlation that remains, will always be correct.

Figure 3-8. Example of Wrap-Around Error

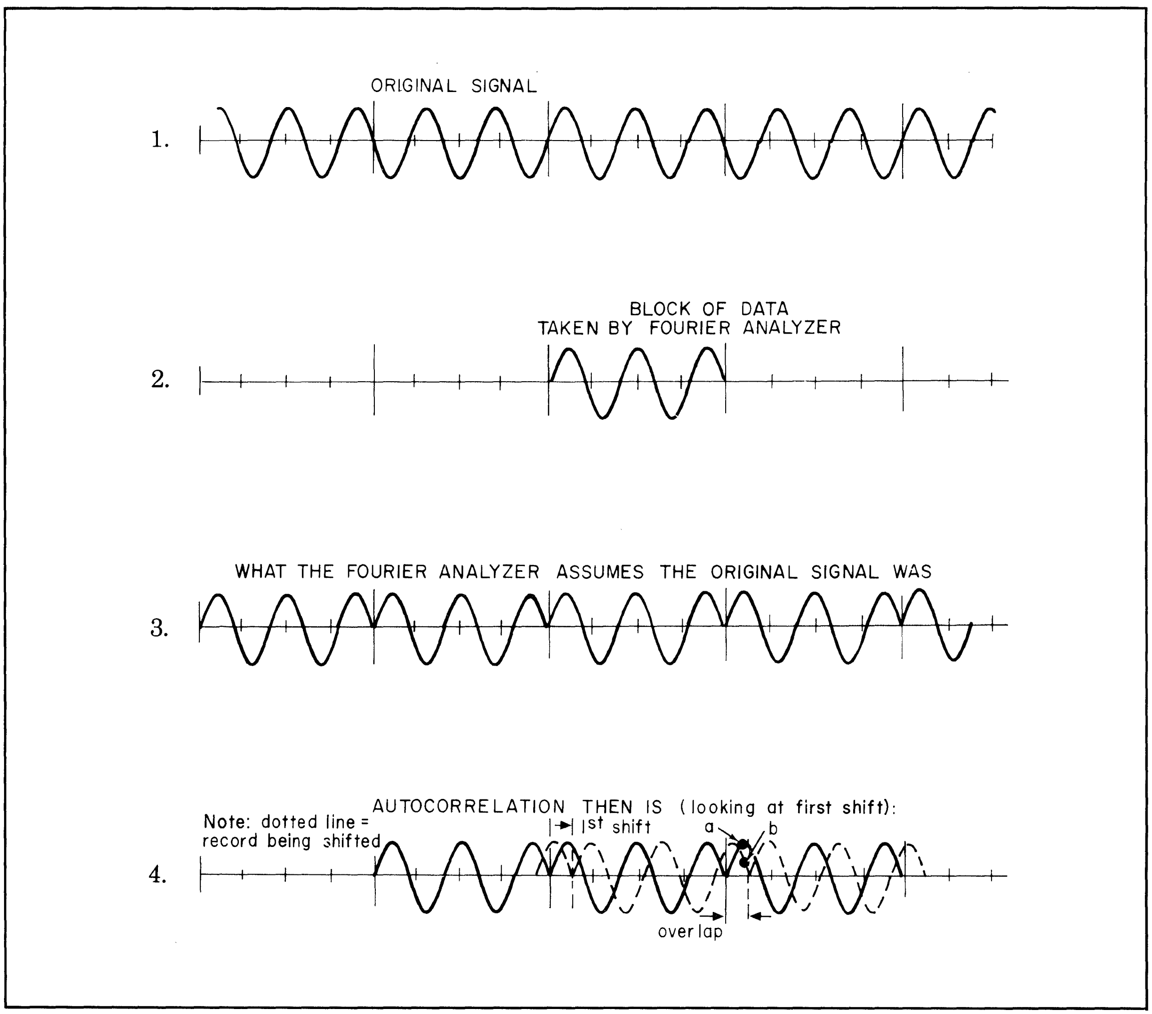
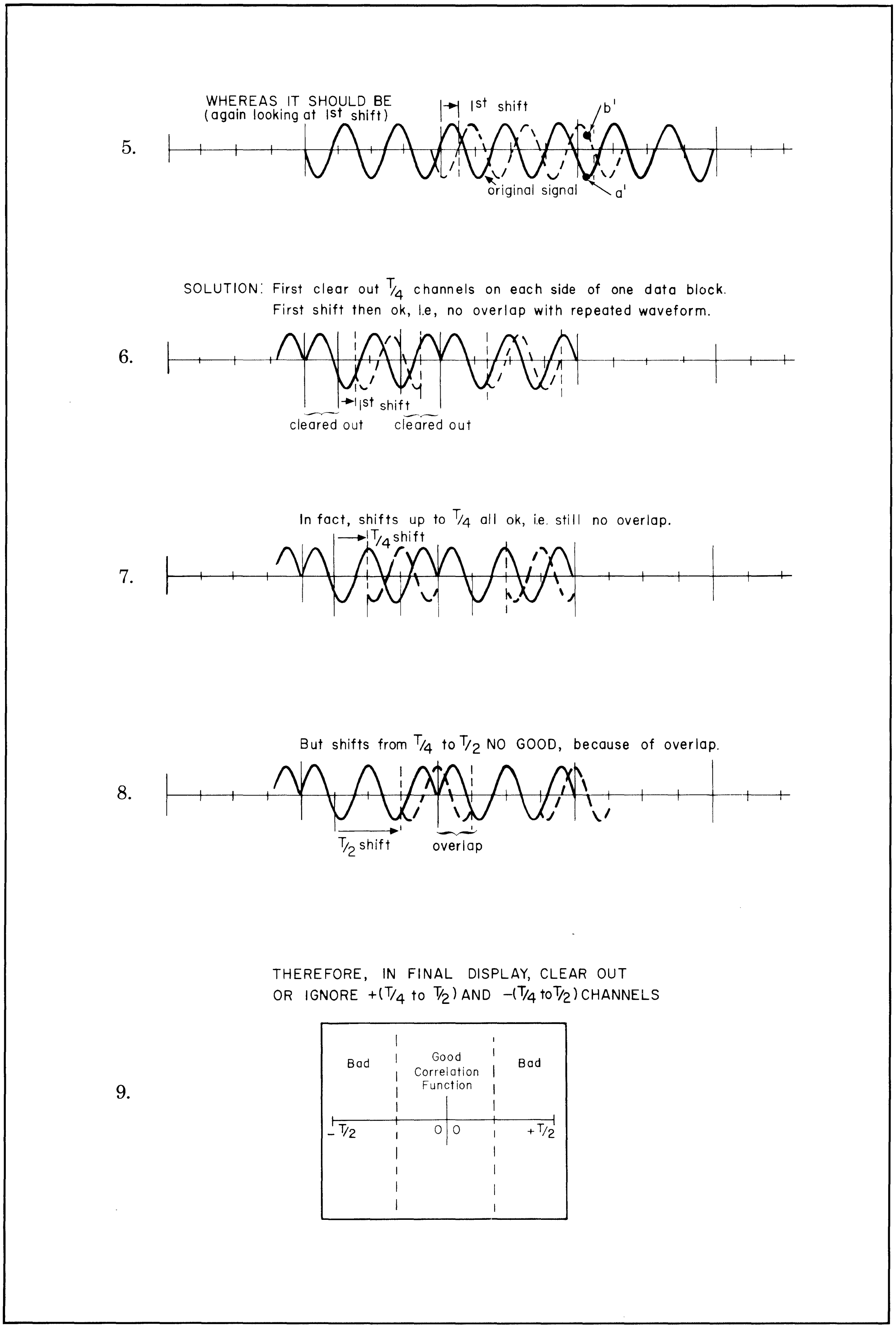


Figure 3-8. Example of Wrap-Around Error (cont'd)



F1

CORR

F1

The gold F1 command calls a prewritten program if one has been placed under control of this key. The command structure is:

F1

USER PROG | CORR | ENTER

For information on how to write programs and how to place them under control of these keys, refer to the Soft Key paragraph in this section.

CONV

Convolution

Convolution, like correlation, is an operation done between two blocks of data, therefore by the rule discussed under the general form of the Keyboard command in this section, one of the blocks must be block 0, and only the second block is named in the command. Results are stored and displayed in block 0. Convolution is performed on single precision blocks so that if any of the data blocks involved are double precision, they will be truncated before the command is executed.

The command structure is:

CONV | BLOCK NI | ENTER

where:

n1 is any data block. Default value = block 0.

The Fourier Analyzer obtains the convolution function as follows: First the system takes the Fourier transform of both blocks or of block 0. The Fourier Analyzer does a block multiply on both blocks or block 0 with itself. Finally, the inverse Fourier transform is taken to produce the convolution function. The convolution function displays are shown in Figure 3-9. (Flow graphs of these computations are shown in Figures 3-10 and 3-11.)

NOTE

When beginning a convolution with data in the time domain, it is necessary to multiply the result by the block size to achieve the proper scaling and calibration.

The final convolution function is in the time domain, but with the horizontal axis representing values of τ(lag), not t (time) as originally, over a range of $\pm T/2$. T is the length of the data window. This is shown in the three sets of axes below:

Figure 3-9. Convolution Function Displays

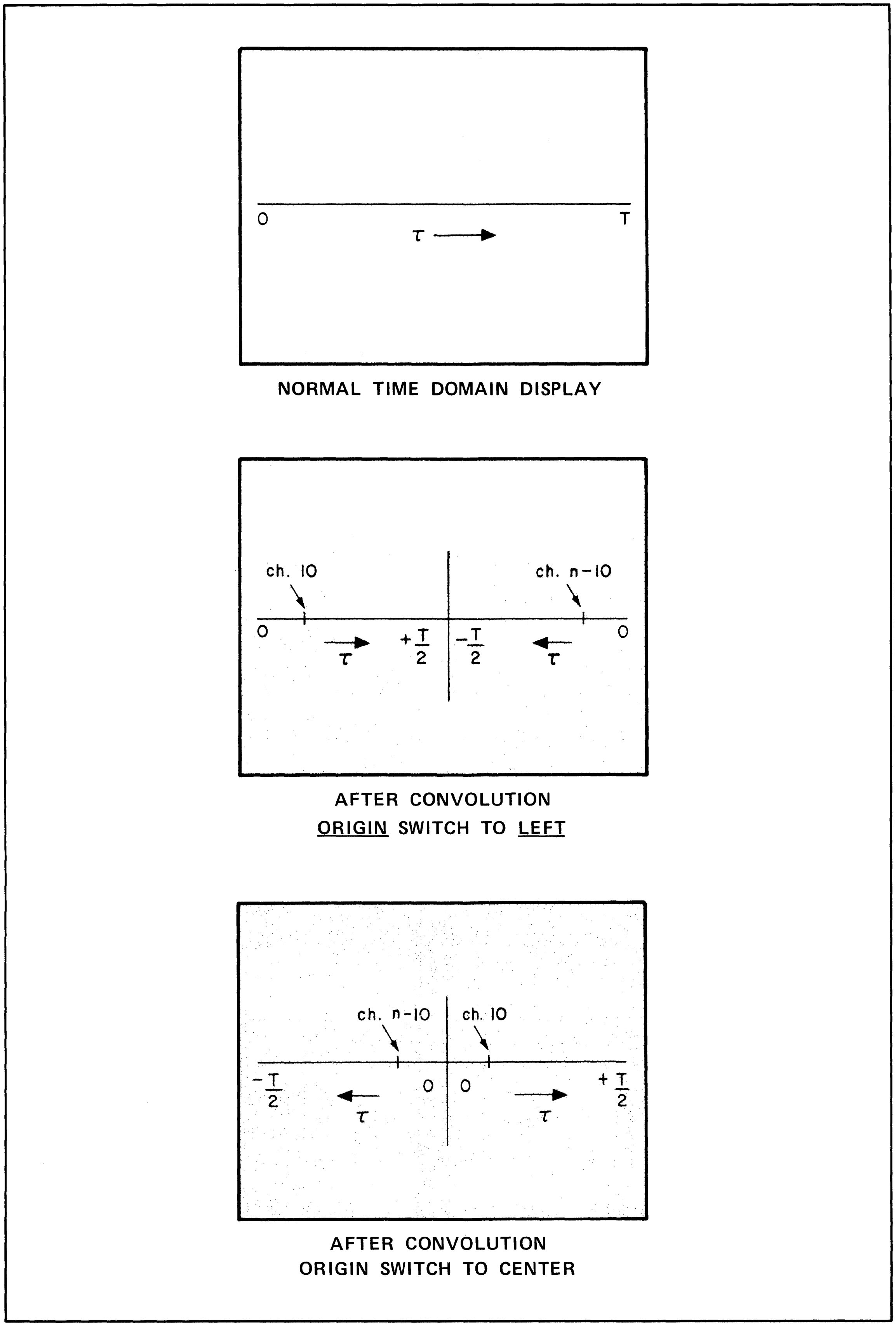
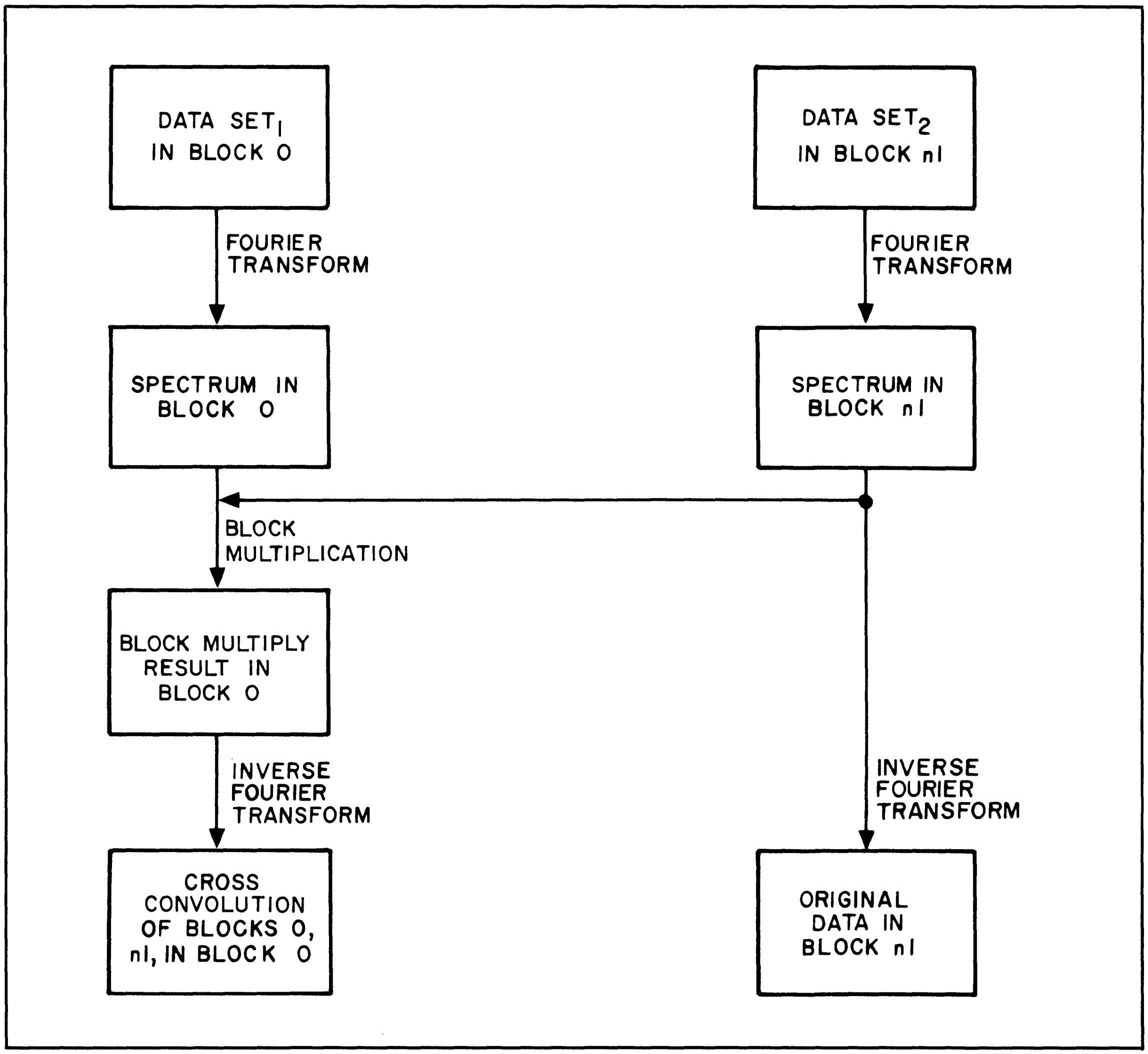


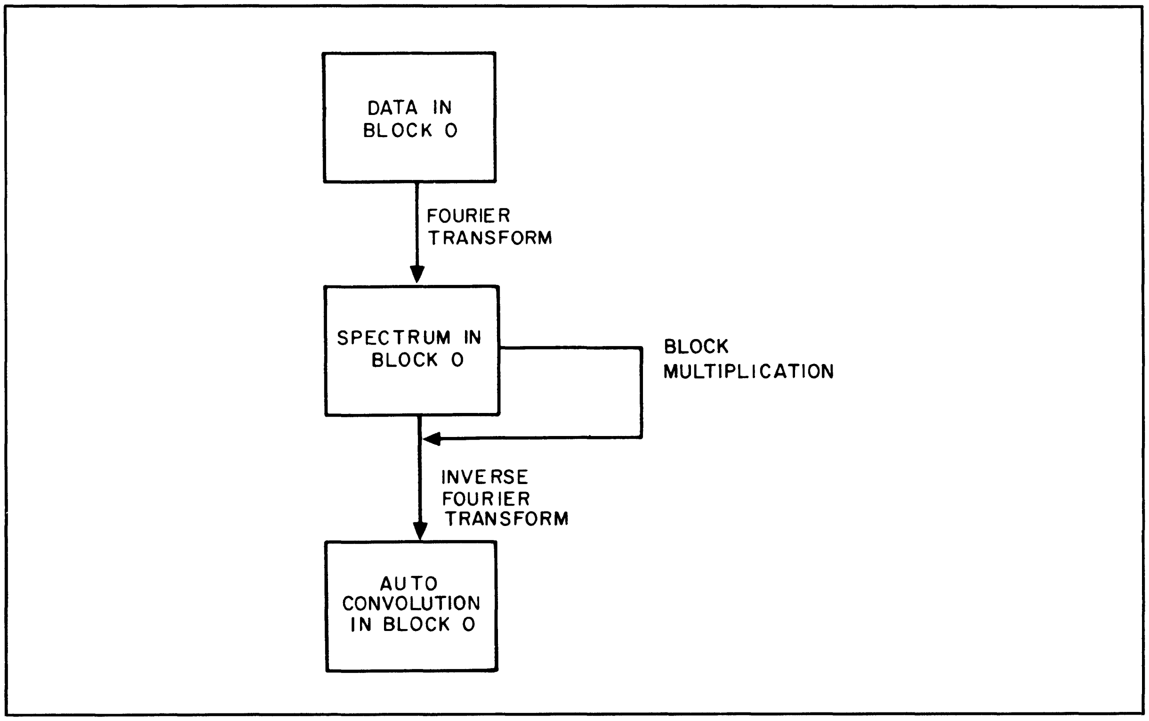
Figure 3-10. Flow Graph of Cross Convolution Computational Procedure



Note that the position of the origin can be switched with the ORIGIN switch on the Display Unit. The switch of origins is a display function, not a change in channel locations in the data block. With the ORIGIN switch to LEFT, the channels run from zero on the left to N–1 on the right.

Be sure to see the remarks on wrap-around error in this section. This error applies to both the convolution and correlation functions.

Figure 3-11. Flow Graph of Convolution Computational Procedure



Example 1: Set a block size of 128, then manually create a rectangular pulse in block 0, 11 channels wide with frequency code 17, via the following commands

KEY BOARD 0 SPACE 0 SPACE 10 ENTER
KEY BOARD -4 SPACE 0 SPACE 17 ENTER
30000 ENTER

Take the convolution of the pulse by pressing:

CONV ENTER

Correct the calibration by multiplying by blocksize.

MULT 0 SPACE 128

The results should be as shown in Figure 3-12(a). Setting the ORIGIN switch to CENTER should give the results shown in Figure 3-12(b).

Example 2: Set the same block size and create the same rectangular pulse as in Example 1. Then transfer the pulse to block 1 by pressing:

STORE 1 ENTER

Shift the pulse in block 1, 100 channels to the left:

← 1 SPACE 1 0 0 ENTER

Convolve blocks 0 and 1 by:

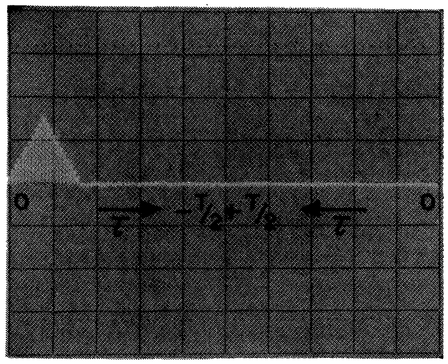
CONV 1 ENTER

Correct the calibration by multiplying by blocksize.

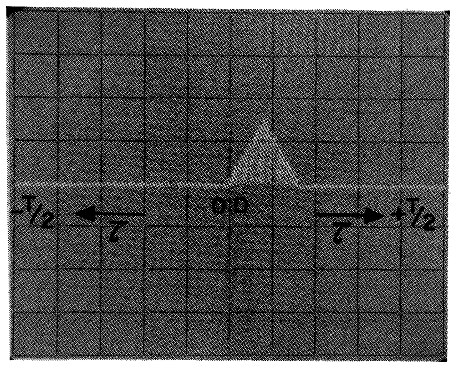
MULT 0 SPACE 128

The result should be as shown in Figure 3-12(c) and indicates a lag between the two signals of 28 channels.

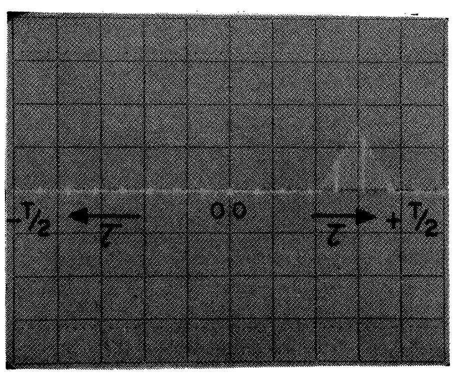
Figure 3-12. Convolution of Pulse (a and b) and Shifted Pulse (c)



(a) ORIGIN switch on LEFT



(b) ORIGIN switch on CENTER



(c) After original pulse was shifted 100 channels
(8 point markers)

CONV **F4**

F4

The gold F4 command calls a prewritten program if one has been placed under control of this key. The command structure is:

USER PROG   CONV (F4)   ENTER

For information on how to write programs and how to place them under control of these keys, refer to the Soft Key paragraph in this section.

HISTO GRAM **Histogram**

This command causes an amplitude histogram to be produced, using time-domain data entered or loaded into block 0.

The calibration of the horizontal scale for the histogram depends on the setting of the ADC input attenuator (OVERLOAD VOLTAGE switch) at the time the data was collected, and is specified on the basis of a fixed-point word format. Any change in the ADC attenuator setting during the time the data is being collected, or a computational change to the block floating-point scale factor associated with this data, will result in invalid calibration of the histogram horizontal scale.

Also, data is (normally) maximized when it is transferred between data blocks (by a LOAD or STORE command). This can cause incorrect histogram calibration, if data is entered from the ADC into some block other than block 0 and then loaded into block 0 to be histogrammed. (For example, this could occur if the data is entered from more than one ADC channel, with data from each channel being stored in a separate data block.) To prevent invalid histogram calibration due to this cause, the maximization routine should be disabled by using USER PROG 3010 before the first LOAD or STORE instruction involving the data to be histogrammed is given. Maximization will remain disabled until re-enabled by use of USER PROG 3010 or the RESTART key.

The general form of the histogram command is:

HISTO GRAM   BLOCK n1   SPACE   CHANNEL n2   SPACE   CHANNEL n3   ENTER

where:

n1 is the data block into which the histogram from block 0 is to be entered. n1 cannot be defaulted or equal to 0.

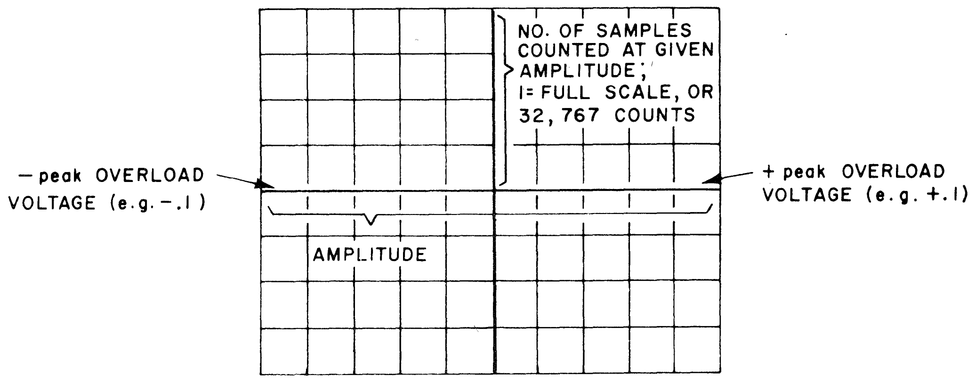
n2 is the channel where the histogram will begin. Default value is 0.

n3 is the channel where the histogram will end. Default value is n2. If n3 is defaulted, then channel n2 will be the beginning and the end of the histogram. If both n2 and n3 are defaulted, the entire block 0 will be histogrammed.

Block 0 must be in the time domain for histogram operation or a WHAT? signal will result. If initially any value in block n1 is equal to 32,767 then the command will not be executed. After programming, the mode of block n1 will be made time-linear rectangular, its scale factor set to zero and block calibrator set equal to 1. Full scale on the histogram is indicated by the value 1.0 or 32,767 counts.

Operational Hint: The smaller the blocksize, the smoother the histogram (due to differential linearity).

The histogram display is automatically set up as follows:



In the Processor memory, each voltage amplitude is an address. Counts are accumulated at each address until the number at any one address is 32,767. When and if that occurs, the histogram stops automatically, and is redisplayed with the correct scale factor. Further counts are ignored. Full scale on the histogram is indicated by the value 1.0. To convert this to actual number of counts, multiply the data block by 32,767 by pressing:

MULT HISTO DATA BLOCK SPACE 3 2 7 6 7 ENTER

Since the histogram command does not automatically clear the data block before accumulating a new set of values, you must be certain to do this yourself, unless additional values are to be collected on an old histogram. To clear the block, press:

CLEAR BLOCK nl ENTER

The program example below shows how two channels of ADC input data can be histogrammed, and includes the commands to disable and enable maximization. The ADC controls are set to enable 2-channel input operation.

LABEL 0 ENTER	
CLEAR 2 ENTER	
CLEAR 3 ENTER	
USER PROG 3010 SPACE 1 ENTER	Disable maximization
LABEL 1 ENTER	
ANALG IN ENTER	ADC inputs go into blocks 0 and 1
HISTO GRAM 2 ENTER	Histogram data in block 0, and put results in block 2
LOAD 1 ENTER	Copy block 1 into block 0
HISTO GRAM 3 ENTER	Histogram data in block 0, and put results in block 3
COUNT 1 SPACE 100 ENTER	Perform steps between "LABEL 1" and here, 100 times
USER PROG 3010 ENTER	Enable maximization
END	

HISTO GRAM **F6**

F6

The gold F6 command calls a prewritten program if one has been placed under control of this key. The command structure is:

USER PROG | HISTO GRAM (F6) | ENTER

For information on how to write programs and how to place them under control of these keys, refer to the Soft Keys paragraph in this section.

POWER SPECT **Power Spectrum**

An averaged power spectrum is obtained by doing the following fundamental operation.

1. Taking N samples of the data (i.e., one record) and storing it in a data block.
2. Doing a Fourier transform on this data.
3. Conjugate multiplying the transform by itself or by another data block to form an auto or cross power spectrum.
4. Averaging the spectrum into an accumulator block.

The POWER SPECTRUM command performs steps 3 and 4 of the above process. The first two steps are done by other steps in the program.

Averaging of a power spectrum can be done by summing each new record into the sum of all past records. When summation averaging is used, the final sum must be divided by the number of averages used to obtain a calibrated result. During the summation, the intermediate sum grows and is not calibrated. If a summation average is stopped before the final step, it must be calibrated by dividing by the actual number of sum terms used. The summation average is described along with a sample program in Appendix C.

A more useful form of average is the stable average. The stable average is an algorithm which effectively divides each term of the summation by a calibrating constant. The stable average is always calibrated and does not grow. This allows a stable, non-growing display as well as the ability to stop the average at any point and have a calibrated power spectrum. The stable average removes the need to divide the average by the number of terms. The algorithm for the stable average is described, and a program for it is given, in Appendix C. The POWER SPECTrum key implements the conjugate multiply and stable averaging steps in one keystroke and at a much higher processing rate.

The POWER SPECTrum key can be used to implement a single channel power spectrum or a dual channel auto and cross spectrum average.

The command format for single channel power spectrum is:

POWER SPECT | n1 | ENTER

where:

n1 is the input channel. (i.e., the data block where the Fourier transform of each input record exists when the POWER SPECTRUM command is used.) Default value of n1 is 0.

The averaged power spectrum is formed in block n1 + 1.

The Fourier transform in block n1 is left unmodified.

The value of N (number of averages) used to calibrate the spectrum is taken from the next COUNT command. The COUNT command that follows the POWER SPECTRUM command should thus be the one that controls the spectrum averaging loop.

The averaged POWER SPECTRUM command leaves the results as follows:

Result	Location of Result
Fourier Transform of last record	Block n1
Averaged Power Spectrum, G_{xx}	Block n1 + 1

If two or more ADC channels are active on the input, the POWER SPECTrum key can be used to form the stable average of the input power spectrum, G_{xx}, the output power spectrum, G_{yy}, and the cross power spectrum G_{yx}. This is called a tri-spectrum ensemble average.

The command for the tri-spectrum ensemble average is:

[POWER SPECT] [n1] [SPACE] [2] [SPACE] [2] [ENTER]

where:

n1 is the block holding the Fourier transform of the input channel.

The first 2 indicates a dual channel input while the second 2 indicates a double precision cross power spectrum. Both 2's must be entered (no default allowed).

NOTE

The POWER SPECTRUM command MUST BE part of a program loop, ~~it~~ must be followed immediately by the loop-forming COUNT command (or a "SP WHAT?" message will result).

The results from this command are held in the following blocks.

Result	Location of Result
Input Auto Power, G_{xx}	Block n1 + 2
Output Auto Power, G_{yy}	Block n1 + 3
Real Part Cross Power, G_{yx} (Stored as double-precision)	Block n1 + 4
Imaginary Part Cross Power, G_{yx} (Stored as double-precision)	Block n1 + 5

Arithmetic operations may be done on the blocks holding the real and imaginary parts of the cross power spectrum just as they would be done on the cross power spectrum when both parts are in one block. The major advantage in using a double-precision cross power spectrum occurs in increased dynamic range during averaging. Once the spectrum, G_{yx}, has been accumulated, it can be converted to single-precision for polar coordinates or other operations. The following operations are used to make this conversion.

LOAD (n1 + 5)

Puts imaginary part of $G_{yx}2$ into block 0.
block 0.

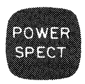
MULT 0 0 1

Multiplies by j. Makes real double precision block into single precision block with 0 real part and imaginary part equal to Im (G_{yx}).

ADD (n1 + 4)

Adds double precision real part of G_{yx} in block (n1 + 4) to complex single precision block 0. Results in complex single precision block containing G_{yx}.

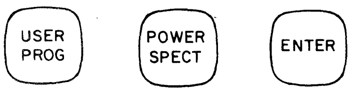
When the cross power spectrum is used to compute the transfer and coherence function, the TRANSfer FunCtioN key does this computation with maximum accuracy.



F5

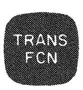
F5

The gold F5 command calls a prewritten program if one has been placed under control of this key. The command structure is:



F5

For information on how to write programs and how to place them under control of these keys, refer to the Soft Keys paragraph in this section.



Transfer Function

A transfer function is a mathematical description of a system, be that system a filter, a jet engine, a vibrating airplane wing, an organ in the human body, or whatever. It can be defined as:

$$\text{transfer function} = \frac{\text{Fourier transform of output}}{\text{Fourier transform of input}}$$

or equivalently,

$$\text{transfer function} = \frac{\text{average cross power spectrum of input and output}}{\text{average power spectrum of input}}$$

The coherence function measures the degree of causality between any two signals. It can therefore be used to check the validity of the transfer function. When a transfer function is computed, we may not be aware of extraneous inputs or whether or not the system is linear. Both of these factors would introduce error in the computed transfer function. The coherence function ranges between 0 and 1. Zero means no coherence between input or output, or in other words, extraneous inputs and/or the system is non-linear; 1 means complete coherence between input and output, or in other words only one input and a linear system.

The equation for calculating the coherence function is:

$$\gamma^2 = \frac{|\overline{G_{YX}}|^2}{\overline{G_{XX}} \cdot \overline{G_{YY}}}$$

where

$|\overline{G_{YX}}|^2$ = square of the magnitude of the cross spectrum

$\overline{G_{XX}}$ = input auto spectrum

$\overline{G_{YY}}$ = output auto spectrum

The TRANSFER FUNCTION command is used to compute the transfer and coherence functions with a minimum of user programming. It assumes dual channel operation, and that the auto and cross power spectra have previously been calculated using the POWER SPECTRUM command. (For a flow graph of transfer and coherence function, see Figure 3-13.) The TRANSFER FUNCTION command structure is:

[TRANS FCN] [n1] [SPACE] [2] [SPACE] [2] [ENTER]

where:

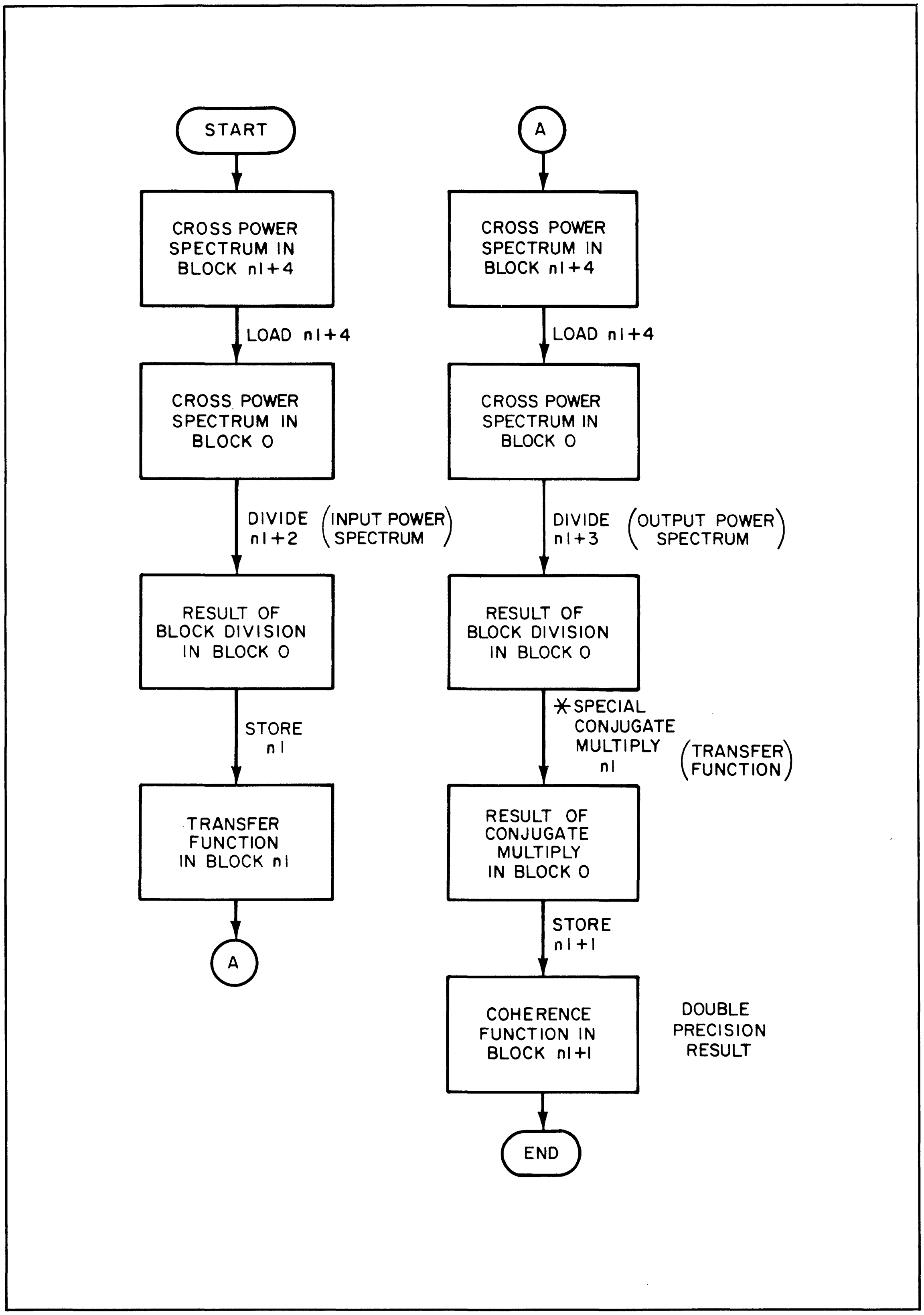
n1 is the first input data block. Default of n1 is not allowed.

n2 is equal to 2 and specifies dual channel input. Default of n2 is not allowed.

n3 is equal to 2 to specify double precision. Default of n3 is not allowed.

Location of Results	Result
Block n1	Transfer Function
Block n1 + 1	Coherence Function
Block n1 + 2	G_{xx}
Block n1 + 3	G_{yy}
Block n1 + 4	Real part of the cross power spectrum, G_{yx}, stored as a double precision block.
Block n1 + 5	Imaginary part of the cross power spectrum, G_{yx}, stored as a double precision block.

Figure 3-13. Flow Graph of Transfer and Coherence Computational Procedures



*Since the coherence function is completely real, a special conjugate multiply is performed with the result being a double precision block.

A program using the POWER SPECTrum and TRANSfer FunCtioN keys is:

List				Keys
L	0			LABEL
CL	2			CLEAR
CL	3			CLEAR
CL	4			CLEAR
CL	5			CLEAR
L	1			
RA	0	3		ANALG IN
H1	0			HANN
H1	1			
F	0	1		F
SP	0	2	2	POWER SPECT
#	1	10	0	COUNT
CH	0	2	2	TRANS FCN
.				END

The TRANSFER FUNCTION command performs an operation similar to the following program steps:

Block n1 : transfer function
n1 + 1: coherence function (γ^2)
n1 + 2: input auto power spectrum (G_{xx})
n1 + 3: output auto power spectrum (G_{yy})
n1 + 4: cross power spectrum, real part (Re (G_{yx}))
n1 + 5: cross power spectrum, imaginary part (Im (G_{yx}))

X<	(n1 + 4)	Load Re (G_{yx})
*		Compute Re^2 (G_{yx})
X>	(n1 + 1)	Store Re^2 (G_{yx}) in n1 + 1, temporarily
X<	(n1 + 5)	Load Im (G_{yx})
*		Compute Im^2 (G_{yx})
A+	(n1 + 1)	Compute $G_{yx}{}^2 = Re^2 (G_{yx}) + Im^2 (G_{yx})$
X>	(n1 + 1)	Store $G_{yx}{}^2$ in n1 + 1, temporarily
X<	(n1 + 3)	Load G_{yy}
*	(n1 + 2)	Computer G_{xx} G_{yy}
X	(n1 + 1)	Load $G_{yx}{}^2$, store $G_{xx}G_{yy}$ in n1 + 1 temporarily
÷	(n1 + 1)	Compute = γ^2
X>	(n1 + 1)	Store γ^2 in n1 + 1
X<	(n1 + 5)	Load Im (G_{yx})
*	0 0 1	Multiply Im (G_{yx}) by 0 + 1j and convert to single precision
A+	(n1 + 4)	Convert Re (G_{yx}) to single precision and reconstruct G_{yx} as a single precision block
÷	(n1 + 2)	Compute G_{yx}, the transfer function
X>	(n1)	Store transfer function in n1

NOTE

The precision of a transfer function or coherence calculation can be improved by invoking USER PROG 3012 to increase the number of scaledowns from 8 (normal) to 12. (See USER PROG 3012 description for details.) For additional examples of the use of the transfer function, refer to Appendix D.

F2

TRANS FCN

F2

The gold F2 command calls a prewritten program if one has been placed under control of this key. The command structure is:

F2

USER PROG | TRANS FCN | ENTER

For information on how to write programs and how to place them under control of these keys, refer to the Soft Keys paragraph in this section.

PROGRAMMING AND EDITING OPERATIONS

These paragraphs explain the PROGRAMMING keys, which allow you to set up your own Keyboard Programs. Some additional keys that pertain to the programming function are discussed at the end of the section. Also, there is an explanation of Variable Parameters, definitions, and examples of each Variable Parameter program and key.

What Is A Program

A program is a sequence of commands that the Fourier Analyzer will perform automatically. Power spectra and averaging functions are two prime examples of programming applications.

A typical printout of a program on the Terminal might appear as follows:

List				Meaning
L	1			Label 1
CL	1			Clear block 1
L	2			Label 2
RA	0	1		Analog into block 0, display block 1
A+	1			Add block 0 to block 1
X>	1			Store block 0 into block 1
#	2	100	0	Count (repeat steps) from label 2, 100 times. 0 steps had been executed when this listing was run.
•				End program

The following paragraphs give an outline of the kind of procedure you use to set up and run a program on the Fourier Analyzer. Details on each of the key commands are provided in this section. A simple, step-by-step demonstration of how to set up and run a program is given in Section 2. Examples of some typical programs can be found in Appendix C.

Setting Up A Program

There are 475 locations in the program memory for storing Keyboard Program steps. When you begin a program these locations may be empty, but most likely they will be partially filled with commands from a previous program. Therefore, setting up a program consists of editing the contents of the program memory. Therefore, every program entry must begin with an editing comand — that is, a command involving one of the PROGRAMMING keys on the Keyboard.

In general, you will enter a program with a REPLACE command, for example:

RPLAC | I | ENTER

This starts to enter the program at line 1 (i.e., at the beginning of the program memory). If this line is not specified (i.e., n1 defaulted), the entire program memory will be cleared. The BUSY light then comes on, indicating the machine is waiting for program commands. The BUSY light must be on when a program is being entered. Subsequent lines (also called steps) that you enter will automatically displace downward in memory any previous lines.

After, or during the setting up of a program, the steps can be listed via a LIST command. The line numbers are automatically assigned by the Fourier Analyzer: the number being the number of elements to the end of the previous line.

Running A Program

To start a program it is necessary to move the internal pointer to the starting point. (This pointer moves down the lines as the program runs.) The starting point may be either a label or a line number. To start a program at a given line number, use the POINT command to set the pointer to a given line number, then press the CONTINUE key. The program then starts running at the line specified, displays the result and stops. To start the program at a given label number, jump to the desired label using the JUMP command; the program runs through from that label, displays results and stops.

The program may be stopped at any time by pressing the STOP key and restarted by pressing the CONTINUE key. To find out where the pointer is after the program is stopped, the POINT key is used; the line number is then printed out on the Terminal. This command is useful if a WHAT? signal should apear during the running of a program. The CONTINUE key will restart the program. The pointer will be set to line 0 when any of the following commands are executed: REPLACE, DELETE, and INSERT.

RPLAC Replace

This command replaces a single line or a range of lines with another line or range of lines. It is the command most often used to write a new program, because it automatically displaces old program steps as the new ones are entered.

The first line to be replaced (for example, line 1 when beginning a new program) is given in this command; the end of the replacement sequence is indicated by a TERM ENTER command.

All lines beyond the replacement range are moved up or down in number, according to whether more or less elements were replaced; in other words, the program memory recompacts itself for most efficient use of memory space. The pointer is automatically set to 1.

The command form is:

RPLAC LINE n1 SPACE LINE n2 ENTER

where:

n1 is the first line number to be replaced. If n1 is defaulted the entire program memory would be cleared.

n2 is the final line number to be replaced. Default value = n1, in which case, only line n1 will be replaced.

If there is a program in memory and nothing beyond it, and you wish to enter another program immediately after it, the REPLACE command must call for the replacement of the last line in the existing program and then this line must be re-entered, as is, and the new program entered following it. The reason for this is that the automatic restacking feature of the REPLACE command will not operate on a blank line (as would be the case if the command were to replace from line 14 on in the following example). The command in that case would go back until it found a full line (line 13) and replace that. Thus, the last line of the previous program must be the one replaced, and then re-entered prior to entering the new program.

Example 1: We will first look at a listing of a sample program. This listing was obtained by the command:

LIST ENTER

The listing is as follows (line numbers on left are automatically assigned):

	List		Meaning
1	RA	0	Analog in block 0, display block 1
6	F	0	Fourier transform block 0
10	*-		Conjugate multiply block 0
13	A+	1	Add block 1 to block 0
17	.		End program

Now it is desired to replace line 6 with commands to perform an interval-centered Hanning function in block 0; do a Fourier transform on the same block, then clear block 0, channel 0. The command sequence to accomplish all this is as follows:

RPLAC 6 ENTER

(Note that the Fourier Analyzer now goes to BUSY state.)

HANN ENTER
F ENTER
CLEAR 0 SPACE 0 ENTER
TERM ENTER

(Note that the Fourier Analyzer now goes to the READY state.)

Now call for a printout of the entire program via the command:

LIST ENTER

and the resulting printout would be:

	List			Keys
1	RA	0	1	
6	H1			(HANN)
9	F			
12	CL	0	0	(CLEAR)
17	*-			
20	A+	1		
24	.			

Example 2: Now, in the preceding program it is desired to replace lines 12 through 17 with a single polar coordinate command. The sequence of commands is:

RPLAC 12 SPACE 17 ENTER

(BUSY light now comes on.)

POLAR ENTER
TERM ENTER

Now a listing, obtained by pressing LIST ENTER, would look as follows:

```
1     RA   0    1
6     H1
9     F
12    TP                  (POLAR)
15    A+   1
19    .
```

INSRT Insert

This command permits the addition of any number of program steps between any two given lines in a program, or following an END command. The pointer is automatically set to line 1. The command form is:

INSRT LINE n1 ENTER

where:

n1 is the number of the line after which the new material will be inserted.

While the additional material is being inserted the BUSY light will be on. After the material is in, a TERM ENTER command must be given. The Fourier Analyzer then goes to READY. Program steps beyond those inserted are shifted farther down in memory.

DELET Delete

This command is used to delete one step or a range of steps from a program. All lines below the delete move up to recompact the program. The pointer is automatically set to line 1. The command form is:

DELET LINE n1 SPACE LINE n2 ENTER

where:

n1 is the number of the first line to be deleted. If n1 is defaulted the entire program memory will be cleared.

n2 is the number of the last line to be deleted. Default value = n1 or in other words, if n2 is defaulted, then only line n1 will be deleted.

No TERM ENTER command is needed after the DELETe command.

NOTE

It is neither necessary nor desirable to delete all elements from program memory before entering a new program. See REPLACE command.

LIST

List

This command causes any program, or any part of a program, to be displayed on the Terminal.

To list an entire program from line 1, use the command form:

[LIST] [ENTER]

The listing will proceed to the first END command, or if there is none, to the first blank line, or if the entire program memory is full, to the last line (line 475).

To list a single line, use the command form:

[LIST] [LINE n1] [ENTER]

where:

n1 is the number of the line to be listed. No default allowed.

To list a range of lines, use the command form:

[LIST] [LINE n1] [SPACE] [LINE n2] [ENTER]

where:

n1 is the first line number to be listed. No default allowed.

n2 is the last line number to be listed. No default allowed.

To list from a certain line to the end of a program, use the command form:

[LIST] [LINE n1] [SPACE] [4] [7] [5] [ENTER]

where:

n1 is the first line to be listed. No default allowed.

475 is the number of the last line in program memory. The listing will proceed to the next END statement and stop there, or, failing such a statement, to the first blank line, or failing that, to the last line — i.e., line 475.

LIST Example 1: It is desired to list all steps in a program starting at line 10. The command is:

[LIST] [1] [0] [SPACE] [4] [7] [5] [ENTER]

LIST Example 2: It is desired to list lines 10 through 13 of a program. The command is:

[LIST] [1] [0] [SPACE] [1] [3] [ENTER]

LIST — List (Variable Parameter)

LIST

The LIST command lists the values of the specified Variable Parameters. The command structure is:

USER PROG   LIST   n1   SPACE   n2   ENTER

LIST

where:

n1 is the first Variable Parameter value to be listed.

n2 is the last Variable Parameter value to be listed. Default value of n2 is n1.

For additional information refer to the Variable Parameter paragraphs (specifically Y 1806) in this section.

POINT — Pointer

The POINT command sets the pointer to any line number (not label number) in the program. Then, when the CONTINUE key is pressed, the program will start at that line number. (Don't press PROGRAMMING keys RPLAC, INSRT, or DELET meanwhile since these return the pointer to line 1). The purpose of this key is to start the program at any point not specified by a label. Line numbers are obtained by listing the program via the LIST command. The command structure is:

POINT   LINE n1   ENTER

where:

n1 is the line number at which the program is to start.

If n1 is defaulted, the current pointer location is printed on the Terminal. This command may be used, for example, after a "WHAT?" signal in order to determine which line produced that signal and caused the program to stop. This printout looks like

P n1

where:

n1 is the pointer location, i.e., the line number.

TERM — Terminate

This command is used to indicate to the Fourier Analyzer the end of a replace or insert editing function, and also to indicate the end of a Keyboard data entry. The system returns to the READY mode. The command is:

TERM   ENTER

Label

This command labels a line in the program irrespective of the line number. The line can then be accessed via the JUMP and COUNT commands. The form of the command is:

[LABEL n1] [ENTER]

where:

n1 is an integer, -32,768 through +32,767. No default allowed.

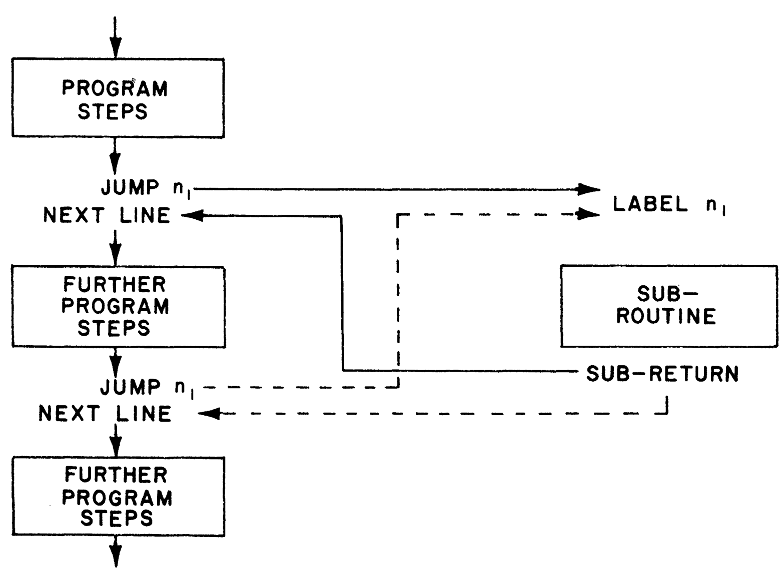
The LABEL command must be executed as part of a program. It has no meaning as an isolated command.

Jump

When a program reaches this command, it immediately goes to a label specified by n1, and proceeds from that point. This command can also be used to start a program running, simply by specifying the label (n1) at the beginning of the program. The command for this type of operation is:

[JUMP] [LABEL n1] [ENTER]

The JUMP command allows a short sequence of steps (i.e., a subroutine) to be executed from any point in a program. The subroutine must have a label n1 to identify its beginning, and a SUBRETURN command to identify its conclusion. When a program reaches a JUMP command, it goes from that line to the first line in the subroutine, executes all the subroutine steps, then returns to the line following the one from which it entered the subroutine. The following diagram illustrates this operation:



Stack-to-Stack Jumps

MASS STORE File 3 is the file containing keyboard stacks (Keyboard Programs). The system allows a JUMP command to jump to a label in a stack on the Disc. The stack being jumped out of is saved on the Disc, the new stack is read into core, and the jump is executed. In this way, the second stack can, when done, subreturn back to the first stack. In addition, jumps and subreturns may be nested (i.e., jump, jump, subreturn, subreturn) as many as 10 levels deep; each of these may involve different stacks on the Disc. Since each stack being jumped out of must be saved, up to 10 stacks may have to be saved by the software. The last 10 records in File 3 are saved for this purpose.

IMPORTANT: Do not use the last 10 records in File 3 to save personal keyboard stacks.

The complete command structure for the JUMP command is:

JUMP | LABEL n1 | SPACE | n2 | SPACE | n3

where:

n1 is a label integer, -32,768 to +32,767. No default allowed.

Parameters n2 and n3 determine whether or not the JUMP command will be included in the subroutine nesting, and if the jump is to a stack on the Disc, whether it is an "absolute" or "relative" jump. These parameters are interpreted as follows:

If n2 and n3 are defaulted, the program will jump to label n1 in the present stack in core. The jump is included in the subroutine nesting.

If a negative n2 is entered and n3 is defaulted, then the program jumps to label n1 in the present stack and the jump is not included in the subroutine nesting.

If a positive n2 is entered, then n2 is the record number in File 3 that the command is to jump to. Thus,

JUMP 3 SPACE 21 ENTER

will save the present stack, read record 21 from File 3, and jump to label 3 in that stack. This jump is included in the subroutine nesting.

If n3 = 0, this command will act as if n3 were defaulted, and n2 were positive (same as previous command).

If n3 $\neq$ 0, then n2 may be greater or less than 0, since it is now not an absolute record in File 3, but a relative positioning factor. The stack jumped to will be CURRENT STACK # + n2 and the jump will be included in the subroutine nesting.

Therefore, if you enter

JUMP SPACE 21 SPACE -1 SPACE 1 ENTER

from the Keyboard, it would not make sense; but if the above command were in the stack in record 23 of File 3, and that stack were executed, then the above jump would save the present stack, read down record 22, and jump to label 21. This jump is included in the subroutine nesting.

Examples:

JUMP 100 ENTER

will jump to label 100 in the present stack in core. It is included in the subroutine nesting; therefore, when a SUBRETURN command is encountered, the command after this JUMP command would be the next step executed by the system. (If the command was executed from the Keyboard, however, a "WHAT?" message would result since there is no step to subreturn to.)

JUMP 100 SPACE -1 ENTER

will jump to label 100 in the present stack in core. This is not included in the subroutine nesting.

JUMP 100 SPACE 1 ENTER

will jump to label 100 in record 1 in File 3. It is included in the subroutine nesting.

JUMP 100 SPACE -1 SPACE 0 ENTER

is illegal. Since n3=0 was entered, the software thinks –1 is a record number in File 3. An MS WHAT? error message will result.

JUMP 100 SPACE 1 SPACE 0 ENTER

will jump to label 100 in record 1 in File 3. This is included in the subroutine nesting.

The next two examples must be commands in keyboard program stacks on the Disc; they should not be executed from the Keyboard.

JUMP 100 SPACE -1 SPACE 1 ENTER

If this command were in a stack in, say, record 30 of File 3, the software would save the present stack, read down record 29 (which is 30 –1), and jump to label 100. The jump is included in the subroutine nesting.

JUMP 100 SPACE 1 SPACE 1 ENTER

If this were in a stack in, say, record 30, then it would save the present stack, read down record 31 from File 3, and jump to label 100. The jump is included in the subroutine nesting.

The example below illustrates how the various JUMP commands may be used. The two Keyboard Programs shown are assumed to be in stacks 10 and 11 in File 3 on the Disc. The programs perform 100 single sweep displays of various data blocks. The arrows signify where the point of execution is being changed either by a JUMP or a SUBRETURN command. The final result of these two programs is that data blocks 0 through 4 are displayed in sequential order, ending with a display of block 4.

To invoke the programs, enter:

JUMP 10 SPACE 10 ENTER

Example of JUMP Commands

```
J   10   10
```

Keyboard Stack 10

```
L   10
D   0    4
#   10   100   0
J   11   1     1
L   20
D   4    4
#   20   100   0
.
```

Keyboard Stack 11

```
L   11
D   1    4
#   11   100   0
J   21
J   31   -1
L   21
D   2    4
#   21   100   0
<
L   31
D   3    4
#   31   100   0
<
.
```

1. 'J 10 10' jumps to label 10 in stack 10 in File 3. The jump is saved in the subroutine nesting.

2. 'J 11 1 1' performs a "relative" jump to label 11 in stack 11 (= current stack + 1). Stack 11 is now automatically read in. The jump is saved in the subroutine nesting.

3. 'J 21' jumps to label 21 in the current stack (stack 11). The jump is saved in the subroutine nesting.

4. '<' returns to the line following 'J 21' (the last jump saved in the nesting).

5. 'J 31 -1' jumps to label 31 in the current stack (stack 11). The jump is not saved in the subroutine nesting.

6. '<' returns to the line following 'J 11 1 1' in stack 10 (stack 10 is now automatically read in).

The program now ends. As there was not a third SUBRETURN command encountered, the 'J 10 10' is still saved in the nesting, but this is of no consequence in this operation.

Count

This command causes a portion of the program whose starting point is label n1, to be repeated n2 times. In other words, this command is used to form a loop. The command form is:

COUNT | FROM LABEL nl | SPACE | n2 TIMES | ENTER

where:

n1 is the label of the starting point (must be -32,768 through +32,767). No default allowed.

n2 is the number of times this portion of the program is to be repeated. n2 must be an integer from 1 through +32,767. No default allowed.

When a program that includes a COUNT command is listed on the Terminal, the COUNT line will have the following form:

n1 n2 n3

where:

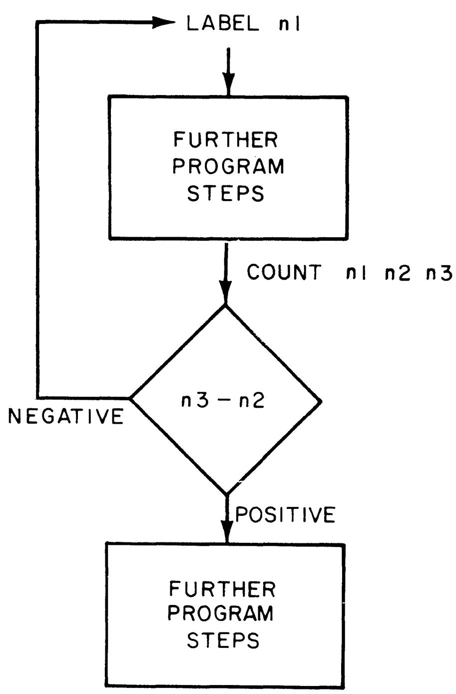
n1 is the label (an integer -32,768 through +32,767).

n2 is the number of times the portion of the program is to be repeated.

n3 is the number of times the loop had been repeated at the time the LIST command was given. In most cases this will be 0, i.e., one does not normally stop a program during execution to ask for a listing. But if this were done, then n3 would state the number of loops that had been completed at the time the LIST command was given.

Note: the COUNT does not automatically reset to 0 after being stopped by a listing, as above. To reset the count to 0, press RESTART or *re-enter the COUNT command* (using a REPLACE command first), then start the program again from the top.

The COUNT command must be executed as part of a program. It has no meaning as an isolated command. A flow graph of a COUNT command is shown as follows:



EXAMPLE: If you wanted to repeat the steps of a program from label 3, 100 times, the following would be placed at the bottom of the list of steps, immediately prior to the END command:

COUNT 3 SPACE 100 ENTER

SKIP Skip

This command allows a skip of the next instruction in the program memory if any of the data points in block n1, between channels n2 and n3, are negative. The command form is:

SKIP | BLOCK n1 | SPACE | CHANNEL n2 | SPACE | CHANNEL n3 | ENTER

where:

n1 is the data block examined; default value = block 0.

n2 is the first channel examined; default value = channel 0.

n3 is the last channel examined; default value = channel n2.

If both n2 and n3 are defaulted, the entire block is examined.

The SKIP command is only executed as part of a program. If you attempt to execute the command directly from the Keyboard, a WHAT? signal will result.

EXAMPLE: It is desired to perform an auto spectrum summation on data blocks which have positive dc values.

	List		Meaning
L	1		Label 1
CL	1		Clear block 1
L	2		Label 2
RA	0	1	Analog into block 0, display block 1
F	0		Fourier transform of block 0
1F	0	0	Skip next program step if channel 0 of block 0 is negative
J	3		Jump to label 3
J	2		Jump to label 2
L	3		Label 3
*			Self-conjugate multiply
A+	1		Add block 1 to block 0
X>	1		Store block 0 in block 1
#	2	100	Count 100 repetitions from label 2
.			End program

SKIP

SKIP Skip (Variable Parameter)

The gold SKIP commands tests Variable Parameter values to determine whether or not to skip keyboard instructions. This command can only be used in Keyboard Programs. The command structure is:

[USER PROG] SKIP [SKIP] [ ] [n1] [SPACE] [n2] [SPACE] [n3] [SPACE] [n4] [ENTER]

where:

The relationship of the values of Variable Parameter n1 and parameter n2 is checked. The relationship that is checked is whether n1 is greater-than n2, greater-than-or-equal, equal-to, less-than-or-equal, or less-than, depending on the value of n4. The program then skips n3 keyboard instructions (in either direction) if the test is true. n4 denotes the following type of test:

n4 = 2 greater than
n4 = 1 greater than or equal
n4 = 0 equal to
n4 = -1 less than or equal
n4 = -2 less than

The default value of n4 is -2. The default value of n3 is 1. The default value of n2 is 0.

For additional information, refer to the Variable Parameter paragraphs (specifically Y 1805) in this section.

SUB RTRN Subroutine Return

This command must be placed at the conclusion of any subroutine in order to return the program control to where it entered the subroutine, i.e., the step immediately following JUMP. A diagram of the subroutine format is shown under the JUMP command. The structure of the SUBRETURN command is:

[SUB RTRN] [ENTER]

END End

This command is used to indicate the conclusion of a program. It should always be used in case there are residual steps from earlier programs still in memory following the new program. The command structure is:

[END] [ENTER]

The END command must be executed as part of a program. It has no meaning as an isolated command.

BLOCK SIZE Block Size

This command establishes the number of channels in each data block. Permissible block sizes are shown above the indicator lights along the top of the Keyboard. The maximum blocksize permitted for most operations is 4096 words, although data block requirements may require that a lower blocksize be used in some instances.

If a blocksize that is not allowed is commanded, the system will give a WHAT? signal. After a BLOCK SIZE command is executed, block 0 is always displayed. The command structure is:

BLOCK SIZE | BLOCK n1 | ENTER

where:

n1 is the number of channels (e.g., 64, 128, 256, 512, etc.).

When block size is decreased, i.e., original block broken into smaller blocks, the new blocks will assume the same data qualifiers (scale factor, block calibrator, frequency code) as the original blocks. However, a coded BLOCK SIZE command can be given which prevents the new blocks from assuming the same data qualifiers. The structure for this command is:

BLOCK SIZE | BLOCK n1 | SPACE | 0 | ENTER

where:

n1 is the number of channels.

0 is a code number for the command.

BS BLOCK SIZE Block Size (Variable Parameter)

The Variable Parameter BS command sets Variable Parameter n1 to the current system block size. The command structure is:

USER PROG | BS BLOCK SIZE | | n1 | ENTER

where:

Variable Parameter n1 is set to the current system block size.

For additional information, refer to the Variable Parameter paragraphs (specifically Y 1807) in this section.

User Program

The User Program feature allows you to write your own relocatable programs in FORTRAN or HP Assembly Language, merge them with the system and execute them using the USER PROGram key. The format of the User Program command is:

USER PROG | n | SPACE | n1 | SPACE | n2 | SPACE | n3 | SPACE | n4 | SPACE | n5 | SPACE | n6 | ENTER

where:

n is a positive number up to four digits (0 through 9999).

n1 through n6 are optional program parameters that can have values from –32,768 to +32,767.

USER PROG User Program (Gold Key Capability)

The Gold Key capability allows you to replace specific User Program numbers with an appropriate mnemonic by pushing a key on the Keyboard. This capability is enabled by pressing the gold-colored USER PROGram key. For example, to set a Variable Parameter to a specific value, press the following keys:

USER PROG | ← (SET) | n1 | SPACE | n2 | SPACE | n3 | ENTER

The same result can be achieved by pressing

USER PROG | 1 | 8 | 0 | 0 | SPACE | n1 | SPACE | n2 | SPACE | n3 | ENTER

Table 3-1 shows the Gold Key functions that have associated User Programs. Note that not all Gold Key functions have related User Programs, and that there are additional User Programs available that do not have related Gold Key functions (e.g., the graphics User Programs covered in Section 6). For a complete list of User Programs, refer to Section 7.

Table 3-1. USER PROG (Gold) Key Program Functions

USER PROG (Gold) Key	User Program
ADD	Y 1801 — add VP
BS	Y 1807 — set VP to block size
DIV	Y 1804 — divide VP
EXP	Y 1827 — exponentiation
F1 thru F6	Prewritten programs placed under control of these keys.
GET	Y 1821 — loads data block information
INPUT	Y 1808 — inputs VP values
LIST	Y 1806 — list VP values
LOG	Y 3022 — changes vertical scale calibration
MULT	Y 1803 — multiply VP
OUTPUT	Y 1809 — prints VP value
PLOT	Y 5800 — plot block
PUT	Y 1822 — stores block information
QUALS	Y 1823 — gets or puts data block qualifiers
SKIP	Y 1805 — test VP value for skip
SUB	Y 1802 — subtract VP
TEXT	Y 5819 — output text message

CURSR Cursor

The CURSOR command enables an x-cursor to appear on the display screen (the display should be in the CONT mode). This is useful for obtaining information about the points on the display; for example, the channel number of the point, and the time (or frequency) and amplitude (or phase) it represents. The command structure is:

CURSR n1 SPACE n2 ENTER

where:

n1 is the starting number of a 3-parameter array of Variable Parameters (preferably of COMPLEX type) where [illegible] time (frequency) or amplitude (phase) are stored when the cursor value is displayed.

If n1 equals –1, the cursor is turned off.

If n1 is defaulted, no values are stored into any Variable Parameters.

n2 is the block number on which the cursor will be displayed. If n2 is defaulted, this cursor will be put on the block that is currently on display.

There are two different modes of operation for the cursor depending upon whether it is invoked from the Keyboard or from within a Keyboard Program.

From the Keyboard:

When the cursor is used from the Keyboard, the cursor will come on the display and the system will go to READY. The cursor may now be controlled using the Processor switch register as follows:

Cursor Switch	Result
BUTTON →	If pressed once, the cursor will move one channel to the right of its current position. If held, the cursor will sweep right.
←	If pressed once, the cursor will move one channel to the left of its current position. If held, the cursor will sweep left.
VALUE	If pressed, the following display appears on the screen: CHANNEL = XXXX TIME = XXXXX AMPLITUDE = XXXXXX These three values are placed into the specified Variable Parameters when the display occurs, if this was specified in the command entry.
ON	Pressing the upper edge of this switch places the cursor bar on the display. Pressing the lower edge of the switch removes the cursor from the display (cursor is still enabled) so as to restore the high-speed display. (The sweep of the display is somewhat slower with the cursor on the display.) The cursor may be switched on and off the display by pressing alternate sides of this switch. Note: To turn off the cursor permanently, the command CURSR - 1 ENTER must be entered. The ON switch only controls the cursor on the display once the cursor has been turned on.
EXPAND	Pressing the upper edge of this switch expands the display around the cursor, making the points in the immediate vicinity of the cursor more visible. Pressing the lower edge of this switch restores the display to its original unexpanded state. If the display is expanded and the → or ← switches are used to move the cursor to either edge of the display, the display will pan past the cursor.

The cursor is turned off from the Keyboard by entering

CURSR - 1 ENTER

From a Keyboard Program:

When the cursor is turned on from a Keyboard Program, the cursor appears on the display and the Processor switches control it as previously described. *The keyboard program will now wait for the VALUE key to be pressed before further commands are executed.*

EXAMPLE: Consider the following program:

L	0		Label 0
CL			Clear block 0
/.	2000		Turn on cursor — put values into VP 2000, 2001, 2002
Y	/L	2002	List VP 2002
/.	-1		Turn off cursor
.			End

When this program is executed, the system will wait for the VALUE switch to be pressed. When it is, the following display will occur:

CHANNEL = XXXX
TIME = XXXXX SEC
AMPLITUDE = .0000
2002D = .0000

When obtaining multiple points with the cursor in a Keyboard Program, it is sometimes useful to have the cursor remain at its previous location rather than reset to channel 0 as it usually does. This is possible to do merely by using the CURSOR command as many times as necessary without turning it off. The following examples will help to clarify this operation:

Example 1:

L	0	Label 0
/.	2000	Turn on cursor. Put values into VP 2000 to 2002.
/.	2003	Turn on cursor. Put values into VP 2003 to 2005.
/.	2006	Turn on cursor. Put values into VP 2006 to 2008.
/.	-1	Turn off cursor.
		End

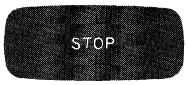
Example 2:

L	0	Label 0
/.	2000	Turn on cursor. Put values into VP 2000 to 2002.
/.	-1	Turn off cursor
/.	2003	Turn on cursor. Put values in VP 2003 to 2005.
/.	-1	Turn off cursor
/.	2006	Turn on cursor. Put values into VP 2006 to 2008.
/.	-1	Turn off cursor
.		End

Both of these programs will wait for the VALUE switch to be pressed three times, and will fill Variable Parameters 2000 to 2008 with the data for three points in the block on display. However, the cursor will not be reset to channel 0 between points in the first program, whereas the second program will reset the cursor to channel 0 for each of the points entered.

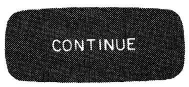
Cursor restrictions:

1. The cursor printout is not valid for horizontal axis display of histogram data.
2. Display ORIGIN switch must be in LEFT or LOG positions, not in the CENTER position, for proper cursor printout.
3. Cursor and the 54420A DAC cannot operate simultaneously.
4. Cursor and the 54470A Fourier Pre-processor cannot operate simultaneously.



Stop

and



Continue

When a program is running, pressing the STOP key will cause the program to stop after it completes the present program line. The STOP command executes automatically, no ENTER keystroke is needed. It differs from the RESTART command in that it always permits the operation to be completed. RESTART interrupts in the middle of an operation, and may therefore leave meaningless data in memory. Neither STOP nor RESTART, however, have any effect on the Fourier operating system; that is, they do not turn off the Fourier Analyzer.

Pressing the CONTINUE key (no ENTER keystroke required) causes the program to resume from the line at which the pointer is set. (Not the following line, but *that* line.)

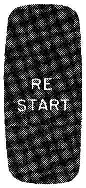
To find out at which line the pointer is set, use the POINT command:

POINT ENTER

The STOP and CONTINUE keys operate in conjunction with the STEP-RUN switch on the Keyboard. When in the STEP position, the program proceeds one step at a time; when in RUN, it automatically proceeds through to the end.

STEP-RUN SWITCH

In the STEP position, this switch causes the program to proceed one step at a time. The step is completed, the appropriate display of results is given, and then the program stops. To make the program continue, the CONTINUE key must be pressed. The next step is then performed, results are displayed, and the program again stops. Thus, the STEP position is useful in debugging a program. In the RUN position, the switch causes the program to continue through all its steps automatically.



Restart

Pushing RESTART key (no ENTER necessary) restarts the entire Fourier program. If it is pushed while a program is running, it interrupts the completion of the program. This may leave meaningless data in memory.

When the Fourier system is initially read from the Disc and started, the following sequence of events occurs:

1. Data space in the Processor is set up and allocated. A message is printed indicating the amount of space available. Blocksize is set to maximum allowable.

2. A brief check of the Fourier Analyzer microcode is made.

3. Software microcode interface is initialized.

4. All currently resident drivers are initialized.

5. Processor "booster" assembly is deactivated (this is the normal mode of Processor operation).

6. The overlay file on the Disc is read and initialized for "invisible" overlay swapping.

7. User Programs 3010, 3011, 3012 and 3016 are set to their normal (default) states:

   Y 3010: Maximization is enabled

   Y 3011: Double precision self-conjugate multiplies are enabled

   Y 3012: Number of division scaledowns is set to 8

   Y 3016: Stable averaging is enabled

8. Loop counters in all COUNT commands are set to zero.

9. Interrupt system is enabled.

10. Processor S-Register is cleared.

11. Block is displayed and system waits for a Keyboard command to be entered.

NOTE

All of the above steps, except 1, 2, and 6, occur whenever RESTART is pressed; the others occur only when the system is read from the Disc and started at P=2.

VARIABLE PARAMETERS

Variable Parameters (VP) are locations im memory containing values that can be used as parameters in Fourier keyboard instructions. Using specific User Programs, the values of these locations can be set, added, subtracted, printed out, entered in via the Terminal, etc., while the system is in the READY mode, or while a Keyboard Program is being executed. This allows for a great deal of flexibility at the keyboard programming level. Using Variable Parameters (which may substitute for any parameter except the first two ASCII characters in the Fourier commands), a Keyboard command can consecutively be called in different ways, JUMP command can jump to any label in the Keyboard Program (simplifying the response to typed-in decisions) and difficult-to-program problems can be simplified.

Types of Variable Parameters

There are three types of Variable Parameters: Integer (range = -32767 to +32767), Floating Point (range = -10^{38} to $+10^{38}$), and Complex (real and imaginary Floating Point). Variable Parameters are referred to by number (for identification, this is not their value), and the number used indicates the VP type being used. For example:

VP No.	Type
0–999	Integer
2000–2999	Floating Point
3000–3999	Complex

Thus, VP 10 is an integer, but VP 2010 is a floating point number, and VP 3010 is a complex floating point number. There are not actually 1000 parameters of each type in the system; the actual amount of each type is decided by a table in core. Operations between mixed types are performed in floating point, then converted as necessary before storing. In the system, the following Variable Parameters are available.

Variable Parameter Configuration

VP No.	Type
0–100	Integer (in common)
2000–2025	Floating Point (in common)
3000–3010	Data (complex) (in common)
101–150	Integer (not in common)
2026–2050	Floating Point (not in common)
3011–3024	Data (complex) (not in common)
2900–2921	Floating Point equivalent to Complex 3000–3010 (in common)
2922–2949	Floating Point equivalent to Complex 3011–3024 (not in common)

The above table shows the order in which the available system Variable Parameters are stored. If the use of any other VP is attempted, an error will result. In addition, some of the floating point Variable Parameters are equivalent to the complex Variable Parameters. For example, VP 2900 and 2901 are equivalent to VP 3000 (complex variables contain two floating point numbers) VP 2902 and 2903 are equivalent to VP 3001, and so on up to VP 2948 and 2949 being equivalent to VP 3024. This equivalency means that operations may be executed on just the real or imaginary part of a complex Variable Parameter by using the floating point Variable Parameter equivalent to the desired part (the lower-numbered floating point parameter is always the real part, while the higher-numbered parameter is the imaginary part). As shown by the table, some of the Variable Parameters are in the common area that may be stored on the Disc. This feature is discussed later in this section.

Use of Variable Parameters

To use the value contained in a specific VP, you must enter the VP number in place of the parameter in a Keyboard Program, and follow the number immediately with the letter D. Thus, VP 6 is entered 6D, distinguishing it from a regular number. To keep from switching to the Terminal and back to the Keyboard each time a VP is entered, you can simply press the DiSPLaY key to enter the D. For example:

F 1 DSPLY ENTER

takes the Fourier transform of the block whose number is in VP 1; if VP 1 = 6, then the transform of block 6 is taken. Whenever a number is followed by a D, it denotes a VP. Thus, 27D means VP 27.

If a negative VP number is used (e.g. –6D), it means take the value of 6D and use that value as the address of the VP to be used.

Example: If 6D = 10
10D = 0

then F –6D
means F 10D
which means F 0

This is a means of indirectly addressing a VP by using a second VP. An actual example of its use would be as follows: let's assume that 50D through 53D contain calibration factors for ADC channels 0 through 3 (or A through D). To multiply data block 0 by the proper calibration factor, set 40D to the value of 50 + (ADC channel number), and execute:

MULT 0 SPACE –40D ENTER

Thus, if ADC channel 1 were being used, 40D = 50 + 1 = 51

and

MULT 0 SPACE –40D ENTER

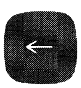
means

MULT 0 SPACE 51D ENTER

which is what is desired.

Variable Parameter User Programs

To manipulate Variable Parameters and use them to perform system functions, a number of User Programs are provided in the system. The Variable Parameter User Programs are listed below in numerical order. The "Y" is equivalent to the USER PROG key on the Keyboard. If a Keyboard key is shown in the margin, you can press this key instead of entering the number. (For example, press SET instead of entering 1800 SPACE. Note that a SPACE is not required between the SET mnemonic and parameter n1. This is true for all Gold key functions. However, SPACE entry is required when the User Program number, e.g., 1800, is used.)


SET

1. **Y 1800 sets a VP**

USER PROG SET n1 SPACE n2 SPACE n3 ENTER

VP n1 is set to the value n2 $\times$ 10^{n3} (Default of n3 = 0)

Examples:

USER PROG SET 23 SPACE 101 ENTER

23D is set to 101.

USER PROG SET 53 SPACE 101D ENTER

53D is set to the value in VP 101.

USER PROG SET 2001 SPACE 2 SPACE 3 ENTER

2000D is set to $2 \times 10^3 = 200$

To "SET" a complex number, the floating point equivalent must be used. For example, to set VP 3000 equal to 3.14 real and 5.12 imaginary, enter:

Y SET 2900 SPACE 314 SPACE –2 ENTER
Y SET 2901 SPACE 512 SPACE –2 ENTER

VARIABLE PARAMETER QUICK REFERENCE

Gold Key	User Program
ADD	1801 — add VP
BS	1807 — set VP to blocksize
DIV	1804 — divide VP
EXP	1827 — exponentiation
GET	1821 — GET
INPUT	1808 — read into VP
LIST	1806 — list VP
MULT	1803 — multiply VP
OUTPUT	1809 — type VP value
PLOT	5800 — plot block
PUT	1822 — PUT
QUALS	1823 — GETQ
SET	1800 — set VP
SKIP	1805 — test VP value
SUB	1802 — subtract VP
TEXT	5819 — output text message

VARIABLE PARAMETER QUICK REFERENCE

Gold Key	User Program
USER PROGRAM	Gold "shift" key
BS	1807 — set VP to blocksize
PLOT	5800 — plot block
INPUT	1808 — read into VP
LIST	1806 — list VP
TEXT	5819 — output text message
OUTPUT	1809 — type VP value
SKIP	1805 — test VP value
QUALS	1823 — GETQ
GET	1821 — GET
PUT	1822 — PUT
EXP	1827 — exponentiation
MULT	1803 — multiply VP
ADD	1801 — add VP
SET	1800 — set VP
DIV	1804 — divide VP
SUB	1802 — subtract VP

ADD

2. **Y 1801 adds values**

USER PROG ADD n1 SPACE n2 SPACE n3 ENTER

VP n1 is set to the sum of the values of n2 and n3. Default value of n2 is n1; n3 is 1.

Examples:

USER PROG ADD 19 SPACE 10 SPACE 23 ENTER

19D is set to the value 33.

USER PROG ADD 19 SPACE 10D ENTER

19D is set to the sum of the value of 10D plus 1.

3. **Y 1802 subtracts values**

SUB

USER PROG SUB n1 SPACE n2 SPACE n3 ENTER

VP n1 is set to the difference between the values n2 and n3. Default value of n2 is n1; n3 is 1.

Examples:

USER PROG SUB 8 SPACE 26 SPACE 53 ENTER

8D is set to -27.

USER PROG SUB 8 SPACE 26D ENTER

MULT

8D is set to the value of 26D minus 1.

4. **Y 1803 multiplies values**

USER PROG MULT n1 SPACE n2 SPACE n3 ENTER

VP n1 is set To the product of the values n2 and n3. Default value of n2 is n1; n3 is 2.

Examples:

USER PROG MULT 6 SPACE 9 ENTER

6D is set to 18.

USER PROG MULT 6D SPACE 9 SPACE 10 ENTER

The VP whose number is the value of 6D is set to 90 (This is an example of indirectly addressing a VP by using a second User Program; i.e., if 6D = 11, then 11D = 90.)

5. **Y 1804 divides values**

DIV

USER PROG DIV n1 SPACE n2 SPACE n3 ENTER

VP n1 is set to the value of n2 divided by n3. Default value of n2 is n1; n3 is 2.

Example:

USER PROG DIV 26 SPACE 10 SPACE 5 ENTER

26D is set to the value 2.

SKIP

6. **Y 1805 tests VP values**

USER PROG SKIP n1 SPACE n2 SPACE n3 SPACE n4 ENTER

For use in Keyboard Programs only, Y 1805 checks the relationship between VP n1 and the value n2 (VP n1 is greater-than n2, greater-than-or-equal, equal-to, less-than-or-equal, less-than, depending on the value n4), and skips n3 keyboard instructions in the keyboard stack if the test is true (or n3 can be negative denoting skip back). n4 denotes the type of test:

n4 = 2 greater than
n4 = 1 greater than or equal
n4 = 0 equal to
n4 = –1 less than or equal
n4 = –2 less than

The default value of n4 is –2. The default value of n3 is 1. The default value of n2 is 0.

Examples:

USER PROG SKIP 5 SPACE 10 SPACE 3 SPACE 1 ENTER

If the value of 5D is greater-than-or-equal to 10, the next 3 instructions in the Keyboard Program will not be executed.

USER PROG SKIP 6 ENTER

If the value of 6D is less than 0, one instruction will be skipped.

n3 may be negative. If so, the stack pointer is decremented appropriately if the test is true. For example, the following listing shows where the stack pointer would position itself depending upon the value assigned to n3 in the Y 1805 User Program if the tested condition is true (only two negative examples are shown, but n3 can be more negative than the values indicated).

Any Keyboard command n3 = –2
Any Keyboard command n3 = –1
Y 1805 n1 n2 n3
Any Keyboard command n3 = 0
Any Keyboard command n3 = 1

You must take care when using Y 1805 that your value of n3 is neither too positive or negative, which will result in the program skipping completely out of the Keyboard Program.

7. **Y 1806 lists VP values on the Terminal**

LIST

USER PROG LIST n1 SPACE n2 ENTER

The values of the Variable Parameters from VP n1 through VP n2 are printed out in the following format:

<n1>D = <value>

The default value of n2 is n1. If 6D = 200, and 7D = 32767, then

Examples:

USER PROG LIST 6 SPACE 7 ENTER

6D = 200
7D = 32767

is printed on the Terminal.

USER PROG LIST 6 ENTER

6D = 200

is printed on the Terminal.

BS

BLOCK SIZE

8. **Y 1807 sets a VP to the blocksize**

USER PROG BS n1 ENTER

VP n1 is set to the current system blocksize.

Examples:

USER PROG BS 21 ENTER

If the blocksize were 256, then 21D is set to the value 256.

INPUT

PHOTO READR

9. **Y 1808 inputs values from Keyboard or Terminal**

USER PROG INPUT n1 SPACE n2 ENTER

Variable Parameters n1 through n2 are set to the values typed into the system Terminal or Keyboard. These values can be typed on one line separated by any non-numeric ASCII character except ".", or may be typed on separate lines. Floating point numbers up to 9,999,999 may be entered, with decimal point. Default of n2 is n1. Complex numbers require two numbers to be entered apiece.

Example:

USER PROG INPUT 1 SPACE 3 ENTER

The software will input three integers entered from the Keyboard or the Terminal.

USER PROG INPUT 2000 ENTER

The software will input one floating point value.

NOTE

Numbers entered may be in floating point form: "64923.445" is a legal number. Since there is no decimal point on the Keyboard, the key labeled END may be used; its ASCII printout is ".b", and the "b" is ignored ("b" is a space). Note that 64923445 SPACE -3 (64923445×10^{-3}) is not a valid format for this command.

Note that all floating point entries must have at least one entry after the decimal point. Therefore, to enter a number such as 3 in floating point you must enter 3.0 and not just 3. by itself.

PUNCH

10. **Y 1809 outputs VP values**

OUTPUT

USER PROG OUTPUT n1 SPACE n2 ENTER

This program prints out the values of Variable Parameters n1 through n2. Default value of n2 is n1. If a floating point VP contains a value too large for the Y 1809 format, all dollar signs ("$") will be printed. If n2 = n1, carriage return and line feed will be suppressed (not true for n2 defaulted).

11. **Y 1810 multiplies/divides integers**

USER PROG 1810 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

Useful only for integers, Y 1810 sets VP n1 to the value of ((n2*n3)/n4).

Example:

USER PROG 1810 SPACE 6 SPACE 32767 SPACE 2 SPACE 3 ENTER

6D is set to ((32767*2)/3), or 21844. Using the programs Y 1803 and Y 1804 instead of Y 1810 to perform this operation ((32767*2)/3), causes an error, since the integer containing the intermediate result (32767*2) can be no larger than 32767.

12. **Y 1817 has the following functions**:

a. Read a selected portion of the Processor Switch Register or Fourier display interface word into a specified Variable Parameter.

b. Write the contents of a specified Variable Parameter into the selected portion of the Switch Register (but not into the Fourier display interface word).

USER PROG 1817 SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

Where: n1 equals the VP number (no default allowed). If VP n1 is not an integer VP, the contents of VP n1 are converted to a one-word integer prior to the execution of the program.

n2 equals the starting bit of the field (0 to 15). No default allowed.

n3 equals number of bits to read or write (1 to 16). If defaulted n3 =1.

If n4 = 0, the Switch Register is selected. If n4 = 1, the display interface word, whose format is given in the System Software Manual, is selected. Default n4 = 0 (Switch Register).

If n5 equals 0, the program reads from the selected input. If n5 equals 1, the program writes to the selected output (Switch Register only). Default n5 = 0 (read).

During a Read operation, the contents of the Switch Register or display word are rotated right until bit n2 is in bit 0 location. All but the first n3 least significant bits are then masked off and the result is stored in VP n1.

During a Write operation, all but the first n3 least significant bits of VP n1 are masked off. The result is then rotated left by n2 bits. This n3 bit field then replaces the corresponding field of the Switch Register. Note that you cannot set Switch Register bit 3 with this program because of additional software considerations.

Examples:

USER PROG 1817 SPACE 0 SPACE 0 SPACE 16 SPACE 0 SPACE 1 ENTER

Writes the value "1" to the Switch Register (assuming that VP 0 contains the value "1").

USER PROG 1817 SPACE 0 SPACE 6 SPACE 1 SPACE 0 SPACE 1 ENTER

Sets bit 6 of the Switch Register (assuming that VP 0 contains the value "1").

USER PROG 1817 SPACE 0 SPACE 0 SPACE 16 ENTER

Reads the decimal value in Switch Register into VP 0.

USER PROG 1817 SPACE 0 SPACE 6 SPACE 1 ENTER

Reads bit 6 value into VP 0

USER PROG 1817 SPACE 0 SPACE 1 SPACE 3 SPACE 1 ENTER

Reads the value of the Display Unit polar ANG/DIV switch into VP 0. (This value is in bits 1-3 of the display interface word.)

GET

LOAD

13. **Y 1821 gets data values from a data block**

USER PROG GET n1 SPACE n2 SPACE n3 ENTER

VP n1 is set to the value of block n2, channel n3 (n1 should be a floating point VP number for time domain data, and a complex VP number for frequency domain data). Default value for n2 and n3 is 0. If the value is greater, or smaller, than the Variable Parameter type will allow, then the Variable Parameter will be set to the largest, or smallest, value. This program is the Keyboard equivalent of the assembly language program "GET" (see the System Software Manual).

NOTE

This program will not work if the data is in LOG MAG format.

When this program detects out-of-range conditions (resulting from mapping the 10^{-512} to 10^{+512} system dynamic range into the 10^{-38} to 10^{+38} standard floating point format), it sets the block status (see QUALS function described below) non-zero and returns the value 2^{127} (positive overflow), -2^{+127} (negative overflow) or 0 (positive or negative underflow).

When using the GET and PUT Gold Key functions, inaccuracies of up to approximately .003% may occur due to the conversion between standard floating point format and floating point by block format.

PUT

14. **Y 1822 puts values into a data block**

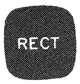
USER PROG PUT n1 SPACE n2 SPACE n3 ENTER

The value of VP n1 is placed into block n2, channel n3. Default values of n2 and n3 are 0. If the information is in the frequency domain, n1 is stored in the real portion and the imaginary portion is set to 0 unless a complex VP is used. This program is equivalent to the assembly language program PUT (see the System Software Manual).

NOTE

This program will not work if the data block is in LOG MAG format. If the block is in POLAR coordinates, then the angle magnitude put into the block must be ≤ 180.

QUALS



15. **Y 1823 gets/puts data block qualifiers**

USER PROG QUALS n1 SPACE n2 SPACE n3 ENTER

This is the counterpart to a GETQ or PUTQ call at assembly-language level (refer to System Software Manual). A table of 5 Variable Parameters starting at n1 gets, or puts, qualifiers for block n2. If n3 = 0, then get qualifiers; if n3 ≠ 0, then put qualifiers. Default values for n2 and n3 are 0.

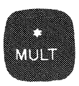
Example:

USER PROG QUALS 6 SPACE 3 ENTER

After the call,

6D = system blocksize
7D = scale factor word, block 3
8D = calibrator, block 3
9D = frequency code, block 3
10D = status, non-zero if GET function was unable to resolve scaling differences between a data point and floating point format. Refer to the discussion of the GET function, above.

EXP



16. **Y 1827 exponentiates**

USER PROG EXP n1 SPACE n2 SPACE n3 ENTER

VP n1 is set to the value of n2 to the power of n3, or n2**n3. No defaults allowed. None of the parameters can be Variable Parameters of complex type. THE PARAMETERS n3 MAY NOT BE NEGATIVE.

Example:

USER PROG EXP 6 SPACE 3 SPACE 2 ENTER

6D is set to 9.

17. **Y 1828 executes frequency/channel conversions**

USER PROG 1828 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

If n4 equals 0, then VP n1 is set to the channel number in block n3 that corresponds to the frequency or time in VP n2.

If n4 does not equal 0, then VP n1 is set to the frequency or time corresponding to the channel number in VP n2, block n3. No parameters may be defaulted.

Example: If 2000D = 6026.9 and 6D = 12, then

USER PROG 1828 SPACE 6 SPACE 2000 SPACE 3 SPACE 0 ENTER

6D is set to the channel number in block 3 that is closest to 6026.9 Hz (the value of 2000D; note that it is understood to be a VP in the call).

USER PROG 1828 SPACE 2000 SPACE 6 SPACE 3 SPACE 1 ENTER

2000 is set to the frequency of channel 12 (the value of 6D), block 3.

18. **Y 1829 sets Variable Parameters to data block header values**

Y 1829 takes the numbers from a data block header (on the Disc or in core) and places them into Variable Parameters so that they can be used at the Keyboard Program level. Refer to Table 3-2 for data block header format.

USER PROG 1829 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

Where n1 equals the beginning VP number.

n2 equals the beginning header word (from 0 to 127). Refer to Table 3-2 for header format.

n3 equals the number of words to be transferred (default equals 1). Note that 1 word must be specified for every integer VP to be transferred, 2 words for every floating point VP, and 4 words for every complex VP.

n4 equals the data block record number on the Disc. If defaulted or <0, the header in core is used.

See Examples after Y 1830.

19. **Y 1830 puts Variable Parameters into a data block header (on the Disc or in core).**

Y 1830 takes the contents of Variable Parameters and places them into a data block header. Refer to Table 3-2 for data header format.

USER PROG 1830 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

Where n1 equals the beginning VP number.

n2 equals the beginning header word.

n3 equals the number of words to be transferred (default equals 1).

n4 equals the data block number on the Disc that is to be changed. If defaulted or <0, change only the header in core.

Table 3-2. Header Formats

	Data Block Header Format	ADC Throughput Header Format
0		ID = 100000_8 = -32768
1	scale factor/coordinate code	
2	calibrator	
3	frequency code	frequency code
4	block size	block size
5		# of channels, n1 (MASS STORE 22)
6		# of records, n2 (MASS STORE 22)
7		record length = n1 × block size
8,9	BSFA center frequency (floating point)	
10,11	BSFA delta frequency (floating point)	
12-79	unshared area for VP storage	
80	length of ASCII search key	length of ASCII search key
81-115	70-character search key	70-character search key
116-127	system reserved area	

The data calibrator and scale factor are obtained by looking at the code in bits 0, 1, 2 of the ADC data word for each ADC channel. The codes are decoded as follows:

Attenuator Setting	Code	Calibrator	Scale Factor
8V	0	26214	1
4V	1	13107	1
2V	2	6554	1
1V	3	32767	0
.125V	4	4096	0
.25V	5	8192	0
.5V	6	16384	0

Examples:

USER PROG 1829 SPACE 2000 SPACE 8 SPACE 4 ENTER

(After BSFA measurement): Reads the values of the zoom center frequency and Δf into floating point Variable Parameters 2000 and 2001 (from the core-resident data block header).

USER PROG 1830 SPACE 0 SPACE 12 SPACE 1 SPACE 100 ENTER

Stores an integer VP value (contained in VP 0) into word 12 of the header of data block 100:

20. **Y 1850 computes and stores the sine of a VP**

USER PROG 1850 SPACE n1 SPACE n2 ENTER

The program computes the sine of the floating point VP n2 (in radians) and stores the results into floating point VP n1.

21. **Y 1851 computes and stores the cosine of a VP**

USER PROG 1851 SPACE n1 SPACE n2 ENTER

The program computes the cosine of the floating point VP n2 (in radians) and stores the results into floating point VP n1.

22. **Y 1852 computes and stores the natural logarithm of a VP**

USER PROG 1852 SPACE n1 SPACE n2 ENTER

The program computes the natural logarithm of the floating point VP n2 and stores the result into floating point VP n1.

Example:

Assume Variable Parameter 2001 contains the value 1.000 (radians).

Then after:	VP 2000 contains:
Y 1850 2000 2001	.8414
Y 1851 2000 2001	.5403
Y 1852 2000 2001	.0000

23. **Y 1853 performs logical AND function**

USER PROG 1853 SPACE n1 SPACE n2 SPACE n3 ENTER

Y 1853 performs the logical AND function between n2 and n3 (on a bit for bit basis), and stores the result into integer VP n1.

Examples:

USER PROG 1853 SPACE 0 SPACE 1 SPACE 2 ENTER

Performs a logical AND function on each bit and stores the result in VP 0 (e.g., showing only the three last significant bits).

001
010
———
000 (therefore, 0 is stored in VP 0).

USER PROG 1853 SPACE 0 SPACE 1 SPACE 3 ENTER

001
011
———
001 (stored in VP 0)

24. **Y 1854 performs logical inclusive OR function**

USER PROG 1854 SPACE n1 SPACE n2 SPACE n3 ENTER

Y 1854 performs the logical OR function between n2 and n3 and stores the result into integer VP n1.

Example:

USER PROG 1854 SPACE 0 SPACE 1 SPACE 2 ENTER

Performs a logical OR function on each bit and stores the result in VP 0 (e.g., showing only the three least significant bits.

001
010
———
011 (therefore, 3 is stored in VP 0)

USER PROG 1854 SPACE 0 SPACE 1 SPACE 3 ENTER

001
011
———
011 (therefore, 3 is stored in VP 0)

Variable Parameters in Common

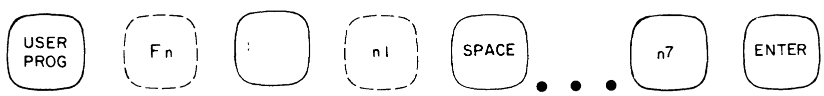
Some of the Variable Parameters reside within the 256-word COMMON in the system. Hence, using Mass Store File 7 commands (explained in Section 4), these Variable Parameters may be saved in a record in File 7 on the Disc, and previously saved Variable Parameters brought into core from another record in File 7. Thus, using Variable Parameters in Keyboard Program calls, and different Commons to control the Variable Parameters, different tests may be executed by the same Keyboard Program simply by reading different COMMONs into core from the Disc. The Variable Parameters not in Common will not be altered by using MASS STORE File 7 commands, but will retain their values unless altered by one of the above User Programs:

Fn

SOFT KEYS (F1 through F6)

Keys F1 through F6 on the Keyboard are "soft" keys which can be used to invoke user-written Keyboard Programs. Essentially, these six functions allow you to define your own keyboard commands.

The command structure is:



F1 through F6 actually perform 'jumps' (similar to using the JUMP key) to specified labels in Keyboard Program stack 0 on the Disc as follows:

Key	Label jumped to in stack 0
F1	L 1
F2	L 2
F3	L 3
F4	L 4
F5	L 5
F6	L 6

Note the numerical correspondence between the number of the soft key and the label jumped to. The label jumped to may, in turn, be programmed to invoke another user-written Keyboard Program, or the measurement may itself be contained in stack 0.

Soft keys can have parameters associated with them. The parameters supplied, as well as the number of parameters given, are passed to specified Variable Parameters, which in turn may be accessed by the Keyboard Program. The Variable Parameters used are as follows:

VP No.	Contents
0	Number of parameters entered
1	Value of first parameter
2	Value of second parameter
3	Value of third parameter
etc.	

Up to 7 parameters may be used with the soft keys. If Variable Parameters are used in the call, they are resolved before they are stored into the above variables.

Example:

If the command

USER PROG F1 1 DSPLY SPACE 2 SPACE 3 ENTER

is given (assuming Variable Parameter 1D = 10), the Variable Parameter contents after this command is executed are:

VP No.	Quantity
0	3 (number of parameters)
1	10
2	2
3	3

Once in the Variable Parameters, the numbers entered in the command may be used by the called Keyboard Program.

PROGRAMMING EXAMPLE

Assume you would like to write a keyboard function to do the following:

1. Take data into a block.
2. Apply the Hanning window to the data.
3. Fourier transform the block.
4. Display the block in log magnitude.

Define this function using the F3 key. The parameters defined are as follows:

GOLD F3 n1 ENTER

where n1 = data block in which the operation is to be performed (default = block 0).

The following keyboard commands should be in the Keyboard Program in keyboard stack 0 in order to implement this function.

NOTE

Soft keys F2 and F5 are programmed at the factory to perform automated transfer function and power spectrum measurements (as explained in Section 2). At the user's option these keys may be reprogrammed to perform different functions.

NOTE

The remainder of stack 0 is not shown. If you wish to operate the prewritten measurement programs (power spectrum and transfer function) supplied with the F5 and F2 keys, you must insert these commands at label 3 while leaving the remainder of stack 0 unchanged.

```
.
.
.
L   3                   Label 3
Y   _    10  0          Set up default case by setting VP 10 (block number) to 0.
Y   IF    0  0   0  1   If 0D = 0, skip one step (parameter defaulted)
Y   _    10  1D         Otherwise, set VP 10 ← VP 1
RA  10D
H1  10D
F   10D
TL  10D
D   10D
.
.
.
```

Further examples of using the soft keys may be found by consulting the prewritten measurement program listings in Appendix D. These programs utilize keys F2 and F5 (without parameters) to initiate various measurement processes.

SECTION 4
MASS STORE

INTRODUCTION

The Mass Store capability controls Disc (and optional Mag Tape) operations. The Command Matrix is illustrated in Figure 4-1. The file structure and individual commands are explained in this section. In addition, the data format, header record format, file pointer concept, and examples of editing ASCII text and sample programs are also covered.

FILE STRUCTURE

The Mass Store devices of the Fourier System are organized into files which are logical, rather than physical, divisions of the storage area. Each file stores a different type of data. The files may all be located on one device, or they may be on several devices. For example, when a system contains a Mag Tape option in addition to the standard Disc, the Data Block File (File 1) will normally be on the Disc, and the ADC Throughput File (File 2) will normally be on the Mag Tape. The files are placed on the various Mass Store devices in an arrangement that optimizes system performance. Since the Disc is the standard Mass Store device for the system, future references will refer to the Disc. When applicable, reference will be made to the optional Mag Tape.

Files are subdivided into records. Records are logical divisions of a file into separate groups of information. For example, each data block stored in File 1 is a separate logical record.

Each file also has a pointer associated with it. This is a logical element in the software which determines the next record of the file to be operated on. The pointer for a file may be set to any record in that file, independently of where the pointer for any other file is set. Also, the pointer allows the user to associate the data in one file with that in another file by setting the pointers. An Index Block File is kept to record such associations. For example, if the data in Record 2 of the Data Block File is associated with Record 3 of the ASCII Text File, then the pointers can be set to print out the data (from the Data File) and the comment text (from the ASCII Text File).

Operations are performed on files and pointers. Files can be read from or written into. Pointers can be moved forward, backward, or to a specific record within a file.

To operate on a file, the user must define (through use of a MASS STORE command) the file, the operation, and the pointer position. In some commands, the pointer position is implied (as in "read from the current pointer position"). Some commands only affect the pointer in a file.

Figure 4-1. Command Matrix

The 8 files shown here represent 8 types of data and program storage. Each file has a fixed number of records of fixed length. These can be changed only by reconfiguring the system software. Each file may be independently positioned, read, and written as though each were on separate devices.

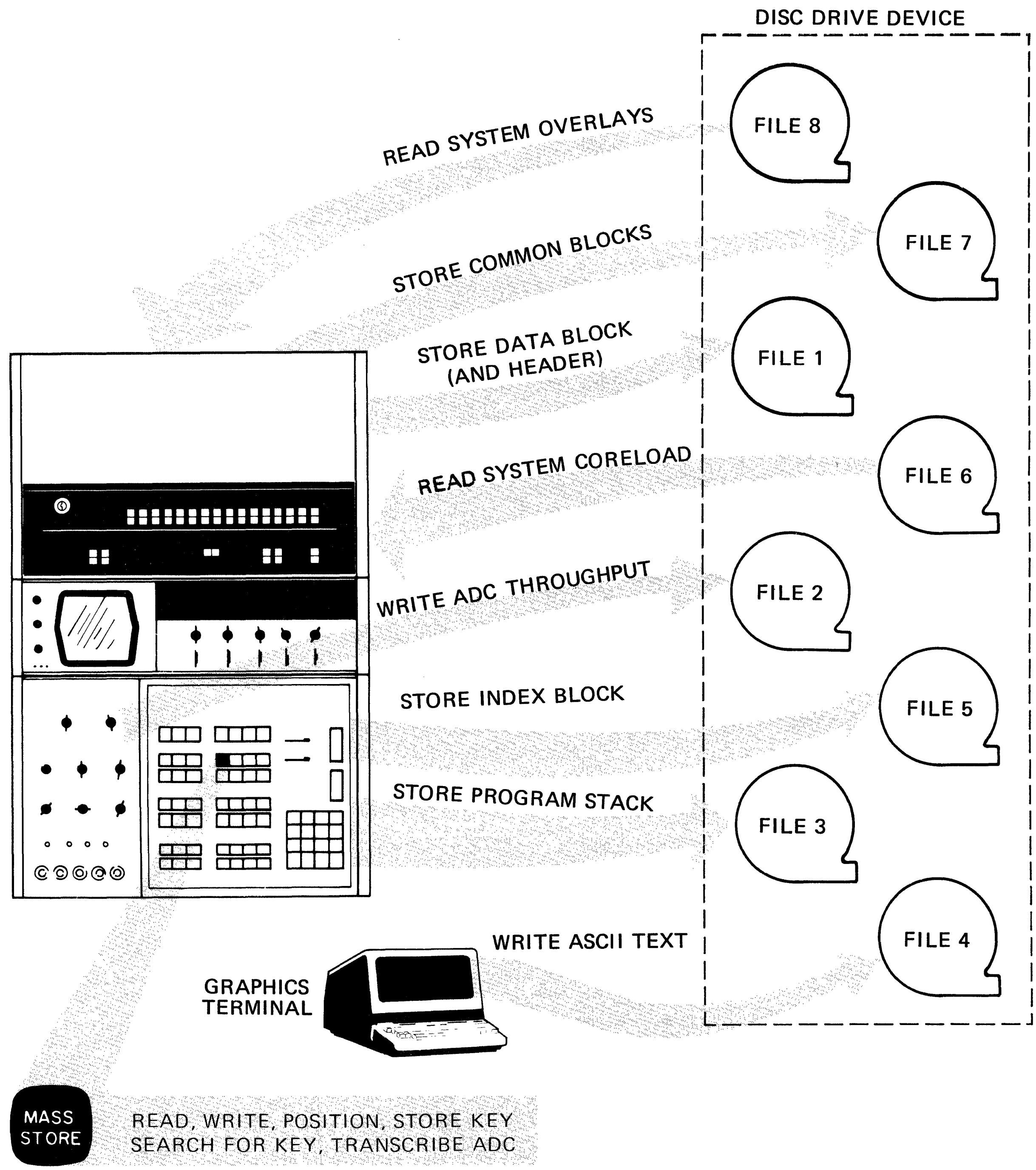
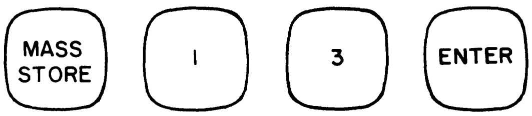


Figure 4-1. Command Matrix (continued)

Disc operations are executed through the MASS STORE key on the Fourier keyboard. The Command Matrix below lists the types of operations permitted.

NOTE

If the system has an optional Mag Tape UNIT, ONLY THE ADC THROUGHPUT FILE (FILE 2) IS ON THE MAG TAPE.



Example of command to READ NEXT RECORD from the PROGRAM STACK file.

FILE → / ↓ OP	1 DATA BLOCK	2 ADC THROUGHPUT	3 PROGRAM STACK	4 ASCII TEXT	5 INDEX BLOCK	6 SYSTEM CORELOAD	7 COMMON	8 OVERLAYS	9 DATA BLOCK HEADERS
1 READ NEXT RECORD	11 Block n1	12 Channel n1 Sample n2	13	14	15	16	17	18 (See Note 1)	*
2 WRITE NEXT RECORD	21 Block n1	22 n1 Channels n2 Records	23	24	25	*	27	*	*
3 POSITION TO RECORD (OR OVERLAY)	31 Record n1 n2 if n1 Relative	32 Record n1 n2 if n1 Relative	33 Record n1 n2 if n1 Relative	34 Record n1 n2 if n1 Relative	35 Record n1 n2 if n1 Relative	36 Record n1 n2 if n1 Relative	37	38 (See Note 2)	*
5 SEARCH FOR KEY	51	52	53	54				58 (See Note 3)	*
6 STORE KEY	61	62	63						*
7 TRANS CRIBE ADC		72							*

* Not Executable From the Keyboard

Note 1:
n1 Defaulted: Force Load Overlay
n1 Given: Load Overlay If It's Not Already in Core

Note 2:
Overlay n1
n2 If n1 Relative

Note 3:
List User Programs in System and/or Overlays

What's an OP? .... This is the operation code number, 1 through 7 (except 4), representing an operation such as read, write, position, etc.

What's a FILE? .... This is a file number, 1 through 8, representing functional storage areas on the disc (mag tape); not separate devices necessarily but separate areas on one or more devices.

The OP code and FILE number are an integral part of every MASS STORE command. All disc (or mag tape) operations are performed by using the Command Matrix codes.

MASS STORE FILES

FILE 1 — DATA BLOCK

This file is set up for storage of standard Fourier blocks of data. The data blocks are identified on the Disc by sequential numbers 0, 1, 2, etc. The positioning commands or the search key may be used to locate data block records within the file.

Data blocks can be transferred between the Disc and system memory any number of times by using the Read and Write commands.

Data block qualifiers are written into the Data Block Header file (file 9).

FILE 2 — ADC THROUGHPUT

This file permits storage of raw ADC data. The maximum rate at which data is acquired is dependent on the system block size and the mass store device used. If there are no sample points lost between the ADC and the Disc or Mag Tape, the acquisition is said to be "real time". Each record consists of a block of sample points. When writing a series of records on this file, the first record is preceded by a header containing calibration, frequency, and search key information. The ADC throughput file is the only file that can be stored on the optional Mag Tape.

FILE 3 — PROGRAM STACK

This file contains Keyboard Program stacks from the Fourier program stack. The number of program stacks and stack sizes are specified in the File Sizes paragraph in this section. The stack size determines the maximum allowable length of a Keyboard Program. The program stacks may be retrieved by any Fourier system using the same file structure. The last 10 program stacks on the Disc are reserved for storing a Keyboard Program when a JUMP command (see Section 3) between stacks is executed.

Records 0 and 51 through 62 are used by the automated power spectrum and transfer function programs described in Section 2.

FILE 4 — ASCII TEXT

This file can be used as a "scratch pad " to store text or numbers during progress of a test. Alphanumeric information is written as one line per record, but may be retrieved as a block of lines by specifying the starting record number or the first line of characters (see Search Key under Special Commands paragraph).

This file also contains storage for Text Buffers, each of which may contain a number of alphanumeric messages. The Text Buffers may be created and edited using a single interactive editor program. ASCII Text and Text Buffers actually "share" the area on the Disc designated for File 4.

Text Buffer operation is described in Section 6. *Text buffers 51 through 55 are used by the automated power spectrum and transfer function programs described in Section 2.*

FILE 5 — INDEX BLOCK

This file stores the current position of all file pointers (i.e., the number of the next record that would be accessed by a read or write command to the file). This information can be recorded at strategic times so that, any time later, the files may be re-positioned to the previous state. The information recorded is equivalent to that printed out when the MASS STORE keys are pressed (see Print File Pointers under SPECIAL COMMANDS paragraph).

AND ENTER.

FILE 6 — SYSTEM CORELOAD

This file contains core images typically of a 'system' plus one 'overlay'. All of core may be dumped to one of the coreloads in this file, then retrieved when desired through a Keyboard command or through the Processor switch panel. Core images may also be constructed and dumped to the Disc using the FSDS package (refer to the System Software Manual for details).

FILE 7 — COMMON

This file contains variables which determine the state of the graphics software (current plot size, plot origin, etc.). It also contains some of the Variable Parameters. This file may be saved on the Disc and then recalled at a later time to recall the saved state of graphics and the Variable Parameters. Refer to Section 6 for a description of the graphics software and Section 3 for Variable Parameters in common. Common record 0 is used by the automated power spectrum and transfer function programs described in Section 2.

FILE 8 — OVERLAYS

This file contains partial core images which may be read from the Disc directly over the current contents of core. Additional User Programs may be stored in up to 30 of these overlays and then read into system memory and used when desired.

FILE 9 — DATA BLOCK HEADERS

This file is used along with File 1 (Data Blocks) and contains information (qualifiers, search key, parameters, etc.) that goes along with the corresponding data block. This file is not directly accessible, but is automatically READ OR written when a data block is read or written. The data block headers occupy a different location on the Disc than the data blocks. There is one core-resident data block header that is read into or written from when a data block is read or written.

COMMAND TYPES

Basic operation of the Mass Store device is achieved through three types of commands. A write command stores the information on the Disc or Mag Tape, a read command retrieves the information, and a positioning command orients the device to the area where the information is to be read or written. See Figure 4-1, Command Matrix, for an abbreviated form of the commands described below.

POSITIONING COMMANDS

Positioning places the "pointer" in front of a desired record. Performing a subsequent operation (e.g., read or write) causes the pointer to increment one logical record. Keep this principle in mind. For instance, to read a record just written, you must enter a –1 relative positioning command or an absolute positioning command to re-position before reading. To position the Disc or Mag Tape in front of the desired record, one of the following positioning commands must be given:

Absolute Positioning

This command positions the Disc to a specific record in a file.

MASS STORE 3 f SPACE n1 ENTER

where:

3 is the op code for positioning

f is a file number (1 through 8).

n1 is the desired record number. Defaulting n1 positions the Disc to record 0.

Relative Positioning

This command positions the Disc n records backward or forward relative to its current position.

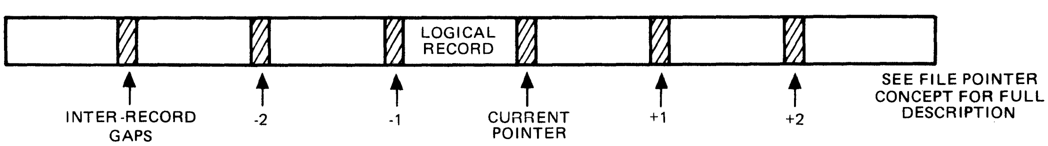
MASS STORE 3 f SPACE n1 SPACE 1 ENTER

where:

n1 is a relative number indicating records forward or backward from the present position (for backward movement enter minus sign first).

1 is a mode parameter signifying relative positioning.

The diagram below shows how the pointers are affected by relative positioning commands. Note the pointers move in increments of one logical record, whether or not this involves one or more physical records in the file being positioned.



See the File Pointer Concept paragraph in this section for full description.

Search Key Positioning

A third type of positioning is by search key. In this case the headers are examined for a particular ASCII code and the Disc stops at the logical record containing the header matching the key. For details see Search for Key command.

READ/WRITE COMMANDS

After positioning to the desired file and record, information can be written on the Disc, a record at a time, by using the write commands described below. To read back these records, the file must be positioned again, then the applicable read command given. If no positioning command is given, either of these commands cause the next record to be read or written, according to where the file pointers were left.

Write Next Record

GENERAL FORM

MASS STORE 2 f SPACE n1 SPACE n2 ENTER

where:

2 indicates a write command.

f is a file number 1 through 8. For additional parameters, see specific commands.

NOTE

The Write Next Record command cannot be used with Files 6 and 8. For error messages during operation, see Table 4-1. Error Messages toward the end of this section.

Read Next Record

GENERAL FORM

MASS STORE 1 f SPACE n1 SPACE n2 ENTER

where:

1 indicates a read command.

f is a file number 1 through 8. For additional parameters, see specific commands.

Data Block

The Read/Write Data Block commands are used to transfer Fourier data blocks between system core and the Disc. A Search for Key command may be used for positioning to a record in the Data Block file.

MASS STORE 1(Read) or 2(Write) 1 SPACE n1 ENTER

where:

n1 is the specified data block in core (0, 1, 2, . . .) to be written from or read into. Default of n1 assumes block 0.

WRITE DATA BLOCK

The Write Data Block command writes data block n1 from core as the next record on File 1 of the Disc.

READ DATA BLOCK

The Read Data Block command reads the next record from File 1 of the Disc into data block n1 in core. Data block n1 is then displayed.

NOTES ON USING THE READ/WRITE DATA BLOCK COMMANDS

For a Read command, a number of words will be read from the Disc equal to the current system blocksize or the Data Block File blocksize, whichever is less.

For a Write command, a number of words will be written to the Disc equal to the current system blocksize or the Data Block File blocksize, whichever is less.

ADC Throughput

NOTE

If the system has an optional Mag Tape UNIT, ONLY THE ADC THROUGHPUT FILE (FILE 2) IS ON THE MAG TAPE.

The ADC Throughput File is used to store raw ADC input data, a channel at a time, or up to four channels sumultaneously. A Write ADC Throughput command is used to store the data. The information can then be transferred to the Data Block File using the Transcribe ADC Throughput command or transferred directly into core using the ADC Throughput command. A Search for Key command may also be used for positioning to a session in this file.

DEFINITION OF TERMS

Sample:

1 data point (i.e., the digital value of any one signal at one instant of time). Represented by one computer word.

Channel:

1 ADC channel (i.e., A, B, C, or D).

Sample Group:

The group of n1 samples consisting of one simultaneous sample from each channel.

Record:

1 physical record on tape or Disc. Contains a number of samples equal to n1 × blocksize.

Session:

All records recorded as the result of a single write command.

WRITE ADC THROUGHPUT

MASS STORE 2 2 SPACE n1 SPACE n2 ENTER

where:

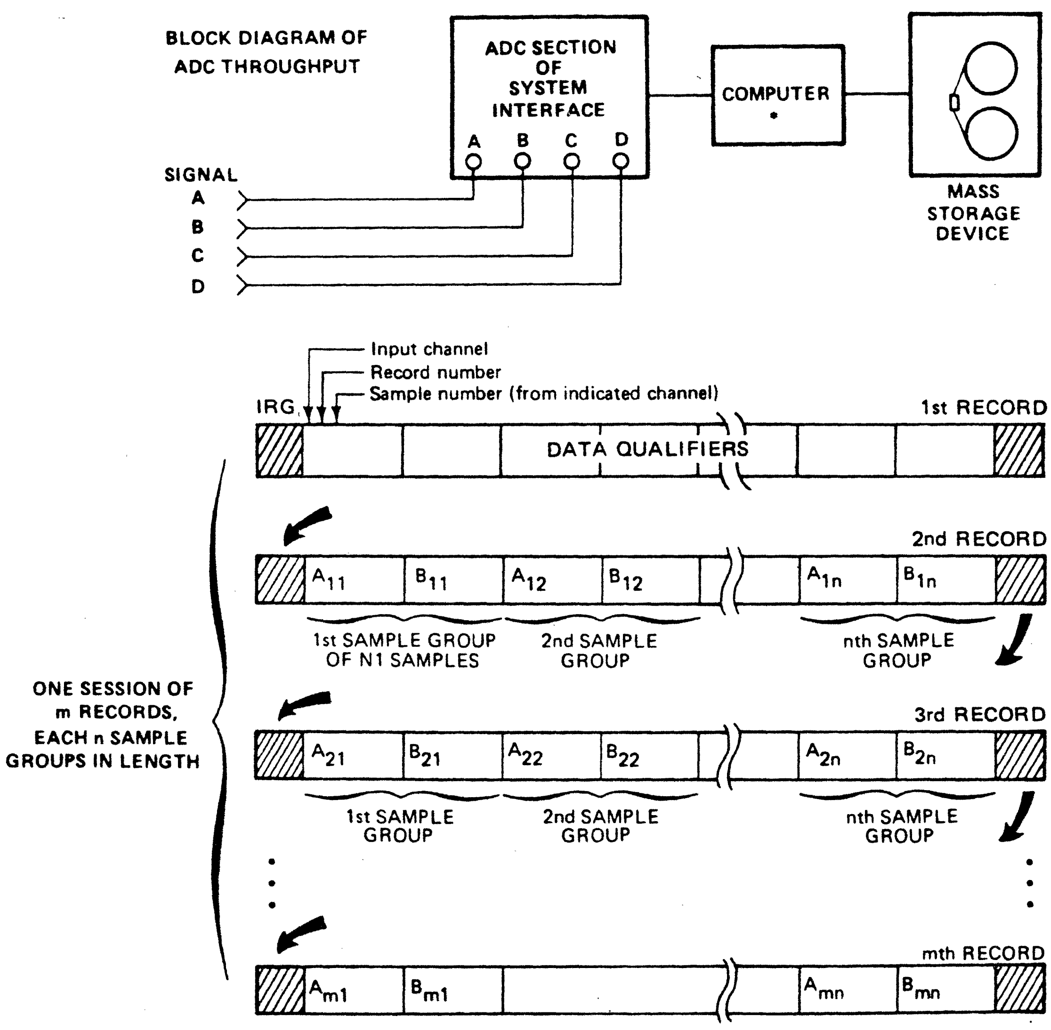
n1 is the number of ADC input channels.

n2 is the number of records taken.

No default allowed.

The Write ADC Throughput command records a data flow of samples from the ADC to the Disc or optional Mag Tape. The number of samples taken to fill one record depends on the system blocksize selected. The number of records taken is selected by the user by specifying n2. The file must first be positioned in front of the session. For dual channel ADC input, records are interwoven on the file: A,B,A,B,A,B, etc. Four-channel inputs are recorded likewise: A,B,C,D,A,B,C,D, etc. See the following figure.

1 ADC THROUGHOUT SESSION AS WRITTEN ON MASS STORAGE DEVICE



m=NUMBER OF RECORDS, N2

n=BLOCK SIZE

CORE REQUIRED FOR DATA BUFFERS = 2XN1XBS

WRITE ADC THROUGHPUT

MASS STORE 2 2 SPACE n1 SPACE n2 ENTER

where:

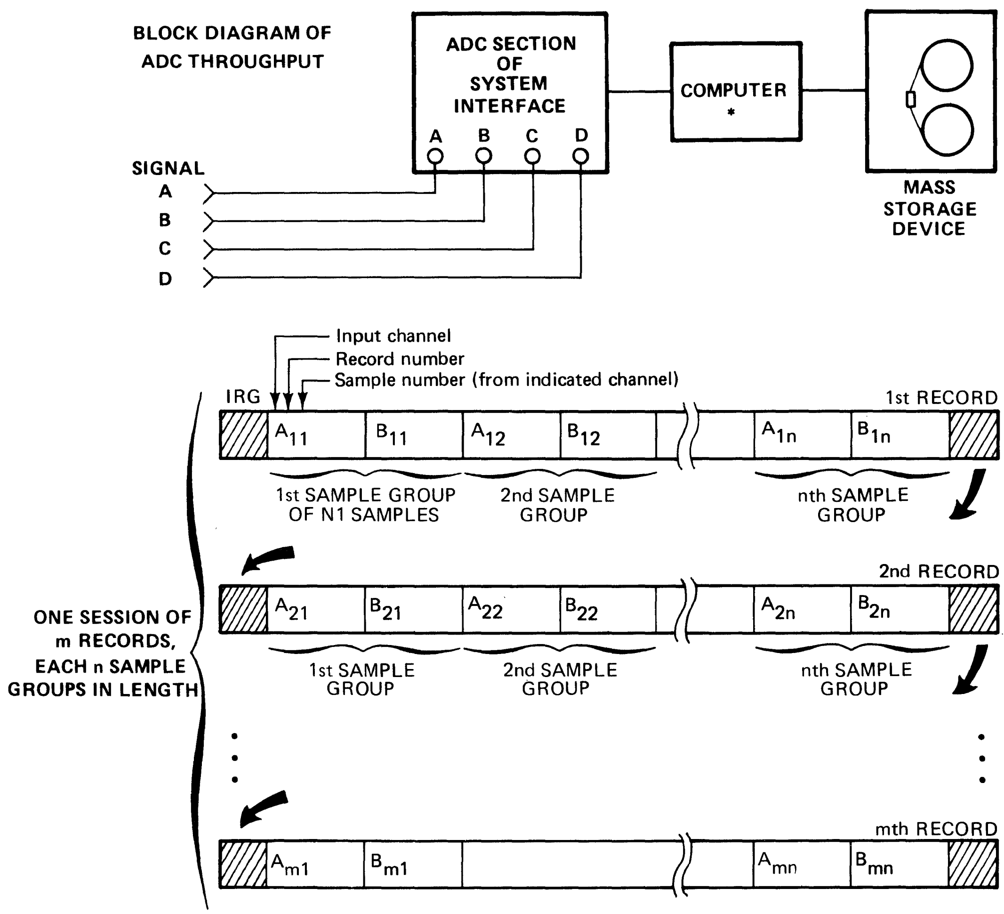
n1 is the number of ADC input channels.

n2 is the number of records taken.

No default allowed.

The Write ADC Throughput command records a data flow of samples from the ADC to the Disc or optional Mag Tape. The number of samples taken to fill one record depends on the system blocksize selected. The number of records taken is selected by the user by specifying n2. The file must first be positioned in front of the session. For dual channel ADC input, records are interwoven on the file: A,B,A,B,A,B, etc. Four-channel inputs are recorded likewise: A,B,C,D,A,B,C,D, etc. See the following figure.

1 ADC THROUGHOUT SESSION AS WRITTEN ON MASS STORAGE DEVICE



m=NUMBER OF RECORDS, N2

n=BLOCK SIZE

CORE REQUIRED FOR DATA BUFFERS = 2XN1XBS

When writing ADC Throughput, there must be at least enough data space to contain two buffers of size:

# of throughput channels × Blocksize

However, if there is sufficient data space, up to 5 such buffers may be used in the data space in order to make the throughput operation more efficient. Any previous data in this area (which starts at the beginning of the data space) will be destroyed by the Write ADC Throughput command.

Perform the following steps to determine which data blocks will be affected by the Write ADC Throughput command:

1. Divide the total data space available by the blocksize to be used during the Write ADC Throughput command. The result should be an even integer.
2. Divide this integer result by the number of throughput channels, and take the integer part of the result.
3. If this number is less than 2, there is insufficient space for the Write ADC Throughput operation — reduce the blocksize or the number of throughput channels.

   If this number is betwen 2 and 5, inclusive, this is the number of buffers that will be used from the data space.

   If this number is greater than 5, only 5 buffers will be declared from the data space.

   Examples: (Assume data space = 16K words)

   a. BLOCKSIZE = 1024 (1K)
      # of chans = 3

      Step 1: 16K/1K = 16

      Step 2: 16/3 = 5-1/3, integer part = 5

      Step 3: 5 buffers of size 3 × 1024 will be used from the data space. Therefore, the first 15K of data space will be altered by a Write ADC Throughput command with these parameters.

   b. BLOCKSIZE = 256
      # of chans = 4

      Step 1: 16K/256 = 64

      Step 2: 64/4 = 16

      Step 3: 16 > 5, therefore a maximum of 5 buffers of size (4 × 256) will be used from the data space. Therefore, the first 5K of data space will be altered by the Write ADC Throughput command.

   c. BLOCKSIZE = 4096
      # of chans = 3

      Step 1: 16K/4K = 4

      Step 2: 4/3 = 1 1/3

      Step 3: 1 < 2, therefore not enough data space is available for this operation.

An additional restriction on the throughput blocksize and # of channels is also illustrated by case "c" above.

For Disc throughput, the product

# of throughput channels × blocksize

should be less than or equal to the ADC Throughput record size as defined in the DIFS table. In the 5451C this record size is 4096 words. This restriction does not exist with Mag Tape throughput operations, as the record size for throughput to Mag Tape is not defined by the DIFS table. Refer to the System Software Manual for additional information.

When writing ADC Throughput, there must be at least enough data space to contain two buffers of size:

# of throughput channels × Blocksize

However, if there is sufficient data space, up to 5 such buffers may be used in the data space in order to make the throughput operation more efficient. Any previous data in this area (which starts at the beginning of the data space) will be destroyed by the Write ADC Throughput command.

Perform the following steps to determine which data blocks will be affected by the Write ADC Throughput command:

1. Divide the total data space available by the blocksize to be used during the Write ADC Throughput command. The result should be an even integer.
2. Divide this integer result by the number of throughput channels, and take the integer part of the result.
3. If this number is less than 2, there is insufficient space for the Write ADC Throughput operation — reduce the blocksize or the number of throughput channels.

   If this number is betwen 2 and 5, inclusive, this is the number of buffers that will be used from the data space.

   If this number is greater than 5, only 5 buffers will be declared from the data space.

   Examples: (Assume data space = 16K words)

   a. BLOCKSIZE = 1024 (1K)
      # of chans = 3

      Step 1: 16K/1K = 16

      Step 2: 16/3 = 5-1/3, integer part = 5

      Step 3: 5 buffers of size 3 × 1024 will be used from the data space. Therefore, the first 15K of data space will be altered by a Write ADC Throughput command with these parameters.

   b. BLOCKSIZE = 256
      # of chans = 4

      Step 1: 16K/256 = 64

      Step 2: 64/4 = 16

      Step 3: 16 > 5, therefore a maximum of 5 buffers of size (4 × 256) will be used from the data space. Therefore, the first 5K of data space will be altered by the Write ADC Throughput command.

   c. BLOCKSIZE = 4096
      # of chans = 3

      Step 1: 16K/4K = 4

      Step 2: 4/3 = 1 1/3

      Step 3: 1 < 2, therefore not enough data space is available for this operation.

   An additional restriction on the throughput blocksize and # of channels is also illustrated by case "c" above.

   For Disc throughput, the product

   # of throughput channels × blocksize

   should be less than or equal to the ADC Throughput record size as defined in the DIFS table. In the 5451C this record size is 4096 words. This restriction does not exist with Mag Tape throughput operations, as the record size for throughput to Mag Tape is not defined by the DIFS table. Refer to the System Software Manual for additional information.

Number of Buffers Used for Write ADC Throughput Command (data space = 16K)
(buffer size = # of throughput channels × blocksize)

# of Throughput Channels	Blocksize						
	64	128	256	512	1024	2048	4096
1	5	5	5	5	5	5	4
2	5	5	5	5	5	4	2[1]
3	5	5	5	5	5	2[1]	**X**
4	5	5	5	5	4	2[1]	**X**

Number of Buffers Used for Write ADC Throughput Command (data space = 28K)

1	5	5	5	5	5	5	5
2	5	5	5	5	5	5	3[1]
3	5	5	5	5	5	4[1]	2[1]
4	5	5	5	5	5	3[1]	**X**

[1]Buffer size exceeds 4096 words and is therefore invalid for disc throughput. This combination may be used for mag tape throughput only.

X = invalid combination of (# of channels) and (blocksize)

READ ADC THROUGHPUT

MASS STORE 1 2 SPACE n1 SPACE n2 ENTER

where:

n1 is the ADC input channel (1 for A, 2 for B, 3 for C, 4 for D).

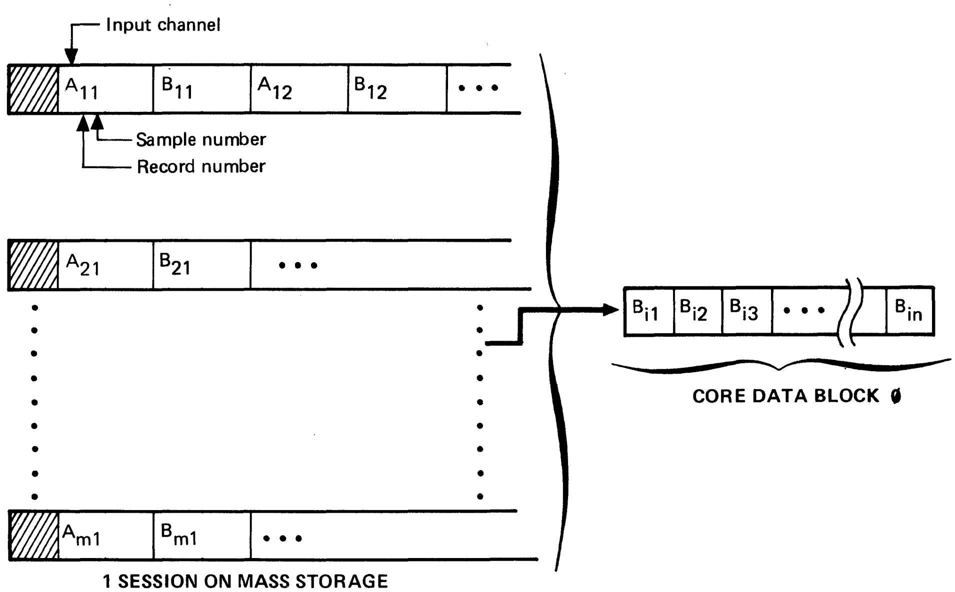
n2 is the record number in the session.

The Read ADC Throughput command retrieves the samples of one channel (n1) from one record (n2) in a session. These samples are then stored in sequential order in data block 0 of system core. The file must first be positioned in front of the session, after which individual records may be addressed or the records may be read sequentially (note for default values below).

Effect of a Read ADC Throughput Command

MASS STORE 1 2 SPACE n1 SPACE n2 ENTER

n1 = 2 $\Longrightarrow$ B CHANNEL
n2 = i $\Longrightarrow$ ith RECORD
IF DEFAULT, n2 INCREMENTS, n1 STAYS THE SAME
(n1 AND n2 CAN NOT BE DEFAULTED ON THE FIRST ENTRY)



NOTE

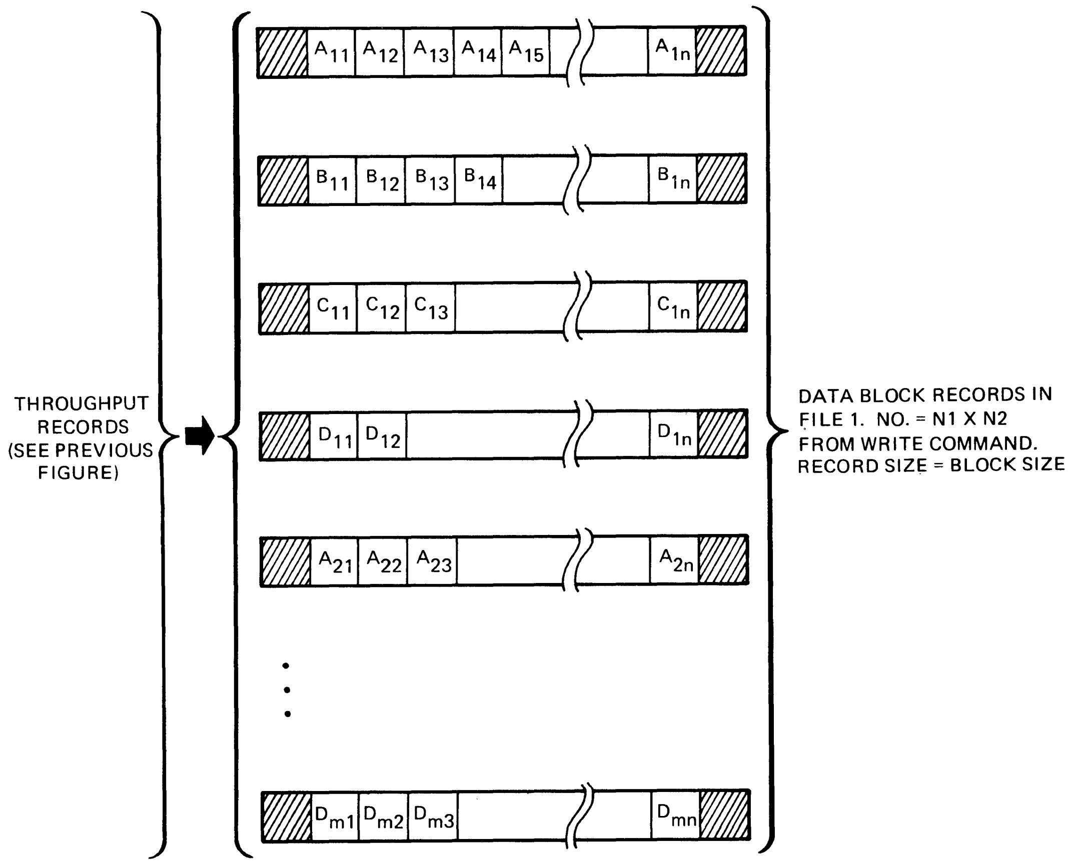
This command uses data blocks 0 through n1 (n = # of ADC Throughput channels) to unwind ADC Throughput data. Any data previously in these blocks will be destroyed.

TRANSCRIBE ADC THROUGHPUT

MASS STORE 7 2 ENTER

The Transcribe ADC Throughput command is an alternate read command that is useful when recording ADC inputs from more than one channel. This command transcribes the interwoven samples from the ADC Throughput File into sequential records in the Data Block File. An entire recording session is transcribed from a single command. The current position of the ADC Throughput File determines the session that is transcribed and the current position of the Data Block File determines the starting point of the data records written. See figure below.

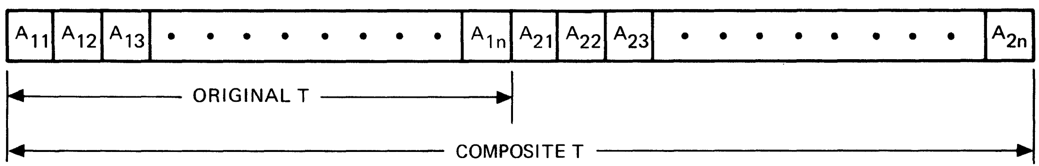
EFFECT OF A TRANSCRIBE ADC THROUGHPUT COMMAND



CORE REQUIRED = (n1+1) X BS
WHERE n1 = THE NUMBER OF CHANNELS.
THE PREVIOUS CONTENTS OF THIS CORE
ARE DESTROYED.

REAL TIME RECORDING

If ADC Throughput data is recorded in real time (i.e., no data loss between records), then these records may be combined to obtain a longer "time window" than the original time window ($T = t \times BS$). To do this, put successive records into successive data blocks as shown. Then, increase the blocksize to form the composite.



NOTE

> Real time recording is a prerequisite for performing offline Band Selectable Fourier Analysis (BSFA).

HIGH SAMPLE RATE, NON REAL TIME RECORDING

When the total ADC data rate (= sample frequency × number of throughput channels) exceeds 300 kHz, User Program 3026 should be invoked to prevent data loss in the first several data records. This User Program is described in Section 7.

Program Stack

The Read/Write Program Stack commands are used to save and restore Keyboard Programs. A Search for Key command may be used to position within this file.

MASS STORE 1 (Read) 3 ENTER
MASS STORE 2 (Write) 3 ENTER

WRITE PROGRAM STACK

The Write Program Stack command transfers the entire Keyboard Program Stack from core into File 3 as the next logical record. This command is useful to save Keyboard Programs for later use. Records 0 and 51 through 62 are used by the automated power spectrum and transfer function programs described in Appendix D. Records 330 through 339 are used for subroutines — see description of JUMP key in Section 3.

READ PROGRAM STACK

The Read Program Stack command returns the Keyboard Program Stack to system core. This command should not be executed within a Keyboard Program. A Keyboard Program may invoke another Keyboard Program using the JUMP command (refer to Section 3).

ASCII Text

The Read/Write ASCII Text commands permit storage and retrieval of text commentary.

DEFINITION OF TERMS

Line:

A single Terminal line of up to 70 characters terminated by RETURN (CR) LINEFEED (LF). One line constitutes one record on the file.

Message:

A group of lines terminated by /*.

MASS STORE 1 (Read) 4 ENTER
MASS STORE 2 (Write) 4 ENTER

WRITE ASCII TEXT

The Write ASCII Text command records an alphanumeric message in File 4. When the command is given, the system enters the BUSY mode and activates the Terminal, allowing you to enter up to 70 ASCII characters followed by RETURN (CR) LINEFEED (LF). Each succeeding line is recorded as a consecutive record until halted by /* as the first two characters of a new line, thus ending the message and removing the system from the BUSY mode.

READ ASCII TEXT

The Read ASCII Text command causes the Terminal to type out a previously entered alphanumeric message. Successive records are read out until a /* is reached. A positioning command or search command may be used to locate the message. The Search Key command will seek any line in the file. A line found by search key will be displayed on the Terminal.

EDIT ASCII TEXT

To edit a message already stored, position to the first line to be edited, using one of the positioning commands. Enter a Write ASCII Text command and reenter the line(s) to be changed. Enter /E to end the edit.

The following is an example of how to write, read, and edit ASCII text, as well as how to position in the ASCII Text File.

Example of Editing ASCII Text

COMMAND **DESCRIPTION**

MS 34 (Position to ASCII Text file, record 0)

MS 24 (Write next record)

```
BARON JEAN BAPTISTE JOSEPH FOURIER FOCUSED HIS REMARKABLE MIND
ON THE QUESTION OF HEAT CONDUCTION.  IN 1822, HE PUBLISHED THE
MATHEMATICAL THEORY THAT WAS TO BECOME KNOWN AS THE FOURIER SERIES.
THIS FAMOUS SERIES IS TODAY THE BASIS FOR HEWLETT-PACKARD'S ELEGANT
FOURIER ANALYZER, DEVELOPED AND MANUFACTURED AT THE SANTA CLARA
DIVISION.  THE SYSTEM IS USED IN UNCOVERING HIDDEN SIGNALS CONTAINED
IN COMPLEX DATA SUCH AS .......
/*
```

MS 24 (Write next record)

```
THE RAIN IN SPAIN FALLS MAINLY IN THE DRAIN
/*
```

MS34 Position to ASCII Text file, record 0)

MS14 (Read next record)

```
BARON JEAN BAPTISTE JOSEPH FOURIER FOCUSED HIS REMARKABLE MIND
ON THE QUESTION OF HEAT CONDUCTION.  IN 1822, HE PUBLISHED THE
MATHEMATICAL THEORY THAT WAS TO BECOME KNOWN AS THE FOURIER SERIES.
THIS FAMOUS SERIES IS TODAY THE BASIS FOR HEWLETT-PACKARD'S ELEGANT
FOURIER ANALYZER, DEVELOPED AND MANUFACTURED AT THE SANTA CLARA
DIVISION.  THE SYSTEM IS USED IN UNCOVERING HIDDEN SIGNALS CONTAINED
IN COMPLEX DATA SUCH AS .......
```

MS34 (Position to record 0)

MS54 (Search for Key in ASCII Text file so that a line may be edited)
IN (Key)

```
IN COMPLEX DATA SUCH AS .......
```

MS24 (Write next record to edit the message)

```
IN COMPLEX DATA SUCH AS VIBRATION SIGNALS FROM ROTATING MACHINERY.
/E
```

MS34 (Position to record 0)

MS14 (Read next record)

```
BARON JEAN BAPTISTE JOSEPH FOURIER FOCUSED HIS REMARKABLE MIND
ON THE QUESTION OF HEAT CONDUCTION.  IN 1822, HE PUBLISHED THE
MATHEMATICAL THEORY THAT WAS TO BECOME KNOWN AS THE FOURIER SERIES.
THIS FAMOUS SERIES IS TODAY THE BASIS FOR HEWLETT-PACKARD'S ELEGANT
FOURIER ANALYZER, DEVELOPED AND MANUFACTURED AT THE SANTA CLARA
DIVISION.  THE SYSTEM IS USED IN UNCOVERING HIDDEN SIGNALS CONTAINED
IN COMPLEX DATA SUCH AS VIBRATION SIGNALS FROM ROTATING MACHINERY.
```

MS34 (Position to record 0)

MS54 (Search for key)
/* (Key: End of message)
/*

MS34 1 1 (Relative position 1 record forward)

MS14 (Read next record)

```
THE RAIN IN SPAIN FALLS MAINLY IN THE DRAIN
```

Index Block

The Index Block File consists of sets of file pointer positions. (See Print File Pointers paragraph.) When a record from this file is read back into core, all file pointers are repositioned to their status when that record was written. This capability is useful to associate records in several files with each other. For example, ASCII commentary in record 11 (File 4) could be tied to related data in Data Block record 5 (File 1) by positioning File 1 to record 5, File 4 to record 11, and executing a Write Index Block command.

MASS STORE 1 (Read) 5 ENTER
MASS STORE 2 (Write) 5 ENTER

WRITE INDEX BLOCK

The Write Index Block command transfers the current file pointer locations (as maintained in core) to the Disc as the next record in File 5.

READ INDEX BLOCK

The Read Index Block command transfers a set of file pointer values from the next record on the Disc to system core. This command has the effect of resetting the current file pointer locations to some previous configuration.

NOTE

Since a Read Index Block command changes the location of all file pointers, successive "read next" commands cannot be used to step sequentially through the records on this file, as its own pointer is affected.

System Coreload

MASS STORE 1 6 ENTER

The Read System Coreload command returns a previously dumped core image to system core. Coreloads on the file are numbered from 0 and the file must be positioned in front of the desired coreload. Since entire coreloads are replaced, remember that all Processor resident data (status, file pointers, etc.) are modified also. Upon loading, the program executes a jump to system starting address 000002_8.

NOTE

Coreloads may also be loaded by using the Processor front-panel switches.

A write command using keystrokes is not available. Therefore, it is impossible to destroy a Disc resident coreload form the system Keyboard.

Common

The Common File consists of some of the parameters used by the graphics software, plus storage for some of the system Variable Parameters. When this file is written to the Disc, the current state of the graphics software, and the current values of the included Variable Parameters are saved. By reading this file at a later time, the values of these variables may be recalled.

MASS STORE 1 (Read) 7 ENTER
MASS STORE 2 (Write) 7 ENTER

WRITE COMMON

The Write Common command stores the current contents of system common to the Disc.

READ COMMON

The Read Common command reads a common file stored on the Disc. These stored values are then made the current values.

Overlays

MASS STORE 1 8 SPACE n1 ENTER

The Read Overlay command returns a previously stored overlay into system core. Overlays on the Disc are numbered from 0 and the file must be positioned in front of the desired overlay. The previous contents of the portion of core occupied by the overlays will be destroyed and replaced by the contents of the new overlay.

If parameter n1 is not given, the overlay will always be read into core. If n1 is given (it may have any value), then the overlay will be read into core only if it is not already in core.

A write command using keystrokes is not available. Therefore, it is impossible to destroy a Disc-resident overlay from the system Keyboard. Overlays may be stored on the Disc only by using the Fourier Software Development System (see 5451C Software Manual for FSDS description).

NOTE

> This command will not normally be necessary due to 'invisible' overlay swapping as discussed in Section 7.

Data Block Header

The Data Block Header is a small storage area (Disc-resident) containing parameters or information associated with data blocks, such as qualifiers, search keys, or variable parameters. This file is not directly accessible to the user via the Mass Store functions, but the contents of this header may be altered by several User Programs (e.g., USER PROG 1829 and USER PROG 1830). Whenever a data block is read to or written from the Disc, the Data Block Header is automatically read or written also. See Header Record Formats paragraph for additional information.

A core-resident Data Block Header exists in the Processor memory at all times. When a data block is written to the Disc, the system looks up the qualifiers (scale factor, calibrator, frequency code, etc.) for that block, stores the qualifiers into the core-resident header, and then writes the core-resident header to the Disc. If the core-resident header contains other information (e.g., parameters) at the time the data block is written, that information is copied to the Disc also. When a data block is read from the Disc, the header is read from the Disc into core (thereby destroying any qualifiers or parameters in the core-resident header) and then the qualifiers from this new core-resident header are assigned to the data block that was just read in. Any other parameters that were in the Disc-resident header are now in the core-resident header.

SPECIAL COMMANDS

The following commands provide features in addition to the basic commands of Position, Write, and Read.

Search Key

The search key feature involves two commands: Store Key and Search For Key. The search key is a line of up to 70 ASCII characters that is entered in header records in the Data Block, ADC Throughput, or Program Stack Files. These files can then be positioned by means of matching the search key rather than by using a positioning command to locate a record.

STORE KEY

MASS STORE 6 f ENTER

where:

f is file number 1, 2, or 3.

The Store Key command causes the system to enter the BUSY mode and activates the Terminal for entry of a line of up to 70 ASCII characters, terminated by RETURN (CR) LINEFEED (LF). This key is then stored in core until changed by another Store Key command.

NOTE

Each time a write command is executed on Files 1, 2, or 3, the current key is written into the header of that file. To avoid duplicate header storage, it is good practice to clear the header when finished. For example, press MASS STORE 6 f ENTER , followed by SPACE ENTER on the system keyboard.

SEARCH FOR KEY

MASS STORE 3 f ENTER (POSITIONING COMMAND)

(As described in Positioning Commands, must be used before the Search for Key command is given. This resets the pointer for the file to 0.)

MASS STORE 5 f ENTER (SEARCH FOR KEY)

where:

f is file number 1, 2, 3, or 4.

(All records in File 4 are in ASCII text, therefore, any record may be found by Search for Key command.)

The Search for Key command causes the system to enter the BUSY mode and allows Terminal entry of the ASCII key to be matched. When terminated by RETURN (CR) LINEFEED (LF), the records in the specified file are read (from the current position) until a match is found or the end-of-file is reached. When a match is found, the key will be read out on the Terminal and the file pointers positioned in front of the matching record. In addition, the record is read into core memory.

A match exists when no character in the search key entered differs from the key stored. This allows entry of partial keys. For example, a search key "A" matches a stored key "ABC", but the converse is not true.

The search key may be segmented into fields separated by decimal points. A search can then be made on some fields, ignoring the others. For example, a key of ABC.XYZ.123 can be located any of the following ways:

ABC or .XYZ or ..123 or A..1, etc.

Print File Pointers

The current position of each file is always known to the system. A file pointer for each file in core is used to keep track of the positions. These positions can be printed on the Terminal at any time by pressing the buttons below.

MASS STORE ENTER

R#	–	2	1	0	0	4	5	10	1	–
(FILE NO.	0	1	2	3	4	5	6	7	8	9)

An example of how the file pointers work is shown in Figure 4-2, File Pointer Concept.

Listing User Programs (by User Program Number)

MASS STORE 5 8 SPACE n1 ENTER

MS 58 ENTER — list User Programs resident in the system.

MS 58 SPACE n1 ENTER — list User Programs in system and overlay n1.

MS 58 SPACE 99 ENTER — list User Programs in system and all existing overlays.

If n1 is out of range (<0 or >19, except 99), or if the overlay specified by n1 does not exist, then no User Programs are printed for that overlay.

This is a special command which, when entered, will list on the system terminal (or optional list device) the User Programs in the system and in each Disc resident overlay. This is particularly useful for locating programs in the system.

The printout is as follows:

SYSTEM (User Programs in system)

1800
1801 (User Program Numbers)
1802
.
.
.

OVERLAY 00 (User Programs in overlay 0)

0005
0006
0100
.
.
.

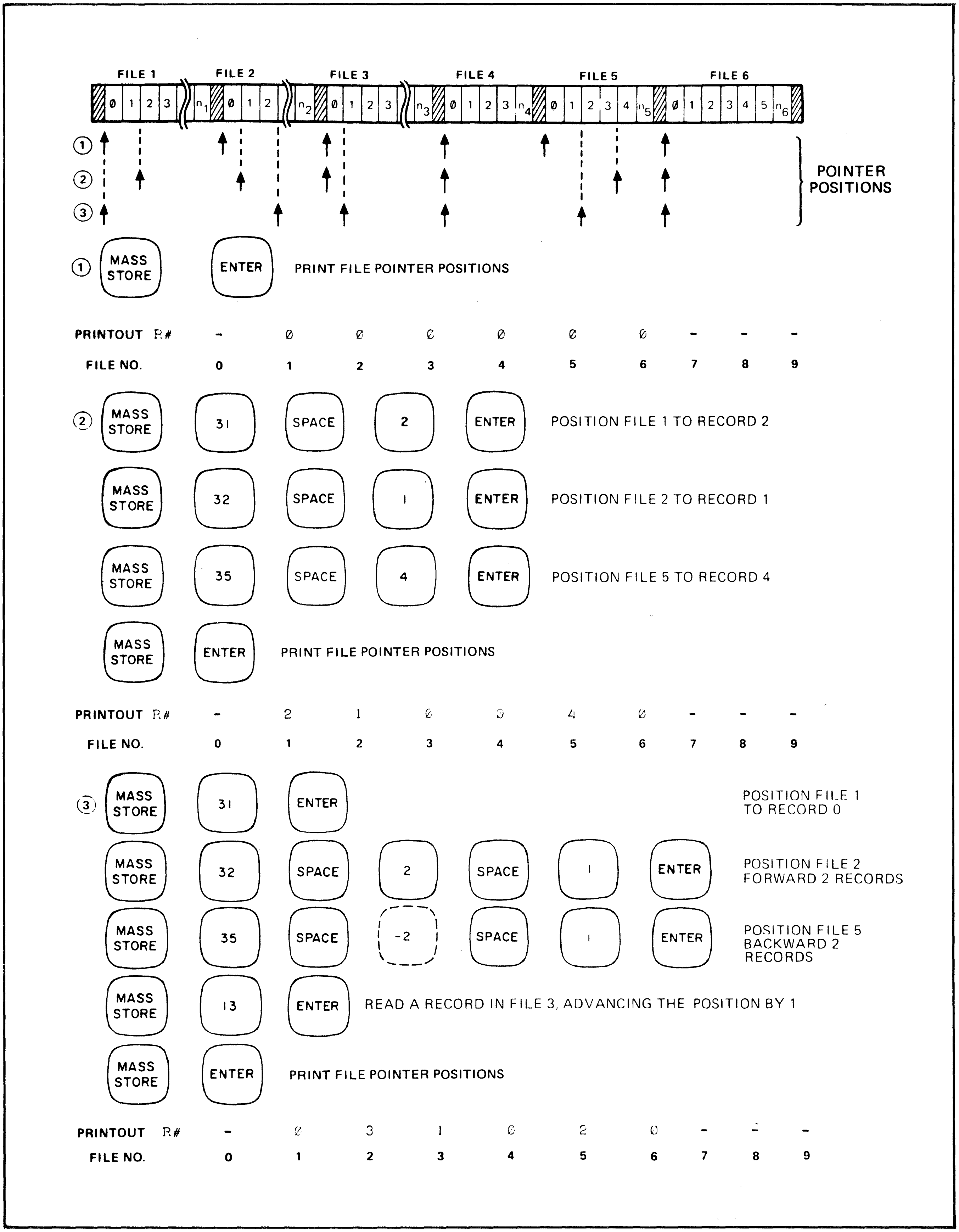
OVERLAY 01 (User Programs in overlay 1)

0005
0041
0045
0141
.
.
.

NOTE

The code to execute the MS 58 command resides in a system overlay. Therefore this overlay may automatically be read into core when the comand is used.

Figure 4-2. File Pointer Concept

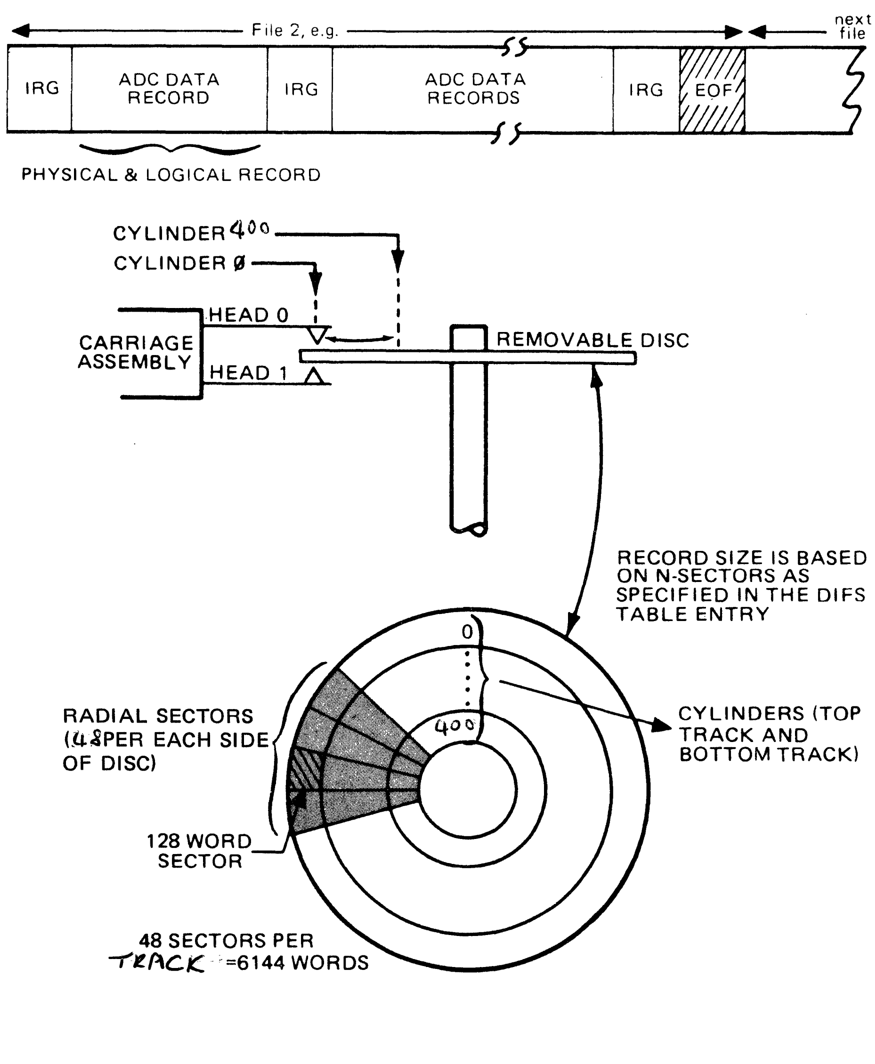


REFERENCE DATA

FORMATS

Files are numbered 1 through 9 and represent specific storage areas on the Disc or optional Mag Tape. The length and number of records in each file is determined when the system is first configured. This fixed number of fixed-length records can be changed only by re-configuring the system (see Organization of the Disc File System). The file numbers 1 through 8 are assigned so that a particular file may be addressed by using the Command Matrix.

Unformatted files on Mag Tape (File 2) contain records of unspecified length. Once a record is written, it is followed by an inter-record-gap for termination. As a consequence, these records cannot be rewritten without destroying following records. Each file on a removable Disc Pack is made up of a predetermined number of records overlapping fixed sectors. See accompanying illustration.



CORE DUMP AND LOAD

Memory may be loaded from or dumped to formatted Disc Packs using a a program called NIBBL. NIBBL resides in the 100_8 words below the memory reserved for the Processor loader. This is an area not used by the system, so that NIBBL and the system software may reside in core simultaneously. It is loaded into this location using the Processor Disc loader. The procedure is:

1. Load NIBBL into core (if it is not already in core) using the Processor Disc loader.

NOTE

Loading NIBBL into core will cause the current contents of core to be altered. Therefore, if a core dump to Disc is to be performed, NIBBL should already be in core before the dump is to be performed if it is desired to dump the current coreload to Disc. For loading core from the Disc, this is not a problem since the core that is loaded will overwrite the core locations modified by loading NIBBL.

The procedure is:

Make sure the system software Disc Pack is in the Disc Drive and the Disc Drive is "READY".

Set the Processor S-register to 011700

Press STORE

Press PRESET

Press IBL/TEST

Press RUN. A Processor halt will occur and the octal number 102076 will be in the T-register.

2. Set the P-register to 77600_8, NIBBL starting address.

   (This step may be omitted if Step 1 was just performed. This address will be set automatically.)

3. Clear the Processor A-register.

   (This step may be omitted if Step 1 was just performed. The A-register will already be clear.)

4. Clear the Processor S-register.

   (This step may be omitted if Step 1 was just performed. The S-register will already be clear.)

5. Use the B-register to indicate the operation to be performed and the starting cylinder for that operation by placing individual bit switches to the following positions:

   BIT 15 = 0 for coreload (Disc to core)
   = 1 for coredump (core to Disc) — see note 1

   BITS 14 through 10 = 0

   BITS 9 AND 8 INDICATE HEAD NUMBER.

   BITS 7 through 0 indicate starting cylinder number, as shown in the examples below:

Coreload #	Starting Cylinder # (octal)
Ø	1 see note 4
1	7
2	15
3	23
any number (n)	$(6 \cdot n + 1)_8$ see note 3

Example: To read coreload Ø from the removable disc to core, the B register must contain 000001_8. To write core to coreload 3 on the removable disc, the B register must contain 100023_8.

6. Press PRESET
7. Press RUN

   Correct performance of the core dump or load operation will result in a Processor HALT, with the number 102077_8 displayed in the T-register.

   If the T-register value is incorrect, repeat Steps 2-7. Refer to System Service manual for more information.

Note 1: Use caution when dumping from core to Disc, as previously stored system software may be destroyed.

HEAD Ø.

Note 2: Storing coreloads on HEADS OTHER THAN ^ is not recommended.

Note 3: The standard system has space allocated for 4 coreloads. Loading from or dumping to the Disc from any other cylinders may read in garbage or destroy system software on the Disc.

Note 4: This coreload typically contains the system operating software and should never be written over.

HEADER RECORD FORMATS

Files 1 and 2 contain Data Block Header records in addition to data records. The contents of the header records in these files are listed as follows. Refer to Section 3, KEY BOARD key for formats of the calibration, scale, and frequency code words.

All Data Block Header records are 128 words long as follows:

Data Block Header Format		ADC Throughput Header Format
0		ID = 100000_8 = -32768
1	scale factor/coordinate code	
2	calibrator	
3	frequency code	frequency code
4	block size	block size
5		# of channels, n1 (MASS STORE 22)
6		# of records, n2 (MASS STORE 22)
7		record length = n1 + block size
8,9	BSFA center frequency (floating point)	
10,11	BSFA delta frequency (floating point)	
12-79	unshared area for VP storage	
80	length of ASCII search key	length of ASCII search key
81-115	70-character search key	70-character search key
116-127	system reserved area	

FILE SIZES

The Mass Store options set up nine files for data and programs. These files are listed below for all possible Mass Store Configurations. The number of records in a file varies according to configuration and purpose of a system. A procedure to determine file lengths is given at the end of this section; the blank spaces in the center column may then be filled in by the user.

NOTE. SEE DISC FILE ALLOCATION IN THIS SECTION

Configuration: HP 7900A Disc Only

Logical File Number	File Name	Number of Logical Records in File	Logical Record Size (maximum)	Physical Unit No. (see Note)*
1	Data Block	*	4096 words	0, 1*
2	ADC Throughput	*	4096 words	0*
3	Program Stack	*	476 words	0, 1*
4	ASCII Text	*	70 characters	0, 1*
5	Index Block	*	128 words	0, 1*
6	System Coreload	*	Same as Coreload (memory) size	1*
7	Common	*	256 words	0, 1*
8	Overlays	*	depends on system size	1*
9	Data Block Headers	*	128 words	0, 1*

Note: Unit 1 = Removable Disc; Unit 0 = Fixed Disc.

*See Figure 4-3, Disc Configuration, for exact layout.

Configuration: Disc Unit and One Mag Tape Unit
(same as 7900A alone except for File 2)

Logical File Number	File Name	Number of Logical Records in File	Logical Record Size (maximum)	Physical Unit No. (see Note)*
1	Data Block	*	4096 words	0, 1*
2	ADC Throughput	Depends on tape length	depends on data space	(Mag Tape)
3	Program Stack	*	476 words	0, 1*
4	ASCII Text	*	70 characters	0, 1*
5	Index Block	*	128 words	0, 1*
6	System Coreload	*	Same as Coreload (memory) size	1*
7	Common	*	256 words	0, 1*
8	Overlays	*	depends on system size	1*
9	Data Block Headers	*	128 words	0, 1*

Note: Unit 1 = Removable Disc; Unit 0 = Fixed Disc.

*See Figure 4-3, Disc Configuration, for exact layout.

Procedure to Determine File Lengths

The number of records in each file depends on the hardware configuration and on the application package, if any, being used. Follow this procedure to determine the number of records in each file for your system (except Files 2, 3, and 5).

Enter the following Keyboard Program in the Fourier Analyzer:

```
LABEL 0 ENTER
MASS STORE 3 n ENTER
LABEL 1 ENTER
MASS STORE 1 n ENTER
JUMP 1 ENTER
END ENTER
TERM ENTER
JUMP 0 ENTER
```

When the Terminal repeatedly prints END-OF-FILE, press:

```
STOP MASS STORE ENTER
```

The Terminal will print the pointer positions of all the files. Note the record number for File n and write it in the center column of the previous table, as appropriate.

Edit the Keyboard Program shown above to set n = the next file, then repeat the procedure for all files except Files 2, 3, and 5. For Files 3 and 5, manual trial-and-error READ operations may be performed to find the last record that does not cause the END-OF-FILE message. For File 2, the above procedure may be used only if some data have been previously throughput to the disc starting at record 0.

ORGANIZATION OF THE DISC FILE SYSTEM

Space allocations for Files 1 through 9 on the Disc file. IS set up at the factory for the general user of the 5451C System. THIS STANDARD DISC LAYOUT (CONFIGURATION) IS SHOWN IN FIG 4-3.

While this organization is acceptable for most applications, it may be desirable to re-define the Disc layout for special applications. This can be done by modifying, at source level, the software tables which define the Disc allocation (see System Software Manual).

DISC FILE ALLOCATIONS

File 1 Data Blocks (4K)

Records 0-799 on Removable Disc
Records 800-1170 on Fixed Disc

File 2 ADC Throughput (4K)

Records 0-769 on Fixed Disc

File 3 Keyboard Program Stacks (512 words)

Records 0-289 on Removable Disc*
Records 290-329 on Fixed Disc
Records 330-339 on Removable Disc

File 4a ASCII Records

Records 0-2499 on Removable Disc*
Records 2500-2699 on Fixed Disc
Records 2700-3699 on Removable Disc*

File 4b Text Buffers (share Disc space with ASCII text file)

Buffers 1-200 on Removable Disc*
Buffers 201-240 on Fixed Disc

File 5 Index (128 words — 2 records used per Write or Read command)

Records 0-149 on Removable Disc
Records 150-159 on Fixed Disc

File 6 Coreloads (32K)

Coreloads 0-3 are on Removable Disc

File 7 Common (256 words)

Records 0-599 on Removable Disc*
Records 600-649 on Fixed Disc

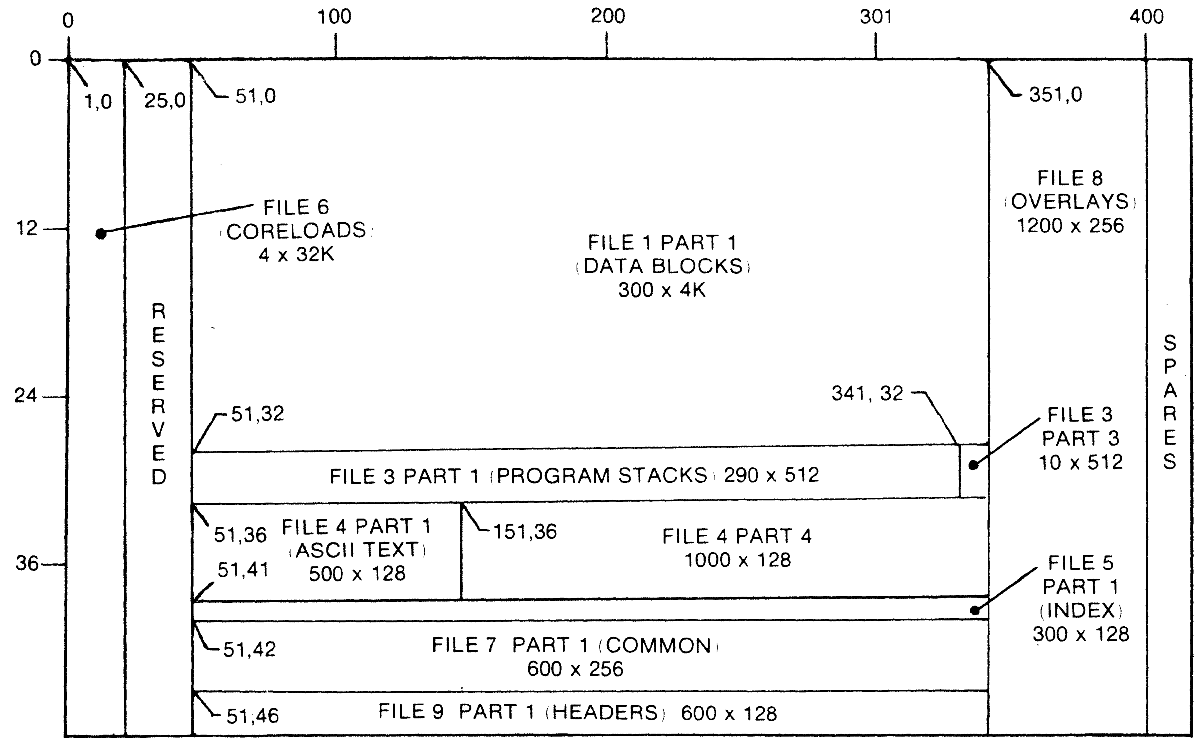
File 8 Overlays (variable size)

All Overlays reside on the Removable Disc

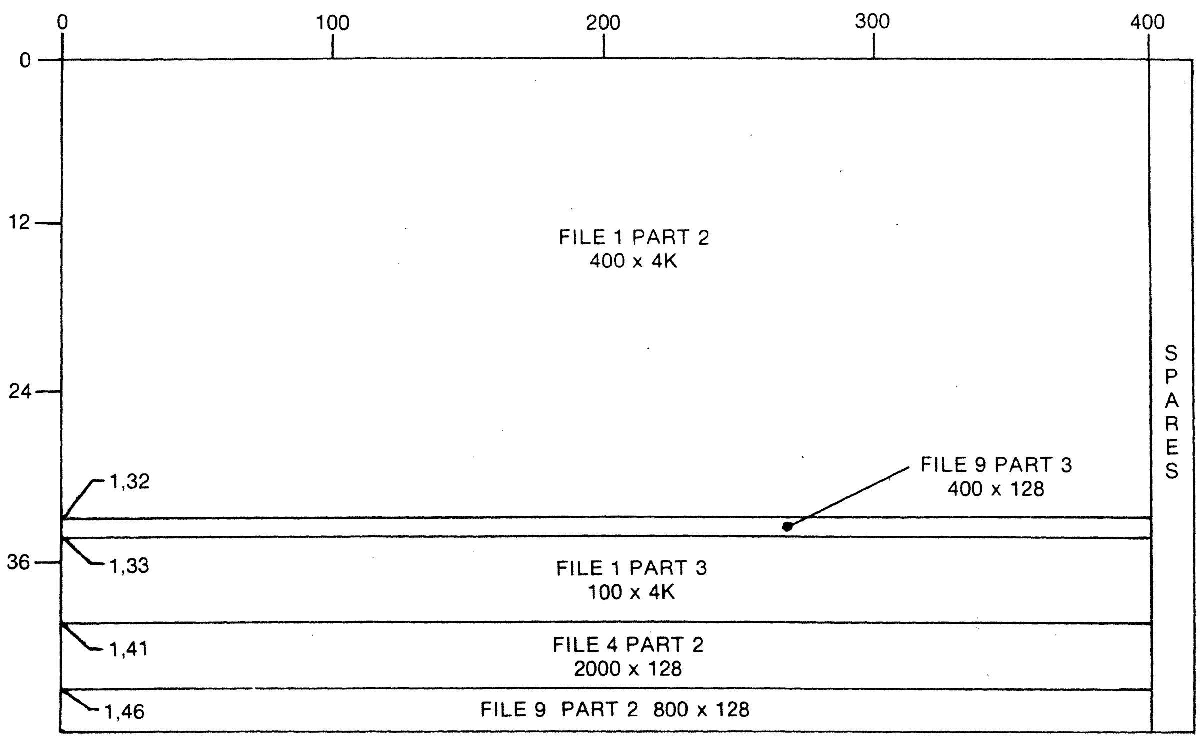
File 9 Data Block Headers (128 words)

Records 0-799 on Removable Disc
Records 800-1170 on Fixed Disc

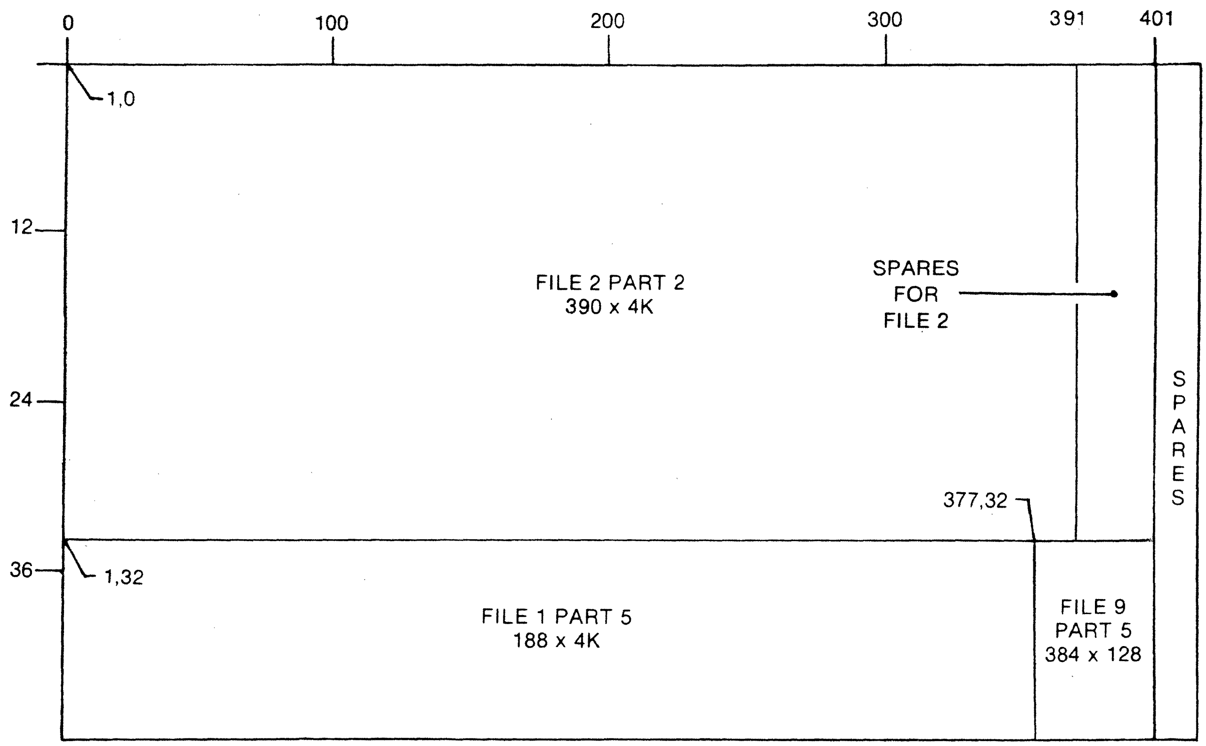
*See Appendix D for areas on the Disc reserved for automated power spectrum and transfer function programs described in Section 2.


0
100
200
301
400
0
12
24
36
1,0
25,0
51,0
351,0
FILE 6
(CORELOADS)
4 x 32K
RESERVED
FILE 1 PART 1
(DATA BLOCKS)
300 x 4K
FILE 8
(OVERLAYS)
1200 x 256
SPARES
51,32
341, 32
FILE 3
PART 3
10 x 512
FILE 3 PART 1 (PROGRAM STACKS) 290 x 512
51,36
FILE 4 PART 1
(ASCII TEXT)
500 x 128
151,36
FILE 4 PART 4
1000 x 128
51,41
FILE 5
PART 1
(INDEX)
300 x 128
51,42
FILE 7 PART 1 (COMMON)
600 x 256
51,46
FILE 9 PART 1 (HEADERS) 600 x 128

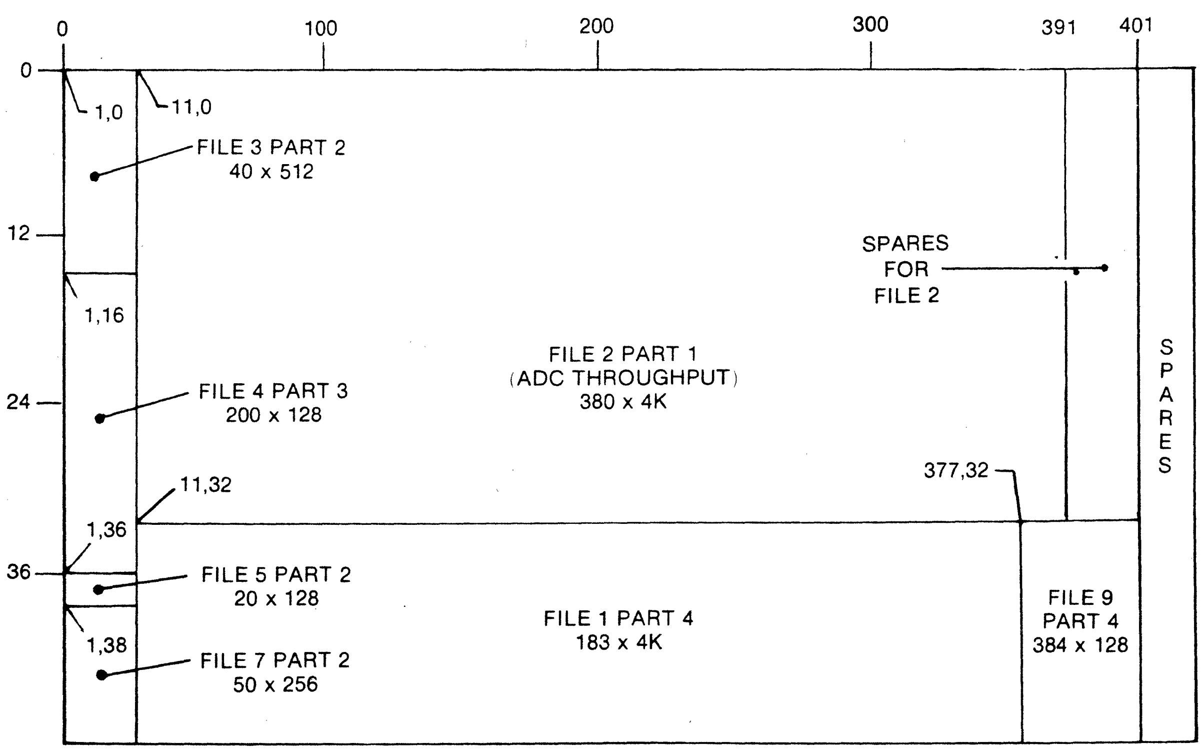
HEAD 0


0
100
200
300
400
0
12
24
36
FILE 1 PART 2
400 x 4K
SPARES
1,32
FILE 9 PART 3
400 x 128
1,33
FILE 1 PART 3
100 x 4K
1,41
FILE 4 PART 2
2000 x 128
1,46
FILE 9 PART 2 800 x 128

HEAD 1


0
100
200
300
391
401
1,0
12
24
36
1,32
FILE 2 PART 2
390 x 4K
SPARES
FOR
FILE 2
SPARES
377,32
FILE 1 PART 5
188 x 4K
FILE 9
PART 5
384 x 128

HEAD 2


0
100
200
300
391
401
0
12
24
36
1,0
11,0
FILE 3 PART 2
40 x 512
1,16
FILE 4 PART 3
200 x 128
11,32
1,36
FILE 5 PART 2
20 x 128
1,38
FILE 7 PART 2
50 x 256
SPARES
FOR
FILE 2
FILE 2 PART 1
(ADC THROUGHPUT)
380 x 4K
SPARES
377,32
FILE 1 PART 4
183 x 4K
FILE 9
PART 4
384 x 128

HEAD 3

ERROR HALTS AND MESSAGES

Table 4-1 lists the error halts and messages. In addition, the paragraphs on Terminal messages and MS 72 operation explain some additional errors that can occur with the Disc but are likely to occur if the optional Mag Tape is being used.

Table 4-1. Error Halts and Messages

Error Halt 102011 Core Dump — Disc Drive Operation

Check "A" register bits:

1	read/write parity error
3	bad track, cylinder cannot be used
6	drive not ready
10	DATA PROTECT lighted
14	repeat dump procedure

Error Messages

Message	Remedy
NOT REAL TIME	ADC Sample rate too fast for real-time (no points lost) ADC Throughput
ADC DMA BUSY	Press RESTART
UNIT OFF LINE	Place unit on-line
END OF TAPE OR FILE	Record number larger than max in FMTXX table
FILE PROTECTED	If Disc, turn off protect switch for upper or lower unit If Mag Tape, install write enable ring if writing operation
I/O ERROR	Repeat, run diagnostics
NO HEADER	Correct positioning in ADC Throughput file
SIZE NOT =	Set blocksize = ADC input during transcribe
BLOCKSIZE TOO BIG	Reduce blocksize. ADC Throughput requires $2 \times$ BS $\times$ no. of channels (BS $\times$ no. of channels must be < 4096)

Terminal Messages

Several Terminal messages indicate errors detected during operations.

The message I/O ERROR is the most common message indicating an error involving an operation. Usually, it indicates that bad data was detected in a record you tried to read into core; bad data records are normally not retrievable.

The message NOT REAL TIME indicates that, for some reason, the stream of ADC Throughput data recorded on the Disc or tape was interrupted, and that some data has been lost. Typically, this happens when the ADC sampling frequency has grossly exceeded the maximum real-time throughput rate for the system. It is possible to lose some data between throughput records prior to receiving this message. Refer to Table 1-1 for real-time throughput specifications.

Error During Transcribe (MS 72) Operation

The MS 72 operation copies data from the ADC Throughput File into the Data Block File, formatting it with appropriate header records while doing so. If, during a transcribe operation, an error is detected, the message I/O ERROR will be printed, followed by a printout of the current positions of the file pointers. One of two error messages will be printed, depending on which file the error occurred in. If the error was in File 2, the following message will be printed:

```
ERROR IN FILE 2 SESSION REC  <X1>
        ERROR IN FILE 2 REC  <X2>
         GOES TO FILE 1 REC  <X3>
```

where:

X1 is the record Number in the session being transcribed,
X2 is the physical File 2 record Number,
X3 is the File 1 record Number in which the error will appear.

If the error occurred during File 1 write, this message will be printed:

```
COMES FROM FILE 2 SESSION REC  <X1>
        COMES FROM FILE 2 REC  <X2>
            ERROR IN FILE 1 REC  <X3>
```

where:

X1, X2, and X3 are the same as in the first message.

The I/O error may be the result of a dropout on the tape in the ADC Throughput File.

The Transcribe command (MS 72) is the only one that provides an automatic printout of the file pointer positions following the I/O ERROR message. At other times, you can use the MS command (with no parameters) to obtain this information.

SECTION 5
BAND SELECTABLE FOURIER ANALYSIS

INTRODUCTION

This section provides operating information on software and hardware Band Selectable Fourier Analysis (BSFA). The software (or off-line) BSFA is the standard BSFA provided with the 5451C.

The Model 54470 Fourier Pre-processor is an option that provides hardware (on-line and off-line) BSFA. This option should not be confused with the the standard BSFA described first in this section. For Pre-processor (hardware) BSFA, refer to the paragraphs on hardware BSFA at the back of this section.

Band Selectable Fourier Analysis is a measurement technique in which Fourier-transform-based digital spectrum analysis is performed over a frequency band whose upper and lower frequencies are independently selectable. A comparison of BSFA and standard (Baseband) Fourier Analysis is presented in Figure 5-1. BSFA can provide an improvement in frequency resolution of more than two orders of magnitude, as well as a 10-dB increase in dynamic range, compared to Baseband Fourier Analysis.

NOTE

DAC-based excitation (Analog Output) is not compatible with BSFA operation.

INCREASED RESOLUTION

In any measurement technique, the resolution achievable in the frequency domain is determined by the length of time that the time-domain function is observed. Specifically, the frequency domain resolution is the reciprocal of the time length of the measurement ($\Delta f = 1/T$).

Standard (Baseband) Fourier Analysis provides uniform frequency resolution from dc to F_{max} (one-half the sampling frequency). Thus, the frequency resolution can also be expressed as $\Delta f = F_{max}/(N/2)$, where N is the block size — the number of samples describing the real time function. There are N/2 complex (magnitude and phase) samples in the frequency domain.

In actual practice, F_{max} is fixed by the frequencies of major experimental interest, and by aliasing considerations. Thus, the only way to improve frequency resolution in Baseband Fourier Analysis is to increase the block size. There are two reasons why this is an inefficient way to increase frequency resolution:

1. Digital processing times increase with block size.
2. The maximum system block size is limited to some relatively small number of samples, based on Processor memory size. The maximum block size in the 5451C, in any case, is 4096 words.

BSFA solves these problems by providing greatly increased resolution about points of interest in the frequency domain, without requiring an increase in the system's block size.

This is done by digitally filtering the incoming time-domain data, and storing only the filtered time-domain data, corresponding to the frequency domain band of interest. Since the frequency resolution is still the reciprocal of the time length of the incoming signal, the digital filters must process T' seconds of data to obtain a frequency resolution in the analysis band of $\Delta f' = 1/T'$. The resolution obtained in the frequency band of interest is approximately equal to:

$$(f_{high} - f_{low})/(N/2) = \text{bandwidth}/(N/2)$$

where

$$N = \text{Blocksize}$$

Thus, by restricting attention to a narrow region of interest below F_{max}, an increase in frequency resolution proportional to F_{max}/BW (where BW is the BSFA measurement bandwidth) can be obtained (see Figure 5-2).

Figure 5-1. BSFA vs. Standard Fourier Analysis

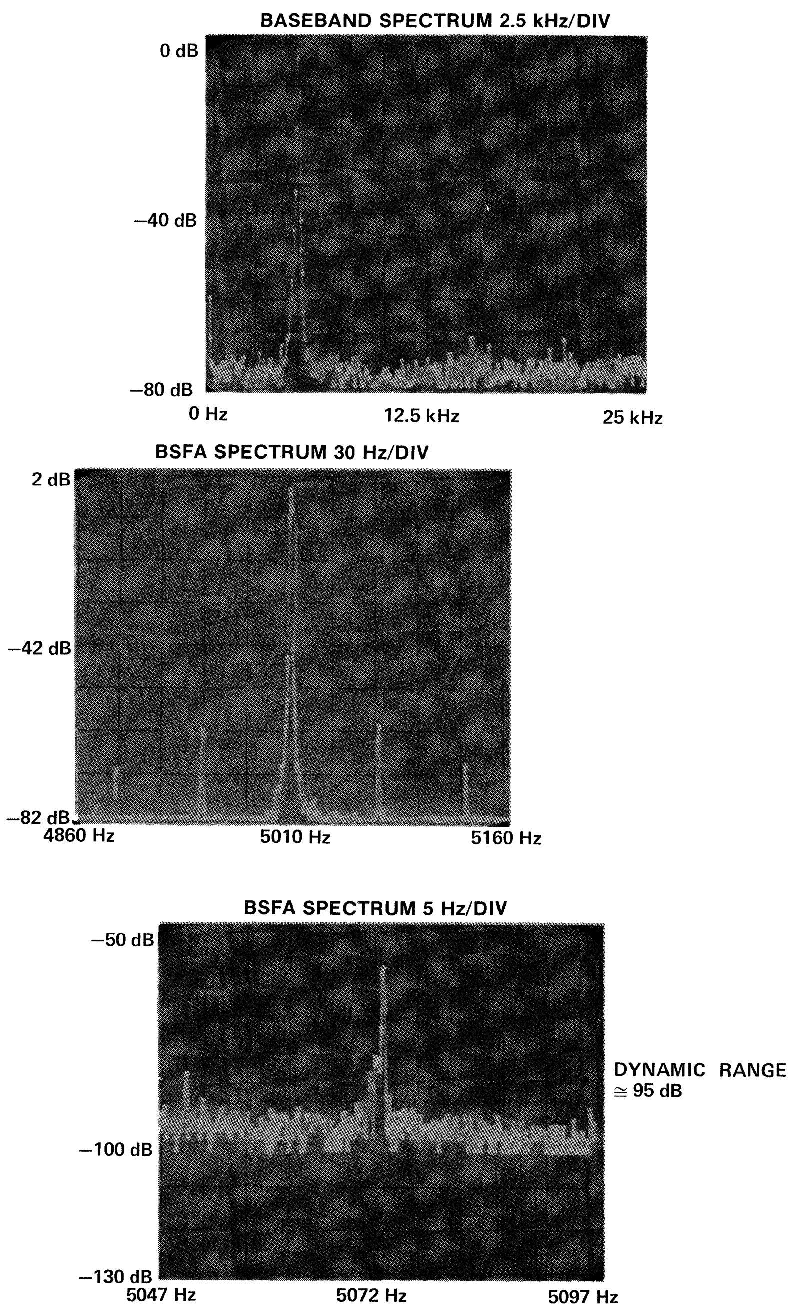
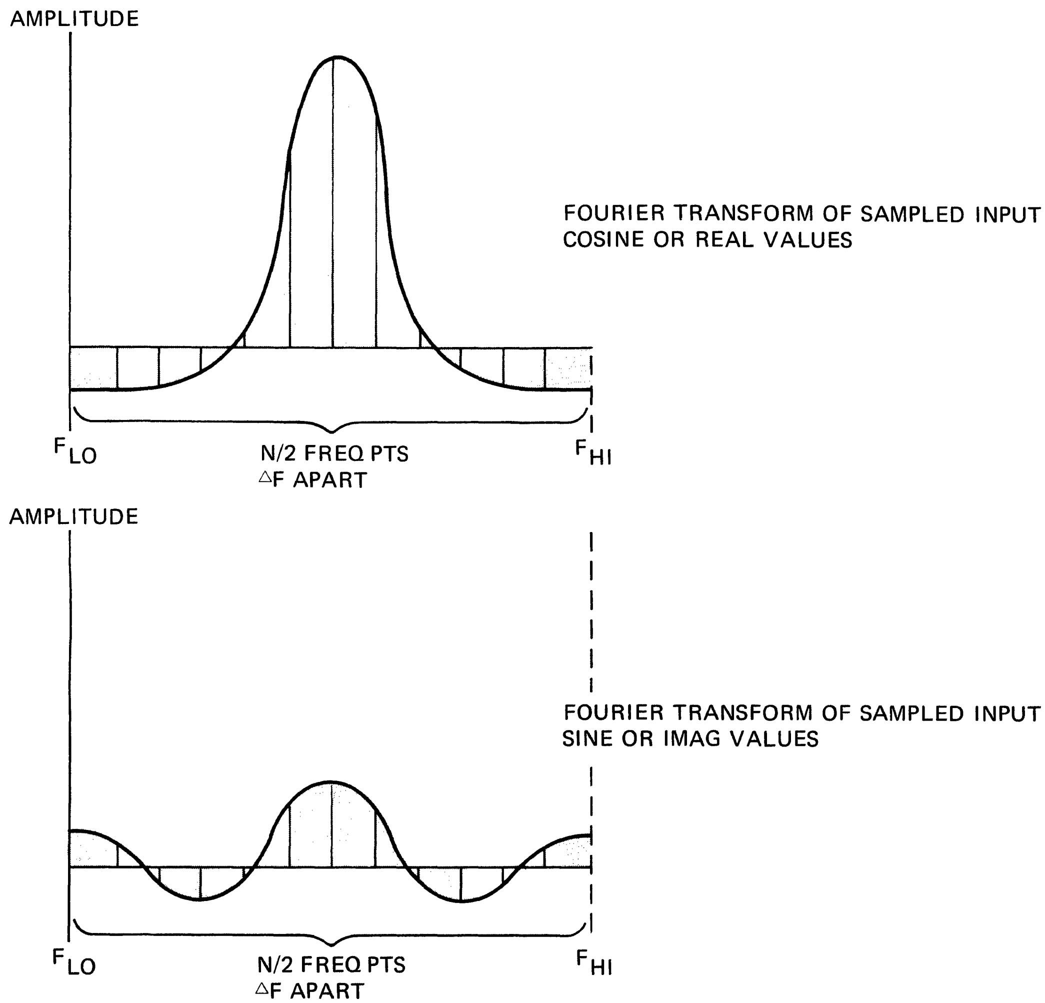


Figure 5-2. Frequency Domain Parameters For a BSFA Measurement



F_{LO} = LOWER FREQUENCY OF BSFA ANALYSIS BAND

F_{HI} = HIGH FREQUENCY OF BSFA ANALYSIS BAND

F_C = CENTER FREQUENCY = $F_{LO} + \dfrac{F_{HI} - F_{LO}}{2}$

$$\Delta F = \frac{(F_{HI} - F_{LO})}{N/2}$$

N/2 = NUMBER OF FREQUENCY POINTS

INCREASED DYNAMIC RANGE

BSFA can provide increased dynamic range relative to Baseband Fourier Analysis. This is due to the increased processor gain on the ADC quantization noise. However, a BSFA system can take advantage of this processor gain only to the extent that is made possible by the noise level and out-of-band rejection of the pre-processing digital filters. The 5451 BSFA system filters provide more than 90 dB of signal-to-noise and out-of-band rejection.

Processor gain refers to the effect of increased frequency resolution on white noise in the presence of a narrow-band signal such as a sine wave. The sine wave energy exists at a single frequency. Its peak value is, therefore, independent of the frequency resolution. The white noise peak amplitude, however, is reduced 3 dB in power each time the frequency resolution is increased by a factor of 2. BSFA provides increased processor gain by increasing frequency resolution in the analysis band (relative to the baseband measurement).

OPERATION

GENERAL

Band Selectable Fourier Analysis operation is controlled by User Programs in the system overlays.

BSFA is performed via a two-step procedure (see Figure 5-3). The first step is initialization, the second is measurement. Each of these steps is performed by calling on a specific User Program.

Initialization Programs

You select the initialization program that will set up for the operation you desire to have performed (i.e., off-line BSFA, or on-line BSFA), and give its User Program number and the parameters indicating the frequency band of interest. The program calculates the best possible fit to this request and prints out these measurement parameters.

Measurement Program

The measurement program performs the actual BSFA measurement, based on the parameters calculated by the most-recently-executed initialization program. The initialization program, once entered, need be called again only if you want to change the parameters to be used by the measurement program.

The Processor memory requirements for BSFA are of two types: fixed and variable. The fixed memory requirements are determined by the sizes of the programs included in the overlay. Additional memory is required for use as computational buffers. This space is a function of the measurement bandwidth, increasing in direct proportion with the ratio F_{max}/(BSFA measurement bandwidth). The variable space required is removed from the system data (block) space at initialization time.

You are provided with the message 'BLOCKS LEFT: n' to indicate how many data blocks of the current size remain available for use. In this manner, the resolution limit of the BSFA system is imposed (see Table 5-1. Specifications). This limit is a function of the Processor memory size. To release this variable space for other use, initialize for Baseband Fourier Analysis or press RESTART.

BSFA can be performed off-line on ADC data stored in the ADC Throughput file on the Disc. The ADC Throughput file data must be recorded at 100% real time (i.e., no samples lost).

Each version of BSFA offered for the Fourier Analyzer System has its own initialization program. There is only one measurement program. The measurement program performs the measurement specified by the parameters entered with the most-recently-executed initialization program.

Table 5-1. Specifications and Supplementary Performance Characteristics

Standard BSFA Initialization Program:
USER PROG 41 (Off-Line)
Optional (Hardware) BSFA Initialization Programs:
USER PROG 40 (On-Line)
USER PROG 43 (Off-Line)

SPECIFICATIONS

Center Frequency Range

USER PROG 41 or USER PROG 43:
dc to one-half the sample rate (up to one-half of the real-time throughput rate of the Disc or optional Mag Tape)

USER PROG 40:
dc to 50 kHz, single or dual channel (25 kHz, dual channel at lowest resolution enhancement only)

Center Frequency Resolution:

USER PROG 41:
Continuous resolution to limit of frequency accuracy for center frequency ≥.02% of sample frequency.

USER PROG 40 or USER PROG 43:
0.015% of the sampling frequency

Bandwidth Selection

USER PROG 41:
Bandwidth = sample frequency/(5 × n) n = 2,3,4 etc.

USER PROG 40 or USER PROG 43:
Bandwidth = sample frequency/(2×2^n) n = 1,2,...8.

Dynamic Range

USER PROG 41:
≥90 dB from peak out-of-band spectral component to the peak level of the pass-band noise after four ensemble averages of block size 1024*.

≥80 dB from peak in-band spectral component to the peak level of the pass-band noise after eight ensemble averages at block size 1024*.

USER PROG 40 or USER PROG 43:
Noise and spurious signals are >75 dB below full scale*.

**Reduced by 10 dB at exact center of band, due to "DC" FFT error.*

Out-of-Band Rejection

USER PROG 41:
Greater than 90 dB.

USER PROG 40 or USER PROG 43:
Greater than 80 dB.

Passband Flatness

USER PROG 41:
Without anti-aliasing filters, ±.01 dB

USER PROG 40 or USER PROG 43:
±0.05 dB

Frequency Accuracy

Equal to ADC clock accuracy.
1 part in 10^6 for HP-supplied clocks.

SUPPLEMENTARY PERFORMANCE CHARACTERISTICS

Maximum BSFA Blocksize

USER PROG 41: 1024
USER PROG 40 or USER PROG 43: 2048

Variable Data Space Required for Computational Buffers

Zoom power is (Δf baseband/Δf BSFA) with blocksize held constant.

USER PROG 41:
One throughput record (= # of channels × thruput BS) + (12.5 words × zoom power)

USER PROG 40:
One data block of current blocksize.

USER PROG 43:
One throughput record (= # of channels × thruput BS) + (one data block of current blocksize)

Max Resolution Enhancement (Zoom Power)

USER PROG 41:
> 400 (less if limited by data storage required)

USER PROG 40 or USER PROG 43:
256

Figure 5-3. Initialization and Measurement Flow

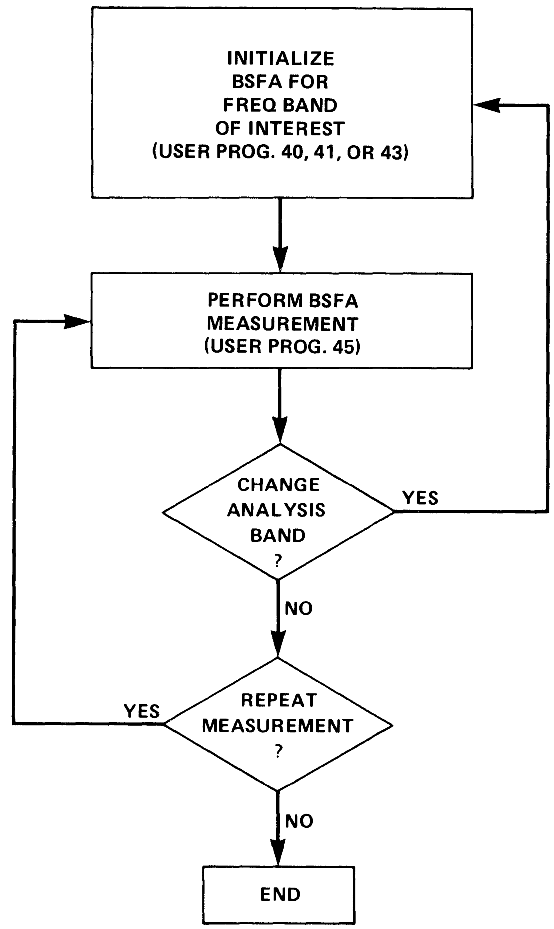


Table 5-2. BSFA User Programs

User Program	Type	Used With
40	On-line Initialization Program	Pre-processor Option
41	Off-line Initialization Program	Standard System
43	Off-line Initialization Program	Pre-processor Option
45	Measurement Program	All Systems

INITIALIZATION PROGRAM DESCRIPTION

The most-recently-executed initialization program determines the type of Band Selectable Fourier Analysis that will be performed by the measurement program. The standard initialization program (USER PROG 41) is described below; the optional initialization programs (USER PROG 40 and 43) are described at the end of this section.

USER PROG 41

This user program initializes off-line BSFA of ADC data stored in the Disc (or optional Mag Tape) ADC Throughput file. Note that you must have T seconds of time data (with no samples missing) stored in the Throughput file in order to obtain a frequency resolution $\Delta f = 1/T$. (See Applications paragraph in this section.)

USER PROG 41 allows you to initialize the processing of one channel of ADC throughput data at a time. Up to six ADC Throughput channels can be processed in any sequence by the BSFA measurement program. If there are more than six ADC channels throughput simultaneously, any six can be processed (by USER PROG 45) in any sequence after USER PROG 41 execution.

The general form of the command is:

USER PROG 41 SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

where:

n1 and n2 are used to specify the frequency band of interest, as indicated in the chart below.

n3 is used to specify the initial Disc ADC Throughput file record number. n3 is initialized to a value of zero.

n4 and n5 are additional parameters required only for the external clock input. These parameters are described in the Extended Capability paragraph of this section.

If	Then
n1 = 0 and n2–n5 defaulted, or n1 & n2 = 0	The measurement program will perform Baseband Fourier Analysis, dc to F_{max}
n1 ≥ n2	n1 specifies the center frequency of the band of interest, and n2 specifies the width of the band of interest, centered on n1. The measurement will perform BSFA.
n1 < n2	n1 specifies the lower limit of the band of interest, and n2 specifies the upper limit of the band of interest. In this case only, n1 and n2 are expressed as percentages relative to the last previous parameters calculated by USER PROG 41; thus, this combination of parameters (n1 < n2) cannot be used the first time USER PROG 41 is invoked.
n3 ≥ 0	n3 is the absolute record number, to which the Disc will be positioned. If this record is a header, the header will be read and the Disc positioned to the next record. If this record is not a header, the previous header information will be retained. For example, a position command to record 10 (Mass Store 32 Space 10) followed by a throughput command for the next 20 records of ADC channel 1 (Mass Store 22 Space 1 Space 20) would be stored on the disc as follows: **10** **11 ------throughput data ------30** **header**
n3 = -1	The Disc will be positioned to the next header. The header will be read, and the Disc will be positioned to the next record.

NOTE

The ADC Throughput File Header specified by n3 on the first USER PROG 41 execution must contain sample frequency and record size information identical to those to be used when BSFA measurement (USER PROG 45) is performed, in order to assure proper initialization.

The setup parameters entered by a USER PROG 41 command are retained until changed by a subsequent USER PROG 41 command. Defaulted (omitted) parameters are left unchanged.

After USER PROG 41 and its parameters have been accepted, you will receive the following message.

CNTR FREQ: n1 or baseband
(Center frequency of band of interest)

HZ/DIV: n2
(Horizontal calibration of display, in HZ per major division.)

ΔF: n3
(Horizontal calibration of display, in Hz per point.)

BLOCKS LEFT: n4
(Number of data blocks available, after space required for BSFA operation has been reserved.)

ZOOM POWER: n5
(Improvement in frequency resolution relative to unfiltered baseband.)

NOTE

If bit 0 of the Processor display register is set (to 1), the center frequency and horizontal calibration terms will be omitted from the readout.

If bit 1 of the Processor display register is set (to 1), there will be no readout.

For off-line BSFA, the number of blocks left is a function of F_{max}/(BSFA bandwidth) and the ADC Throughput record size. Increasing F_{max}/(BSFA bandwidth) and/or increasing the ADC Throughput record size results in fewer blocks left (see Table 5-1. Specifications).

USER PROG 41 Examples

The sequence of USER PROG 41 examples below, indicates how you can select an optimum analysis band, through a series of BSFA measurements. It is assumed that the measurement program (USER PROG 45) is executed following each USER PROG 41 example. (USER PROG 45 is described and examples provided later in this section.)

1. First set the ADC for single channel, F_{max} = 10 kHz, and then perform an ADC Throughput operation.

   BLOCK SIZE 2048 ENTER

   MASS STORE 32 SPACE 0 ENTER

   MASS STORE 22 SPACE 1 SPACE 50 ENTER

   BLOCK SIZE 512 ENTER

2. Then initialize off-line BSFA for a baseband measurement.

   USER PROG 41 SPACE 0 ENTER

3. After performing the measurement based on the initialization command given in step 2, observe a region of interest (resonance, harmonic, order, etc.) at 5 kHz. Now initialize for off-line BSFA with a bandwidth of 1 kHz, centered on 5 kHz starting at record 0.

   USER PROG 41 SPACE 5000 SPACE 1000 ENTER

4. After performing the BSFA measurement initialized in step 3, observe that the actual region of interest lies between the 5th and 7th major oscilloscope divisions (50% and 70% relative to the previous measurement). BSFA is initialized to cover this region as shown below starting at record 0.

   USER PROG 41 SPACE 50 SPACE 70 ENTER

   For examples of measurement Keyboard Programs, see the Applications paragraphs of this section.

   For additional USER PROG 41 capabilities, see Extended Capabilities paragraph of this section.

THE MEASUREMENT PROGRAM (USER PROG 45)

This User Program causes a BSFA measurement to be made, based on the parameters provided by the most-recently-executed initialization program (i.e., USER PROG 41). Time-domain data is processed; the processing provides a frequency-domain (linear spectrum) result. On the standard BSFA, the lowest and highest 10% of the frequency domain output of USER PROG 45 is set to zero, to remove aliased frequency components. Similarly, for filtered baseband mode, the highest 20% of the frequency domain output is set to zero for aliasing.

The general form of the measurement command is:

USER PROG 45 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

where:

n1 specifies the number of the data block where the result from processing the first ADC channel is to be stored. In simultaneous two-channel procesing, the result from processing the second ADC channel is stored in block n1 + 1.

NOTE

One additional working block is required for each result data block. In single-channel operation, the working block is n1 – 1. In simultaneous two-channel operation, blocks n1 – 2 and n1 – 1 are required as working blocks. Your n1 entry should allow for this. Also, in offline 2-channel operation, you should work from the highest value n1 to the lowest value to ensure good data is not destroyed in the working blocks.

n2 specifies the type of window to be used, as indicated below

If	Then
$0 \leq n2 \leq 10$	n2 is the number of times the HANNing window will be applied to the data.
$n2 < 0$	The time window to be applied is stored in block –n2. (In this manner, the operator can specify any time window desired.)

n3 specifies the ADC channel in the ADC Throughput file that is to be processed by USER PROG 45.

1 = Channel A
2 = Channel B

and so forth.

NOTES

In On-line BSFA, n3 specifies the data block to be displayed.

USER PROG 45 processes only one ADC Throughput channel at a time. Thus, performing multi-channel analysis requires multiple application of USER PROG 45 commands, which differ only in the specification of USER PROG 45 n3 value. For more information see the Applications paragraph in this section.

n4 is described in the Extended Capabilities paragraph of this section.

NOTE

Parameters entered by a USER PROG 45 command are retained until changed by a subsequent USER PROG 45 command. Defaulted parameters are left unchanged.

Each time USER PROG 45 is asked to process a given channel (parameter n3) in the ADC Throughput file, it starts processing one sample beyond the last sample of the channel previously processed by USER PROG 45 (unless the position of the Throughput file pointers has been modified by invoking the Initialization program). Up to six channels can be processed alternately in this manner after execution of an initialization program. (See Applications paragraph.)

The ADC REPEAT scan switch is operable only when USER PROG 45 has been initialized to perform Baseband Fourier Analysis on data coming directly from the ADC.

Examples of USER PROG 45 in actual measurement situations are given in the Applications paragraph.

For additional USER PROG 45 capability, see the Extended Capability paragraph.

BSFA Qualifier Storage

When USER PROG 45 is used to perform BSFA measurements, it does the following:

1. Stores the floating point values of the Center Frequency (CF) and Delta Frequency (Δf) of the BSFA measurement(s) into reserved locations in the core-resident data block header (see Section 4 for further details and format of the data block header).

2. Assigns all BSFA measurement data blocks a special frequency code of 99.

   The special frequency code of 99 signifies that the frequency axis of the data block is not to be determined by the settings of the ADC (as is usually the case), but rather from the center frequency and Δf parameters stored in the core-resident data block header.

In contrast to the normal mode of data block qualifier storage where each data block has an individual set of qualifiers, BSFA measurements in core have only one set of "global" qualifiers. This means that all BSFA data blocks in core are assumed to have the same qualifiers.

Therefore, two BSFA measurements with different qualifiers (i.e., different center frequency and Δf from two different BSFA measurement sessions) will be assumed to have the same qualifiers (the qualifiers currently residing in the core-resident data block header). This will result in an error if the two measurements are compared via plotting, or using the cursor where the frequency axis calibration must be printed out.

The BSFA qualifiers associated with any BFSA currently in core may be saved by saving the relevant data blocks on the Disc. This is possible since the core-resident data block header containing the BSFA qualifiers will be stored to the Disc along with the data. These blocks can later be read from the Disc, thereby restoring the "global" BSFA qualifiers to the values for that data.

RULE: The BSFA global qualifiers in core (from which all data with frequency code = 99 are calibrated) are either:

a. The qualifiers (CF and Δf) from the last call to USER PROG 45;

or

b. The qualifiers from the last data block read from the Disc, whichever was done last.

APPLICATIONS — STANDARD OFF-LINE BSFA

The examples presented in this section include the MASS STORE commands required to record time-domain data in the ADC Throughput file on the Disc (or optional Mag Tape). The ADC Throughput operation (i.e., putting ADC data into the Throughput file) requires up to five data blocks for each ADC channel being used. You must be careful not to write-over intermediate results during the throughput operation.

The amount of data stored in the ADC Throughput file limits the frequency resolution obtainable in the BSFA measurement ($\Delta f = 1/T$). If resolution is to be improved by a factor of 100, relative to a Baseband measurement, then the number of samples that must be recorded is 100 times the Block Size, in real time, with no samples missing. The exact number of ADC Throughput file data records required for a given bandwidth and F_{max} is:

$$\text{Number of Throughput Records} = \text{Zoom Power} \times \frac{\text{BSFA Blocksize}}{\text{Throughput Blocksize}} \times \text{Number of Averages}$$

$$\text{Zoom Power} = \frac{F_{max}}{\text{Bandwidth}}$$

For example, if Zoom Power = 20, the baseband (throughput) blocksize = 2048 and the BSFA block-size = 256, then,

$$\text{Number of Records} = 20 \times \frac{256}{2048} \times \text{Number of Averages}$$

$$= 20 \times 1/8 \times \text{Number of Averages}$$

$$= 2.5 \times \text{Number of Averages}$$

Note that the BSFA measurement block size can be different than the block size used during the throughput operation. Choosing a large block size for the throughput operation maximizes the real-time throughput rate, as well as the amount of data stored in the Throughput file (at the expense of increasing the computational buffer space required for BSFA operation). For storage considerations, see Table 5-1. Specifications.

NOTE

In the following program listings, the correlation between the printout abbreviation and the Keyboard pushbuttons is explained in Appendix B.

The program listings below are intended to serve only as examples, and are not intended to illustrate all possible BSFA keyboard programs.

1. Single-channel Power Spectrum Average

   a. The program listing below illustrates a continuously-excited response into the ADC Throughput file. This long time record is then divided into shorter time records, which are processed by BSFA and averaged.

L 0	Establish Label 0.
BS 2048	Set data block size for throughput operation.
MS 32	Position ADC Throughput file to record 0.
MS 22 1 60	Perform ADC Throughput operation for one ADC channel, 60 records.
Y 41 n1 n2 0	Initialize BSFA to record 0 where n1 and n2 define the analysis band. Some data for the BSFA operation must be put through the ADC to the mass storage device before the off-line BSFA initialization program is called so that the header contains current information.
L 1	Establish label 1.
BS 256	Set data block size for BSFA operation.
CL 2	Clear data block 2
L 2	Establish Label 2.
Y 45 1 1 1	Perform BSFA into block 1, perform one HANNING operation on channel 1.
D 2 4	Single sweep display of block 2 (this step is optional, it may be omitted).
SP 1	Perform power spectrum average.
# 2 8	Perform steps between label 2 and here a total of eight times.
.	End of program.

This program can be operated by giving the command:

JUMP 0 ENTER

Data block 0 is used as a working block by USER PROG 45.

The actual measurement band is determined by the state to which USER PROG 45 has been initialized by the most-recently-executed initialization program.

b. The program listing below illustrates placing several transient responses on the ADC Throughput file; each response is several throughput records long. Power spectrum average is performed on the stored data.

L 0	Establish label 0.
BS 1024	Set data block size for ADC Throughput operation.
MS 32	Position ADC Throughput file to record 0.
L 1	Establish label 1.
MS 22 1 10	Perform ADC Throughput operation one ADC channel, 10 records (on ADC trigger).
# 1 5	Perform steps from label 1 to here a total of 5 times.
BS 512	Change block size.
L 2	Establish label 2.
Y 41 n1 n2 0 (or Y 141 0)	Initialize BSFA to record 0. Parameters n1 and n2 in the "Y41" lines define the BSFA analysis band. Note that n1 must be $\geq$ n2 for this measurement, i.e., you must call Y41 with center frequency and bandwidth instead of % screen divisions. (to position only)
CL 2	Clear data block 2 (used for G_{xx} average).
L 3	Establish label 3.
Y 45 1 0 1	Perform BSFA into data block 1, no HANNING operation.
D 2 4	Single sweep display block 2 (this step is optional, it can be omitted).
SP 1	Perform power spectrum average.
Y 41 n1 n2 -1 (or Y 141 -1)	Position to next header. (to position only)
# 3 5	Perform steps between label 3 and here a total of 5 times.
•	End of program.

This program can be operated by giving the command:

JUMP 0 ENTER

Block 0 is used as a working block.

c. The program listing below illustrates entering new data (either transient or continuous) into the ADC Throughput file for each average. The total time record is not stored at once.

L 0	Establish label 0.
CL 0	Clear data block 0.
MS 31	Position to start of data block file.
MS 21 0	Write Block 0 to file 1, record 0 (this is the averaging block).
L 1	Establish label 1.
Y 41 0	Release BSFA buffers (this step is optional, it allows more data space for throughput operations).
MS 32	Position to start of ADC Throughput file.
MS 22 1 50	Perform ADC Throughput for one ADC channel, 50 records.
MS 31	Position to start of data block file.
MS 11 2	Read from data block file into data block 2.
Y 41 n1 n2 0 (or Y 141 0)	Initiate BSFA operation (n1 and n2 specify analysis band). (to position only)
Y 45 1 1 1	Perform BSFA operation into block 1, perform one HANNING operation, for ADC channel 1.
D 2 4	Single sweep display of block 2 (this step is optional, it can be omitted).
SP 1	Power spectrum average.
MS 31	Position to start of file 1.
MS 21 2	Write average (block 2) into data block file.
# 1 8	Perform steps between label 1 and here a total of eight times.
.	End of program.

This program can be operated by giving the command:

JUMP 0 ENTER

2. Double-precision tri-spectrum average, and transfer and coherence function

Dual channel operation requires two USER PROG 45 commands, one for each channel. The operator must direct the first channel processed into the next higher numbered block relative to the result of the second channel processed in order to allow use of the POWER SPECT command (see example program listings). In the first example below, BSFA is performed on data in Block 2 before it is performed on data in Block 1.

a. This program listing illustrates recording data from two ADC channels in the ADC Throughput file. The input data for one of the ADC channels is the input signal to a device under test; the input signal for the other channel is the output signal of the device under test. The input and output signals of the device under test are continuous during the period during which the recording is made, and no data points are lost. Long time records are used during the recording period; these records are later sub-divided, and spectral quantities averaged.

L	0				Establish label 0.
BS	1024				Set data block size for ADC Throughput operation.
MS	32				Position to start of ADC Throughput file.
MS	22	2	60		Perform ADC Throughput operation, for two ADC channels, for 60 records.
Y	41	n1	n2	0	Initiate BSFA operation (see previous program).
L	1				Establish label 1.
BS	512				Set block size for BSFA operation.
CL	3				Clear data block 3 (used for G_{xx} average).
CL	4				Clear data block 4 (used for G_{yy} average).
CL	5				Clear data block 5 (used for G_{yx} average — real part).
CL	6				Clear data block 6 (used for G_{yx} average — imaginary part).
L	2				Establish label 2.
Y	45	2	1	2	Perform BSFA on Channel 2, into block 2; do one HANNING operation (block 0 is used as a working block).
Y	45	1	1	1	Perform BSFA on channel 1, into block 1; do one HANNING operation (block 0 is used as a working block).
D	5	4			Single sweep display of block 5 (this step is optional, it can be omitted).
SP	1	2	2		Perform double-precision power spectrum average.
#	2	8			Perform steps between label 2 and here a total of eight times.
CH	1	2	2		Perform transfer and coherence functions.
.					End of program.

This program can be operated by giving the command:

JUMP 0 ENTER

The actual measurement band is determined by the state to which USER PROG 45 has been initialized by the most-recently-executed initialization program.

b. The program listing below illustrates recording several transient excitation and response sessions, then analyzing them.

When using transient excitation, the two lines denoted by the stars (*) govern the maximum amount of resolution enhancement and the maximum number of averages obtainable. In this example, the throughput operation records six records for each ADC channel and nine such sessions are recorded.

Example: F_{max} = 5000 Hz — BSFA Bandwidth = 500 Hz
Throughput Blocksize = 1024 — BSFA Blocksize = 512

Thus the resolution enhancement due to BSFA in this example is:

$$\text{Zoom Power} = .8 \times \frac{F_{max}}{\text{BSFA Bandwidth}} = .8 \left(\frac{5000}{500}\right) = 8$$

and the total number of records necessary for 9 averages is:

$$\text{Number of records} = \left[\begin{array}{c}\text{Zoom}\\ \text{Power}\end{array}\right] \times \left[\frac{\text{BSFA Blocksize}}{\text{Throughput Blocksize}}\right] \times \text{Number of Averages}$$

= 8 (1/2) (9) = 4 (9) = 36 records of blocksize 1024 to achieve a BSFA resolution enhancement of 8 with 9 averages. The program below records 54 total records (6 records × 9 sessions) and adequately meets the criteria.

L 0	Establish label 0.
BS 1024	Set data block size for ADC Throughput operation.
MS 32	Position to start of Throughput file.
L 1	Establish label 1.
* MS 22 2 6	Perform ADC Throughput operation, for two ADC channels, six records.
* # 1 9	Perform steps between label 1 and here 9 times.
BS 512	Set data block size for analysis.
Y 41 n1 n2 0	Initialize BSFA to record 0 (n1 and n2 specify the actual measurement band).
(or Y 141 0)	(to position only)
CL 3	Clear data block 3 (used for G_{xx} average).
CL 4	Clear data block 4 (used for G_{yy} average).
CL 5	Clear data block 5 (used for G_{yx} average — real part).
CL 6	Clear data block 6 (used for G_{yx} average — imaginary part).
L 3	Establish label 3.
Y 45 2 0 2	Perform BSFA into block 2 (block 1 used for scratch), no HANNING, ADC channel 2 data.
Y 45 1 0 1	Perform BSFA into block 1 (block 0 used for scratch), no HANNING, ADC channel 1 data.
D 5 4	Single sweep display of block 5 (this step is optional, it can be omitted).
SP 1 2 2	Perform power spectrum average.
Y 41 n1 n2 -1	Initialize BSFA to next header (n1 and n2 specify the actual measurement band).
(or Y 141 -1)	(to position only)
# 3 9	Perform steps between label 3 and here a total of 9 times.
.	End of program.

To operate this program, give the command:

JUMP 0 ENTER

c. This program listing illustrates entering new data (either transient or continuous) into the ADC Throughput file for each average.

L 0	Establish label 0.
CL 0	Clear block 0.
MS 31	Position to start of data block file.
MS 21 0	Write block 0 into file 1, next record (0) (G_{xx} average).
MS 21	Write block 0 into file 1, next record (1) (G_{yy} average).
MS 21	Write block 0 into file 1, next record (2) (G_{yx} average — real part).
MS 21	Write block 0 into file 1, next record (3) (G_{yx} average — imaginary part).
L 1	Establish label 1.
Y 41 0	Release BSFA buffers (this step is optional, it allows more data blocks for ADC Throughput operation).
MS 32	Position to start of ADC Throughput file.
MS 32 2 50	Perform ADC Throughput operation, from two ADC channels, 50 records.
MS 31	Position to start of data block file.
MS 11 3	Read file 1, record 0 into block 3 (G_{xx} average).
MS 11 4	Read file 1, next record (1) into block 4 (G_{yy} average).
MS 11 5	Read file 1, next record (2) into block 5 (G_{yx} average — real part).
MS 11 6	Read file 1, next record (3) into block 6 (G_{yx} average — imaginary part).
Y 41 n1 n2 0 (or Y 141 0)	Initialize BSFA to start of ADC Throughput file (n1 and n2 specify the actual measurement band). (to position only)
Y 45 2 1 2	Perform BSFA into block 2, perform one HANNING operation, use ADC channel 2 data (block 1 is used for scratch).
Y 45 1 1 1	Perform BSFA into block 1 (block 0 is used for scratch), perform one HANNING, use ADC channel 1 data.
D 5 4	Single sweep display of block 5 (this step is optional, it can be omitted).
SP 1 2 2	Perform double-precision power spectrum average.
MS 31	Position to start of data block file.
MS 21 3	Write block 3 into file 1, next record (0).

MS	21	4		Write block 4 into file 1, next record (1).
MS	21	5		Write block 5 into file 1, next record (2).
MS	21	6		Write block 6 into file 1, next record (3).
#	1	4		Perform steps between label 1 and here a total of 4 times.
CH	1	2	2	Perform transfer and coherence functions.
.				End of program.

This program can be operated by giving the command:

JUMP 0 ENTER

The MASS STORE ENTER command can be used to locate the position of any of the six ADC Throughput channel pointers. Simply use this command after a

USER PROG 45 SPACE n1 SPACE n2 SPACE n3 ENTER

command. The third (ADC Throughput file) pointer value readout on the Terminal by the MASS STORE command is the number of the next record to be read by throughput channel n3. Then use a

USER PROG 41 SPACE n1 SPACE n2 SPACE n3 ENTER

command followed by a MASS STORE ENTER command to show the Throughput file pointer position for all six BSFA Throughput channel pointers.

OPTIONAL HARDWARE BSFA

This option provides real time (on-line) BSFA using the 54470A Fourier Pre-processor (FPP). In addition, the off-line processing of ADC Throughput files stored on the Disc can be performed at a rate significantly faster than that possible with the standard system.

Optional Initialization Program Format

The most-recently-executed initialization program determines the type of BSFA Fourier Analysis that will be performed by the measurement program. Each optional initialization program is discussed below.

USER PROG 40

This user program initializes a real-time BSFA measurement of ADC data. For a real-time single or dual channel measurement, the maximum ADC sample rate is 100 kHz (F_{max} = 50 kHz).* USER PROG 40 allows the ADC to run one or two channels of input data. The number of channels is taken into account by program and need not be set at initialization time.

The general form of this command is

USER PROG 40 SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

where:

n1 to n3 specify the frequency band of interest and n4 and n5 are special parameters covered in the Extended Capabilities paragraph.

*Except at the lowest resolution enhancement (ZOOM POWER = 2).

If	Then
n1 to n5 defaulted	No change in current setup. The current setting of the 54470A FPP is printed for the information of the user.
n1 is any value and n2 to n5 defaulted	The 54470A FPP is set to a baseband throughput mode with an F_{max} as indicated by the ADC settings as if a straight ADC input were used.
$n1 < n2$	The BSFA band is defined to be n1% to n2% of the last frequency range as covered on the CRT display.
$n1 > n2$ n3 defaulted	n1 = center frequency in Hz. n2 = bandwidth in Hz.
$n1 > n2$ n3 given	Center frequency = $n1 \times 10^{n2}$ n3 = bandwidth in Hz

After USER PROG 40 has been entered, a printout will indicate the current setting of the 54470A FPP. If bit 0 of the Processor register is set to 1, the printout will be eliminated.

USER PROG 40 EXAMPLES

The following sequence of USER PROG 40 examples indicates how you might adjust the measurement band of interest.

As an example, start with the ADC set to the Δf, F_{max} of 5 kHz. The command

USER PROG 40 SPACE 0 ENTER

will set for a direct input from the ADC with no filtering. The parameters will be set to:

CNTR FREQ: BASEBAND
HZ/DIV: 500 Hz
ΔF: 5000/(Block Size/2)
ZOOM POWER: 1

As in all direct measurements, signals above F_{max} (5000 Hz) will alias back as frequencies below 5000 Hz unless they are eliminated with an anti-aliasing filter. The safest practice is to set the filter to 1/2 F_{max} (2500 Hz) so that all components that can be aliased back into frequencies below 1/2 F_{max} (2500 Hz) will be attenuated by >75 dB, a value in agreement with the dynamic range of the 54470A FPP. Thus in the Baseband Throughput mode, only frequencies up to 1/2 F_{max} are usable.

The command

USER PROG 40 SPACE 0 SPACE 25 ENTER

will set the analysis bandwidth to 0 to 25% of the previous range (i.e., from 0 to 1250 Hz). This is an example of digital lowpass filtering.

The printout will now indicate:

CNTR FREQ: BASEBAND
HZ/DIV: 125
ΔF: 1250/(Block Size/2)
ZOOM POWER: 4 (filter set to reduce bandwidth by 4)

Since a complex (dual block) transform is used (as explained under Measurement Program Description) when any filtered baseband or BSFA measurement is made, all the resulting data is valid. In this case, the total data from 0 to 1250 Hz is valid and does not contain aliasing terms if the analog filter on the input is set to 1/4 the ADC sample rate of 10 kHz. That is, the aliasing filter should remain set to 2500 Hz. The digital filtering of the 54470A FPP will now remove components between 1250 and 2500 Hz.

The command

USER PROG 40 SPACE 50 SPACE 100 ENTER

will result in a BSFA analysis band which is from 50 to 100% of the last settings. This will give:

CNTR/FREQ: 937.5 Hz
HZ/DIV: 62.5 Hz
ΔF: 625/(Block Size/2)
ZOOM POWER: 8

The command

USER PROG 40 SPACE 1250 SPACE 625 ENTER

will set a BSFA measurement independent of the last setting with

CNTR/FREQ: 1250 Hz
HZ/DIV: 62.5
ΔF: 625/(Block Size/2)
ZOOM POWER: 8

This same command could be given as

USER PROG 40 SPACE 125 SPACE 1 SPACE 625 ENTER

or as

USER PROG 40 SPACE 12500 SPACE -1 SPACE 625 ENTER

NOTE

If your system contains the high-speed ADC and you are operating at the maximum frequency (200 kHz sample rate for single channel or 100 kHz for dual channel) a Zoom Power of more than 2 must be used or a "DL WHAT?" will result.

USER PROG 43

This user program initializes off-line BSFA of ADC data stored in the mass storage device (Disc or Mag Tape) ADC Throughput file. Note that you must have T seconds of time data (with no samples missing) stored in the Throughput file in order to obtain a frequency resolution $\Delta f = 1/T$.

USER PROG 43 allows you to initialize the processing of one channel of ADC Throughput data at a time. Up to six ADC Throughput channels can be processed in any sequence by the BSFA measurement program. If there are more than six ADC channels throughput simultaneously, any six can be processed (by USER PROG 45) in any sequence after USER PROG 43 execution.

The general form of the command is:

USER PROG 43 SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

where:

n1 and n2 are used to specify the frequency band of interest, as indicated in the chart below.

n3 is used to specify the initial Mass Store Throughput file record number, as indicated in the second chart below. n3 is initialized to a value of zero.

n4 and n5 are additional parameters, and may be omitted. These parameters are described in the Extended Capabilities paragraph.

If	Then
n1 = 0	The measurement program will perform throughput (no filter) Baseband Fourier Analysis.
n1 ≥ n2	n1 specifies the center frequency of the band of interest, and n2 specifies the width of the band of interest, centered on n1. The measurement program will perform BSFA.
n1 < n2	n1 specifies the lower limit of the band of interest, and n2 specifies the upper limit of the band of interest. In this case only, n1 and n2 are expressed as percentages relative to the last previous parameters calculated by USER PROG 43; thus, this combination of parameters (n1 < n2) cannot be used the first time USER PROG 43 is invoked.

n3 is used to specify the initial Mass Store Throughput file record number, as indicated in the chart below. n3 is initialized to a value of zero. (Positions all six ADC Throughput channel pointers simultaneously.) If defaulted, n3 assumes its last value (initially zero).

If	Then
n3 ≥ 0	n3 is the absolute record number, to which the Mass Store device will be positioned. If this record is a header, the header will be read and the Mass Store device positioned to the next record. If this record is not a header, the previous header information will be retained. For example, a position command to record 10 (Mass Store 32 Space 10) followed by a throughput command for the next 20 records of ADC channel 1 (Mass Store 22 Space 1 Space 20) would be stored on the disc as follows: **10** | **11 — — — — — — — — — 30** **header** **throughput data**
n3 = -1	The Mass Store device will be positioned to the next header. The header will be read, and the Mass Store device will be positioned to the next record.

NOTE

In order to assure proper initialization, the Throughput File Header specified by n3 on the first USER PROG 43 execution must contain sample frequency and record size information identical to those to be used when the BSFA measurement (USER PROG 45) is performed.

The setup parameters entered by a USER PROG 43 command are retained until changed by a subsequent USER PROG 43 command.

After USER PROG 43 and its parameters have been accepted, the following message is provided:

CNTR FREQ: n1 or BASEBAND
(Center frequency of band of interest.)

HZ/DIV: n2
(Horizontal calibration of display, in Hz per major division.)
ΔF: n3
(Horizontal calibration of display, in Hz per point.)

BLOCKS LEFT: n4
(Number of data blocks available, after space required for BSFA operation has been reserved.)

ZOOM POWER:
(Resolution improvement. This is a power of 2 from 1 to 256.)

NOTE

If bit 0 of the Processor display register is set to 1, the printout is suppressed.

USER PROG 43 EXAMPLES

The sequence of USER PROG 43 examples below indicates how you can select an optimum anlysis band through a series of BSFA measurements. It is assumed that the measurement program (USER PROG 45) is executed following each USER PROG 43 example. (USER PROG 45 is described and examples provided in this section.)

1. First set the ADC for single channel, $F_{max} = 10$ kHz, and perform an ADC Throughput operation.

   BLOCKSIZE 2048 ENTER

   MASS STORE 32 SPACE 0 ENTER

   MASS STORE 22 SPACE 1 SPACE 50 ENTER

   BLOCKSIZE 512 ENTER

2. Then initialize off-line BSFA for a baseband measurement.

   USER PROG 43 SPACE 0 ENTER

3. After performing the measurement based on the initialization command given in step 2, observe a region of interest (resonance, harmonic, order, etc.) at 5 kHz. Now initialize for off-line BSFA with a bandwidth of 1 kHz, centered on 5 kHz starting at header record 0.

   USER PROG 43 SPACE 5000 SPACE 1000 SPACE ENTER

4. After performing the BSFA measurement initialized in step 3, observe that the actual region of interest lies between the 5th and 7th major oscilloscope divisions (50% and 70% relative to the previous measurement). BSFA is initialized to cover this region as shown starting at header record 0.

   USER PROG 43 SPACE 50 SPACE 70 ENTER

   For example measurements of keyboard programs, refer to the Optional Applications paragraphs.

   For additional USER PROG 43 capabilities, refer to the Extended Capabilities paragraphs.

OPTIONAL APPLICATIONS

This subsection provides examples of BSFA keyboard programs and other relevant application information. This subsection is divided into on-line BSFA and off-line BSFA operations.

Optional On-Line BSFA

The following is an example of a keyboard program for single-channel power spectrum average. USER PROG 40 must be called to initialize the 54470A FPP. The ADC is set to single channel operation when this program is run.

L 0	Establish label 0
CL 2	Clear data block 2 (used for averaging)
L 1	Establish label 1
Y 45 1 1	Perform BSFA into block 1, perform one HANNING operation. Display of block 2 (power spectrum) while inputting this parameter is optional for on-line BSFA.
SP 1	Perform Power Spectrum Averaging.
# 1 8	Perform steps between label 1 and here a total of 8 times.
.	End of Program.

This program is operated by giving the keyboard command:

JUMP 0 ENTER

Functionally, USER PROG 45 takes the place of ANALOG INPUT, HANNING, and FOURIER TRANSFORM commands.

The actual measurement band is determined by the state to which USER PROG 45 has been initialized by the most-recently-executed initialization program.

The following to an example of a keyboard program for double-precision tri-spectrum average, and coherence function. The ADC must be set for dual-channel operation for initialization and measurement.

L 0	Establish label 0.
CL 4	Clear data block 4 (used for G_{xx} average).
CL 5	Clear data block 5 (used for G_{yy} average).
CL 6	Clear data block 6 (used for G_{yx} average — real part).
CL 7	Clear data block 7 (used for G_{yx} average — imaginary part).
L 1	Establish label 1.
Y 45 2 1 4	Perform BSFA into blocks 2 and 3, perform one HANNING operation. Display of block 4 while inputting this parameter is optional for on-line BSFA.
SP 2 2 2	Perform double-precision power spectrum average.
# 1 4	Perform steps between label 1 and here a total of 4 times.
CH 2 2 2	Perform transfer and coherence function.
	End of program.

This program is operated by giving the command:

JUMP 0 ENTER

Note that data blocks 0 and 1 are used for working blocks by USER PROG 45.

The actual measurement band is determined by the state to which USER PROG 45 has been initialized by the most-recently-executed initialization program. One call to USER PROG 45 is sufficient for 2-channel operation of BSFA (off-line BSFA requires two calls).

Optional Off-Line BSFA

The only difference in the standard and optional off-line BSFA programs is the User Program number of the initialization program. Refer to the standard off-line BSFA examples and Table 5-2.

EXTENDED CAPABILITY

The system BSFA overlays include User Programs and user capabilities not described in detail in the Operation paragraph in this section. While these additional programs, and program parameters, are not required for BSFA operation, they add capabilities and flexibility beyond that provided by the basic BSFA programs described in the Operation paragraph.

Additional User Programs

USER PROG 44

This program allows you to re-define the horizontal calibration of the most recent BSFA measurement. This is accomplished by means of an automatic partial block display. USER PROG 44 is useful for interpreting and recording (by photograph or plotting) the results of BSFA measurements.

The general form of the command is

USER PROG 44 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

where:

n1 is the number of the block on which USER PROG 44 is to operate.

$n2 \times 10^{n4}$ is the desired center frequency of the display, in Hz.

$n3 \times 10^{n4}$ is the desired horizontal calibration of the display, in Hz per major division.

NOTE

Defaulted parameters are left unchanged.

USER PROG 44 evaluates the inputs and performs the appropriate partial block display of the block specified by n1. The accuracy of USER PROG 44 is $\pm\ \Delta f/2$.

USER PROG 3017

This program performs a complex time to complex frequency fast Fourier transform.

The general form of the command is

USER PROG 3017 SPACE n1 ENTER

where:

n1 is used to locate the data blocks to be transformed, as indicated in the chart below.

If	Then
The data to be transformed is in the time domain.	n1 is the number of the block containing the real part of the time function. The imaginary part is in block n1+1. The frequency domain result is stored with the negative frequencies and dc in block n1 and positive frequency components in block n1+1. Frequency increases from left to right (displayed).
The data to be transformed is in the frequency domain.	n1 is the number of the block containing the negative frequency components and dc. The positive frequency components are in block n1+1. The time domain result is stored with the real part in block n1 and the imaginary part in block n1+1.

NOTE

Scale factors and calibrators of blocks n1 and n1+1 must be equal. The system blocksize must be less than or equal to 2048.

USER PROG 141

This program positions the ADC Throughput file for processing by Y45.

The general form of the command is

USER PROG 141 SPACE n1 SPACE n2 ENTER

where:

If	Then
$n1 \geq 0$ n2 defaulted	Position throughput file to record n1.
$n1 < 0$ n2 defaulted	Position throughput file to next header.
n2 given	n1 = relative record number.

USER PROG 41

Only parameters n4 and n5 of USER PROG 41 are explained in this paragraph.

The general form of this command is

USER PROG 41 SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

where:

n1, n2, and n3 are discussed in the Operation paragraph of this section.

n4 and n5 are used to specify the F_{max} of the ADC Throughput session, if the session was recorded with the ADC in the EXT clock position. Use of n4 and n5 is indicated in the chart below. n4 is initialized to a value of zero.

If	Then
$n4 = 0$	The sample rate of the throughput session is read from the header.
$n4 > 0$	The value of F_{max} for the throughput session is $n4 \times 10^{n5}$. (This entry is used when the ADC clock is in the EXTernal position.) n5 must be greater than 0.

EXAMPLE:

To initialize for off-line BSFA, using data recorded with an external (25 kHz) clock (F_{max}) = 12.5 kHz, the command is:

USER PROG 41 SPACE n1 SPACE n2 SPACE n3 SPACE 125 SPACE 2 ENTER

USER PROG 45

Only parameter n4 of USER PROG 45 is explained in this paragraph.

The general form of this command is

USER PROG 45 SPACE N1 SPACE N2 SPACE N3 SPACE N4 ENTER

where:

n1, n2, and n3 are discussed in the Operation paragraph of this section.

n4 is used to specify changes in the standard number of right-shifts (down-scales) performed in BSFA processing, as indicated in the chart below. The initial value of n4 is zero.

If	Then
$0 \leq n4 \leq 8$	n4 additional downscales will be performed. The result will remain calibrated.
$-3 \leq n4 < 0$	n4 fewer downscales will be performed. The result will remain calibrated.

By using a negative value of n4 (i.e., n4 < 0), the dynamic range of the BSFA analysis band is improved, allowing the detection (and display) of signals that are low in level (compared to a peak signal outside the analysis band). This improvement is because the standard number of shifts (n4 = 0) is calculated to prevent arithmetic overflow of data representing the largest signal in the analysis band. If the "OV WHAT?" message occurs when the measurement program (USER PROG 45) is executed, the value of n4 in this program (USER PROG 41) must be increased.

By using a positive (non-zero) value for n4, you can insure that no overflows will occur. The standard number of shifts (n4 = 0) is calculated so that the probability of an overflow occurring is less than 1 in 1000. However, if an overflow does occur (indicated by the message "OV WHAT?"), the situation can be corrected by either increasing the ADC attenuator setting (if this is possible), or by setting n4 = 1.

USER PROG 40 AND 43 (n4 AND n5 PARAMETERS)

Only parameters n4 and n5 of USER PROG 40 and 43 are covered in this paragraph.

The general form of this command is

USER PROG 40 (or 43) SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

where:

n1, n2, and n3 are discussed in the Optional Hardware BSFA paragraph of this section.

n4 and n5 are used to specify the sample rate when an external clock is used. Use of n4 and n5 is indicated in the chart below. n4 is initialized to a value of zero. A value of zero is assumed when n4 is given and n5 is defaulted.

If	Then
n4 = 0	The sample rate is determined by the ADC setting (Y 40), or by the throughput session header (Y 43).
n4 > 0	The sample rate is n4 $\times 10^{n5}$.

EXAMPLE:

To initialize for off-line BSFA, using data recorded with an external (25 kHz) clock, F_{max} = 12.5 kHz, the command is

USER PROG 43 SPACE n1 SPACE n2 SPACE n3 SPACE 25 SPACE 3 ENTER

Note that when USER PROG 40 is used and n1 > n2, n3 must be given or the bandwidth will be assumed to be zero (i.e., 3-parameter mode must be used).

USING BSFA WITH OVERLAY SWAPPING

Since most of the programs in the BSFA package reside in system overlays, and because these overlays may be swapped in and out "invisibly" by performing various functions, care must be taken when making BSFA measurements.

Here is an example of a potentially INCORRECT operation:

1. Set up a BSFA measurement using USER PROG 41. (Uses Y 41 in overlay n1.)
2. Make the BSFA measurement using USER PROG 45. (Uses Y45 in overlay n1.)
3. Plot the results (automatically reads in overlay n2 to access Graphics software.)
4. Call USER PROG 45 to make another measurement. (Overlay used = ??)

This operation may provide incorrect results at step 4. Since we plotted some BSFA results in step 3, we now find ourselves no longer in the overlay from which we were making the measurement. (If we had not executed any commands that necessitated an overlay swap then no problem would exist.) Now, when we call Y45 in step 4, we again force the system to look for an overlay containing Y45, which is eventually found and read into the Processor.

The potential problem is that this overlay does not necessarily contain the BSFA initialization program used to initialize the measurement at step 1. As a matter of fact, it may not contain any intialization program. Not having an initialization program can be catastrophic since the BSFA initialization programs, besides initializing the proper BSFA measurement, also contain the software to control that measurement when it is performed — USER PROG 45 is merely configured to invoke the correct section of the initialization program that initializes the measurement.

In order to avoid any potential problems when performing BSFA measurements in an overlay swapping environment, keep the following rules in mind:

1. When generating system overlays containing BSFA programs, always put at least one BSFA initialization program (Y41, Y40, or Y43) in the overlay along with the measurement program (Y45). Never put the measurement program in an overlay that does not contain an initialization program, and vice versa.

2. If you perform a BSFA measurement, swap overlays, and now desire to resume the previous BSFA measurement, execute a call to the appropriate initialization program *with no parameters* before invoking the measurement program. This will cause an overlay containing the desired initialization program to be read in, at which time the previous measurement may be resumed.

NOTE

The initial overlay used to perform the BSFA measurement need not be read in. A different overlay containing the same programs may be used, as all important BSFA software parameters are resident in the system, not the overlays. Therefore, if overlays 1 and 3 (for example) both contained Y41 and Y45, the BSFA measurement could be started in overlay 1 and later continued in overlay 3.

The correct operation of our initial example is now:

1. Set up a BSFA measurement using USER PROG 41. (Uses Y41 in overlay n1.)
2. Make the BSFA measurement using USER PROG 45. (Uses Y45 in overlay n1.)
3. Plot the results. (Automatically reads in overlay n2 to access Graphics software.)
4. Call USER PROG 41 ENTER. (Reads in overlay containing Y41, and prints out current BSFA parameters.)
5. Call USER PROG 45 to make another measurement.

Overlays With More Than One Initialization Program

Any BSFA initialization program, when used, re-defines the amount of available data space, based on the zoom power specified by the User Program command parameters given when the program is called.

Before any other BSFA initialization program is used, the data space requirements resulting from use of any previously-used BSFA initialization program should be reset by setting up a baseband measurement using that User Program. For example, the following command

USER PROG 41 SPACE 500 SPACE 250 SPACE 0 ENTER

requests a zoom power of 4.

Before any other BSFA initialization program is used, you should give the command

USER PROG 41 SPACE 0 ENTER

to set up a baseband (zoom power equals 1) measurement.

Then you can give your new BSFA initialization program, such as

USER PROG 43 SPACE 500 SPACE 250 SPACE 0 ENTER

Not resetting the programs as shown above will have the effect of decreasing the data space available for subsequent measurements. Operationally, however, it has no effect.

Pressing RESTART also resets the data space as explained above.

ERROR MESSAGES

Table 5-3 lists error messages that can occur as a result of BSFA operation.

Table 5-3. Error Messages

MESSAGE	PROGRAM OCCURRENCE	MEANING
Y WHAT?	USER PROG 41	Incorrect keyboard entry, or center frequency out of range, or the previously displayed block is no longer available.
BLOCKS LEFT = 1		Requested bandwidth is too narrow for present block size. (Reduce system block size or increase bandwidth.)
BLOCKS LEFT ≤ 0		Requested bandwidth is too narrow for system. (Increase bandwidth.)
ER WHAT?		USER PROG 40 External Clock out of range $F_{max} < 5$ (see Extended Capabilities) $F_{max} > 3200$
IO WHAT?		.IOC. Status error on Mass Store Throughput File
Y WHAT?	USER PROG 40 *or* USER PROG 43	Incorrect Keyboard entry, or center frequency out of range, or the previously displayed block is no longer available.
BLOCKS LEFT = 1		Requested bandwidth is too narrow for present block size. (Reduce system block size or increase bandwidth.)
BLOCKS LEFT ≤ 0		Requested bandwidth too narrow for system. (Increase bandwidth.)
IO WHAT?		.IOC. status error on Mass Store Throughput File (USER PROG 43).
NG WHAT?		Illegal negative parameter.
NB WHAT?		Bandwidth 0 or too small overflow in bandwidth computation.
BW WHAT?		Bandwidth too large.
CF WHAT?		If message occurs before printout of setup parameters, then center frequency is above F_{max}. If message occurs after printout, then center frequency and bandwidth combine to put upper band edge above F_{max} (check filter setting).
AD WHAT?		ADC codeword error (check ADC settings).
C3 WHAT?		ADC set to 3 channels when 2-channel is max.
C4 WHAT?		ADC set to 4 channels when 2-channel is max.

Table 5-3. *Error Messages* (continued)

MESSAGE	PROGRAM OCCURRENCE	MEANING
Y WHAT?	USER PROG 45	Incorrect Keyboard entry, or USER PROG 45 not properly initialized (initialization either has not occurred or hasn't been run to completion).
SH WHAT?		Shift parameter value (n4) is not legal. (Y 41 only)
BL WHAT?		Block required for BSFA doesn't exist. Needs n1, n1–1 for single channel. n1, n1+1, n1–1, n1–2 for dual channel.
BZ WHAT?		Current block size is larger than maximum BSFA block size selected during overlay generation.
OV WHAT?		Overflow occurred during BSFA processing. Increase value entered for n4 or ADC attenuator setting. (see Extended Capabilities) (Y 41 only)
M0 WHAT?		Illegal Mass Store channel number (n3).
M1 WHAT?		More than six different Mass Store channels were requested by USER PROG 45. (Only six may be active at any one time.)
M2 WHAT? @ WHAT?		Next file header or End of Session encountered. (Not enough data in throughput file.)
IO WHAT?		.IOC. status error on Mass Store Throughput File.
DL WHAT?		FPP data word rate too high. Reduce rate with narrower bandwidth. (Y 40 only)

SUPPLEMENT NUMBER 1
for
5451C SYSTEM OPERATING MANUAL

INTRODUCTION

This supplement provides replacement pages for the 5451C System Operating Manual.

CONTENTS AND INSTRUCTIONS

This supplement contains replacement pages for Section 5 (pages 5-1 through 5-33).

Remove Section 5 from the 5451C System Operating Manual HP Part Number 05451-90529 and replace with pages 5-1 through 5-63 attached.

PART NUMBER

The part number of this supplement is **05451-90553**.

July 1980

HEWLETT PACKARD

SECTION 5
BAND SELECTABLE FOURIER ANALYSIS

INTRODUCTION

This section provides operating information on software and hardware Band Selectable Fourier Analysis (BSFA). The off-line software BSFA is the standard BSFA provided with the 5451C.

The Model 54470 Fourier Pre-processor is an option that provides hardware (on-line and off-line) BSFA. This option should not be confused with the the standard BSFA described first in this section. For Pre-processor (hardware) BSFA, refer to the paragraphs on hardware BSFA later in this section.

Band Selectable Fourier Analysis is a measurement technique in which Fourier-transform-based digital spectrum analysis is performed over a frequency band whose upper and lower frequencies are independently selectable. A comparison of BSFA and standard (baseband) Fourier Analysis is presented in Figure 5-1. BSFA can provide an improvement in frequency resolution of more than two orders of magnitude, as well as a 10-dB increase in dynamic range compared to Baseband Fourier Analysis.

NOTE

DAC-based excitation (Analog Output) is not compatible with BSFA operation.

INCREASED RESOLUTION

In any measurement technique, the resolution achievable in the frequency domain is determined by the length of time that the time-domain function is observed. Specifically, the frequency domain resolution is the reciprocal of the time length of the measurement (Df = 1/T), where Df is the frequency resolution.

Standard (baseband) Fourier Analysis provides uniform frequency resolution from dc to Fmax (one-half the sampling frequency). Thus, the frequency resolution can also be expressed as Df=Fmax/(N/2), where N is the block size -- the number of samples describing the real time function. There are N/2 complex (magnitude and phase) samples in the frequency domain.

In actual practice, Fmax is fixed by the frequencies of major experimental interest, and by aliasing considerations. Thus, the only way to improve frequency resolution in Baseband Fourier Analysis is to increase the block size. There are two reasons why this is an inefficient way to increase frequency resolution:

1. Digital processing times increase with block size.

2. The maximum system block size is limited to some relatively small number of samples, based on Processor memory size. The maximum block size in the 5451C, in any case, is 4096 words.

BSFA solves these problems by providing greatly increased resolution about points of interest in the frequency domain, without requiring an increase in the system's block size.

This is done by digitally filtering the incoming time-domain data, and storing only the filtered time-domain data, corresponding to the frequency domain band of interest. Since the frequency resolution is still the reciprocal of the time length of the incoming signal, the digital filters must process T' seconds of data to obtain a frequency resolution in the analysis band of Df' = 1/T'. The resolution obtained in the frequency band of interest is approximately equal to:

(f high - f low)/(N/2) = bandwidth/(N/2)

where

N = Blocksize

Thus, by restricting attention to a narrow region of interest below Fmax, an increase in frequency resolution proportional to Fmax /BW (where BW is the BSFA measurement bandwidth) can be obtained (see Figure 5-2).

Figure 5-1. BSFA vs. Standard Fourier Analysis

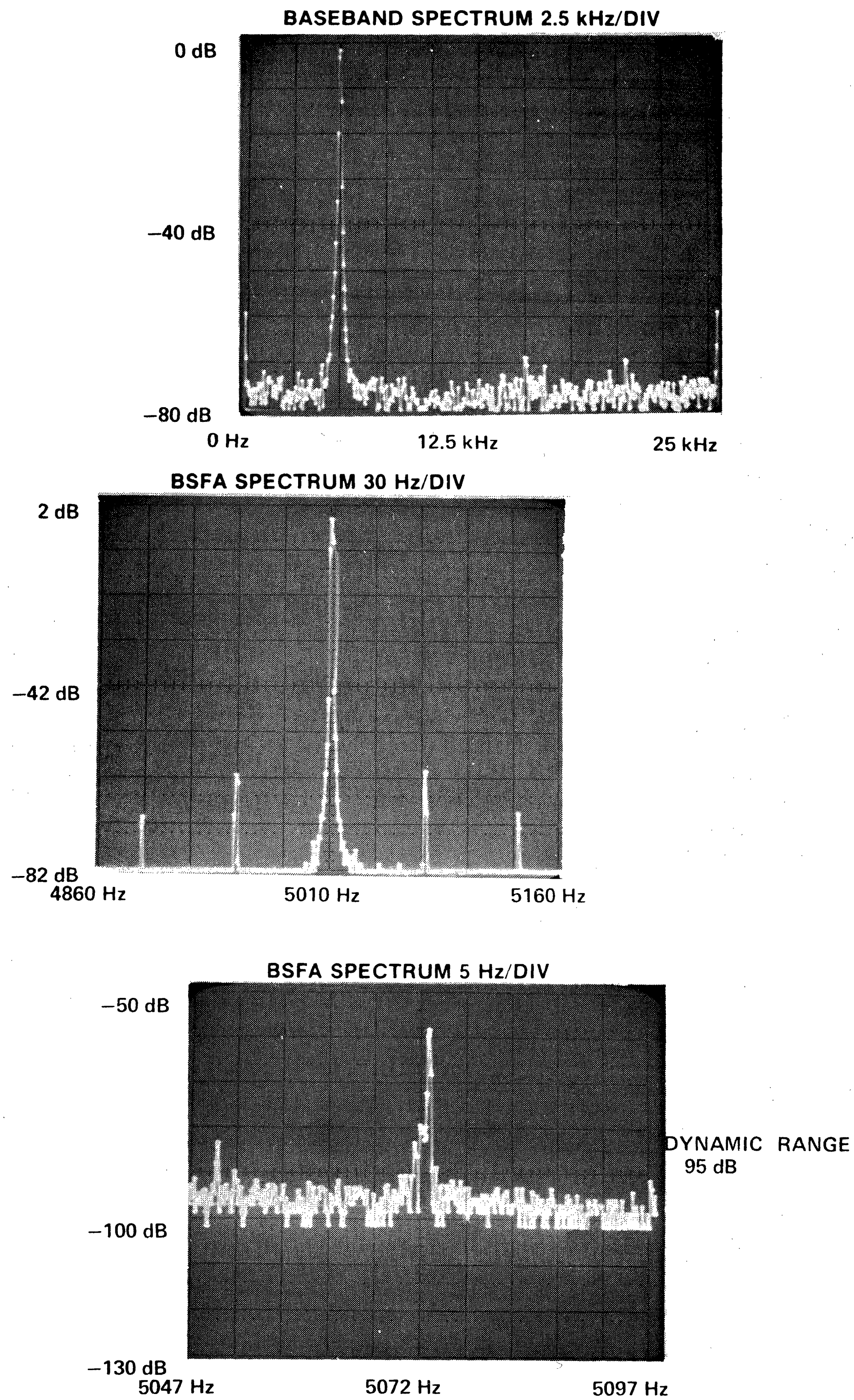
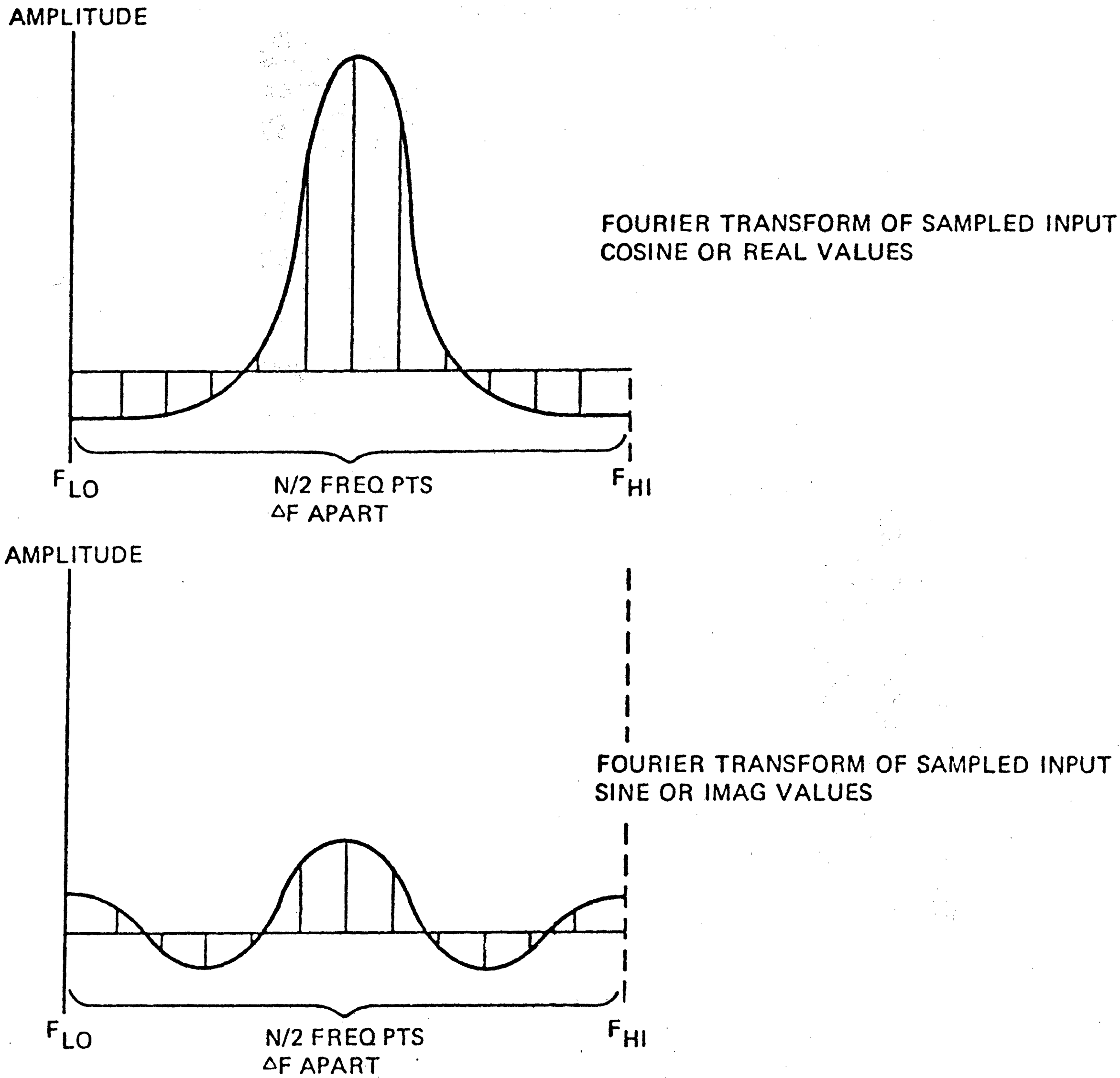


Figure 5-2. Frequency Domain Parameters For a BSFA Measurement



F_{LO} = LOWER FREQUENCY OF BSFA ANALYSIS BAND

F_{HI} = HIGH FREQUENCY OF BSFA ANALYSIS BAND

F_C = CENTER FREQUENCY = $F_{LO} + \dfrac{F_{HI} - F_{LO}}{2}$

$$\Delta F = \frac{(F_{HI} - F_{LO})}{N/2}$$

N/2 = NUMBER OF FREQUENCY POINTS

INCREASED DYNAMIC RANGE

BSFA can provide increased dynamic range relative to Baseband Fourier Analysis. This is due to the increased processor gain on the ADC quantization noise. However, a BSFA system can take advantage of this processor gain only to the extent that is made possible by the noise level and out-of-band rejection of the pre-processing digital filters. The 5451 BSFA system filters provide more than 90 dB of signal-to-noise and out-of-band rejection.

Processor gain refers to the effect of increased frequency resolution on white noise in the presence of a narrow-band signal such as a sine wave. The sine wave energy exists at a single frequency. Its peak value is, therefore, independent of the frequency resolution. The white noise peak amplitude, however, is reduced 3 dB in power each time the frequency resolution is increased by a factor of 2. BSFA provides increased processor gain by increasing frequency resolution in the analysis band (relative to the baseband measurement).

OPERATION

GENERAL

Band Selectable Fourier Analysis operation is controlled by User Programs in the system overlays.

BSFA is performed via a two-step procedure (see Figure 5-3). The first step is initialization, the second is measurement. Each of these steps is performed by calling on a specific User Program.

Initialization Programs

You select the initialization program that will set up for the operation you desire to have performed and give its User Program number and the parameters indicating the frequency band of interest. The program calculates the best possible fit to this request and prints out these measurement parameters.

Measurement Program

The measurement program performs the actual BSFA measurement, based on the parameters calculated by the most-recently-executed initialization program. The initialization program, once entered, need be called again only if you want to change the parameters to be used by the measurement program.

BSFA can be performed off-line on ADC data stored in the ADC Throughput file on the Disc. The ADC Throughput file data must be recorded at 100% real time (i.e., no samples lost).

Each version of BSFA offered for the Fourier Analyzer System has its own initialization program. There is only one measurement program. The measurement program performs the measurement specified by the parameters entered with the most-recently-executed initialization program.

The measurement program calling sequence depends on which initialization program is used. In the following sections, the measurement program is described separately for each initialization program.

NOTE

> During BSFA measurements a portion of the computer data space is temporarily required by the BSFA programs for calculations and for storage of intermediate results. The initialization program provides a message "BLOCKS LEFT: n" to indicate how many data blocks of the current size remain available for other uses. The optional Preprocessor (hardware) BSFA requires additional temporary data space for the input buffer. The user should understand this allocation of data space so that when BSFA measurements are being made, the allocated blocks are not disturbed. A detailed discussion of this topic, "BSFA Data Space Requirements" is at the end of this section.

Table 5-1. Specifications and Supplementary Performance Characteristics

Standard BSFA Initialization Program:
USER PROG 41 (Off-Line)

Optional (Hardware) BSFA Initialization Programs:
USER PROG 40 (On-Line)
USER PROG 43 (Off-Line)

SPECIFICATIONS

Center Frequency Range

USER PROG 41 or USER PROG 43:
dc to one-half the sample rate (up to one-half of the real-time throughput rate of the Disc or optional Mag Tape)

USER PROG 40:
dc to F_{max} specified in tables below:

Y40 F_{max} (Hz) with Standard 100 kHz Digitizers

Zoom Power	# Channels 1	2	4
2	50K	25K	12.5K
4	50K	50K	25K
≥8	50K	50K	31.2K

Y40 F_{max} (Hz) with Optional 200 kHz Digitizers

Zoom Power	# Channels 1	2	4
2	50K	25K	12.5K
4	100K	50K	25K
≥8	100K	62.5K	31.2K

Center Frequency Resolution

USER PROG 41:
Continuous resolution to limit of frequency accuracy for center frequency ≥.02% of sample frequency.

USER PROG 40 or USER PROG 43:
0.015% of the sampling frequency

Bandwidth Selection

USER PROG 41:
Bandwidth = sample frequency/(5 × n), n = 2, 3, 4 etc.

USER PROG 40 or USER PROG 43:
Bandwidth = sample frequency/(2×2^n), n = 1, 2, . . . 8.

Dynamic Range

USER PROG 41:
≥90 dB from peak out-of-band spectral component to the peak level of the passband noise after four ensemble averages of block size 1024*.

≥80 dB from peak in-band spectral component to the peak level of the pass-band noise after eight ensemble averages of block size 1024*.

USER PROG 40 or USER PROG 43:
Noise and spurious signals are >75 dB below full scale*.

Out-of-Band Rejection

USER PROG 41: Greater than 90 dB.

USER PROG 40 or USER PROG 43: Greater than 80 dB.

Passband Flatness

USER PROG 41: Without anti-aliasing filters, ± .01 dB.

USER PROG 40 or USER PROG 43: ± 0.05 dB.

Frequency Accuracy

Equal to ADC clock accuracy. 1 part in 10^6 for HP supplied clocks.

Maximum 54470A Input Word Rate

Input word rate = 2 × F_{max} × # ADC channels.

Max input rate = 250,000 words/sec

$$\left[\, 2 \times F_{max} \times \#\ \text{ADC channels} \le 250\ \text{kHz} \,\right]$$

Maximum 54470A Output Word Rate

$$\text{Output word rate} = \frac{4 \times F_{max} \times \#\ \text{ADC channels}}{\text{Zoom Power}}$$

Max output rate = 100,000 words/sec

$$\left[\, \frac{4 \times F_{max} \times \#\ \text{ADC channels}}{\text{Zoom Power}} \le 100\ \text{kHz} \,\right]$$

SUPPLEMENTARY PERFORMANCE CHARACTERISTICS

Maximum BSFA Blocksize

USER PROG 41: 1024

USER PROG 40 or USER PROG 43: 2048

Variable Data Space Required for Computation

Zoom power is (Δ frequency baseband/Δ frequency BSFA) with blocksize held constant.

USER PROG 41: One throughput record = (# of channels × throughput BS) + (12.5 words × zoom power)

USER PROG 40: (non-buffered mode) One data block of current blocksize.

USER PROG 43: (non-buffered mode) One throughput record = (# of channels × throughput BS) + (one data block of current blocksize)

Max Resolution Enhancement (Zoom Power)

USER PROG 41: > 400 (less if limited by data storage required)

USER PROG 40 or USER PROG 43: 256

**Reduced by 10 dB at exact center of band, due to "dc" FFT error.*

Figure 5-3. Initialization and Measurement Flow

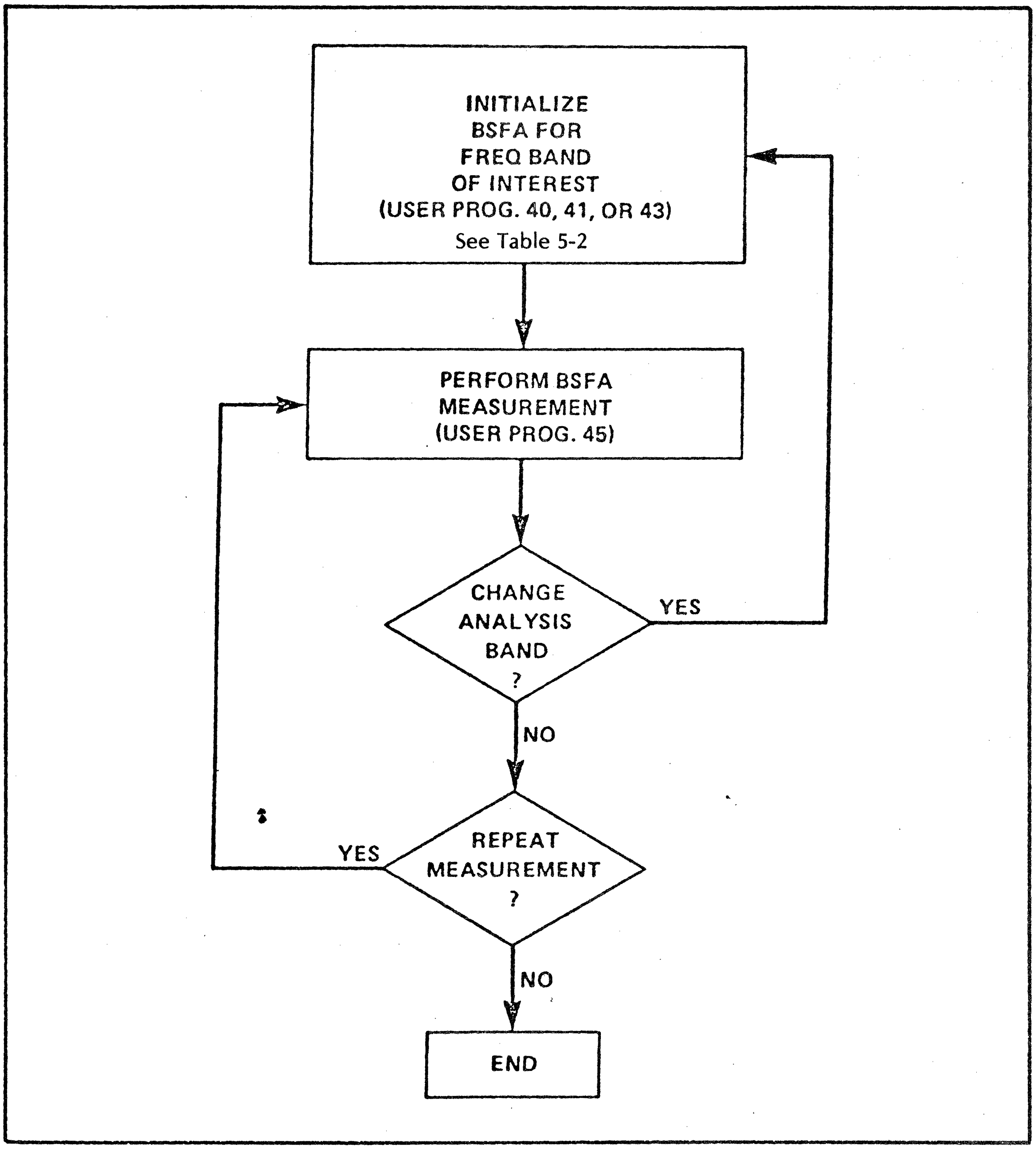


Table 5-2 BSFA User Programs

User Program	Type	Used With
40	On-line Initialization	54470A
41	Off-line Initialization	Standard System
43	Off-line Initialization	54470A
45	Measurement	Standard & 54470A

INITIALIZATION PROGRAM DESCRIPTION

The most-recently-executed initialization program determines the type of Band Selectable Fourier Analysis that will be performed by the measurement program. The standard initialization program (USER PROG 41) is described below; the optional initialization programs (USER PROG 40 and 43) are described at the end of this section.

USER PROG 41

This user program initializes off-line BSFA of ADC data stored in the Disc (or optional Mag Tape) ADC Throughput file. Note that you must have T seconds of time data (with no samples missing) stored in the Throughput file in order to obtain a frequency resolution Df = 1/T. (See Applications paragraph in this section.)

USER PROG 41 allows you to initialize the processing of one channel of ADC throughput data at a time. Up to six ADC Throughput channels can be processed in any sequence by the BSFA measurement program. If there are more than six ADC channels throughput simultaneously, any six can be processed (by USER PROG 45) in any sequence after USER PROG 41 execution.

The general form of the command is:

USER PROG 41 SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

where:

n1 and n2 are used to specify the frequency band of interest, as indicated in the chart below.

n3 is used to specify the initial Disc ADC Throughput file record number. n3 is initialized to a value of zero.

n4 and n5 are additional parameters required only for the external clock input. These parameters are described in the Extended Capability paragraph of this section.

IF	THEN
n1=0 and n2-n5 defaulted,or, n1 and n2=0	The measurement program will perform Baseband Fourier Analysis, dc to Fmax, and release all reserved BSFA data space.
n1>=n2	n1 specifies the center frequency of the band of interest, and n2 specifies the width of the band of interest centered on n1. The measurement will perform BSFA. Center frequencies greater than 32767 must be specified by performing a baseband measurement, then using the n1<n2 format below.
n1<n2	n1 specifies the lower limit of the band of interest, and n2 specifies the upper limit of the band of interest. In this case only, n1 and n2 are expressed as percentages relative to the last previous parameters calculated by USER PROG 41; thus, this combination of parameters (n1<n2) cannot be used the first time USER PROG 41 is invoked.
n3>=0	n3 is the absolute record number, to which the Disc will be positioned. If this record is a header, the header will be read and the Disc positioned to the next record. If this record is not a header, the previous header information will be retained. For example, a position command to record 10 (Mass Store 32 Space 10) followed by a throughput command for the next 20 records of ADC channel 1 (Mass Store 22 Space 1 Space 20) would be stored on the disc as follows: ¦ 10 ¦ 11 ----throughput data---- 30 ¦ header
n3=-1	The Disc will be positioned to the next header. The header will be read, the Disc will be positioned to the next record.

NOTE

The ADC Throughput File Header specified by n3 on the first USER PROG 41 execution must contain sample frequency and record size information identical to those to be used when BSFA measurement (USER PROG 45) is performed, in order to assure proper initialization.

The setup parameters entered by a USER PROG 41 command are retained until changed by a subsequent USER PROG 41 command. Defaulted (omitted) parameters are left unchanged.

After USER PROG 41 and its parameters have been accepted, you will receive the following message.

CNTR FREQ: n1 or baseband
(Center frequency of band of interest)

HZ/DIV: n2
(Horizontal calibration of display, in HZ per major division.)

DF: n3
(Horizontal calibration of display, in Hz per point.)

BLOCKS LEFT: n4
(Number of data blocks available, after space required for BSFA operation has been reserved. Refer to paragraph "BSFA Data Space Requirements".)

ZOOM POWER: n5
(Improvement in frequency resolution relative to unfiltered baseband.)

NOTE

If bit 0 of the Processor display register is set (to 1), the center frequency and horizontal calibration terms will be omitted from the readout.

If bit 1 of the Processor display register is set (to 1), there will be no readout.

For off-line BSFA, the number of blocks left is a function of Fmax /(BSFA bandwidth) and the ADC Throughput record size. Increasing Fmax /(BSFA bandwidth) and/or increasing the ADC Throughput record size results in fewer blocks left (see Table 5-1. Specifications).

USER PROG 41 Examples

The sequence of USER PROG 41 examples below indicates how you can select an optimum analysis band through a series of BSFA measurements. It is assumed that the measurement program (USER PROG 45) is executed following each USER PROG 41 example. (USER PROG 45 is described and examples provided later in this section.)

1. First set the ADC for single channel, Fmax = 10 kHz, and then perform an ADC Throughput operation.

   BLOCK SIZE 2048 ENTER

   MASS STORE 32 SPACE 0 ENTER

   MASS STORE 22 SPACE 1 SPACE 50 ENTER

   BLOCK SIZE 512 ENTER

2. Then initialize off-line BSFA for a baseband measurement.

   USER PROG 41 SPACE 0 ENTER

3. After performing the measurement based on the initialization command given in step 2, observe a region of interest (resonance, harmonic, order, etc.) at 5 kHz. Now initialize for off-line BSFA with a bandwidth of 1 kHz, centered on 5 kHz starting at record 0.

   USER PROG 41 SPACE 5000 SPACE 1000 ENTER

4. After performing the BSFA measurement initialized in step 3, observe that the actual region of interest lies between the 5th and 7th major oscilloscope divisions (50% and 70% relative to the previous measurement). BSFA is initialized to cover this region as shown below starting at record 0.

   USER PROG 41 SPACE 50 SPACE 70 ENTER

   For examples of measurement Keyboard Programs, see the Applications paragraphs of this section.

   For additional USER PROG 41 capabilities, see Extended Capabilities paragraph of this section.

THE MEASUREMENT PROGRAM (USER PROG 45)

This User Program causes a BSFA measurement to be made, based on the parameters provided by the most-recently-executed initialization program (e.g., USER PROG 41). Time-domain data is processed; the processing provides a frequency-domain (linear spectrum) result. On the standard BSFA, the lowest and highest 10% of the frequency domain output of USER PROG 45 is set to zero, to remove aliased frequency components. Similarly, for filtered baseband mode, the highest 20% of the frequency domain output is set to zero for aliasing.

When initialized by USER PROG 41, the command format for USER PROG 45 is:

USER PROG 45 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

where:

n1 specifies the number of the data block where the result from processing the Mass-Store throughput channel is to be stored.

NOTE

One additional working block (n1-1) is required. Your n1 entry should allow for this. Refer also to Error messages.

n2 specifies the type of window to be used, as indicated below

IF	THEN
0=<n2=<10	n2 is the number of times the HANNing window will be applied to the data.
n2<0	The time window to be applied is stored in block -n2. (In this manner, the operator can specify any time window desired.)

n3 specifies the ADC channel in the ADC throughput file that is to be processed by USER PROG 45.

1 = Channel A
2 = Channel B

and so forth.

NOTE

When initialized by USER PROG 41 for off-line software BSFA, USER PROG 45 processes only one ADC Throughput channel at a time. Thus, performing multi-channel analysis requires multiple application of USER PROG 45 commands, which differ only in the specification of USER PROG 45 n3 value. When performing multi-channel operations with Y41, you should work from the highest value n1 to the lowest value to ensure that the working blocks do not overwrite the data from other channels. See examples under "Applications - Standard OffLine BSFA" in this section. For more information see the Applications paragraph in this section.

n4 is used only to change the number of down-scales performed in BSFA processing.

n4 is described in the Extended Capabilities paragraph of this section.

NOTE

Parameters entered by a USER PROG 45 command are retained until changed by a subsequent USER PROG 45 command. Defaulted parameters are left unchanged.

Each time USER PROG 45 is asked to process a given channel (parameter n3) in the ADC Throughput file, it starts processing one sample beyond the last sample of the channel previously processed by USER PROG 45 (unless the position of the Throughput file pointers has been modified by invoking the Initialization program). Up to six channels can be processed alternately in this manner after execution of an initialization program. (See Applications paragraph.)

Examples of USER PROG 45 in actual measurement situations are given in the Applications paragraph.

For additional USER PROG 45 details, see the BSFA Qualifier Storage and Extended Capability paragraphs.

STANDARD OFF-LINE BSFA APPLICATIONS

The examples presented in this section include the MASS STORE commands required to record time-domain data in the ADC Throughput file on the Disc (or optional Mag Tape). The ADC Throughput operation (i.e., putting ADC data into the Throughput file) requires up to five data blocks for each ADC channel being used. You must be careful not to write-over intermediate results during the throughput operation.

The amount of data stored in the ADC Througput file limits the frequency resolution obtainable in the BSFA measurement (Df = 1/T). If resolution is to be improved by a factor of 100, relative to a Baseband measurement, then the number of records that must be recorded is 100 times the Block Size, in real time, with no samples missing. The exact number of ADC Throughput file data records required for a given bandwidth and Fmax is:

Number of Throughput Records=

$$\text{Zoom Power} \times \frac{\text{BSFA Blocksize}}{\text{Throughput Blocksize}} \times \text{Number of Averages}$$

$$\text{Zoom Power} = \frac{\text{Fmax}}{\text{Bandwidth}}$$

For example, if Zoom Power = 20, the baseband (throughput) blocksize = 2048 and the BSFA blocksize = 256, then,

Number of Records = 20 x 256/2048 x Number of Averages

= 20 x 1/8 x Number of Averages

= 2.5 x Number of Averages

Note that the BSFA measurement block size can be different than the block size used during the throughput operation. Choosing a large block size for the throughput operation maximizes the real-time throughput rate, as well as the amount of data stored in the Throughput file (at the expense of increasing the computational buffer space required for BSFA operation). For storage considerations, see Table 5-1. Specifications.

NOTE

In the following program listings, the correlation between the printout abbreviation and the Keyboard pushbuttons is explained in Appendix B.

The program listings below are intended to serve only as examples, and are not intended to illustrate all possible BSFA keyboard programs.

1. Single-channel Power Spectrum Average

   a. The program listing below illustrates a continuously-excited response into the ADC Throughput file. This long time record is then divided into shorter time records, which are processed by BSFA and averaged.

L	0	Establish Label 0.
BS	2048	Set data block size for throughput operation.

Y 41 0	Release all reserved BSFA data space (optional).
MS 32	Position ADC Throughput file to record 0.
MS 22 1 60	Perform ADC Throughput operation for one ADC channel, 60 records.
BS 256	SET DATA BLOCK SIZE FOR BFSA OPERATION.
Y 41 n1 n2 0	Initialize BSFA to record 0 where n1 and n2 define the analysis band. Some data for the BSFA operation must be put through the ADC to the mass storage device before the off-line BSFA initialization program is called so that the header contains current information.
L 1	Establish label 1.
CL 2	Clear data block 2
L 2	Establish Label 2.
Y 45 1 1 1	Perform BSFA into block 1, perform one HANNING operation on channel 1.
D 2 4	Single sweep display of block 2 (this step is optional, it may be omitted).
SP 1	Perform power spectrum average.
# 2 8	Perform steps between label 2 and here a total of eight times.
	End of program.

This program can be operated by giving the command:

JUMP 0 ENTER

Data block 0 is used as a working block by USER PROG 45.

The actual measurement band is determined by the state to which USER PROG 45 has been initialized by the most-recently-executed initialization program.

b. The program listing below illustrates placing several transient responses on the ADC Throughput file; each response is several throughput records long. Power spectrum average is performed on the stored data.

L 0	Establish label 0.
BS 1024	Set data block size for ADC Throughput operation.
MS 32	Position ADC Throughput file to record 0.
Y 41 0	Release BSFA buffers (optional).
L 1	Establish label 1.
MS 22 1 10	Perform ADC Throughput operation one ADC channel, 10 records (on ADC trigger).
# 1 5	Perform steps from label 1 to here a total of 5 times.
BS 512	Change block size.
L 2	Establish label 2.
Y 41 n1 n2 0	Initialize BSFA to record 0. Parameters n1 and n2 in the "Y41" lines define the BSFA analysis band. Note that n1 must be >= n2 for this measurement, i.e., you must call Y41 with center frequency and bandwidth instead of % screen divisions.
(or Y 141 0)	(to position only)
CL 2	Clear data block 2 (used for G_{xx} average).
L 3	Establish label 3.
Y 45 1 0 1	Perform BSFA into data block 1, no HANNING operation.
D 2 4	Single sweep display block 2 (this step is optional, it can be omitted).
SP 1	Perform power spectrum average.

Y 41 n1 n2 -1 (or Y 141 -1)	Position to next header. (to position only)
# 3 5	Perform steps between label 3 and here a total of 5 times.
	End of program.

This program can be operated by giving the command:

JUMP 0 ENTER

Block 0 is used as a working block.

c. The program listing below illustrates entering new data (either transient or continuous) into the ADC Throughput file for each average. The total time record is not stored at once.

L 0	Establish label 0.
CL 0	Clear data block 0.
MS 31	Position to start of data block file.
MS 21 0	Write Block 0 to file 1, record 0 (this is the averaging block).
L 1	Establish label 1.
Y 41 0	Release BSFA buffers (this step is optional, it allows more data space for throughput operations).
MS 32	Position to start of ADC Throughput file.
MS 22 1 50	Perform ADC Throughput for one ADC channel, 50 records.
MS 31	Position to start of data block file.
MS 11 2	Read from data block file into data block 2.
Y 41 n1 n2 0 (or Y 141 0)	Initiate BSFA operation (n1 and n2 specify analysis band). (to position only)

Y 45 1 1 1	Perform BSFA operation into block 1, perform one HANNING operation, for ADC channel 1.
D 2 4	Single sweep display of block 2 (this step is optional, it can be omitted).
SP 1	Power spectrum average.
MS 31	Position to start of file 1.
MS 21 2	Write average (block 2) into data block file.
# 1 8	Perform steps between label 1 and here a total of eight times.
	End of program.

This program can be operated by giving the command:

JUMP 0 ENTER

2. Double-precision tri-spectrum average and transfer and coherence functions

   Dual channel operation requires two USER PROG 45 commands, one for each channel. The operator must direct the first channel processed into the next higher numbered block relative to the result of the second channel processed in order to allow use of the POWER SPECT command (see example program listings). In the first example below, BSFA is performed on data in Block 2 before it is performed on data in Block 1.

   a. This program listing illustrates recording data from two ADC channels in the ADC Throughput file. The input data for one of the ADC channels is the input signal to a device under test; the input signal for the other channel is the output signal of the device under test. The input and output signals of the device under test are continuous while the recording is made, and no data points are lost. Long time records are used during the recording period; these records are later subdivided, and spectral quantities averaged.

L 0	Establish label 0.
BS 1024	Set data block size for ADC Throughput operation.
Y 41 0	Release BSFA buffers (optional).
MS 32	Position to start of ADC Throughput file.
MS 22 2 60	Perform ADC Throughput operation, for two ADC channels, for 60 records.
BS 512	SET BLOCK SIZE FOR BSFA OPERATION
Y 41 n1 n2 0	Initiate BSFA operation (see previous program).
L 1	Establish label 1.
CL 3	Clear data block 3 (used for G_{xx} average).
CL 4	Clear data block 4 (used for G_{yy} average).
CL 5	Clear data block 5 (used for G_{yx} average -- real part).
CL 6	Clear data block 6 (used for G_{yx} average -- imaginary part).
L 2	Establish label 2.
Y 45 2 1 2	Perform BSFA on Channel 2, into block 2; do one HANNING operation (block 1 is used as a working block).
Y 45 1 1 1	Perform BSFA on channel 1, into block 1; do one HANNING operation (block 0 is used as a working block).
D 5 4	Single sweep display of block 5 (this step is optional, it can be omitted).
SP 1 2 2	Perform double-precision power spectrum average.

```
#   2   8             Perform steps between label 2 and
                      here a total of eight times.

CH  1  2  2           Perform transfer and coherence
                      functions.

                      End of program.
```

This program can be operated by giving the command:

JUMP 0 ENTER

The actual measurement band is determined by the state to which USER PROG 45 has been initialized by the most-recently-executed initialization program.

b. The program listing below illustrates recording several transient excitation and response sessions, then analyzing them.

When using transient excitation, the two lines denoted by the stars (*) govern the maximum amount of resolution enhancement and the maximum number of averages obtainable In this example, the throughput operation records six records for each ADC channel and nine such sessions are recorded.

Example: Fmax = 5000 Hz; Throughput Blocksize = 1024; BSFA Bandwidth = 500 Hz; BSFA Blocksize = 512

Thus, the resolution enhancement due to BSFA in this example is:

$$\text{Zoom Power} = .8 \times \frac{\text{Fmax}}{\text{BSFA Bandwidth}} = .8 \times \frac{5000}{500} = 8$$

and the total number of records necessary for 9 averages is:

$$\text{Number of records} = \begin{matrix}\text{Zoom}\\ \text{Power}\end{matrix} \times \frac{\text{BSFA Blocksize}}{\text{Throughput Blocksize}} \times \begin{matrix}\text{Number of}\\ \text{Averages}\end{matrix}$$

= 8 (1/2)(9) = 4(9) = 36 records of blocksize 1024 to achieve a BSFA resolution enhancement of 8 with 9 averages. The program below records 54 total records (6 records x 9 sessions) and adequately meets the criteria.

	L 0	Establish label 0.
	BS 1024	Set data block size for ADC Throughput operation.
	Y 41 0	Release BSFA buffers (optional).
	MS 32	Position to start of Throughput file.
	L 1	Establish label 1.
*	MS 22 2 6	Perform ADC Throughput operation for two ADC channels, six records.
*	# 1 9	Perform steps between label 1 and here 9 times.
	BS 512	Set data block size for analysis.
	Y 41 n1 n2 0	Initialize BSFA to record 0 (n1 and n2 specify the actual measurement band).
	(or Y 141 0)	(to position only)
	CL 3	Clear data block 3 (used for G_{xx} average).
	CL 4	Clear data block 4 (used for G_{yy} average).
	CL 5	Clear data block 5 (used for G_{yx} average -- real part).
	CL 6	Clear data block 6 (used for G_{yx} average -- imaginary part).
	L 3	Establish label 3.
	Y 45 2 0 2	Perform BSFA into block 2 (block 1 used for scratch), no HANNING, ADC channel 2 data.
	Y 45 1 0 1	Perform BSFA into block 1 (block 0 used for scratch), no HANNING, ADC channel 1 data.
	D 5 4	Single sweep display of block 5 (this step is optional, it can be omitted).

SP 1 2 2	Perform power spectrum average.
Y 41 n1 n2 -1	Initialize BSFA to next header (n1 and n2 specify the actual measurement band).
(or Y 141 -1)	(to position only)
# 3 9	Perform steps between label 3 and here a total of 9 times.
	End of program.

To operate this program, give the command:

JUMP 0 ENTER

c. This program listing illustrates entering new data (either transient or continuous) into the ADC Throughput file for each average.

L 0	Establish label 0.
CL 0	Clear block 0.
MS 31	Position to start of data block file.
MS 21 0	Write block 0 into file 1, next record (0) (G_{xx} average).
MS 21	Write block 0 into file 1, next record (1) (G_{yy} average).
MS 21	Write block 0 into file 1, next record (2) (G_{yx} average -- real part).
MS 21	Write block 0 into file 1, next record (3) (G_{yx} average -- imaginary part).
L 1	Establish label 1.
Y 41 0	Release BSFA buffers (this step is optional, it allows more data blocks for ADC Throughput operation).
MS 32	Position to start of ADC Throughput file.
MS 22 2 50	Perform ADC Throughput operation, from two ADC channels, 50 records.

MS 31	Position to start of data block file.
MS 11 3	Read file 1, record 0 into block 3 (G_{xx} average).
MS 11 4	Read file 1, next record (1) into block 4 (G_{yy} average).
MS 11 5	Read file 1, next record (2) into block 5 (G_{yx} average -- real part).
MS 11 6	Read file 1, next record (3) into 6 (G_{yx} average -- imaginary part).
Y 41 n1 n2 0	Initialize BSFA to start of ADC Throughput file (n1 and n2 specify the actual measurement band).
(or Y 141 0)	(to position only)
Y 45 2 1 2	Perform BSFA into block 2, perform one HANNING operation, use ADC channel 2 data (block 1 is used for scratch).
Y 45 1 1 1	Perform BSFA into block 1 (block 0 is used for scratch), perform one HANNING, use ADC channel 1 data.
D 5 4	Single sweep display of block 5 (this step is optional, it can be omitted).
SP 1 2 2	Perform double-precision power spectrum average.
MS 31	Position to start of data block file.
MS 21 3	Write block 3 into file 1, next record (0).
MS 21 4	Write block 4 into file 1, next record (1).
MS 21 5	Write block 5 into file 1, next record (2).
MS 21 6	Write block 6 into file 1, next record (3).

# 1 4	Perform steps between label 1 and here a total of 4 times.
CH 1 2 2	Perform transfer and coherence functions.
	End of program.

This program can be operated by giving the command:

JUMP 0 ENTER

The MASS STORE ENTER command can be used to locate the position of any of the six ADC Throughput channel pointers. Simply use this command after a

USER PROG 45 SPACE n1 SPACE n2 SPACE n3 ENTER

command. The third (ADC Throughput file) pointer value readout on the Terminal by the MASS STORE command is the number of the next record to be read by throughput channel n3. Then use a

USER PROG 41 SPACE n1 SPACE n2 SPACE n3 ENTER

command followed by a MASS STORE ENTER command to show the Throughput file pointer position for all six BSFA Throughput channel pointers.

OPTIONAL HARDWARE BSFA

This option provides real time (on-line) BSFA using the 5447OA Fourier Pre-processor (FPP). In addition, the off-line processing of ADC Throughput files stored on the Disc can be performed at a rate significantly faster than that possible with the standard system.

Optional Initialization Program Format

The most-recently-executed initialization program determines the type of BSFA Fourier Analysis that will be performed by the measurement program. Each optional initialization program is discussed below.

USER PROG 40

This user program initializes a real-time BSFA measurement of ADC data. Consult Table 5-1 for maximum sampling frequencies. USER PROG 40 allows the ADC to run one, two, or four channels of input data. The number of channels is taken into account by program and need not be set at initialization time.

The general form of this command is

USER PROG 40 SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

where:

n1 to n3 specify the frequency band of interest and n4 and n5 are special parameters covered in the Extended Capabilities paragraph toward the back of this section.

IF	THEN
n1 to n5 defaulted	No change in current setup. The current setting of the 54470A FPP is printed for the information of the user.
n1 is any value and n2 to n5 defaulted	The 54470A FPP is set to a baseband throughput mode with an Fmax as indicated by the ADC settings as if a straight ADC input were used.
n1<n2	The BSFA band is defined to be n1% to n2% of the last frequency range as covered on the CRT display.
n1>n2 n3 defaulted	n1=center frequency in Hz. n2=bandwidth in Hz.
n1>n2 n3 given	Center frequency = n1 x 10(n2) n3 = bandwidth in Hz

After USER PROG 40 has been entered, a printout will indicate the current setting of the 54470A FPP. If bit 0 of the Processor register is set to 1, the printout will be eliminated.

USER PROG 40 Examples

The following sequence of USER PROG 40 examples indicates how you might adjust the measurement band of interest.

As an example, start with the ADC set to Fmax of 5 kHz. The command

USER PROG 40 SPACE 0 ENTER

will set for a direct input from the ADC with no filtering. The parameters will be set to:

CNTR FREQ: BASEBAND
HZ/DIV: 500 Hz
DF: 5000/(Block Size/2)
ZOOM POWER: 1

As in all direct measurements, signals above Fmax (5000 Hz) will alias back as frequencies below 5000 Hz unless they are eliminated with an anti-aliasing filter. The safest practice is to set the filter to 1/2 Fmax (2500 Hz) so that all components that can be aliased back into frequencies below 1/2 Fmax (2500 Hz) will be attenuated by >75 dB, a value in agreement with the dynamic range of the 54470A FPP. As with baseband measurements, anti-aliasing filters are equally important in BSFA measurements, with the filters set at 1/2 the Fmax specified by the 5466B ADC (1/4 the sample rate).

The command

USER PROG 40 SPACE 0 SPACE 25 ENTER

will set the analysis bandwidth to 0 to 25% of the previous range (i.e., from 0 to 1250 Hz).

The printout will now indicate:

CNTR FREQ: BASEBAND
HZ/DIV: 125
DF: 1250/(Block Size/2)
ZOOM POWER: 4 (reduce bandwidth by 4)

When any hardware BSFA measurement is made, all the resulting data is valid (as opposed to User Program 41 software BSFA, which clears 20% of the band). In this case, the total data from 0 to 1250 Hz is valid and does not contain aliasing terms if the analog filter on the input is set to 1/4 the ADC sample rate of 10 kHz. That is, the aliasing filter should remain set to 2500 Hz. The digital filtering of the 54470A FPP will now remove components between 1250 and 2500 Hz. This is an example of digital lowpass filtering.

The command

USER PROG 40 SPACE 50 SPACE 100 ENTER

will result in a BSFA analysis band which is from 50 to 100% of the last settings. This will give:

CNTR/FREQ: 937.5 Hz
HZ/DIV: 62.5 Hz
DF: 625/(Block Size/2)
ZOOM POWER: 8

The command

USER PROG 40 SPACE 1250 SPACE 625 ENTER

will set a BSFA measurement independent of the last setting with:

```
CNTR/FREQ:  1250 Hz
HZ/DIV:  62.5
DF:  625/(Block Size/2)
ZOOM POWER:  8
```

This same command could be given as

USER PROG 40 SPACE 125 SPACE 1 SPACE 625 ENTER

or as

USER PROG 40 SPACE 12500 SPACE -1 SPACE 625 ENTER

The Measurement Program (USER PROG 45)

This section describes the operation of USER PROG 45 when initialized for on-line hardware BSFA.

USER PROG 45 causes a BSFA measurement to be made, based on the parameters provided by the most-recently-executed initialization program (e.g. USER PROG 40). Time-domain data is processed, the processing provides a frequency-domain (linear spectrum) result.

When initialized by USER PROG 40, the command format for USER PROG 45 is:

USER PROG 45 SPACE n1 SPACE n2 SPACE n3 ENTER

where:

n1 specifies the number of the data block where the result from processing the first ADC channel is to be stored. In simultaneous two channel processing, the result from processing the second ADC channel is stored in block n1+1. In simultaneous four-channel operation, the result from processing the third ADC channel is stored in bock n1+2, and the result from processing the fourth ADC channel is stored in block n1+3.

NOTE

One additional working block is required for each result data block. In single-channel operation, the working block is n1-1. In simultaneous two channel operation, blocks n1-2 and n1-1 are required as working blocks. In simultaneous four channel operation, blocks n1-1, n1-2, n1-3, and n1-4 are required as working blocks. Your n1 entry should allow for this.

n2 specifies the type of window to be used as indicated below.

IF	THEN
0=<n2=<10	n2 is the number of times the HANNing window will be applied to the data.
n2<0	The time window to be applied is stored in block -n2. (In this manner, the operator can specify any time window desired.)
-999 =< n2 =< -911	Special code for n2 which allows the user to specify two separate window blocks, which is useful for impact testing. See the Extended Capabilities section for details on using this feature.

n3 specifies the data block to be displayed during the BSFA measurement.

When a BSFA measurement is made with a large output word rate from the Preprocessor (i.e. high Fmax, low zoom power) it is possible that some data will be lost when block n3 is displayed, resulting in an error message (DL WHAT). If this occurs, then parameter n3 should not be given.

When the BSFA measurement is executed in buffered input mode (see description of User Program 142 later in this section), block n3 is displayed only at the end of each execution of Y45. For buffered input, it is better to include a Single Sweep high-speed display command within the measurement count loop and not use parameter n3. See the keyboard program examples in the Optional Applications subsection.

NOTE

Parameters entered by a USER PROG 45 command are retained until changed by a subsequent USER PROG 45 command. Defaulted parameters are left unchanged.

The number of channels to be simultaneously processed (on-line) is set by the Input Selector switch on the ADC. This switch may be set to 1,2 or 4 channels. If three channels are selected, an error message will result.

For BSFA, the ADC REPEAT scan switch is operable only when USER PROG 45 has been initialized to perform Baseband Fourier Analysis on data coming directly from the ADC.

Examples of USER PROG 45 in actual measurement situations are given in the Applications paragraph.

For additional USER PROG 45 details, see the BSFA Qualifier Storage section and the Extended Capability paragraph.

USER PROG 43

This user program initializes off-line BSFA of ADC data stored in the mass storage device (Disc or Mag Tape) ADC Throughput file. Note that you must have T seconds of time data (with no samples missing) stored in the Throughput file in order to obtain a frequency resolution Df = 1/T.

USER PROG 43 allows you to initialize the processing of any combination of one, two or four channels of ADC Throughput data at a time. Up to six combinations of ADC Throughput channels can be processed in any sequence by the BSFA measurement program. If there are more than six ADC channels throughput simultaneously, any six combinations can be processed (by USER PROG 45) in any sequence after USER PROG 43 execution.

The general form of the command is:

USER PROG 43 SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

where:

n1 and n2 are used to specify the frequency band of interest, as indicated in the chart below.

n3 is used to specify the initial Mass Store Throughput file record number, as indicated in the second chart below. n3 is initialized to a value of zero.

n4 and n5 are additional parameters, and may be omitted. These parameters are described in the Extended Capabilities paragraph.

IF	THEN
n1=0	The measurement program will perform Baseband Fourier Analysis and release all reserved BSFA data space.
n1=>n2	n1 specifies the center frequency of the band of interest, and n2 specifies the width of the band of interest, centered on n1. The measurement program will perform BSFA.
n1<n2	n1 specifies the lower limit of the band of interest and n2 specifies the upper limit of the band of interest. In this case only, n1 and n2 are expressed as percentages relative to the last previous parameters calculated by USER PROG 43; thus, this combination of parameters (n1<n2) cannot be used the first time USER PROG 43 is invoked.

n3 is used to specify the initial Mass Store Throughput file record number, as indicated in the chart below. n3 is initialized to a value of zero. (Positions all six ADC Throughput channel pointers simultaneously.) If defaulted, n3 assumes its last value (initially zero).

IF	THEN
n3=>0	n3 is the absolute record number, to which the Mass Store device will be positioned. If this record is a header, the header will be read and the Mass Store device positioned to the next record. If this record is not a header, previous header information will be retained. For example, a position command for the next 20 records of ADC channel 1 (Mass Store 22 Space 1 Space 20) would be stored as follows: | 10 | 11 --------------------30 | header throughput data
n3=-1	The Mass Store device will be positioned to the next header. The header will be read, and the Mass Store device will be positioned to the next record.

NOTE

In order to assure proper initialization, the Throughput File Header specified by n3 on the first USER PROG 43 execution must contain sample frequency and record size information identical to those to be used when the BSFA measurement (USER PROG 45) is performed.

The setup parameters entered by a USER PROG 43 command are retained until changed by a subsequent USER PROG 43 command.

After USER PROG 43 and its parameters have been accepted, the following message is provided:

CNTR FREQ: n1 or BASEBAND
(Center frequency of band of interest.)

HZ/DIV: n2
(Horizontal calibration of display, in Hz per major division.)

DF: n3
(Horizontal calibration of display, in Hz per point.)

BLOCKS LEFT: n4
(Number of data blocks available, after space required for BSFA operation has been reserved. Refer to paragraph "BSFA Data Space Requirements" toward the end of this section.)

ZOOM POWER:
(Resolution improvement. This is a power of 2 from 1 to 256.)

NOTE

If bit 0 of the Processor display register is set to 1, the printout is suppressed.

USER PROG 43 Examples

The sequence of USER PROG 43 examples below indicates how you can select an optimum anlysis band through a series of BSFA measurements. It is assumed that the measurement program (USER PROG 45) is executed following each USER PROG 43 example. (USER PROG 45 is described and examples provided in this section.)

1. First set the ADC for single channel, Fmax = 10 kHz, and perform an ADC Throughput operation.

   BLOCKSIZE 2048 ENTER

   MASS STORE 32 SPACE 0 ENTER

   MASS STORE 22 SPACE 1 SPACE 50 ENTER

   BLOCKSIZE 512 ENTER

2. Then initialize off-line BSFA for a baseband measurement.

   USER PROG 43 SPACE 0 ENTER

3. After performing the measurement based on the initialization command given in step 2, observe a region of interest (resonance, harmonic, order, etc.) at 5 kHz. Now initialize for off-line BSFA with a bandwidth of 1 kHz, centered on 5 kHz starting at header record 0.

   USER PROG 43 SPACE 5000 SPACE 1000 SPACE ENTER

4. After performing the BSFA measurement initialized in step 3, observe that the actual region of interest lies between the 5th and 7th major oscilloscope divisions (50% and 70% relative to the previous measurement). BSFA is initialized to cover this region as shown starting at header record 0.

USER PROG 43 SPACE 50 SPACE 70 ENTER

For example measurements of keyboard programs, refer to the Optional Applications paragraphs.

For additional USER PROG 43 capabilities, refer to the Extended Capabilities paragraphs.

The Measurement Program (USER PROG 45)

When initialized by USER PROG 43, the command format for USER PROG 45 is:

USER PROG 45 SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 SPACE n6 ENTER

where:

n1 specifies the number of the data block where the result from processing the first ADC throughput channel is to be stored. In simultaneous two-channel processing the result from processing the second throughput channel is stored in block n1+1. In simultaneous four-channel operation, the result from processing the third ADC throughput channel is stored in block n1+2 and the result from processing the fourth ADC throughput channel is stored in block n1+3.

NOTE

One additional working block is required for each result data block. In single-channel operation, the working block is n1-1. In simultaneous two-channel operation, blocks n1-1 and n1-2 are required as as working blocks. In simultaneous four-channel operation, blocks n1-1, n1-2, n1-3 and n1-4 are required as working blocks. Your n1 entry should allow for this.

n2 specifies the type of window to be used, as indicated below.

IF	THEN
0=<n2=<10	n2 is the number of times the HANNing window will be applied to the data.
n2<0	The time window to be applied is stored in block -n2. (In this manner, the operator can specify any time window desired.)
-999 =< n2 =< -911	Special code for n2 which allows the user to specify two separate window blocks, which is useful for impact testing. See the Extended Capabilities section for details on using this feature.

n3 through n6 specify the ADC channels in the ADC Throughput file that are to be processed simultaneously by USER PROG 45.

1 = Channel A
2 = Channel B

and so forth.

When initialized for hardware, off-line BSFA, USER PROG 45 can process 1,2 or 4 channels simultaneously. The order in which the throughput channels are entered (parameters n3 to n6) determines the order in which the BSFA results are stored in the result blocks. For example, the following call,

```
USER PROG 45 4 1 1 2 3 4 ENTER
```

performs simultaneous processing of throughput channels: 1,2,3, and 4 uses one HANNing window, and stores the results as indicated below:

BLOCK	CONTENTS
4	BSFA result for channel 1
5	BSFA result for channel 2
6	BSFA result for channel 3
7	BSFA result for channel 4

The following call,

```
USER PROG 45 4 1 4 1 3 2 ENTER
```

stores the BSFA results as indicated below:

BLOCK	CONTENTS
4	BSFA result for channel 4
5	BSFA result for channel 1
6	BSFA result for channel 3
7	BSFA result for channel 2

As an example of two-channel processing, the following call,

USER PROG 45 2 1 3 1

performs simultaneous two-channel BSFA and stores the results as indicated below:

BLOCK	CONTENTS
2	BSFA result for channel 3
3	BSFA result for channel 1

NOTE

Parameters entered by a USER PROG 45 command are retained until changed by a subsequent USER PROG 45 command. Defaulted parameters are left unchanged.

Each time USER PROG 45 is asked to process a given combination of channels (parameters n3 to n6) in the ADC Throughput file, it starts processing one sample beyond the last sample of the channel combination previously processed by USER PROG 45 (unless the position of the Throughput file pointers has been modified by invoking the Initialization program). Up to six combinations of channels can be processed alternately in this manner after execution of an initialization program.

USER PROG 45 implements this feature by maintaining a table in memory in which each table entry has two parts: a keyword and an ADC Throughput file pointer. The keyword indicates the channel combination and is made up of parameters n3 to n6 (in order). The associated ADC Throughput file pointer indicates the next sample in the Throughput file to begin processing whenever USER PROG 45 is called where parameters n3 to n6 "match" the keyword. If no match is found a new entry is made in the table. Whenever the initialization program (USER PROG 43) is called, this table is initialized to zero.

The following examples demonstrate this feature. For these examples it is assumed that:

USER PROG 43 has been called with n3=0
BSFA blocksize = ADC Throughput blocksize
Zoom Power = 8

Example measurement program calls:

USER PROG 45 4 1 1 2 3 4 ENTER

This call simultaneously processes Throughput records 1 through 8 for channels 1,2,3 and 4. Since the file pointer table is empty, a new entry is made with keyword = 1234 and the associated file pointer indicates the first sample of the 9th throughput record (next sample to be processed).

If the following call is made next:

USER PROG 45 2 1 1 2 ENTER

The file pointer table is searched for keyword = 12. SInce no such entry exists, this call again processes throughput channels 1 through 8 for channels 1 and 2. Then a new table entry is made with keyword = 12 and file pointer = first sample of 9th throughput record.

If the following call is made next:

USER PROG 45 4 1 1 2 3 4 ENTER

Then the file pointer table is searched for keyword = 1234. In this case a match occurs and the throughput file is positioned to the file pointer associated with the "matched" keyword, then throughput records 9 through 16 are processed for channels 1,2,3, and 4.

If the following call is made next:

USER PROG 45 4 1 4 3 2 1 ENTER

Then the file pointer table is searched for keyword = 4321. This keyword does not match the entry with keyword = 1234, since both the throughput channel numbers and their order of entry (n3 to n6) determine a unique keyword, therefore a new entry is made for this combination of channels, and throughput records 1 through 8 are processed for the channel combination: 4,3,2,1.

At this point the file pointer table has 3 entries with the following keywords: 1234, 12 and 4321. The file pointer table has room for six entries. If USER PROG 43 is called at any time, the file pointer table is reset to zero.

Examples of USER PROG 45 in actual measurement situations are given in the Applications paragraph.

For additional USER PROG 45 details, see the BSFA Qualifier Storage section and the Extended Capability paragraph.

Buffered Input and Overlap Processing (USER PROG 142)

GENERAL

USER PROG 142 allows the user to specify buffered input with or without overlap processing for subsequent BSFA measurements using the 54470A Preprocessor. For on-line BSFA measurements, when the processing time for a particular operation is less than the time to input a new record from the preprocessor, then it is possible to process all of the values sampled by the ADC (i.e. real time processing). Real time processing is possible by buffering the input data. The first data record is read into the buffer and transferred into the true input block. The next input record is then started into the buffer block. While the buffer block is being filled, the processing operation (e.g. power spectrum average or tri-power spectrum average) is performed on the data in the true input block. In non-overlap processing, if a whole new record has not been read into the input buffer when the processing loop is ready for a new record, processing will stop until a whole new record can be transferred to the true input block. In overlap processing the input buffer is transferred to the true input block before a whole new record is read into the input buffer. Hence, the true input block contains some new data and some "old" (overlapped) data. Overlap processing can only occur at or below real time data rates.

In the text that follows, the command format is given along with a short summary of the parameter functions. Following the summary is a more detailed explanation of each input parameter.

Calling Sequence

The command format for USER PROG 142 is:

USER PROG 142 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

where:

IF	THEN
0=<n1=<100	Buffered input mode is selected and n1 is the desired record-to-record % overlap.
n1<0	Disables all input parameters and resets to non-buffered, non-overlapped mode.
n2>0	n2 is an integer variable parameter location into which is stored the actual % overlap obtained after each subsequent execution of USER PROG 45.
n2<0	Disables storing actual % overlap into the variable parameter location.
n3>0	n3 is an integer variable parameter location into which is stored a real time indicator. The realtime indicator is one of two possible values: +1 for a realtime measurement session -1 for a non-realtime measurement session.
n3<0	Disables storing the realtime indicator into the variable parameter location.
n4>0	n4 is a floating point variable parameter location into which is stored a running counter of the number of independent samples per channel in the measurement session.
n4<0	Disables storing the independent sample count into the variable parameter location.

USER PROG 142 is an additional initialization program which may be invoked at any time and it affects all subsequent BSFA measurements.

NOTES

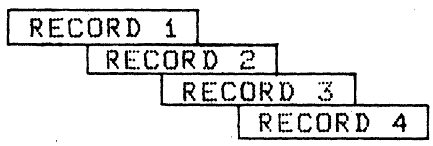
- Parameters entered by a USER PROG 142 command are retained until changed by a subsequent USER PROG 142 command. Defaulted parameters are left unchanged. USER PROG 142 may be called at any time, either before or after the initialization program (Y40 or Y43) is called.

- If the BSFA measurement program (USER PROG 45) has been initalized for a baseband measurement, (dc to Fmax), then USER PROG 142 parameters are ignored. USER PROG 45 performs all baseband measurements (dc to Fmax) in non-buffered, non-overlapped mode. However, BSFA measurements from DC to some higher frequency below Fmax (filtered baseband) can utilize the features of USER PROG 142.

Percent Overlap (Parameters n1 and n2)

An example of 50% record-to-record overlap for four averages is shown below:

50% OVERLAP



For on-line BSFA measurements, the user-specified % overlap (parameter n1) represents an upper boundary for the amount of overlap. In other words, the actual % overlap obtained will never be greater than n1 but it may, of course, be less than n1 if the measurement loop time is too long. The measurement loop time is a function of ADC sample rate, zoom power and number of commands within the keyboard program measurement count loop. The software will not allow 100% overlap to occur. However, the user may specify n1=100 in order to obtain the maximum possible % overlap. If buffered input without overlap processing is desired, set n1=0 for 0% overlap. A keyboard program which performs buffered or overlap processing must contain a COUNT command since the buffered input will continue until runout of the count loop.

For any given on-line BSFA keyboard program, each pass through the measurement count loop can have a slightly different loop time due to variations (i.e., interrupts, branches, etc.) in the amount of program code executed during each count loop. This in turn, can product slight variations in the record-to-record % overlap. Therefore, the value placed in variable parameter location n2 is a running average of the actual % overlap obtained for each loop and is updated each time through the keyboard program count loop. In simultaneous multi-channel processing, the % overlap is the same for all channels.

For off-line BSFA measurements, the software calculates the actual % overlap, which is as close as possible to the value given for parameter n1. This calculated % overlap (optionally stored in VP n2) will always be either equal to or less than n1. Note that if n1=25,50 or 75, then the actual % overlap obtained will be exactly equal to n1. For off-line BSFA measurements, the measurement loop time is irrelevant so the record-to-record % overlap will be constant throughout the measurement keyboard program. As in on-line BSFA, the software will not allow 100% overlap to occur.

Realtime Indicator (Parameter n3)

Selecting buffered input mode with USER PROG 142 (0<=n1<=100) permits the input of 1,2 or 4 channels of data into a buffer block while simultaneously processing the data in the true input block. If the processing time is less than the time to input a full buffer block, then no data will be lost between successive inputting of records and a real time analysis will be performed. If a USER PROG 142 command is given with parameter n3, then a realtime indicator will be placed into n3. This indicator will have a value of +1 designating real time if absolutely no data was lost between any of the input records in the measurement average. If there was at least one gap (i.e. lost data) between input records then the measurement average is non-real time and the real time indicator is set to a value of -1. The measurement average is, of course, still valid even if it is not realtime. If the indicator value is -1, it is simply an indication to the user that some data was lost between input records.

NOTE

The realtime indicator only applies to data lost between input records, if some data is lost within an input record, the measurement is invalid and an error message (DL WHAT) results.

Independent Sample Count (Parameter n4)

The software keeps a running count of the number of independent samples per channel, which is updated each time through the BSFA measurement loop. When a USER PROG 142 command is given with parameter n4, this updated independent sample count is stored in the floating point variable parameter location n4 each time USER PROG 45 is excecuted within the keyboard program loop.

The number of independent samples per channel can be used to calculate the exact % overlap for the measurement session as shown below:

where:

ISAMP = number of independent samples/channel
BS = BSFA blocksize
#AVG = number of averages in the measurement session

$$(1) \quad \% \text{ OVERLAP} = \frac{\#AVG \times BS - ISAMP}{\#AVG \times BS}$$

Note that the above calculation includes the samples in the first record of the average. The first record of each channel contains no overlapped data and therefore contains only independent samples. To calculate the % overlap excluding the first record, the equation becomes:

$$(2) \quad \% \text{ OVERLAP} = \frac{(\#AVG - 1) \times BS - (ISAMP - BS)}{(\#AVG - 1) \times BS}$$

The equation (2) given above can be used to calculate a more accurate value for the record-to-record % overlap than that given in variable parameter location n2.

BUFFER SPACE REQUIREMENTS

When BSFA measurements using buffered input, either with or without overlap processing, are performed, additional buffer blocks are needed at the upper end of the data space for the input buffer. These buffer blocks are not removed from the available data blocks so the user must be sure not to store into these blocks during buffered input measurements.

The number of buffered blocks required is determined by the following equation:

input buffer blocks = 2 x # of BSFA channels

Note that the number of blocks available which is indicated by the initialization program (USER PROG 40 or USER PROG 43) applies only to non-buffered measurements. For buffered measurements the number of blocks available can be determined by subtracting the number of input buffer blocks from the number of "blocks left" given by the initialization program.

If the user specified BSFA result blocks (determined by USER PROG 45 parameter n1) conflict with the input buffer blocks, an error message (BL WHAT) will result.

OPTIONAL APPLICATIONS

This subsection is divided into two parts. The first part contains a familiarization procedure to demonstrate the buffered input and overlap processing capability. The second part contains example keyboard programs for on-line and off-line hardware BSFA measurements.

Familiarization Procedure

In this procedure, a single-channel power spectrum measurement will b: made on the CHECK PULSE of ADC channel A.

(1) Enter the following keyboard program:

L 0	Establish label 0
CL 2	Clear data block 2 (used for averaging)
L 1	Establish label 1
Y 45 1 1	Perform BSFA into block 1, perform one HANNING operation.
SP 1	Perform Power Spectrum Averaging.
D 2 4	Single Sweep display of Input Power Spectrum
# 1 8	Perform steps between label 1 and here a total of 8 times.
	End of program

(2) Set the ADC controls as follows:

Sample Mode to:	KHz/us
Multiplier to:	100/10/5
EXT/INT Switch to:	INT
Input Selector to:	A
Channel A Overload Voltage:	CHECK
Trigger Source to:	FREE RUN

(3) Enter the commands:

BS 512 ENTER

USER PROG 40 SPACE 2000 SPACE 1500 ENTER

This will cause Y45 to be initialized to a center freq. = 2 kHz and BSFA bandwidth = 2.5 kHz. Notice that the bandwidth of 2.5 KHz is automatically calculated to be the bandwidth greater than or equal to the requested bandwidth (1500 Hz) which meets the bandwidth selection criteria described in Table 5-1.

(4) Enter the command:

USER PROG 142 SPACE 100 SPACE 1 SPACE 2 ENTER

where:

the first parameter (100) is the desired % overlap. Since the software will not allow 100% overlap, this is effectively a request for the highest possible overlap; the second parameter (1) is the Variable Parameter location into which the % overlap actually obtained will be stored; the third parameter (2) is the Variable Parameter location into which the real time indicator will be stored.

(5) Enter the command:

JUMP 0 ENTER

This starts the measurement keyboard program.

(6) When the measurement is complete, enter the commands

USER PROG LIST 1 ENTER
USER PROG LIST 2 ENTER

These commands list, on the system terminal, the contents of V.P.#'s 1 and 2. Note that the % overlap = 0 and the real time indicator = -1 designating a non-real time measurement. Overlap is not possible when the BSFA bandwidth exceeds the real time bandwidth. BSFA bandwidth = 10 x Hz/DIV value given by the initialization program printout.

(7) Enter the command:

USER PROG 40 SPACE 2000 SPACE 600 ENTER

This initializes to a center freq. = 2 kHz and BSFA Bandwidth = 625.

(8) Enter the command:

JUMP 0 ENTER

(9) After the measurement is complete, enter the commands:

USER PROG LIST 1 ENTER
USER PROG LIST 2 ENTER

Note that the % overlap = 57 and the real time indicator = 1, designating a real time measurement.

(10) Enter the command:

USER PROG 142 SPACE 25 ENTER

This will limit the % overlap to 25%.

(11) Enter the command:

JUMP 0 ENTER

(12) After the measurement is completed, enter the command:

USER PROG LIST 1 ENTER

Note that the % overlap is now 25%.

(13) This completes the familiarization procedure.

Example Keyboard Programs

The following is an example of a keyboard program for double-precision tri-spectrum average, transfer function and coherence function. The ADC must be set for dual-channel operation.

L 0	Establish label 0.
CL 4	Clear data block 4 (used for Gxx average)
CL 5	Clear data block 5 (used for Gyy average).
CL 6	Clear data block 6 (used for Gyx average - real part).
CL 7	Clear data block 7 (used for Gyx average - imaginary part).
L 1	Establish label 1.
Y 45 2 1 4	Perform BSFA into blocks 2 and 3, perform one HANNING operation. Display of block 4 while inputting data (this parameter is optional).
SP 2 2 2	Perform double-precision power spectrum average.
# 1 4	Perform steps between label 1 and here a total of 4 times.
CH 2 2 2	Perform transfer and coherence functions.
	End of program.

This program is operated by giving the command:

JUMP 0 ENTER

Note that data blocks 0 and 1 are used for working blocks by

USER PROG 45

The actual measurement band is determined by the state to which USER PROG 45 has been initialized by the most-recently-executed initialization program.

The following is an example keyboard program for on-line, four-channel power spectrum average. The blocksize must be equal to or less than 1024 (due to data space limitations). USER PROG 40 must be called to initialize the 54470A FPP. The system must have a 4-channel ADC that is setup for 4-channel operation when this program is run.

L	0	Establish label 0.
CL	8	Clear block 8 (used for chan. 4 G_{44} average).
CL	10	Clear block 10 (used for chan. 3 G_{33} average).
CL	12	Clear block 12 (used for chan. 2 G_{22} average).
CL	14	Clear block 14 (used for chan. 1 G_{11} average).
L	1	Establish label 1.
Y	45 4 1	Perform BSFA into blocks 4,5,6 and 7; perform one HANNING operation.
SP	7	Perform Power Spectrum Average for channel 4.
X<	6	Load channel 3 BSFA result into block 0.
X>	9	Store channel 3 BSFA result into block 9.
SP	9	Perform Power Spectrum Average for channel 3.
X<	5	Load channel 2 BSFA result into block 0.
X>	11	Store channel 2 BSFA result into block 11.
SP	11	Perform Power Spectrum Average for channel 2.
X<	4	Load channel 1 BSFA result into block 0.
X>	13	Store channel 1 BSFA result into block 13.
SP	13	Perform Power Spectrum Average for

channel 1.

D 14 4	Single Sweep display of channel 1 Power Spectrum (optional).
# 1 8	Perform steps between label 1 and here a total of 8 times.
TL 8	Take Log Magnitude of channel 4 Power Spectrum average.
TL 10	Take Log Magnitude of channel 3 Power Spectrum average.
TL 12	Take Log Magnitude of channel 2 Power Spectrum average.
TL 14	Take Log Magnitude of channel 1 Power Spectrum average.
.	End of program.

This program is operated by giving the keyboard command:

JUMP 0 ENTER

The actual measurement band is determined by the state to which USER PROG 45 has been initialized by the most-recently-executed initialization program.

Note that data blocks 0,1,2 and 3 are used as working blocks by USER PROG 45.

This keyboard program can be used to perform an off-line, four-channel power spectrum measurement simply by changing the USER PROG 45 command to:

Y 45 4 1 1 2 3 4

Before running the program for an off-line measurement, 4 channels of data must be throughput to the Mass Store device, and USER PROG 43 must be called to initialize the 54470A FPP for off-line BSFA.

Also, the above keyboard Program can be run using buffered input and overlap processing by either calling USER PROG 142 (with appropriate parameters) before running the program or inserting the USER 142 command into the program before LABEL 1.

The following is an example keyboard program for an on-line, tri-axial transfer function measurement. The blocksize must be equal to or less than 1024 (due to data space limitations) and USER PROG 40 must be called to initialize the 54470A FPP. The system must have a 4-channel ADC that is setup for 4-channel operation when this program is run. The input signal goes into ADC channel A, and the three response signals go into ADC channels B,C, and D.

L	0	Establish label 0.
CL	8	Clear block 8 (used for G_{11} average).
CL	9	Clear block 9 (used for G_{44} average).
CL	10	Clear block 10 (used for G_{41} - real part average).
CL	11	Clear block 11 (used for G_{41} -imag. part average).
CL	14	Clear block 14 (used for G_{11} average).
CL	15	Clear block 15 (used for G_{33} average).
CL	16	Clear block 16 (used for G_{31} - real part average).
CL	17	Clear block 17 (used for G_{31} - imag. part average).
CL	20	Clear block 20 (used for G_{11} average).
CL	21	Clear block 21 (used for G_{22} average).
CL	22	Clear block 22 (used for G_{21} - real part average).
CL	23	Clear block 23 (used for G_{21} - imag. part average).
L	1	Establish label 1.
Y	45 4 1	Perform BSFA into blocks 4,5,6 and 7; perform one HANNING operation.
X<	6	Load channel 3 BSFA result into block 0.
X>	13	Store channel 3 BSFA result into block 13.
X<	4	Load channel 1 BSFA result into block 0
X>	6	Store channel 1 BSFA result into block 6.
SP	6 2 2	Perform tri-spectrum ensemble average for channels 1 and 4.

X> 12	Store channel 1 BSFA result into block 12.
SP 12 2 2	Perform tri-spectrum ensemble average for channels 1 and 3.
X> 18	Store channel 1 BSFA result into block 18.
X< 5	Load channel 2 BSFA result into block 0.
X> 19	Store channel 2 BSFA result into block 19.
SP 18 2 2	Perform tri-spectrum ensemble average for channels 1 and 2.
D 14 4	Single Sweep Display of Input Power Spectrum (optional).
# 1 8	Perform steps between Label 1 and here a total of 8 times.
CH 6 2 2	Perform transfer and coherence function for channels 1 and 4.
CH 12 2 2	Perform transfer and coherence function for channels 1 and 3.
CH 18 2 2	Perform transfer and coherence function for channels 1 and 2.
	End of program.

This program is operated by giving the keyboard command:

JUMP 0 ENTER

The actual measurement band is determined by the state to which USER PROG 45 has been initialized by the most-recently-executed initialization program.

Note that data blocks 0,1,2 and 3 are used as working blocks by USER PROG 45.

This keyboard program can be used to perform an off-line, four-channel, tri-axial transfer function measurement simply by changing the USER PROG 45 command to:

Y 45 4 1 1 2 3 4

Before running the program for an off-line measurement, 4 channels of data must be throughput to the Mass Store device, and USER PROG 43 must be called to initialize the 54470A FPP for off-line BSFA.

Also, the above keyboard Program can be run using buffered input and overlap processing by either calling USER PROG 142 (with appropriate parameters) before running the program or inserting the USER PROG 142 command into the program before LABEL 1. When buffered input is used with this program, the blocksize must be equal to or less than 512, because the available data space has been further reduced by the required input buffer.

BSFA QUALIFIER STORAGE

When USER PROG 45 is used to perform BSFA measurements, it does the following:

1. Stores the floating point values of the Center Frequency (CF) and Delta Frequency (Df) of the BSFA measurement(s) into reserved locations in the core-resident data block header (see Section 4 for further details and format of the data block header).

2. Assigns all BSFA measurement data blocks a special frequency code of 99.

   The special frequency code of 99 signifies that the frequency axis of the data block is not to be determined by the settings of the ADC (as is usually the case), but rather from the center frequency and Df parameters stored in the core-resident data block header.

   In contrast to the normal mode of data block qualifier storage where each data block has an individual set of qualifiers, BSFA measurements in core have only one set of "global" qualifiers. This means that all BSFA data blocks in core are assumed to have the same qualifiers.

   Therefore, two BSFA measurements with different qualifiers (i.e., different center frequency and Df from two different BSFA measurement sessions) will be assumed to have the same qualifiers (the qualifiers currently residing in the core-resident data block header). This will result in an error if the two measurements are compared via plotting, or using the cursor where the frequency axis calibration must be printed out.

   The BSFA qualifiers associated with any BFSA currently in core may be saved by saving the relevant data blocks on the Disc. This is possible since the core-resident data block header containing the BSFA qualifiers will be stored to the Disc along with the data. These blocks can later be read from the Disc, thereby restoring the "global" BSFA qualifiers to the values for that data.

RULE: The BSFA global qualifiers in core (from which all data with frequency code = 99 are calibrated) are either:

a. The qualifiers (CF and Df) from the last call to USER PROG 45;

or

b. The qualifiers from the last data block read from the Disc, whichever was done last.

EXTENDED CAPABILITY

The system BSFA overlays include User Programs and user capabilities not described in detail in the Operation paragraph in this section. While these additional programs and program parameters are not required for BSFA operation, they add capabilities and flexibility beyond that provided by the basic BSFA programs described in the Operation paragraph.

Additional User Programs

USER PROG 44

This program allows you to re-define the horizontal formatting of the most recent BSFA measurement. This is accomplished by means of an automatic partial block display. USER PROG 44 is useful for interpreting and recording (by photograph or plotting) the results of BSFA measurements.

The general form of the command is

USER PROG 44 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

where:

n1 is the number of the block on which USER PROG 44 is to operate.

n2 x 10E+n4 is the desired center frequency of the display, in Hz.

n3 x 10E+n4 is the desired horizontal calibration of the display, in Hz per major division.

NOTE
Defaulted parameters are left unchanged.

USER PROG 44 evaluates the inputs and performs the appropriate partial block display of the block specified by n1. The accuracy of USER PROG 44 is +/- Df/2.

USER PROG 3017

This program performs a complex time to complex frequency fast Fourier transform.

The general form of the command is

USER PROG 3017 SPACE n1 ENTER

where:

n1 is used to locate the data blocks to be transformed, as indicated in the chart below.

IF	THEN
The data to be transformed is in the time domain.	n1 is the number of the block containing the real part of the time function. The imaginary part is in block n1+1. The frequency domain result is stored with the negative frequencies and dc in block n1 and positive frequency components in block n1+1. Frequency increases from left to right (displayed).
The data to be transformed is in the frequency domain.	n1 is the number of the block containing the negative frequency components and dc. The positive frequency components are in block n1+1. The time domain result is stored with the real part in block n1 and the imaginery part in block n1+1.

Scale factors and calibrators of blocks n1 and n1+1 must be equal. The system blocksize must be less than or equal to 2048.

USER PROG 141

This program positions the ADC Throughput file for processing by Y45, when initialized for off-line software BSFA (Y41).

The general form of the command is

USER PROG 141 SPACE n1 SPACE n2 ENTER

where:

IF	THEN
n1>=0 n2 defaulted	Position throughput file to record n1.
n1<0 n2 defaulted	Position throughput file to next header.
n2 given	n1 = relative record number.

USER PROG 143

This program positions the ADC throughput file for processing by Y45 when initialized for off-line hardware BSFA (Y43). This is the same function that Y141 performs for off-line software BSFA.

The general form of the command is:

USER PROG 143 SPACE n1 SPACE n2 ENTER

where:

IF	THEN
n1=>0 n2 defaulted	Position through put file to record n1.
n1<0 n2 defaulted	Position throughput file to next header
n2 given	n1=relative record number.

USER PROG 41

Only parameters n4 and n5 of USER PROG 41 are explained in this paragraph.

The general form of this command is

USER PROG 41 SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

where:

n1, n2, and n3 are discussed in the Operation paragraph of this section.

n4 and n5 are used to specify the Fmax of the ADC Throughput session, if the session was recorded with the ADC in the EXT clock position. Use of n4 and n5 is indicated in the chart below. n4 is initialized to a value of zero.

IF	THEN
n4=0	The sample rate of the throughput session is automatically read from the header of the througput session.
n4>0	The value of Fmax for the throughput session is n4 x 10(n5). (This entry is used when the ADC clock is in the EXTernal position.) n5 must not be defaulted or equal to zero if n4>0.

EXAMPLE:

To initialize for off-line BSFA, using data recorded with an external 25 kHz clock (Fmax = 12.5 kHz), the command is:

USER PROG 41 SPACE n1 SPACE n2 SPACE n3 SPACE 125 SPACE 2 ENTER

USER PROG 45

Only parameters n4 and the special case of n2 of USER PROG 45 are explained in this paragraph.

The general form of this command is

USER PROG 45 SPACE N1 SPACE N2 SPACE N3 SPACE N4 ENTER

where:

n1, n2, and n3 are discussed in the Operation paragraph of this section.

When performing multi-channel BSFA measurements while using impact testing, it may be desirable to use one type of window on the input channel and a different window on the response channels.

n2 can be used to specify two separate window blocks as explained below:

IF	THEN
-999 =< n2 =<-911	the "tens" digit of n2 designates the block containing the window for the first result (input) block. The "ones" digit of n2 designates the block containing the window for all the other result (response) blocks. Block 0 cannot be used as a window block. Therefore, when using n2 to specify 2 windows, neither the "tens" digit nor the "ones" digit can be 0.

For example, if n2 = -912 then:

Block 1 is used to window the first BSFA result block. Block 2 is used to window all the remaining BSFA result blocks.

Remember, Y45 requires some temporary buffer blocks, therefore the user must ensure that the specified window blocks do not conflict with the Y45 buffer blocks.

n4 is used to specify changes in the standard number of right-shifts (down-scales) performed in off-line, software BSFA processing, as indicated in the chart below. The initial value of n4 is zero.

IF	THEN
0=<n4=<8	n4 additional downscales will be performed. The result will remain calibrated.
-3=<n4=<0	n4 fewer downscales will be performed. The result will remain calibrated.

By using a negative value of n4 (i.e., n4 < 0), the dynamic range in the BSFA analysis band is improved, allowing the detection (and display) of signals that are low in level (compared to a peak signal outside the analysis band). This improvement is because the standard number of shifts (n4 = 0) is calculated to prevent arithmetic overflow of data representing the largest signal in the analysis band. If the "OV WHAT?" message occurs when the measurement program (USER PROG 45) is executed, the value of n4 in this program must be increased.

By using a positive (non-zero) value for n4, you can insure that no overflows will occur. The standard number of shifts (n4 = 0) is calculated so that the probability of an overflow occurring is less than 1 in 1000. However, if an overflow does occur (indicated by the message "OV WHAT?"), the situation can be corrected by either increasing the ADC attenuator setting and performing the ADC throughput again (if this is possible), or by setting n4 = 1.

USER PROG 40 and 43 (n4 and n5 Parameters)

Only parameters n4 and n5 of USER PROG 40 and 43 are covered in this paragraph.

The general form of this command is

USER PROG 40 (or 43) SPACE n1 SPACE n2 SPACE n3 SPACE n4 SPACE n5 ENTER

where:

n1, n2, and n3 are discussed in the Optional Hardware BSFA paragraph of this section.

n4 and n5 are used to specify the sample rate when an external clock is used. Use of n4 and n5 is indicated in the chart below. n4 is initialized to a value of zero. A value of zero is assumed when n4 is given and n5 is defaulted.

IF	THEN
n4=0	The sample rate is determined by the ADC setting (Y40) or the thruput session header (Y43)
n4>0	The sample rate is n4 x 10(n5)

EXAMPLE:

To initialize for off-line BSFA using data recorded with an external 25 kHz clock (sample rate = 25 kHz), the command is

USER PROG 43 SPACE n1 SPACE n2 SPACE n3 SPACE 25 SPACE 3 ENTER

Note that when USER PROG 40 is used and n1 > n2, n3 must be given or the bandwidth will be assumed to be zero (i.e., 3-parameter mode must be used).

BSFA DATA SPACE REQUIREMENTS

The 5451C has a fixed amount of available data space of either 16,384 or 28,672 words. During BSFA measurements a portion of this data space is temporarily required by the BSFA programs for calculations and storage of intermediate results. After the BSFA operations are performed, this borrowed data space is once again available for other operations. The user should understand this allocation of data space so that when BSFA measurements are being made, these allocated blocks are not disturbed.

The Processor BSFA data block requirements are of three types:

Type 1: Variable Computation Blocks

The number of these blocks required to perform a particular measurement is calculated automatically based on the formulas given in Table 5-1 (under "Variable Data Space Required for Computation"); the message "BLOCKS LEFT:n" is printed out during the BSFA initialization step, indicating the total number of data blocks minus the Type 1 Computation Data Blocks. These blocks are allocated to the highest numbered data blocks in the data space (see example below).

To protect certain of these BSFA Computation Blocks, the system sets up a software boundary which prevents user access of the protected blocks until they are released. Protected blocks are released whenever a BSFA initialization program is set up for baseband (Y40 0, Y41 0, or Y43 0) or when the RESTART button is pushed. Type 2 and Type 3 blocks are not protected in this manner and, therefore, do not need to be released.

Type 2: BSFA Working Data Blocks

A number of Working Data Blocks (equal to the number of analysis data blocks) are required and are located adjacent to the result blocks. User Program 41 can analyze only one channel per call of User Program 45, and therefore uses only one working block. User Programs 40 and 43 will use 1, 2, or 4 working blocks corresponding to the number of channels analyzed. See paragraphs on User Program 45 for further details.

Type 3: BSFA Buffer Data Blocks

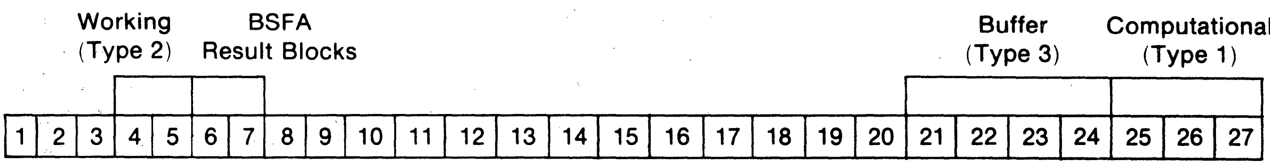
When User Program 142 is used, additional data blocks are required for the buffering operations. These blocks are located in upper data space just below the Type 1 Computation Blocks. The number of Buffer Blocks required is twice the number of BSFA analysis channels.

Example:

A user performs a two-channel ADC throughput operation using a blocksize of 1024 and wishes to do a buffered two-channel BSFA analysis using the off-line hardware BSFA program (User Program 43).

If the system has 28,672 words of data space and if he selects a BSFA blocksize of 1024, the allocation of data blocks would look like this:

Figure 5-3. Data Block Allocation



The total number of data blocks=28672/1024=28, numbered 0 through 27. The number of Type 1 blocks required will be (from the Y43 formula in Table 5-1) =

$$\frac{(\#\ \text{throughput channels} \times \text{throughput blocksize})}{\text{current blocksize}} + 1$$

$$= \frac{2 \times 1024}{1024} + 1 = 3$$

Blocks 27, 26, and 25 will be allocated to Type 1, and the message "BLOCKS LEFT: 25" will be printed (blocks 0-24 are available).

The buffering operation will require another four blocks (blocks 24, 23, 22, and 21). If block 6 is specified as the first result block (Y 45 6 etc.), then blocks 4 and 5 will be used as the Type 2 Working Blocks.

USING BSFA WITH OVERLAY SWAPPING

Since most of the programs in the BSFA package reside in system overlays, and because these overlays may be swapped in and out "invisibly" by performing various functions, care must be taken when making BSFA measurements.

Here is an example of a potentially INCORRECT operation:

1. Set up a BSFA measurement using USER PROG 41. (Uses Y 41 in overlay n1.)

2. Make the BSFA measure using USER PROG 45. (Uses Y45 in overlay n1.)

3. Plot the results (automatically reads in overlay n2 to access
   Graphics software.)

4. Call USER PROG 45 to make another measurement. (Overlay used = ??)

This operation may provide incorrect results at step 4. Since we plotted some BSFA results in step 3, we now find ourselves no longer in the overlay from which we were making the measurement. (If we had not executed any commands that necessitated an overlay swap then no problem would exist.) Now, when we call Y45 in step 4, we again force the system to look for an overlay containing Y45, which is eventually found and read into the Processor.

The potential problem is that this overlay does not necessarily contain the BSFA initialization program used to initialize the measurement at step 1. As a matter of fact, it may not contain any intialization program. Not having an initialization program can be catastrophic since the BSFA initialization programs, besides initializing the proper BSFA measurement, also contain the software to control that measurement when it is performed
-- USER PROG 45 is merely configured to invoke the correct section of the initialization program that initializes the measurement.

In order to avoid any potential problems when performing BSFA measurements in an overlay swapping environment, keep the following rules in mind:

1. When generating system overlays containing BSFA programs, always put at least one BSFA initialization program (Y41, Y40, or Y43) in the overlay along with the measurement program (Y45). Never put the measurement program in an overlay that does not contain an initialization program, and vice versa.

2. If you perform a BSFA measurement, swap overlays, and now desire to resume the previous BSFA measurement, execute a call to the appropriate initialization program with no parameters before invoking the measurement program. This will cause an overlay containing the desired initialization program to be read in, at which time the previous measurement may be resumed.

NOTE

The initial overlay used to perform the BSFA measurement need not be read in. A different overlay containing the same programs may be used, as all important BSFA software parameters are resident in the system, not the overlays. Therefore, if overlays 1 and 3 (for example) both contained Y41 and Y45, the BSFA measurement could be started in overlay 1 and later continued in overlay 3.

The correct operation of our initial example is now:

1. Set up a BSFA measurement using USER PROG 41. (Uses Y41 in overlay n1.)

2. Make the BSFA measurement using USER PROG 45. (Uses Y45 in overlay n1.)

3. Plot the results. (Automatically reads in overlay n2 to access Graphics software.)

4. Call USER PROG 41 ENTER. (Reads in overlay containing Y41, and prints out current BSFA parameters.)

5. Call USER PROG 45 to make another measurement.

Overlays With More Than One Initialization Program

Any BSFA initialization program, when used, re-defines the amount of available data space, based on the zoom power specified by the User Program command parameters given when the program is called.

Before any other BSFA initialization program is used, the data space requirements resulting from use of any previously-used BSFA initialization program should be reset by setting up a baseband measurement using that User Program. For example, the following command

USER PROG 41 500 250 0 ENTER

initializes an off-line software BSFA measurement.

Before any other BSFA initialization program is used, you should give the command

USER PROG 41 0 ENTER

to set up a baseband (zoom power equals 1) measurement.

Then you can give your new BSFA initialization program, such as

USER PROG 43 500 250 0 ENTER.

Not resetting the programs as shown above will have the effect of decreasing the data space available for subsequent measurements. Operationally, however, it has no effect.

Pressing RESTART also resets the data space as explained above.

ERROR MESSAGES

Table 5-3 lists error messages that can occur as a result of BSFA operation.

Table 5-3. Error Messages (Part 1 of 3)

MESSAGE	PROGRAM OCCURRENCE	MEANING
Y WHAT?	USER PROG 41	Incorrect keyboard entry, or center frequency out of range, or the previously displayed block is no longer available.
BLOCKS LEFT=1		Requested bandwidth is too narrow for present block size. (Reduce system block size or increase bandwidth.)
BLOCKS LEFT=<0		Requested bandwidth is too narrow for system. (Increase bandwidth.)
IO WHAT?		.IOC. Status error on Mass Store Throughput File
Y WHAT?	USER PROG 40 or USER PROG 43	Incorrect Keyboard entry, or center frequency out of range.
BLOCKS LEFT=1		Requested bandwidth is too narrow for present block size. (Reduce system block size or increase bandwidth.)
BLOCKS LEFT=0		Requested bandwidth too narrow for system. (Increase bandwidth.)
IO WHAT?		.IOC. status error on Mass Store Throughput File (USER PROG 43).
NG WHAT?		Illegal negative parameter.
NB WHAT?		Bandwidth 0 or too small overflow in bandwidth computation.
BW WHAT?		Bandwidth too large.
CF WHAT?		If message occurs before printout of setup parameters, then center frequency is above Fmax. If message occurs after printout, then center frequency and bandwidth combine to put upper band edge above Fmax (check filter setting).

Table 5-3. Error Messages (Part 2 of 3)

MESSAGE	PROGRAM OCCURRENCE	MEANING
Y WHAT?	USER PROG 45	Incorrect keyboard entry, or USER PROG 45 not properly initialized (initialization either has not occurred or hasn't been run to completion).
AD WHAT?		ADC codeword error, check ADC settings (Y40 only).
TO WHAT?		The number of ADC input channels was changed during a buffered or overlapped BSFA measurement keyboard program.
C3 WHAT?		ADC set to 3 channels. Must be 1,2, or 4 channels.
SH WHAT?		Shift parameter value (n4) is not legal. (Y 41 only)
BL WHAT?		Block required for BSFA doesn't exist. Needs n1, n1-1 for single channel; n1, n1+1, n1-1, n1-2 for dual channel; n1-4 thru n1+3 for four channels.
BZ WHAT?		Current block size is larger than maximum BSFA block size during overlay generation.
OV WHAT?		Overflow occurred during BSFA processing. Increase value for n4 or ADC attenuator setting. (See Extended Capabilities)(Y 41 only).
M0 WHAT?		Illegal Mass Store channel number (n3).
M1 WHAT?		More than six different Mass Store channels were requested by USER PROG 45. (Only six may be active at any one time.)
M2 WHAT? @ WHAT?		Next file header or End of Session encountered. (Not enough data in throughput file.)

Table 5-3. Error Messages (Part 3 of 3)

MESSAGE	PROGRAM OCCURRENCE	MEANING
IO WHAT?		.IOC. status error on Mass Store Throughput File.
DL WHAT?		FPP output data word rate too high. Reduce rate by increasing zoom power (Y 40 only). The quantity: (4 x Fmax x #channels)/zoom power must be equal to or less than 100 khz.
FS WHAT?		The quantity: 2 x Fmax x #channels must be equal to or less than 250 kHz. Either reduce Fmax or # of input channels.
M9 WHAT?		Y0045 called with 5 parameters for processing 3 channels of throughput data. Only 1,2 or 4 channels can be processed simultaneously with the optional hardware BSFA.
T1 WHAT?		The BSFA blocksize is too big. Must have: Blocksize =(1024 for software BSFA (Y41). Blocksize =(2048 for hardware BSFA (Y40 or Y43).
M6 WHAT?	USER PROG 142	Invalid floating pt. V.P. # given for for input parameter n4.
M7 WHAT?		Invalid integer V.P. # given for input parameter n2 or n3.
M8 WHAT?		A value greater than 100 was given for input parameter n1.

SECTION 6
GRAPHICS

INTRODUCTION

In general, the graphics capability constitutes a series of software user programs which provide plots of data currently being displayed in the Fourier Analyzer system environment. These programs are normally accessed via User Program commands within the structure of a keyboard program (refer to Section 3 KEYBOARD and USER PROG keys).

Once data has been acquired, you can program a particular plot through a series of calls to the various graphic programs. You use the system Keyboard or the Terminal to enter the graphics commands.

Graphics commands can be executed singly or within the structure of a Keyboard Program. A summary of the graphics commands and error messages is provided at the end of this section.

FUNCTIONS AVAILABLE

The following functions are available using the graphics commands:

1. Automatic plotting — Data being displayed will be automatically plotted with axes and vertical and horizontal annotations and scale factor.
2. Specification of plot size, location, and orientation.
3. Text editing — Text may be created to be used later as descriptive annotations for a particular plot.
4. Specification of grid lines — Major grid lines or all grid lines may be specified and then drawn with the execution of one command.
5. Graphics Terminal control — Controls echoing of system terminal input and system response, erases the screen, and makes a hard copy of the screen.
6. Specification of MODE, ORIGIN, and SCALE to be used to produce a particular plot.
7. Plot boundary alignment — On the plot device (terminal or optional plotter), you are allowed to define the horizontal and vertical boundary of the plotting area.
8. Horizontal and vertical range specification — you can specify a portion of the data being displayed and eliminate the remaining portion when the plot is finally made.
9. Text message disc storage and recall.

GRAPHICS INITIALIZATION

After power has been applied to the system it is ready to execute any of the graphics commands. **However, to ensure proper operation, execute the following command before executing any other graphics commands:**

USER PROGRAM 5821 SPACE n1 ENTER

where n1 is equal to 6, 10, or 35. Entering a 6 for n1 indicates that the plotting device to be used will be the Terminal. Entering a 10 for n1 indicates that the plotting device to be used will be an HP 7210 (digital plotter without HP-IB). Entering a 35 for n1 indicates that the plotting (or printing) device will be an HP-IB compatible device such as the HP 9872 or 7245.

GRAPHICS OPERATION

The following paragraphs explain in detail each graphics command available. Included in each discussion is a description of each command function, command format, and a familiarization procedure.

NOTES

If graphics commands are attempted without calling User Program 5821, then the Processor may HALT with 102000_8 in the S Register.

For HP-IB plotters: The standard software is configured for plotter HP-IB address = 5. Therefore, the address switches on the rear of the 9872 or 7245 plotters should be set to 5. To change the HP-IB address requires system regeneration as described in the 5451C System Software Manual.

5800 AUTOMATIC PLOTTING

NOTE

The 5800 automatic plotting can also be initiated using the "Gold Key" capability of the PLOT (KEYBOARD) key. To implement this function, press USER PROG PLOT instead of USER PROG 5800, and then enter the required parameters as explained below.

User Program 5800 is used to execute, automatically, the following User Program commands:

1. User Program 5804 — Set Plot Size
2. User Program 5805 — Set Location of Plot on Page
3. User Program 5806 — Set Increment of Plot Location
4. User Program 5807 — Set Plot Orientation
5. User Program 5809 — Set Mode, Origin, and Scale Parameters
6. User Program 5829 — Set Horizontal Range for full or partial block plot per parameters n3,n4
7. User Program 5814 — Erase Screen
8. User Program 5815 — Plot Data Block per parameter n2
9. User Program 5816 — Draw Axis
10. User Program 5817 — Annotate Plot Axes
11. User Program 5821 — Initialize Plot Device per parameter n1

Referencing forward in this section, you will find that most of the above programs (except for User Program 5821) can be executed with default values that produce a standard plot as shown in Figure 6-1. When User Program 5800 is executed, most of the programs are always executed with default values except where the specified parameters are used to modify the calling sequence.

If 5800 is executed with parameter n1 equal to –1, only 5815, 5816, and 5817 are executed with their default values. The parameters of the remaining programs, mentioned above, are retained from the last time each one of these programs was called and 5814 is not called at all. For the "–1" parameter to be used, 5800 must have been previously called at least once with a valid device parameter. This is true even if 5821 has already been called.

Command Format

USER PROG 5800 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

n1 = -1: execute programs with current parameters except 5815, 5816, and 5817 which retain default values.

n1 = device type: (6 for 2648 Terminal, 10 for 7210 optional plotter, 35 for 9872 or 7245 optional plotters)
default = 6 or the last configured plot device.

n2 = block number to plot
default = block on display

n3 = starting channel of plot
default = first channel in block

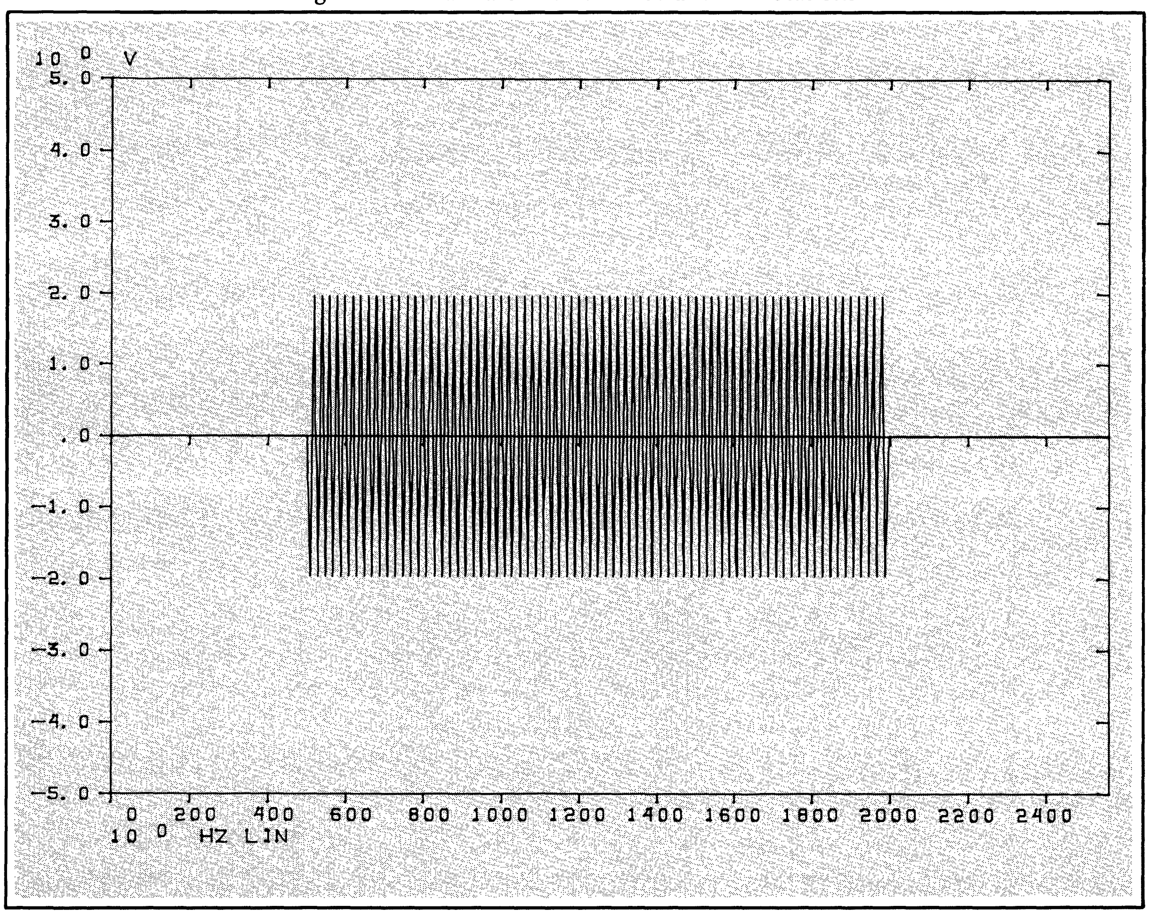
n4 = ending channel of plot
default = last channel in block

Error Conditions

L6 WHAT?	n1 isn't a valid device code
L7 WHAT?	n2 is an invalid block number

The other possible error messages that may be received are output by each individual program that is called when 5800 is executed. Each error message is discussed under the appropriate User Program description in this section and is summarized in Table 6-1 (at the end of the section).

Figure 6-1. Standard Default Plot and Annotations



Familiarization Procedure

Perform the following steps to gain a better understanding of the 5800 functions available.

1. Using the Keyboard, enter the following commands to create some data to plot.

   BLOCK SIZE 64 ENTER
   CLEAR ENTER
   KEYBOARD 0 SPACE 0 SPACE 5 ENTER
   KEYBOARD 5 SPACE 0 SPACE 9 ENTER
   1 ENTER
   F ENTER

   NOTE

   If the Terminal is to be used for plotting in any of the following procedures, it must be set to Graphics mode (refer to the Turn On Procedure in Section 2).

   If the optional Plotter is used in any of the following procedures it must be set up prior to performing the procedure (i.e., paper installed, pen positioned, etc.)

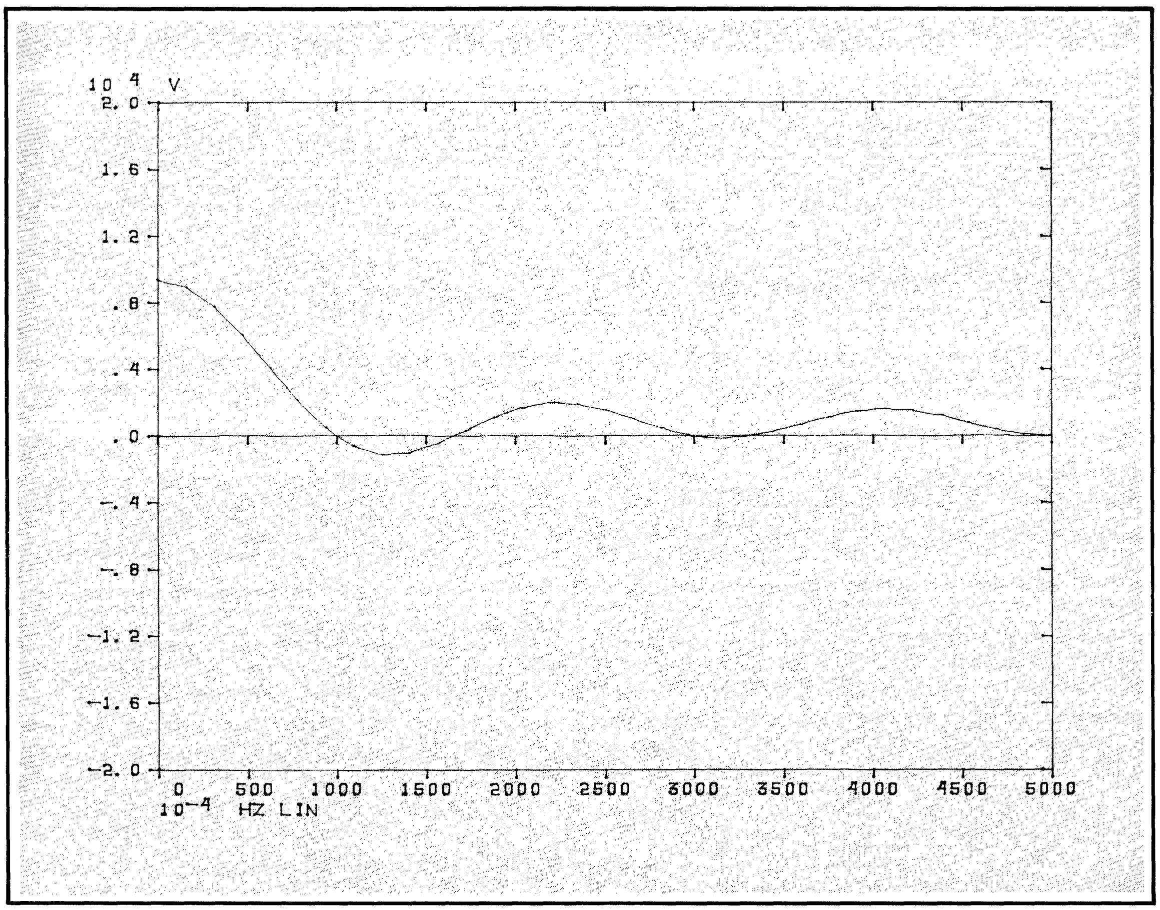
2. Enter

   USER PROG 5800 SPACE n1 ENTER

   where n1 is either 6 (or default) to specify the Terminal, or 10 or 35 to specify the Plotter.

   Figure 6-2 is an example of the plot that will be produced when 5800 is executed as above.

Figure 6-2. Familiarization Program Plot



NOTE

The following steps use programs (explained later in this section) to rescale the previous plot and demonstrate the n1 = -1 function.

3. Now change the size from full page to quarter page plot by entering the following commands. (Remember to change the paper in the Plotter.)

   USER PROG 5804 SPACE 480 SPACE 500 ENTER
   USER PROG 5813 SPACE 1 ENTER (suppress entry & system response echo on Terminal)
   USER PROG 5814 ENTER (erase Terminal screen)
   USER PROG 5800 SPACE -1 ENTER

4. Without changing the paper in the Plotter, enter the following commands.

   USER PROG 5805 SPACE 0 SPACE 500 ENTER
   USER PROG 5800 SPACE -1 ENTER

   Notice that the same plot is moved to the lower right quadrant of the page and that the parameters previously entered for 5804 (which changed the size of the plot) remain the same because 5800 was executed with the n1 parameter equal to -1.

5. Now move the plot to the upper right quandrant by executing the following commands.

   USER PROG 5805 SPACE 520 SPACE 500 ENTER
   USER PROG 5800 SPACE -1 ENTER

6. To return the echo on the Terminal, use the following command:

   USER PROG 5813 ENTER

5803 TEXT BUFFER EDITOR

This command is used to create text and/or modify text that is later used to annotate plots or print messages on the terminal. User Program 5803 can be used to edit text in the core-resident text buffer, replace specific text in a specified disc text buffer, or add text to a specified disc text buffer.

The format that must be used when entering text is as follows:

nn
line 1 containing ASCII text
    through
line xx containing ASCII text
/*

where nn is the text identification number between 01 and 99, and /* is the text entry termination command. Be sure two numbers are entered for nn (i.e., 06, not just 6).

NOTE

The /* and the text identification number are not output when the text is printed on the Terminal or the Plotter with the Gold TEXT command (Y 5819).

Text buffers 51-55 are reserved for automatic measurement programs listed in Appendix D.

Once text has been entered, it may be used to annotate plots by executing User Programs 5808 (points to where text is to begin) and 5819 (writes text). It may also be used to print text on the terminal by invoking only User Program 5819. Also, User Program 5817 is used to write vertical and horizontal unit text (labels). User Programs 5838 and 5839 are used to read and write text buffers between disc and core. Refer to these User Program descriptions in this section for more information.

Command Format

USER PROG 5803 SPACE n1 SPACE n2 ENTER

n1 = disc text buffer number (default or 0 will cause the system to use the core-resident text buffer).

n2 = text message identification number between 01 and 99 to indicate where the text (about to be entered) is to replace existing text or is to be inserted in the disc text buffer indicated by n1.

n2 = default, you may use text editing commands.

Example Command Usage

USER PROG 5803 ENTER

Uses the core-resident text buffer for saving text entered via edit commands. Text is not automatically stored on the Disc and may be lost if 5803 is subsequently executed using parameter n1 to reference a disc text buffer.

USER PROG 5803 SPACE n1 ENTER

Loads specified disc text buffer n1 to core-resident text buffer, then uses core-resident text buffer for editing purposes via edit commands and automatically saves text in disc text buffer indicated by n1.

USER PROG 5803 SPACE 0 SPACE n2 ENTER

Uses core-resident text buffer for saving the text about to be entered. Replaces or inserts text into the core-resident text buffer. Text may be lost if 5803 is subsequently executed using parameter n1 to reference a disc text buffer.

USER PROG 5803 SPACE n1 SPACE n2 ENTER

Loads disc text buffer n1 into core-resident text buffer, then uses the core-resident text buffer to receive the text about to be entered. The new text is then added to or inserted into the data that was in disc text buffer n1, according to n2 — the text message identification number.

NOTE

When n2 is >0 and <100, the system reads a single line of text (70 characters) from the terminal and stores it in the core-resident text buffer and disc text buffer (if n1 is specified). Note that the end of the message and the signal to store is indicated by the RETURN key on the terminal where, in text editing mode, the TERM key (/) is used to end the entry of text.

When you perform an edit with n1 defaulted, you are editing the core-resident text buffer. Therefore, if a disc text buffer is to be edited, you must call appropriate disc text buffer into processor memory (i.e., USER PROG 5803 SPACE n1 ENTER).

The use of the RESTART command ends the edit session without restoring the text buffer to the Disc.

Error Conditions

IRRECOVERABLE ERRORS

When an irrecoverable error is detected, 5803 outputs the following messages and returns to the READY state.

B1 WHAT?	Disc text buffer number (n1) is <0.
B2 WHAT?	Text message ID number (n2) is ≤0 or >99.
B3 WHAT?	An error has been detected during editing.
B8 WHAT?	System is unable to configure.
B9 WHAT?	An illegal disc text buffer is being accessed.
C4 WHAT?	A disc I/O error has occurred.

All of the above errors can be corrected by entering the 5803 command again. If any of the above errors continue to occur, servicing may be necessary.

WARNING MESSAGES

WARNING — CLEAR NEW DISC BUFFERS

The disc text buffer just transferred from Disc to the core-resident text buffer area may need to be cleared of extraneous text before continuing. However, the text just transferred may be of some use. In this case, input LIST ENTER to obtain a listing of the existing text. If it is not what you want, then input CLEAR ENTER.

ILLEGAL BUFFER

This message indicates that the text buffer data just read is not valid. To clear the data enter CLEAR ENTER.

???

When the system detects an error in the input stream, it issues the above message. Simply type a correct command to continue.

Editing Commands

If n2 is defaulted, the following commands are available on the Keyboard or on the Terminal.

Keyboard	Terminal	Function
CLEAR	CL	Clears the current core-resident text buffer of all information.
DELET nn	/D nn	Deletes text message identified as nn from the core-resident text buffer.
INSRT	/I	Insert all subsequent text messages that are not already present in the core-resident text buffer.
INSRT nn	/I nn	Insert text message nn into the core-resident text buffer.
LIST	/L	List all messages in the core-resident text buffer.
LIST nn	/L nn	List text message nn.
RPLAC nn	/R nn	Replace text message nn in the core-resident text buffer.
STORE nn	>X nn	Store the current core-resident text buffer on Disc in text buffer nn. This command allows you to store the core-resident text buffer in more than one place on the Disc.
TERM	/	Terminates the edit mode after storing the current core-resident text buffer on Disc in text buffer n1.
RSTART	No equivalent	Terminates the edit mode without storing the core-resident text buffer on Disc.

Familiarization Procedure

Perform the following steps to gain a better understanding of the 5803 functions available. (CR is carriage return on the Terminal, i.e., RETURN with AUTO LF depressed.)

1. Enter the following commands to create the necessary annotations needed for the plot description.

    USER PROG 5803 SPACE 1 ENTER
    CLEAR ENTER (enter from Keyboard)
    INSERT ENTER
    01 CR
    SINE X/X FUNCTION CR
    /* CR
    02 CR
    THIS IS THE REAL OR COSINE CR
    VALUES OF THE SYNC FUNCTION CR (enter from Terminal)
    /* CR
    03 CR
    LIN AMP UNITS CR
    /* CR
    04 CR
    DC TO 50 HZ CR (enter from Terminal)
    /* CR
    TERM ENTER (enter from Keyboard)

    The above text messages (01, 02, 03, and 04) are now stored on the Disc in text buffer number 1. If you had wanted to, you could have also stored the text messages in another disc text buffer by executing the STORE command, before the TERM command, indicating a disc buffer other than number 1.

    To check if the text buffers were entered correctly, you can list them as follows:

    USER PROG 5803 SPACE 1 ENTER
    LIST ENTER
    TERM ENTER

2. Enter the following Keyboard commands to create some data for plotting.

    BLOCK SIZE 512 ENTER
    CLEAR ENTER
    KEYBOARD 0 SPACE 0 SPACE 5 ENTER
    KEYBOARD 5 SPACE 0 SPACE 9 ENTER
    1 ENTER
    F ENTER

3. Initialize the system to plot on the Plotter or the Terminal.

    USER PROG 5821 SPACE n1 ENTER (where n1 is 6 for the Terminal, 35 or 10 for the Plotter)

    The Plotter pen should move to the upper left-hand corner if the Plotter is indicated.

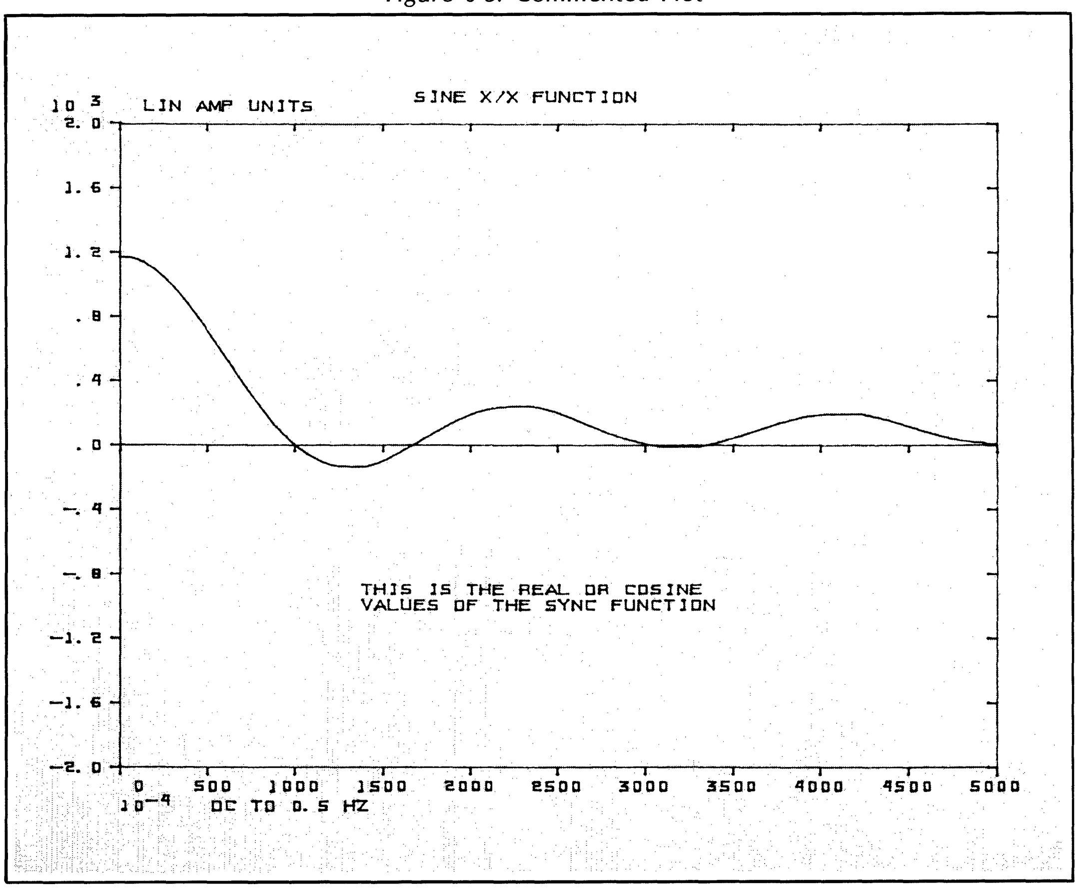
4. Enter the following Keyboard Program using standard keyboard program editing (refer to KEYBOARD key in Section 3).

```
RPLAC  ENTER
  L       Ø
  Y    5814                ◄──────── (Not necessary if you are using the Plotter)
  Y    5804
  Y    5805
  Y    5806
  Y    5807
  Y    5809
  Y    5810
  Y    5811
  Y    5815
  Y    5816
  Y    5817     3      4   ◄──────── Writes text messages 3 and 4
  Y    5808   98Ø    375
  Y    5819     1          ◄──────── Writes text message 1
  Y    5808   3ØØ    32Ø
  Y    5819     2          ◄──────── Writes text message 2

TERM  ENTER
```

5. Enter JUMP 0 ENTER to produce the full page plot shown in Figure 6-3.

Figure 6-3. Commented Plot



Use of the Back Slash

The use of the back slash (\) at the end of a line of text and before CR is executed, allows you to eliminate the execution of the CR when the text is finally printed (i.e., the line of text is read out and the cursor is not returned to the left-hand margin and is not advanced a line; the cursor remains at the end of the line of text). This function is useful when you wish to add more information to a plot, say, after having viewed the results of a test that produces variable results.

Use of the back slash (\) is restricted in the following ways:

1. It only prevents a CR from executing. You must use the on-line text entry feature of 5803 to activate the Terminal.

2. The back slash only works within the structure of a Keyboard Program.

3. If the back slash falls on an odd-numbered character (column), the system will automatically add a space. For example, when the following messages are executed, the responses will start at column 14, even though the first message is 12 characters and the second is 13.

   THE VALUE IS

   THE VALUE IS:

5804 SET PLOT SIZE

User Program 5804 allows you to specify the size of your plot in terms of percent of full page or screen. The size of the plot can be specified to include space for labels or exclude space for labels. This command is normally used in combination with 5805, 5806, and 5807 to place several plots on a single paper.

NOTE

The parameters entered using 5804 are not actualized until User Program 5815, 5816, and 5817 are executed. 5804 must be called before 5815, 5816, and 5817 are executed if you want to change the size of your plot.

Command Format

USER PROG 5804 SPACE n1 SPACE n2 SPACE n3 ENTER

n1 = 1 to 1000 for vertical dimension where 1000 = 100% or top of screen or page. Default = full scale.

n2 = 1 to 1000 for horizontal dimension where 1000 = 100% or right edge of screen or page. Default = full scale.

n3 ≠ 0, do not include space for labels. When 5817 is executed, the labels for the horizontal and vertical axes are output outside the boundaries specified by n1 and n2.

n3 = 0 or default, include space for labels within the boundaries defined by n1 and n2.

NOTE

When full page is specified (i.e., n1 and n2 = 1000 or default), you should default n3 as there will be no room for labels outside of full scale boundaries. When n1 and/or n2 are smaller than 100 (i.e., size is less than 10%), n3 should be set ≠ 0, since if n3 is defaulted the program will reserve a fixed space for labels which may be larger than the total size specified by the parameters n1 and/or n2. Once the size of the plot has been changed, it remains the indicated plot size until 5804 is executed again with different parameters.

Error Conditions

B7 WHAT? Either n1 or n2 is <0 or >2000. Repeat the 5804 command to recover.

Familiarization Procedure

Perform the following steps to gain a better understanding of the 5804 functions available.

1. Enter the data that is to be plotted with the following commands:

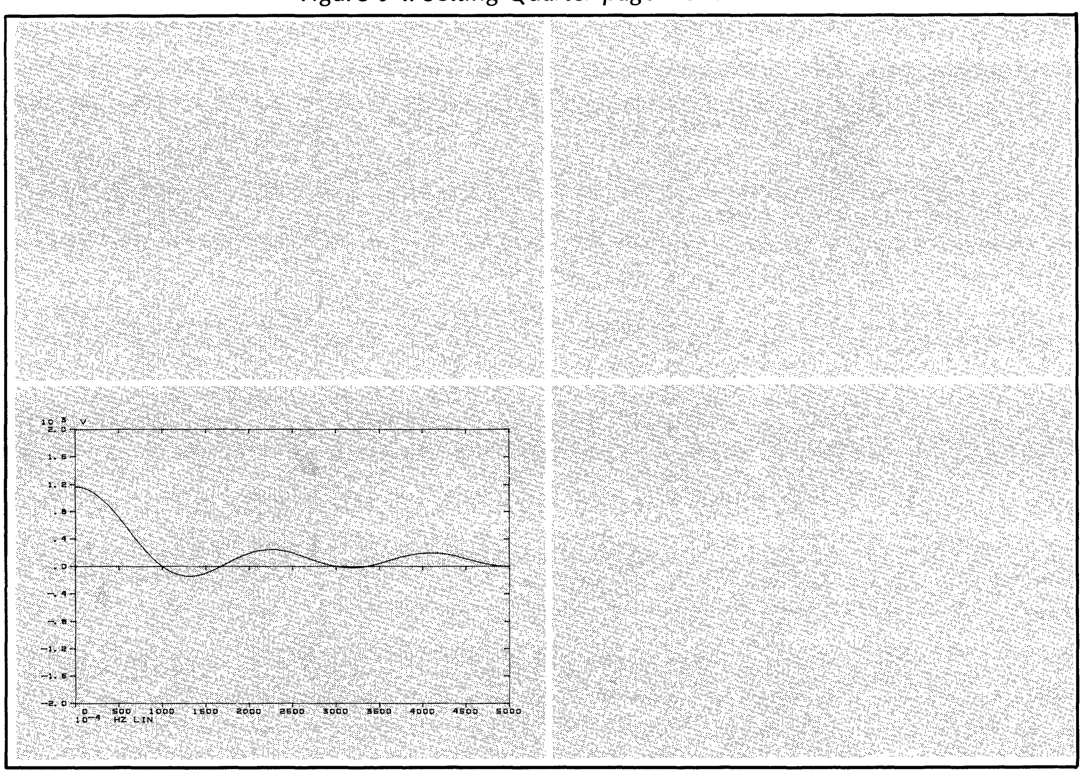
   BLOCK SIZE 512 ENTER
   CLEAR ENTER
   KEYBOARD 0 SPACE 0 SPACE 5 ENTER
   KEYBOARD 5 SPACE 0 SPACE 9 ENTER
   1 ENTER
   F ENTER

2. To create a plot that is a quarter of full page, located in the lower left corner, and **includes** space for labels, execute the following commands:

   USER PROG 5809 ENTER
   USER PROG 5813 SPACE 1 ENTER (supress echo)
   USER PROG 5814 ENTER (use if you are plotting on the terminal)
   USER PROG 5804 SPACE 450 SPACE 450 ENTER
   USER PROG 5805 ENTER
   USER PROG 5815 ENTER (plot display screen)
   USER PROG 5816 ENTER (draw axis)
   USER PROG 5817 ENTER (label axes)

   Figure 6-4 is an example of the plot that should be made when the above commands are through executing.

Figure 6-4. Setting Quarter-page Plot Size

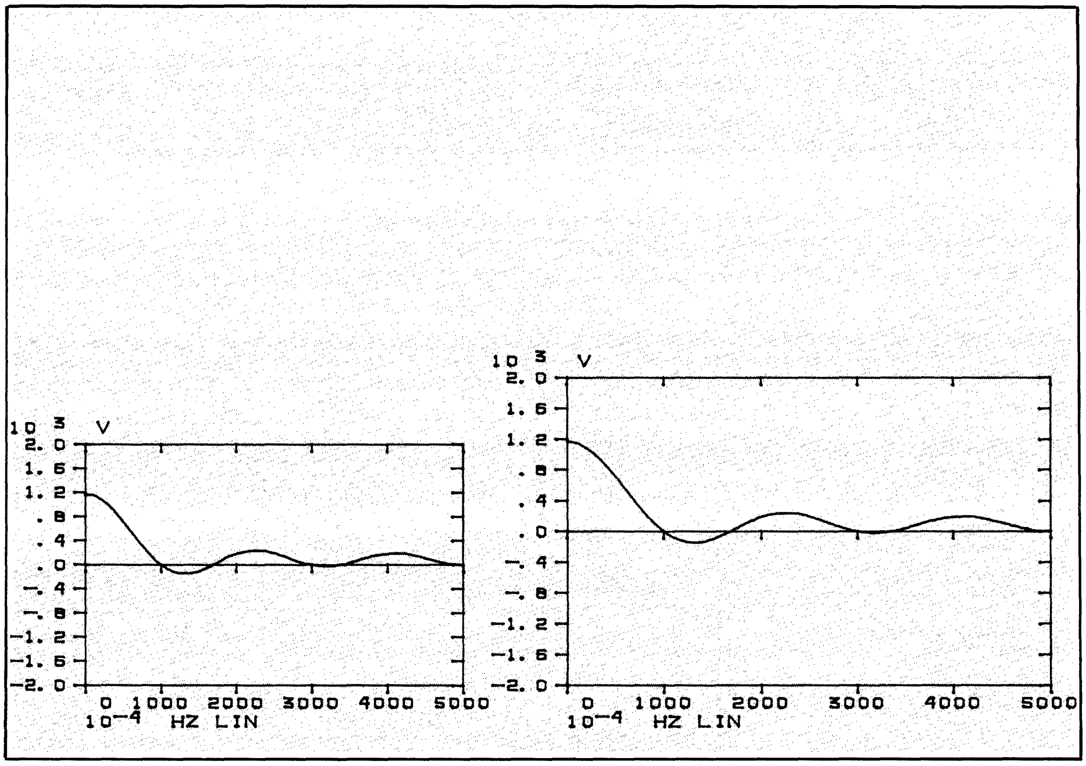


3. To create a plot that is a quarter of full page, located in the lower right corner, and does **not** include space for labels, execute the following commands. If you are using the Plotter as your plotting device, do not change paper.

   USER PROG 5805 SPACE 0 SPACE 450 ENTER
   USER PROG 5804 SPACE 450 SPACE 450 SPACE 1 ENTER
   USER PROG 5815 ENTER
   USER PROG 5816 ENTER
   USER PROG 5817 ENTER

   Figure 6-5 is an example of the plot that should be made when the above commands are through executing.

Figure 6-5. Changing Plot Size by Excluding Labels



Notice that the second plot area for the data is larger as the labels were not included in calculating the plot area and were placed outside the boundaries specified by User Program 5804.

NOTES

When specifying your plot size excluding space for labels you must be sure that your plot size is small enough so that enough room is still left on the Terminal screen or on the page you defined via the Plotter controls.

In Figure 6-4, X_o,Y_o is the origin of the plot area which includes the plot labels. X_p,Y_p is the origin of the plot area, excluding labels. The space required for labels is always the same, regardless of plot size, since character size is independent of plot size. On the Terminal and Plotter the space required for labels is as follows (in device units):

	Horizontal (X_p-X_o)	**Vertical (Y_p-Y_o)**
Terminal	72	61
Plotter	704	596

5805 SET PLOT LOCATION ON PAGE

User Program 5805 allows you to specify the location of your plot on the page or the terminal screen, taking into account the space for vertical and horizontal labels. You are actually specifying where to place the X-Y axes origin in terms of vertical distance from the bottom (in tenths of % of full page) and horizontal distance from the left (in tenths of % of full page).

NOTE

> The parameters entered using 5805 are not actualized until User Programs 5815, 5816, and 5817 are executed. 5805 must be called before 5815, 5816, or 5817 is executed if you want to change the location of your plot on the page or screen.

The 5805 command is normally used in combination with 5804, 5806, and 5807 to place several plots on a single paper.

Command Format

USER PROG 5805 SPACE n1 SPACE n2 ENTER

n1 = Vertical distance from bottom in tenths of % of full page (range = 1 to 1000)

n2 = Horizontal distance from left in tenths of % of full page (range = 1 to 1000)

If n1 and n2 are defaulted, the lower left corner is assumed.

Error Conditions

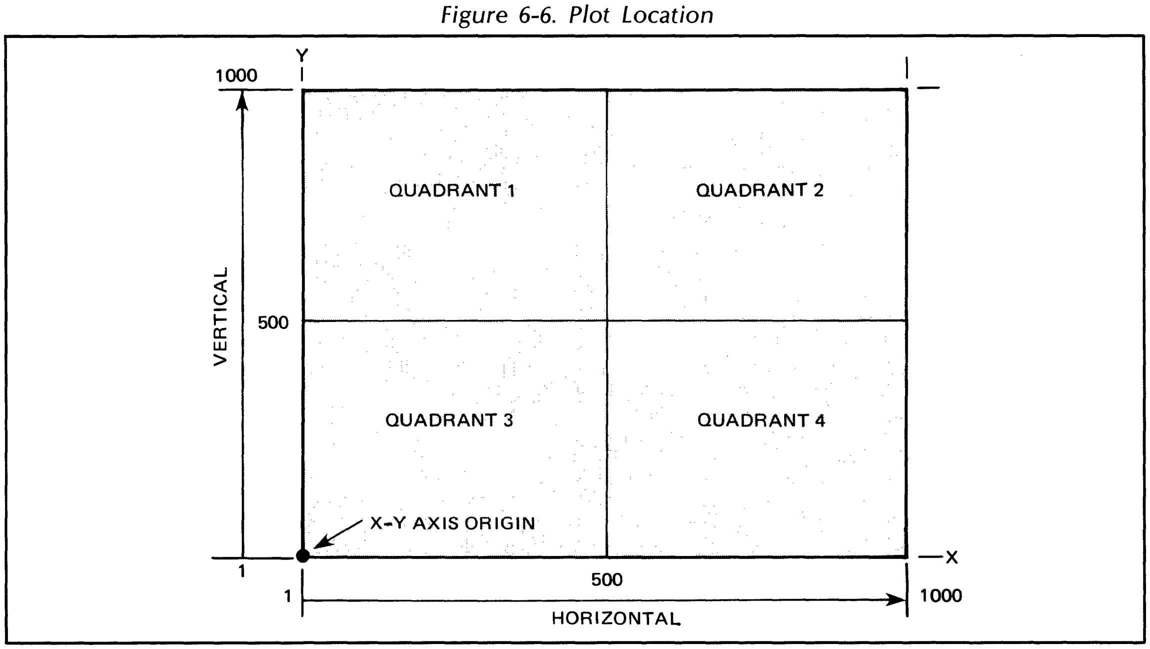
B7 WHAT? Either n1 or n2 is >2000. Repeat the command.

Note that it is possible to enter up to 2000 for n1 and n2, however, anything specified above 1000 becomes meaningless when you try to plot your data (User Program 5815, 5816, 5817).

Familiarization Procedure

5805 has been used several times in explaining 5800 and 5804. Study Figure 6-6 and associate it with 5805 usage in the preceding familiarization procedures for 5800 and 5804.

Figure 6-6. Plot Location



To place a plot in any one of the four quadrants depicted in the preceding figure, you must use 5804 and 5805 in combination as follows:

Quandrant 1:

USER PROG 5804 SPACE 500 SPACE 500 ENTER
USER PROG 5805 SPACE 500 SPACE 0 ENTER

Quandrant 2:

USER PROG 5804 SPACE 500 SPACE 500 ENTER
USER PROG 5805 SPACE 500 SPACE 500 ENTER

Quandrant 3:

USER PROG 5804 SPACE 500 SPACE 500 ENTER
USER PROG 5805 SPACE 0 SPACE 0 ENTER

Quandrant 4:

USER PROG 5804 SPACE 500 SPACE 500 ENTER
USER PROG 5805 SPACE 0 SPACE 500 ENTER

Remember, it is the X-Y axes origin that we are describing with the n1 and n2 parameters in 5805 where the origin is the lower left corner of the plot area.

NOTE

Once you have executed 5805 with a certain set of parameters, it will not be necessary to execute it again until you wish to change the parameters, thus moving the plot to a different portion of the page.

5806 SET PLOT POSITION INCREMENT

User Program 5806 allows you to specify incremental positioning of the X-Y axes origin. This command is useful in presenting data in a "waterfall" display for trend analysis purposes.

NOTE

The parameters entered using 5806 are not actualized until User Programs 5815, 5816, or 5817 are executed. 5806 must be called before 5815, 5816, or 5817 is executed if you wish to change the amount of plot origin incrementation.

The 5806 command is normally used in combination with 5804, 5805, and 5806 to place several plots on a single sheet of paper.

Command Format

USER PROG 5806 SPACE n1 SPACE n2 ENTER

n1 = Vertical axis increment in 0.1% of full scale (range = 0 to 1000)

n2 = Horizontal axis increment in 0.1% of full scale (range = 0 to 1000)

When 5806 is executed with no parameters, the origin of the plot is incremented automatically by the amount set by a previous call to 5806. Therefore, if you do not want incrementing to take place, you must execute 5806, originally, with n1 and n2 equal to 0 (no default is allowed).

NOTE

> Executing 5806 with n1 and n2 equal to zero does not reset the plot origin to the lower left corner. 5805 must be used to do this.

Error Conditions

B7 WHAT? Either n1 or n2 <0 or >2000. Repeat the command.

Familiarization Procedure

Perform the following steps to gain a better understanding of the 5806 functions available. Note that the following procedure requires the use of a function generator.

1. To produce data similar to that shown in the plot examples:

   Set the 5466 ADC as follows:

   SAMPLE MODE to INT
   MAX FREQ to kHz/μs
   MULTIPLIER to 200/5/2.5
   INPUT SELECTOR to A
   OVERLOAD VOLTAGE A to 2
   TRIGGER SOURCE to FREE RUN

   Set the 3330A Function Generator (or comparable device) as follows:

   CHANNEL A OUTPUT to CHANNEL A INPUT on ADC
   DIAL to 1
   RANGE to X1K
   CHANNEL A knob to SQUARE

   Set REPEAT/SINGLE switch to REPEAT on Keyboard

   Enter ANALG IN ENTER on Keyboard

   On 3330A, adjust AMPLITUDE knob to about 2.4 volts peak-to-peak (just below overload)

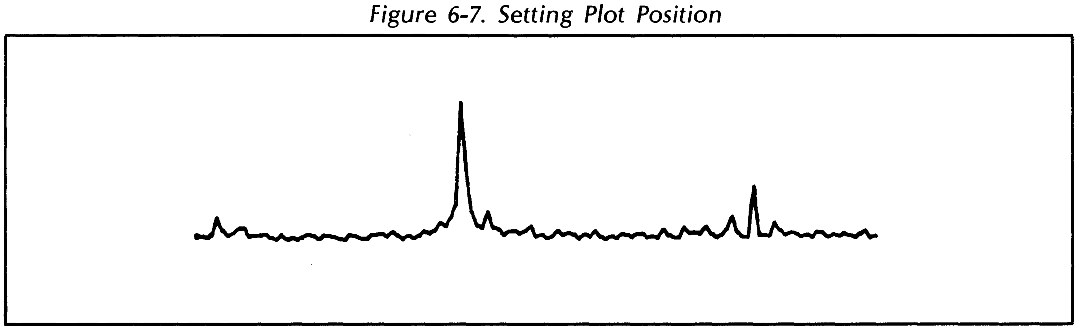
   Set REPEAT/SINGLE switch to SINGLE position on 5475A

   Enter the following keyboard commands on the 5475A:

   ANALG IN ENTER
   F ENTER
   POLAR ENTER

   Figure 6-7 is a plot example of the data that should now be on display. Note that the data may vary slightly.

Figure 6-7. Setting Plot Position

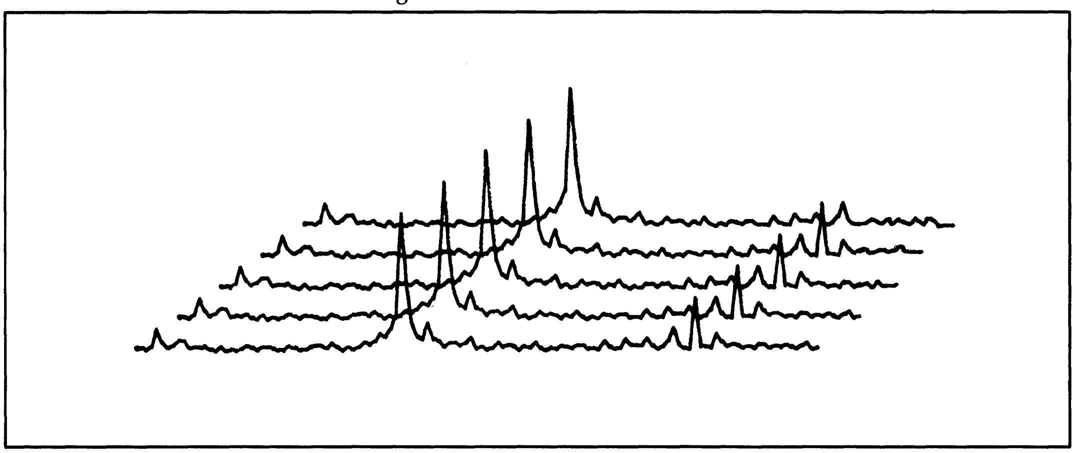


2. Enter the following Keyboard Program to produce position incremented plots of the above data:

    REPLACE 0 ENTER
    LABEL 0 ENTER
    USER PROG 5821 SPACE n1 ENTER (where n1 is 10 or 6)
    USER PROG 5814 ENTER (only used if Terminal is specified)
    USER PROG 5804 SPACE 450 SPACE 450 ENTER (to ensure plot
    USER PROG 5805 ENTER origin is initially lower left corner)
    USER PROG 5806 SPACE 100 SPACE 100 ENTER
    USER PROG 5815 ENTER
    USER PROG 5806 ENTER
    USER PROG 5815 ENTER
    USER PROG 5806 ENTER
    USER PROG 5815 ENTER
    USER PROG 5806 ENTER
    USER PROG 5815 ENTER
    USER PROG 5806 ENTER
    USER PROG 5815 ENTER
    END ENTER
    TERM ENTER
    JUMP 0 ENTER

   Figure 6-8 shows an example of incrementing several plots.

Figure 6-8. Incremented Plots



In this plot the data was offset or incremented by 10 percent in both the horizontal and the vertical axis directions. Notice that in subsequent calls to 5806 both n1 and n2 were defaulted, thus, each subsequent plot was offset by 10%. To return to a zero, a zero increment situation, you must specify n1 and n2 as 0 in the next call to 5806.

5807 SET PLOT ORIENTATION

User Program 5807 allows you to specify a horizontal or vertical orientation on the screen or page.

NOTE

> The parameters entered using 5807 are not actualized until User Programs 5815, 5816, and 5817 are executed. 5807 must be called before 5815, 5816, or 5817 is executed if you want to alter the orientation of a plot.

The 5807 command is normally used in combination with 5804, 5805, and 5806 to place several plots on a single piece of paper.

Command Format

USER PROG 5807 SPACE n1 ENTER

n1 = 0 or default, the standard orientation from left to right is plotted.

n1 = <0, rotate plot 90° clockwise

n1 = >0, rotate plot 90° counterclockwise

Familiarization Procedure

Perform the following steps to gain a better understanding of the 5807 functions available. Note that the following procedure requires that you specify whether you are using the Plotter or the Terminal via the 5821 command.

1. To create some data to be plotted, enter (via the Keyboard) the following Keyboard commands:

   BLOCK SIZE 64 ENTER
   CLEAR ENTER
   KEYBOARD 0 SPACE 0 SPACE 5 ENTER
   KEYBOARD 5 SPACE 0 SPACE 9 ENTER
   1 ENTER
   F ENTER

2. To make a plot of the above data that is standard horizontal orientation, enter the following commands:

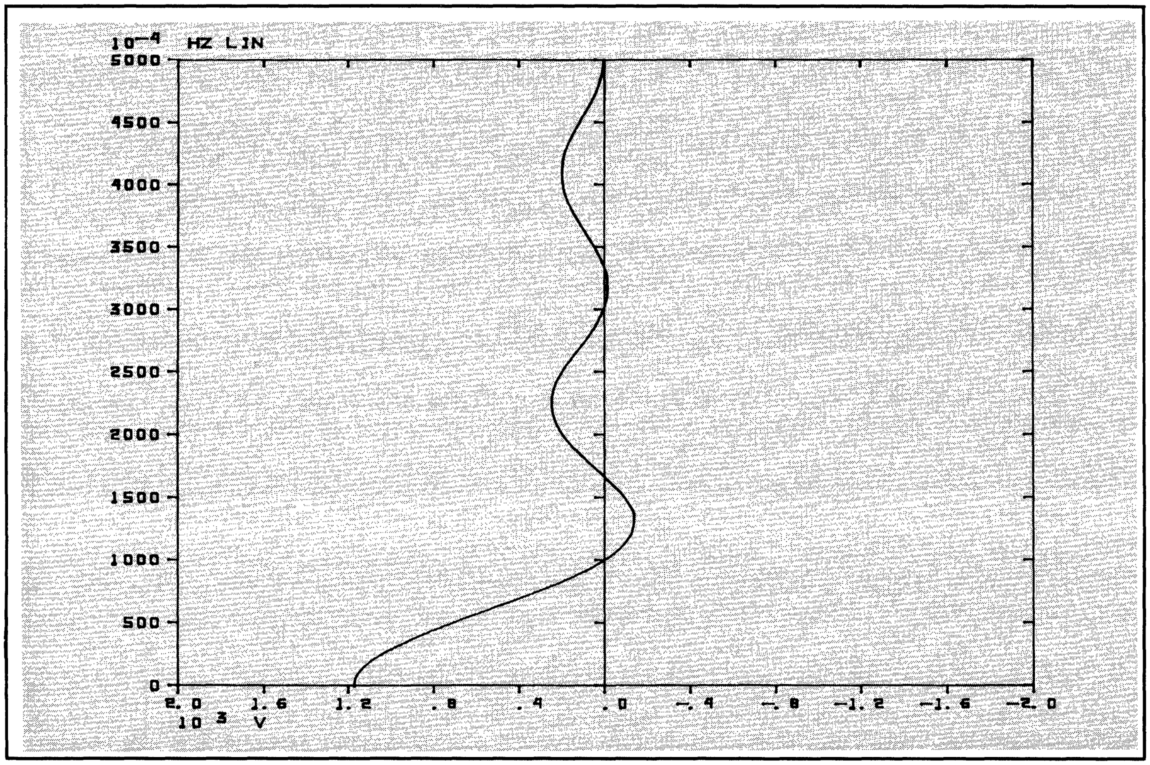
   USER PROG 5804 ENTER
   USER PROG 5805 ENTER
   USER PROG 5806 SPACE 0 SPACE 0 ENTER
   USER PROG 5807 ENTER
   USER PROG 5813 SPACE 1 ENTER
   USER PROG 5814 ENTER (used only if Terminal was specified)
   USER PROG 5800 SPACE -1 ENTER

3. Enter the following commands to make a plot that has a 90° rotation counterclockwise from standard orientation. Change the paper in the Plotter if appropriate.

   USER PROG 5804 ENTER
   USER PROG 5805 ENTER
   USER PROG 5806 ENTER
   USER PROG 5807 SPACE 1 ENTER
   USER PROG 5813 SPACE 1 ENTER
   USER PROG 5814 ENTER (Terminal only)
   USER PROG 5800 SPACE -1 ENTER

Figure 6-9 is an example of the plot that should be produced.

Figure 6-9. Changing Plot Orientation



5808 SET MESSAGE POSITION

User Program 5808 is used in subsequent calls to specify the position of text messages previously stored in the text buffer by User Program 5819.

NOTE

> The position of the message is defined as the bottom left corner of the first character.

Command Format

USER PROG 5808 SPACE n1 SPACE n2 ENTER

- n1 = Vertical distance from bottom in tenths of % of full scale (range = 1 to 980). Default = top side of screen/page.
- n2 = Horizontal distance from left in tenths of % of full scale (range = 1 to 999). Default = left side of screen/page.

If n1 is defaulted, n2 must be defaulted; otherwise the text printed by 5819 will be placed erroneously on the screen/page.

Error Conditions

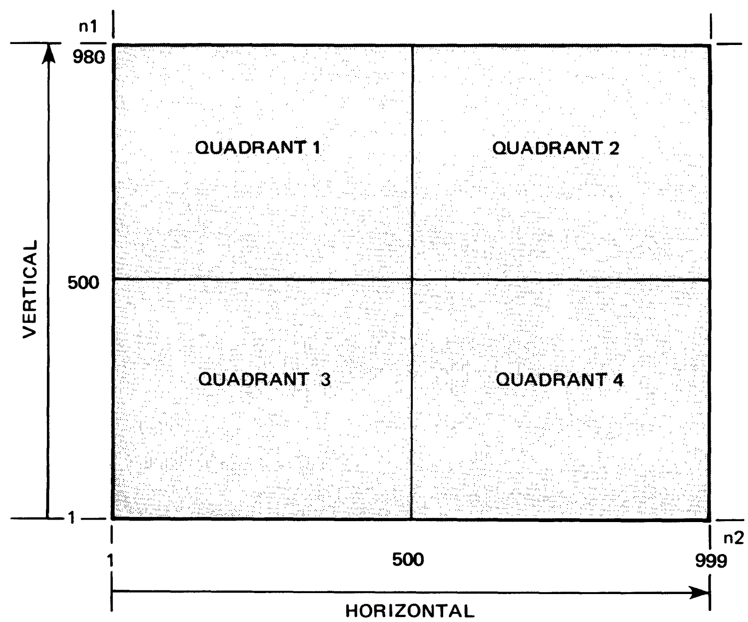
B7 WHAT? Either n1 or n2 is <0 or >2000. Repeat the command.

Note that it is possible to enter up to 2000 for n1 and n2. However, anything specified above 999 becomes meaningless (in the case of n2), and above 980 (in the case of n1), when you try to plot your data (User Program 5815, 5816, 5817).

Familiarization Procedure

To place text at a specified point on a plot you must use 5808 and 5819 in combination. Use Figure 6-10 and commands to gain a clearer understanding of 5808 usage.

Figure 6-10. Message Position Quadrant Example



Enter some text using User Program 5803. Give your text message ID number 01.

Example Command Combinations:

Quadrant 1:

USER PROG 5808 ENTER
USER PROG 5819 SPACE 1 ENTER

Quandrant 2:

USER PROG 5808 SPACE 980 SPACE 500 ENTER
USER PROG 5819 SPACE 1 ENTER

Quandrant 3:

USER PROG 5808 SPACE 500 ENTER
USER PROG 5819 SPACE 1 ENTER

Quandrant 4:

USER PROG 5808 SPACE 500 SPACE 500 ENTER
USER PROG 5819 SPACE 1 ENTER

5809 SET MODE, ORIGIN, AND SCALE PARAMETERS

The use of this command supplies the display mode, origin, and scale parameters to User Program 5815. This user program must be executed at least once before User Program 5815, 5816, or 5817 is executed.

Command Format

USER PROG 5809 SPACE n1 SPACE n2 SPACE n3 ENTER

MODE Switch

n1 = 0, Real/Magnitude is plotted

n1 = default, use the MODE switch to control plot mode

n1 = 1, Imaginary/Phase is plotted

n1 = -1, Complex is plotted

ORIGIN Switch

n2 = 0, linear horizontal origin (LEFT) is plotted

n2 = default, use the HORIZONTAL ORIGIN switch to control plot origin

n2 = 1, origin center is plotted, e.g., for correlation and convolution type data

n2 = -1 to -5, Log horizontal origin is plotted with the parameter number entered corresponding to the number of decades (i.e., 1 to 5 decades)

If n2 is defaulted with the ORIGIN switch in the LOG position, 5 decades are plotted.

SCALE Switch

n3 = 0, plot the data as if the SCALE switch were in its vertical position (12 o'clock)

n3 = default, use the SCALE switch to control the plot scale.

n3 = >0 up to 8, plot data to the scale indicated by the parameter entered. Note that counterclockwise rotation is not allowed. Refer to foldout Figure 2-8 for an explanation of the SCALE switch function.

In conclusion, in order to change mode, origin, and scale, 5809 must be executed before 5815, 5816, and/or 5817 are called. Mode, origin, and scale can be indicated by:

1. Entering the 5809 command with n1, n2, and n3 specified.
2. Entering the 5809 command without parameters so that Display Unit switches are used to indicate mode, origin, and scale.

Error Conditions

C1 WHAT?	If any of the parameters are not those listed above, this error occurs. Repeat the command.
C0 WHAT?	If too many parameters are passed, this occurs. Repeat the command.

Familiarization Procedure

Use the following procedure to become more familiar with the 5809 command.

1. Set the Display Unit controls as follows:

   MODE to REAL/MAGNITUDE
   SCALE to 12 o'clock position
   HORIZONTAL ORIGIN to LEFT

   The remaining switches and knobs should be in their nominal positions as specified in Section 2.

2. Enter the data that is to be plotted using the following commands:

   BLOCK SIZE 512 ENTER
   CLEAR ENTER
   KEYBOARD 0 SPACE 0 SPACE 5 ENTER
   KEYBOARD 5 SPACE 0 SPACE 9 ENTER
   1 ENTER
   F ENTER

3. Enter the following command to indicate which plotting device you will be using.

   USER PROG 5821 SPACE n1 ENTER (n1 = 6 for the Terminal or 10 for the Plotter)

4. Enter the following commands to make a full page plot of the data as it now appears on the display screen.

   USER PROG 5814 ENTER (used only if the Terminal is the plotting device)
   USER PROG 5804 ENTER
   USER PROG 5805 ENTER
   USER PROG 5809 ENTER (with this command, system copies Display Unit panel)
   USER PROG 5813 SPACE 1 ENTER
   USER PROG 5814 ENTER (needed for Terminal only) USER PROG 5815 ENTER (draw data)
   USER PROG 5816 ENTER (draw axes)

   Example 1 of Figure 6-11 shows the plot that should be made.

5. To reduce the plot size and plot the IMAGINARY/PHASE mode, enter the following commands:

   USER PROG 5814 ENTER (use to clear Terminal display)
   USER PROG 5804 SPACE 450 SPACE 450 ENTER
   USER PROG 5809 SPACE 1 ENTER (specifies imaginary mode)
   USER PROG 5815 ENTER
   USER PROG 5816 ENTER

   If the Plotter is used and you did not change the paper, then example 2 of Figure 6-11 shows the plot that should be produced.

   Notice that you were not requested to alter the MODE knob on the Display Unit in order to cause the imaginary mode of the data being displayed to be plotted.

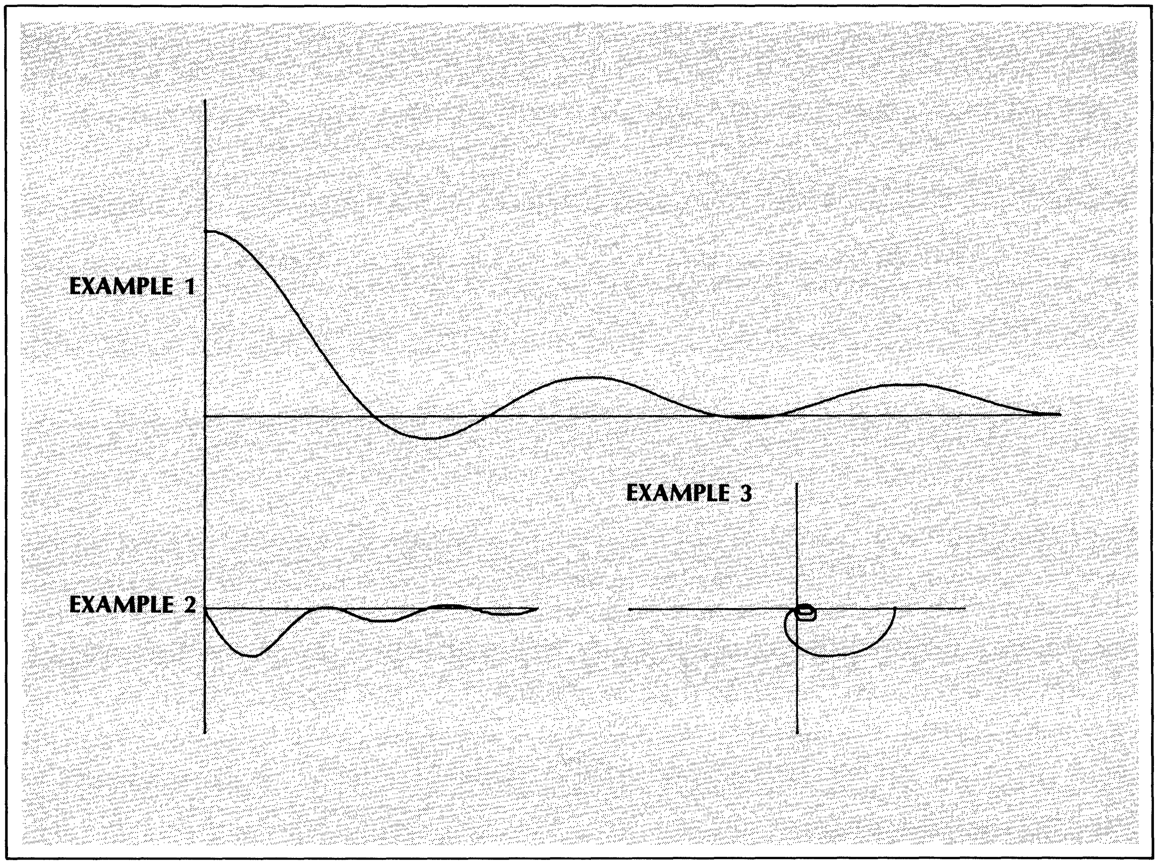
6. A complex mode plot of the data on display can be made by either changing the MODE knob to COMPLEX position and executing 5809 without any parameters, or by executing 5809 with n1 equal to –1. Perform the following commands to obtain a complex mode plot.

   USER PROG 5814 (used if Terminal is plotting device)
   USER PROG 5805 SPACE 0 SPACE 450 ENTER
   USER PROG 5809 SPACE –1 ENTER
   USER PROG 5815 ENTER
   USER PROG 5816 ENTER

   Notice that when User Program 5816 is executed without a parameter that it automatically selects the appropriate axes.

Example 3 of Figue 6-11 shows the final plot that is produced if you are using the Plotter and you did not change the paper during the entire procedure.

Figure 6-11. Example Settings of Mode, Origin, and Parameters



NOTE

When 5815 is executed in succession, without 5809 being executed in between, the parameters from the last execution of 5809 are used in each succeeding plot until 5809 is executed again with different parameters.

5810 SET HORIZONTAL RANGE USING FREQUENCY OR TIME UNITS

User Program 5810 is used before 5815, 5816, or 5817 are called to let the system know the length of the horizontal range that is to be plotted.

In the process of making plots, 5815, 5816, and 5817 look at the frequency code and coordinate code (stored with the data being displayed), and the parameters entered using 5810, to determine the appropriate labels and the horizontal range.

Normally 5810 is executed without parameters to define the total length of the horizontal range. However, using the n1, n2, n3, and n4 parameters, you can indicate that only a portion of the total horizontal range is to be plotted and then labels are applied appropriately when 5815, 5816, and 5817 are executed.

NOTE

User Program 5810 allows you to specify the horizontal range in terms of frequency or time units, whereas User Program 5829 allows you to specify the horizontal range in terms of channel numbers (i.e., from some starting channel number to some ending channel number of the data being displayed). Refer to the discussion on 5829 for more information.

Command Format

USER PROG 5810 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

n1 = horizontal maximum (frequency or time), where n1 is between 0 and 9999.

n2 = horizontal maximum exponent, where the horizontal maximum is set to $n1 \times 10_{n2}$ Hz or seconds.

n3 = horizontal minimum (frequency or time), where n3 is between 0 and 9999.

n4 = horizontal minimum exponent, where the horizontal minimum is set to $n3 \times 10_{n4}$ Hz or seconds.

If 5810 is executed with no parameters, then subsequent plots will have the default range of a full data block.

NOTES

n2, n3, and n4 can be defaulted to specify a range from 0 to n1.

When 5817 is executed, it truncates the labels to integers. Therefore, n1 should have a trailing zero to avoid loss of significant digits in the labels.

Whenever a non-default range is specified, User Program 5817 uses the frequency code of the displayed data block to determine the range.

Error Conditions

A6 WHAT?	Either n1 or n3 is <0 or >10000. Repeat the command.
A7 WHAT?	n4 is illegal. Repeat the command.
A8 WHAT?	$n3 \times 10_{n4}$ is $\geq$ $n1 \times 10_{n2}$

Familiarization Procedure

Perform the following steps to gain a better understanding of the 5810 functions available. Note that the following procedure requires that you specify whether you are using the Plotter or the Terminal via the 5821 command.

1. To create some data to be plotted, enter (via the Keyboard) the following Keyboard commands:

   BLOCK SIZE 512 ENTER
   CLEAR ENTER
   KEYBOARD 0 SPACE 0 SPACE 50 ENTER
   KEYBOARD 5 SPACE 0 SPACE 14 ENTER
   1 ENTER
   F ENTER

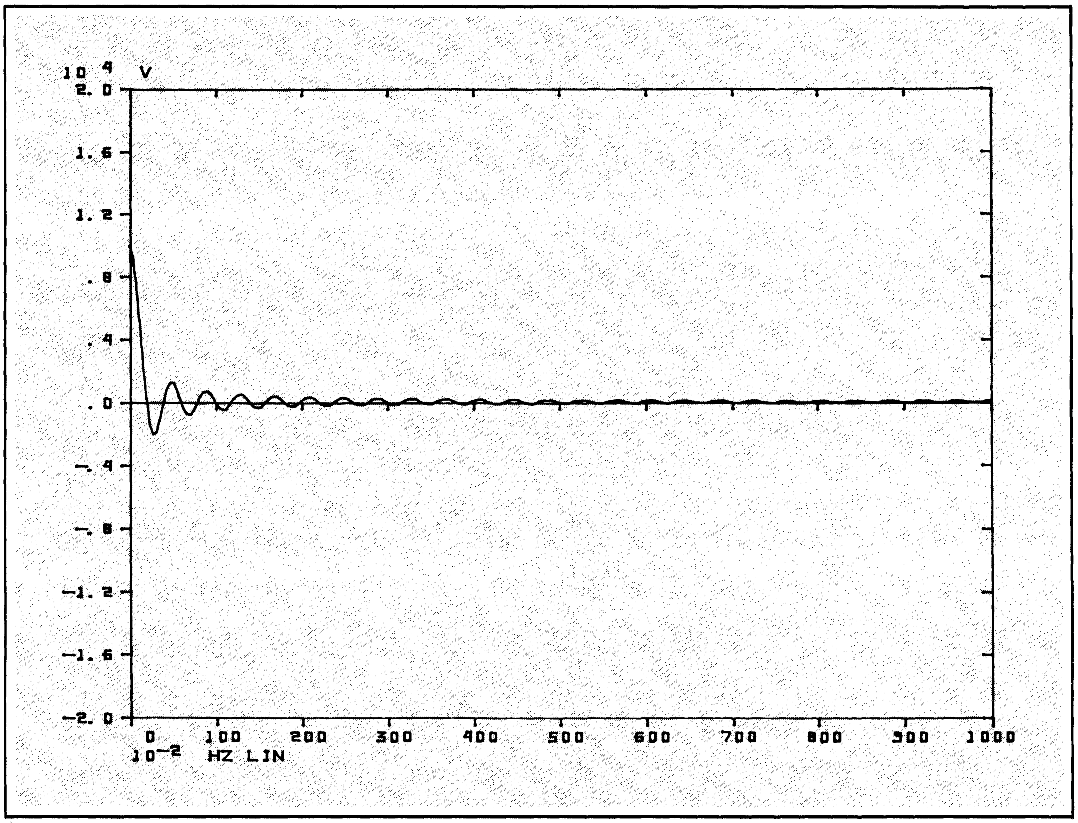
2. To make a standard plot of the displayed data for comparison purposes, enter the following command:

   USER PROG 5800 SPACE n1 ENTER

   Where n1 = 6 (Terminal) or 10 (Plotter)

   Figure 6-12 is an example of the plot that should be produced.

Figure 6-12. Setting Horizontal Range

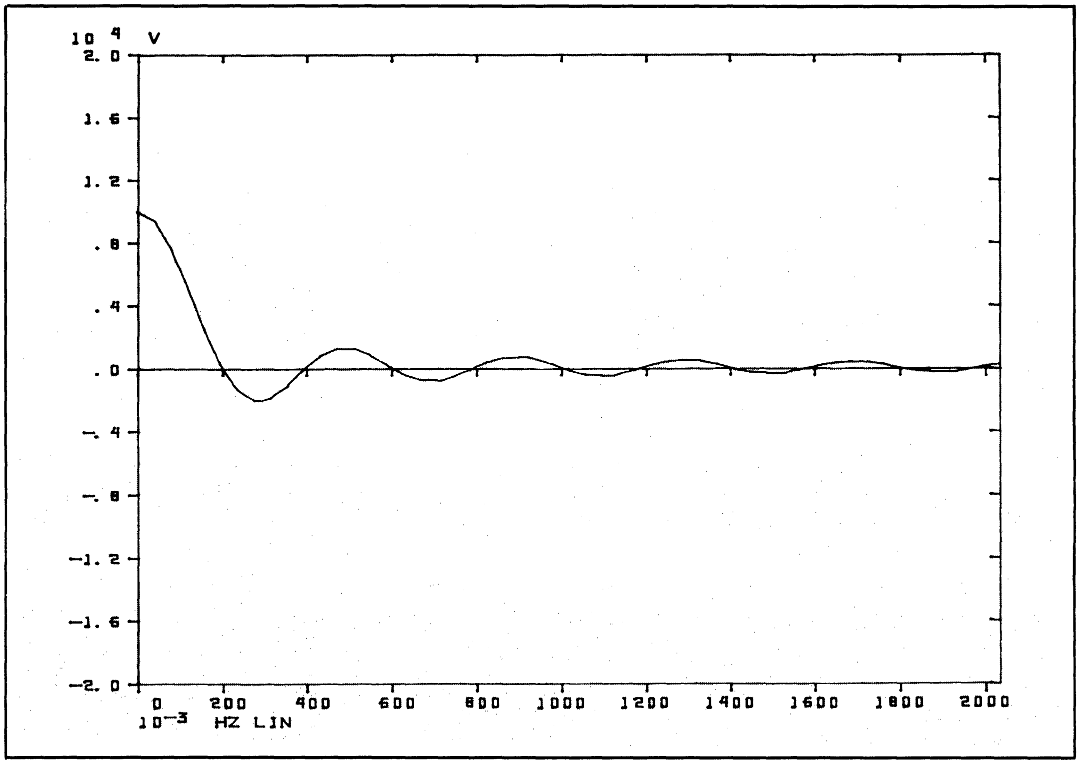


3. To make a plot of the data from dc to 2 Hz, enter the following commands:

   USER PROG 5810 SPACE 2 SPACE 0 ENTER
   USER PROG 5813 SPACE 1 ENTER
   USER PROG 5814 ENTER (used only if Terminal was specified)
   USER PROG 5800 SPACE –1 ENTER

   Figure 6-13 is an example of the plot that should be made.

Figure 6-13. Changing Horizontal Range



5811 PLOT BOUNDARY ALIGNMENT

User Program 5811 allows you to calibrate the 7210 digital plotter (see User Program 5883 for HP-IB plotters). Through the use of the 5811 parameter you indicate the total plot area, exclusive of the space required for labels. This command is useful for aligning the plot axes origin on preprinted paper.

Command Format

USER PROG 5811 SPACE n1 ENTER

n1 = 0, move pen to left center of plot area.

n1 > 0, move pen to upper right of plot area.

n1 < 0, move pen to lower right of plot area.

n1 = default, move to the next pen position in sequence.

When 5811 is executed with any one of the above parameters, the raised pen is moved immediately to the position indicated, but nothing is plotted.

Error Conditions

None.

Familiarization Procedure

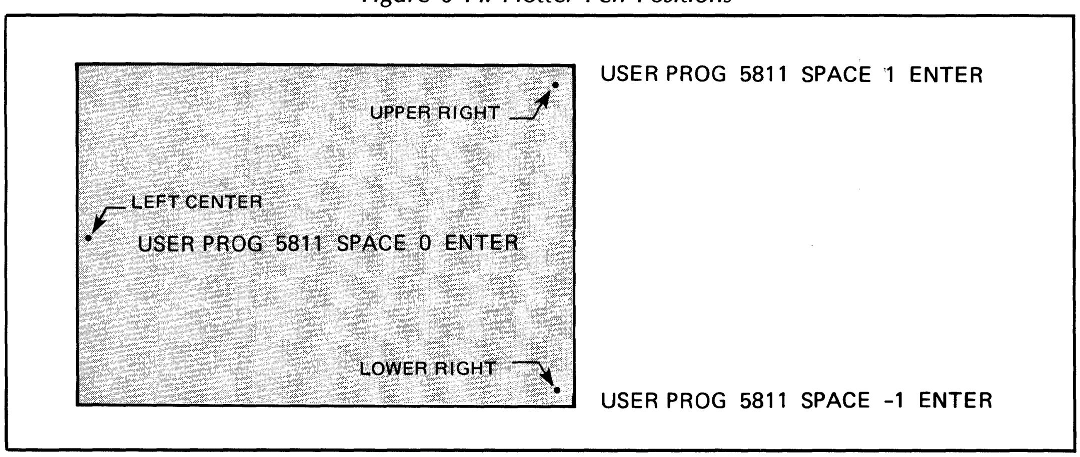
Perform the following to become more familiar with the actions that result from the execution of the 5811 command.

1. Indicate that the Plotter is the plotting device via the following command:

   USER PROG 5821 SPACE 10 ENTER

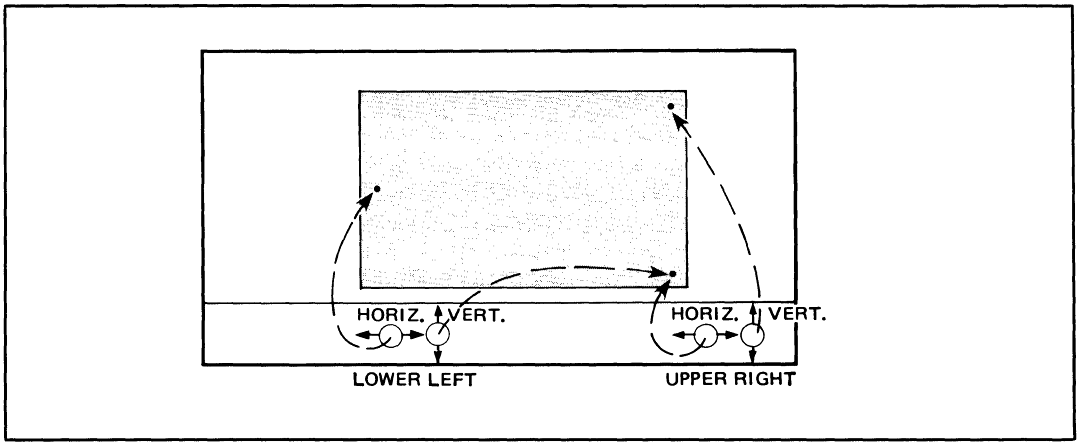
2. Using Figure 6-14 and the following commands, observe the movement of the pen as each command is executed.

Figure 6-14. Plotter Pen Positions



3. Now, using the 5811 commands in combination with the adjustment knobs on the Plotter (see Figure 6-15), calibrate your plotting device using the following commands:

Figure 6-15. Plotter Calibration



USER PROG 5811 SPACE 0 ENTER	(Adjust lower left horizontal control knob to position the pen over the desired vertical grid line)
USER PROG 5811 SPACE -1 ENTER	(Adjust lower left vertical control knob to position the pen over desired horizontal grid line; then adjust the upper right horizontal control knob to position the pen over the desirired right hand vertical grid line)
USER PROG 5811 SPACE 1 ENTER	(Adjust upper right vertical control knob to position pen over the desired upper horizontal grid line)

Repeat the above sequence to assure that the pen positions are accurate. If not, repeat the above sequence for final adjustment.

4. At this point you can enter some data and plot it, if you so desire.

NOTE

When plotting log horizontal data in the F_{max} mode on preprinted paper, it may be necessary to adjust the left horizontal knob to place the pen slightly to the left of the last desired printed line (on the preprinted paper) to compensate for the non-integer value of Δf.

5812 DRAW VECTOR

User Program 5812 draws a vector from one X,Y point on the plot to another. This is useful in annotating plots and underlining text.

Command Format

USER PROG 5812 SPACE n1 SPACE n3 SPACE n4 ENTER

n1 = X_2

n2 = Y_2, default is last Y_2

n3 = X_1, default is last X_1

n4 = Y_1, default is last Y_1

The range for X_1, X_2, Y_1, and Y_2 is specified in terms of "device units". For the 7210 plotter, the range for X and Y is 0 to 9999; for the 9872 plotter, the range for X is 0 to 16000 and Y is 0 to 11400. For the Terminal, the range for X is 0 to 1023; for Y it is 0 to 777.

NOTE

Once 5812 has been executed with a certain vector line specified, that same line will be drawn in subsequent calls to 5812 unless a different line is specified.

Error Conditions

A3 WHAT?	No parameters were passed.
A4 WHAT?	Vector line is outside of device limits.
	Please note that n1 must always be entered whenever 5812 is executed.

Familiarization Procedure

Perform the following steps to become more familiar with the actions that result from the execution of the 5812 command. If you are using the Terminal as your plot device, use procedure number 1. If you are using the Plotter as the plot device, use procedure number 2.

PROCEDURE 1

1. Create text to be underlined with the following command.

   USER PROG 5821 SPACE 6 ENTER (indicates the Terminal is to be used.
   USER PROG 5803 SPACE 0 SPACE 1 ENTER

   Enter the following line of text:

   SINE X/X FUNCTION DEMONSTRATION PLOT CR

   The data is now stored on Disc in record 0 of the Data Block File and in the core-resident text file with the text message ID of 01.

2. Write a Keyboard Program to position and print the above text message on the screen, underlined.

   RPLAC ENTER
   LABEL 0 ENTER
   USER PROG 5814 ENTER
   USER PROG 5808 ENTER
   USER PROG 5819 SPACE 1 ENTER
   USER PROG 5812 SPACE 500 SPACE 760 SPACE 0 SPACE 760 ENTER
   END ENTER
   TERM ENTER

3. Execute the Keyboard Program by entering JUMP 0 ENTER.

   The text message 01, entered in step 1, should be output to the Terminal and underlined.

NOTE

The range for X is 0 to 1023; for Y it is from 0 to 777 on the Terminal.

PROCEDURE 2

1. Create text to be underlined with the following commands:

   USER PROG 5821 SPACE 10 ENTER (indicates the Plotter is to be used)
   USER PROG 5803 SPACE 0 SPACE 01 ENTER

   Enter the following line of text:

   SINE X/X FUNCTION DEMONSTRATION PLOT CR

   The data is now stored on Disc in record 0 of the Data Block File and in the core-resident text file with the text message ID of 01.

2. Enter the following Keyboard Program to position and write the above text message on the screen:

   RPLAC ENTER
   LABEL 0 ENTER
   USER PROG 5808 ENTER
   USER PROG 5819 SPACE 1 ENTER
   END ENTER
   TERM ENTER

3. Execute the Keyboard Program by entering:

   JUMP 0 ENTER

4. Underline the text using the following command:

   USER PROG 5812 SPACE 4850 SPACE 9800 SPACE 0 SPACE 9800 ENTER

NOTE

The range for X and Y is 0 to 9999 on the Plotter.

5813 ECHO CONTROL

User Program 5813 enables or suppresses the echo in read operations on the Terminal. This suppression prevents entries from appearing on a plot that is being displayed on the Terminal screen.

NOTE

> If 5813 has been used to suppress the display of entries, restarting the system will enable the echo again. If the plotting device is not the Terminal, the execution of this command has no effect.

Command Format

USER PROG 5813 SPACE n1 ENTER

n1 = 0 or default, entries are displayed

n1 $\neq$ 0, entries are not displayed (i.e., echo is suppressed)

Error Conditions

None.

Familiarization Procedure

Perform the following steps to become more familiar with the actions that result from the execution of the 5813 command.

1. Enter the following commands:

   USER PROG 5821 SPACE 6 ENTER
   USER PROG 5803 SPACE 0 SPACE 1 ENTER

   Enter the following line of text:

   SINE X/X FUNCTION CR

   This line of text, via the following steps, will appear at various places on the Terminal screen without the appearance of the intervening commands that are necessary to place them.

2. Enter the 5813 command to suppress echo.

   USER PROG 5813 SPACE 1 ENTER

3. Enter the 5814 command to erase the screen (USER PROG 5814 ENTER). Notice that this command does not appear on the screen and that the screen is erased.

4. Enter the following commands to place the line of text in the upper left corner of the screen:

   USER PROG 5808 ENTER
   USER PROG 5819 SPACE 1 ENTER

   The line of text should be written on the Terminal screen in the upper left corner.

5. Now, place the line of text in the lower left corner.

   USER PROG 5808 SPACE 1 SPACE 0 ENTER
   USER PROG 5819 SPACE 1 ENTER

   The line of text should now be placed in the lower left corner of the screen.

6. Enter the 5813 command again to enable the echo.

   USER PROG 5813 ENTER

7. Place the line of text in the upper right corner of the screen.

   USER PROG 5808 SPACE 900 SPACE 550 ENTER
   USER PROG 5819 SPACE 1 ENTER

   Notice that the text is placed in the upper right of the screen and that the above commands were also output to the screen.

5814 ERASE SCREEN

User Program 5814 erases the Terminal screen and leaves the graphics cursor in the home position (if it is on). If the Terminal is not being used as the plotting device, 5814 will have no effect when it is executed.

This command is useful primarily in a Keyboard Program, as the SHIFT CLEAR keys on the Terminal accomplish the same function.

Command Format

USER PROG 5814 ENTER

Error Conditions

None.

Familiarization Procedure

Perform the following steps to become more familiar with the actions that result from the execution of the 5814 command:

1. Enter the following commands using the Keyboard:

   BLOCK SIZE 512 ENTER
   CLEAR ENTER

2. Erase the screen by entering:

   USER PROG 5814 ENTER

   The system should blank out the screen and, if the graphics cursor is on, it should return the cursor to the home position in the upper left corner of the screen.

3. Enter the following commands using the Keyboard:

   USER PROG 5800 SPACE 6 ENTER

   The system should plot the straight line being displayed on the Display Unit screen.

4. Erase the screen by entering:

   USER PROG 5814 ENTER

   The system should again blank out the screen and return the cursor (if it is on) to the home position in the upper left corner of the screen.

5815 PLOT DISPLAY SCREEN

When 5815 is executed, a plot of the data being displayed is executed. Note that 5816 draws the axes and 5817 appropriately labels the data being displayed.

The following exceptions apply to the 5815 command:

1. User Program was previously executed specifying different mode, origin, or scale than the data being displayed. You must execute 5809 again to re-specify the appropriate mode, origin, and scale, otherwise the system will plot the previous specifications.

2. *If a request for a plot which is not within the bound of 5810 or 5829 is given, then no plotting occurs, nor does the system output an error.*

Command Format

USER PROG 5815 SPACE n1 SPACE n2 SPACE n3 ENTER

n1 = core-resident data block number (data plotted from specified data block)

n1 = default or zero, data plotted from data block 0

n2 = first channel to be plotted

n3 = last channel to be plotted.

If n2 is specified, then n3 must also be specified. Of the data to be plotted, delineated by channel numbers n2 and n3, only that portion of the data that lies within the range previously specified by 5810 or 5829 will be plotted. Note that the axis (5816) and labels (5817) are drawn for the entire block whether n2 and n3 are specified or not.

A default of n2 and n3 results in a plot whose data block end points are bounded by 5810 or 5829, or from 0 (dc) to F_{max} (if in the frequency domain), or from 0 to T (if in the time domain), whichever is the more restrictive.

Error Conditions

B6 WHAT?	An illegal frequency code is being used. Correct the code and repeat the command. Often times a data block may be inadvertently specified that does not contain valid data, especially if data is being keyed in manually on the Keyboard.
C2 WHAT?	All three parameters or just n1 must be present in the command. Just two or more than three are not accepted. Repeat the command to recover.
C7 WHAT?	Invalid core-resident data block number. Re-specify the current data block number.
C8 WHAT?	Frequency domain, origin center plot attempted. This plot is not allowed. Repeat the command after you have re-specified the origin with the 5809 command.

Familiarization Procedure

Use the following procedure to become more familiar with the 5815 command.

1. Initialize the system to plot on either the Plotter or the Terminal by entering the following:

   USER PROG 5821 SPACE n1 ENTER

2. Enter the following data via the Keyboard:

   BLOCK SIZE 512 ENTER
   CLEAR ENTER
   KEYBOARD 0 SPACE 0 SPACE 5 ENTER
   KEYBOARD 5 SPACE 0 SPACE 9 ENTER
   1 ENTER
   F ENTER

3. Initialize the mode, origin, and scale of the data by entering the following command, after setting the Display Unit switches properly.

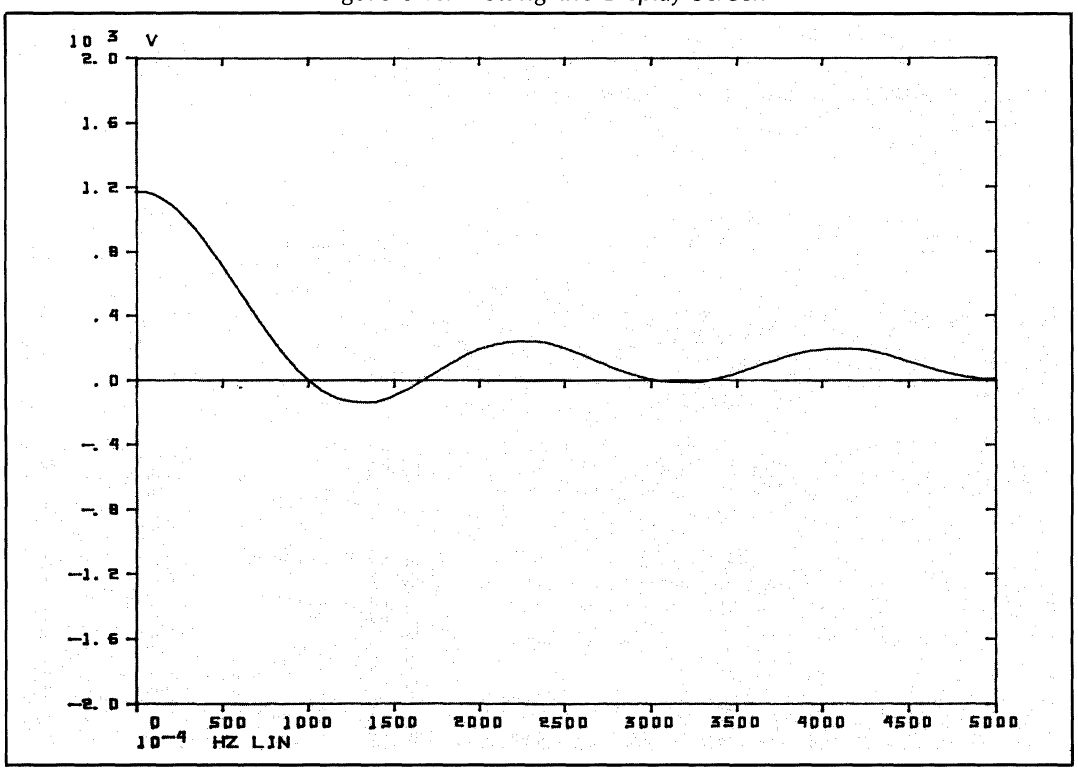
   USER PROG 5809 ENTER

4. Execute the following commands to obtain a fully annotated plot:

   USER PROG 5813 SPACE 1 ENTER
   USER PROG 5814 ENTER
   USER PROG 5815 ENTER
   USER PROG 5816 ENTER
   USER PROG 5817 ENTER

   Figure 6-16 is an example of the final plot.

Figure 6-16. Plotting the Display Screen

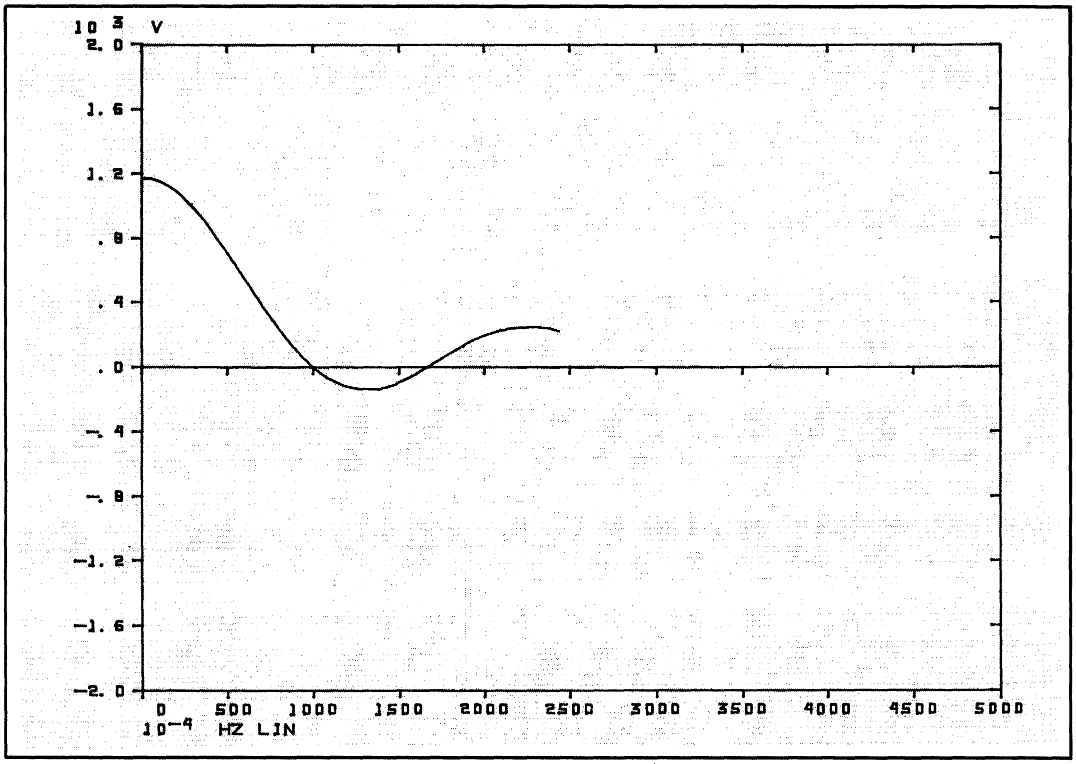


5. Execute the following commands to obtain a fully annotated plot that shows data from channels 1 to 125.

   USER PROG 5815 SPACE 0 SPACE 1 SPACE 125 ENTER
   USER PROG 5816 ENTER
   USER PROG 5817 ENTER

   Figure 6-17 is an example of the final plot. Notice that the axes and the label annotations still reflect the full data block.

6-17. Plotting Channels 1 to 125

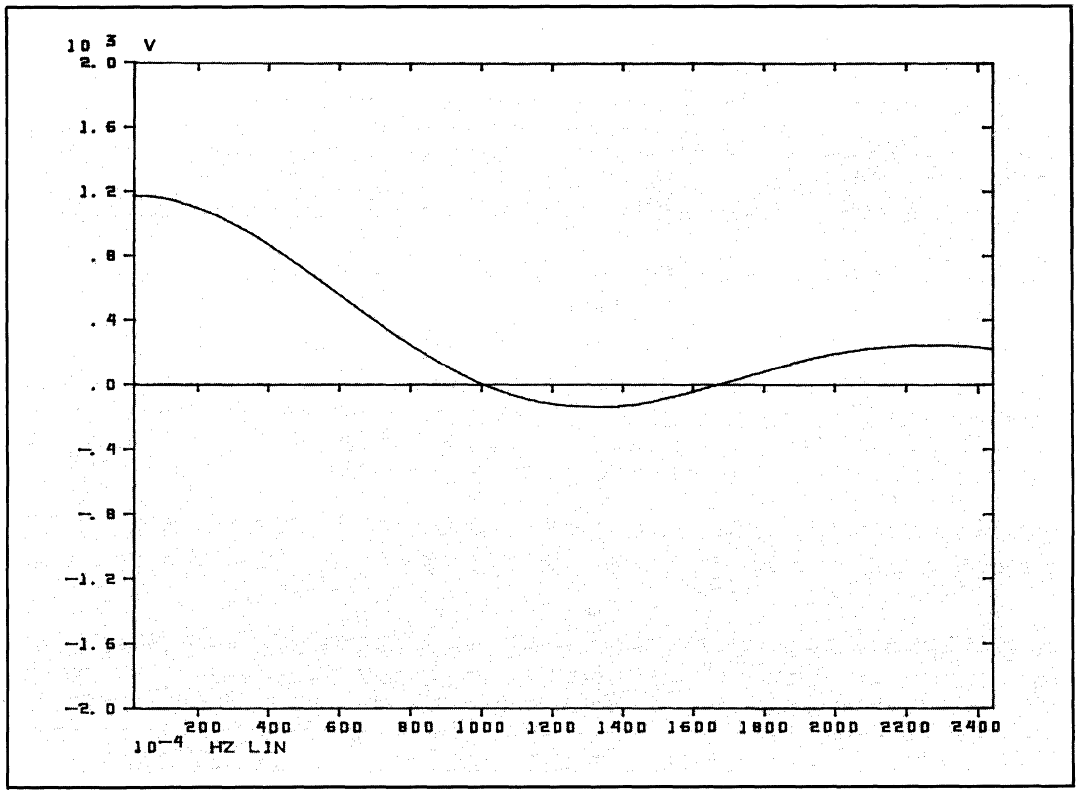


6. Now execute the following commands to fully align the axes and label annotation information with data being plotted.

   USER PROG 5814 ENTER
   USER PROG 5829 SPACE 0 SPACE 1 SPACE 125 ENTER
   USER PROG 5815 ENTER
   USER PROG 5816 ENTER
   USER PROG 5817 ENTER

   Figure 6-18 is an example of the final plot.

Figure 6-18. Aligning Channels 1 to 125



5816 DRAW AXIS

User Program 5816 draws the axes. The axis size is determined by the most recent calls to 5804, 5805, 5806 and 5807. The type of axis is determined by the data type unless it is overridden by the n1 parameter. Note that User Program 5809 must be called to specify the mode, origin, and scale which also influence the axis type to be used.

NOTE

5816 must be called prior to calling 5817.

Command Format

USER PROG 5816 SPACE n1 ENTER

where:

n1 = 1 ├──

= 2 ─┼─

= 3 ┌──

= 4 ─┬─

= 5 └──

= 6 ─┴─

If n1 is defaulted, the axis type is determined as follows:

1. For frequency or time rectangular data:

   ├──

2. For complex data:

   ─┼─

3. For log vertical data:

   ┌──

4. For complex or origin center, negative Y-half plane
   (using USER PROG 5864 SPACE -1 ENTER):

   ─┬─

5. For single precision* power spectra
   (Using USER PROG 5864 SPACE 1 ENTER) or for polar data:

   └──

6. For complex or origin center, positive Y-half plane
   (using USER PROG 5864 SPACE 1 ENTER):

   ─┴─

Refer to USER PROG 5864 for more information concerning Y-half plane plotting.

Error Conditions

A0 WHAT? Either n1 is <1 or >6. Repeat the command.

**Double precision power spectra can be drawn by specifying n1 = 5 also.*

Familiarization Procedure

Use the following procedure to become more familiar with the 5816 command.

1. Acquire an analog pulse using the procedure in Section 2.
2. Next, transform the data to polar coordinates by entering the following keyboard commands:

   F ENTER
   POLAR ENTER
3. Ensure the system is set up properly by entering the following commands:

   USER PROG 5809 ENTER
   USER PROG 5810 ENTER
   USER PROG 5829 ENTER
4. If the Terminal is being used as your plotting device, clear the screen by entering:

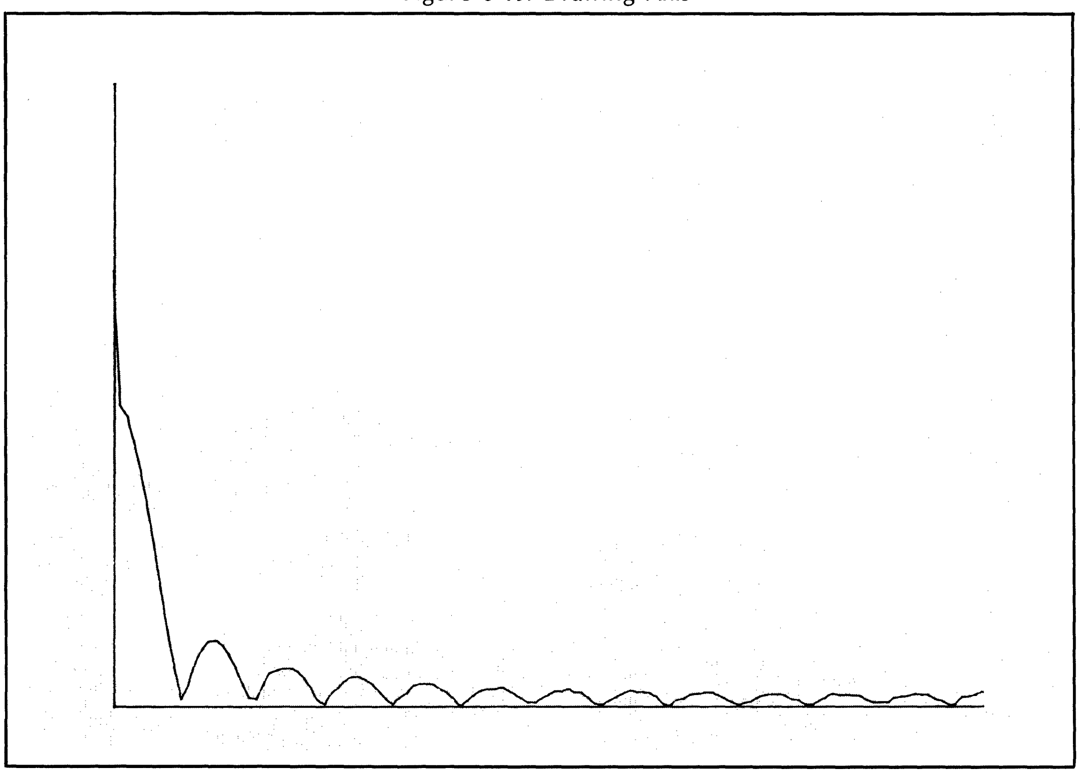
   USER PROG 5814 ENTER
5. Plot the data and draw the axis via the following commands:

   USER PROG 5815 ENTER
   USER PROG 5816 ENTER

   Notice that the axis type number 5 is automatically chosen by the system.

   Figure 6-19 is an example of the plot that should result.

Figure 6-19. Drawing Axis



5817 ANNOTATE PLOT AXES

User Program 5817 annotates the vertical and horizontal divisions of the axes and writes the scale factors on the horizontal and vertical axes. Tic marks are also provided for the vertical and horizontal axes depending upon the size of the plot which is set by a call to User Program 5804.

NOTE

5817 must be preceded by a call to 5816.

Command Format

USER PROG 5817 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

n1 = Text message ID for vertical axis units where the text message was created using 5803.

n2 = Text message ID for horizontal axis units where the text message was created using 5803.

If n1 and n2 are 0 or defaulted, the system uses the standard default units (refer to the familiarization procedure in this discussion for additional information on default unit designations).

n3 = <0, do not label vertical axis (only the horizontal axis will be labeled).

n3 = 0 or default, label both axes.

n4 = 0 or default, the system will automatically use one of the following label types, depending upon the data being displayed and the switch settings on the Display Unit:

Default Units are:

Vertical: DG (degrees), DB (decibels), V (volts), V SQR (volts squared)

Horizontal Frequency: HZ LIN (Hertz linear), HZ LOG (Hertz log)

Horizontal Time: SEC (seconds)

n4 = 1, the vertical units may be DEG, DB RE (relative to) G (acceleration units), G SQR/HZ, or DB RE MSG #n1, depending upon the data being displayed, switch settings, and the most recent call to 5865

NOTE

User Program 5865 allows you to select the vertical range for log vertical plots and whether that range is to specify dB or decades. (Refer to 5865 description for more information.)

n4 = 2, the horizontal units are SEC (applicable only in Signature Analysis operations Option 450).

n4 = 3, the horizontal units are RPM (applicable only in Signature Analysis operations Option 450).

The following table summarizes labeling when 5817 is called. The familiarization procedures for 5817 demonstrate how these labels are obtained.

VERTICAL UNITS		
	Single Precision	**Double Precision**
LINEAR	V, G	V SQR, or G SQR/HZ
LOG dB (5865 n1, where n1 = 10, 20 etc.)	DB*, DB RE G	DB*, DB RE G, G SQR/HZ
LOG DECADES (5865 n1, where n1 = 1, 2,...8)	V, G	V SQR, G SQR/HZ
PHASE	DEG	DEG
HORIZONTAL UNITS		
COMPLEX	Same as any of the Vertical units	
FREQUENCY	HZ LINEAR or HZ LOG (depending upon ORIGIN setting)	
TIME	SEC	

**DB is all that is output. The inference, however, is DB relative to volts.*

Error Conditions

C5 WHAT? n1 or n2 < 0, n4 > 3. Repeat the command.

Familiarization Procedure

Use the following procedure to become more familiar with the 5817 command.

NOTE

User Program 5865 description should be read to facilitate your understanding, as its parameters affect the function of 5817.

1. Create some single precision data so that V (volts) is the default vertical unit.

   BLOCK SIZE 256 ENTER
   CLEAR ENTER
   KEYBOARD 0 SPACE 5 SPACE 5 ENTER
   KEYBOARD -4 SPACE 0 SPACE 43 ENTER
   10000 ENTER
   F ENTER

2, Save this data on the Disc via the following commands:

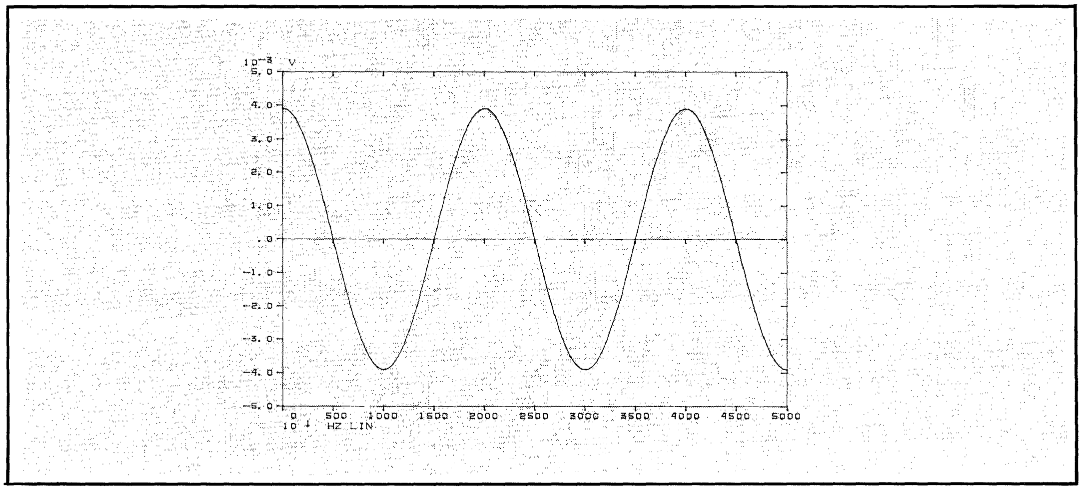
   MASS STORE 31 SPACE 1 ENTER
   MASS STORE 21 ENTER

3. Plot the data using the following command:

   USER PROG 5800 SPACE n1 ENTER (n1 = 6, 10, or 35) — Plots a full page

Figure 6-20 is an example of the plot that should be produced.

Figure 6-20. Example 1 of Annotating Plot Axis

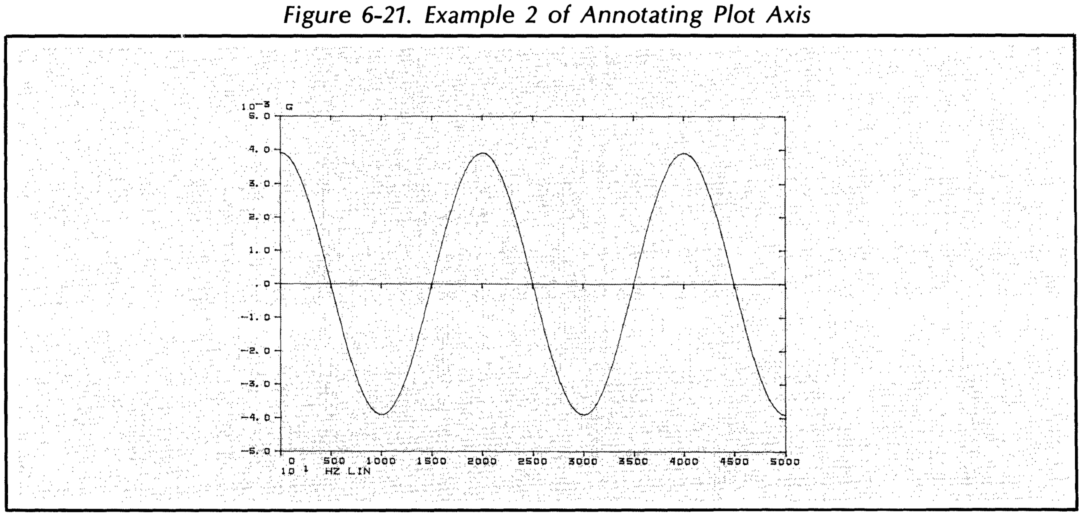


4. Entering the following commands will now produce a plot with Gs as the vertical axis annotation. This is always possible when you are working with single precision linear data.

   USER PROG 5813 SPACE 1 ENTER
   USER PROG 5814 ENTER
   USER PROG 5815 ENTER
   USER PROG 5816 ENTER
   USER PROG 5817 SPACE 0 SPACE 0 SPACE 0 SPACE 1 ENTER

   Figure 6-21 is an example of the plot that should result.

Figure 6-21. Example 2 of Annotating Plot Axis

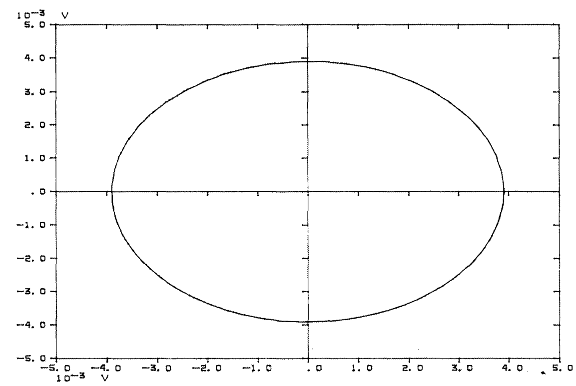


5. Now, using the same data, turn the MODE knob on the Display Unit to the COMPLEX position and execute the following commands.

   USER PROG 5809 ENTER
   USER PROG 5814 ENTER
   USER PROG 5800 SPACE -1 ENTER

Notice that both the horizontal and vertical axes are annotated with V (volts).

Figure 6-22. Example 3 of Annotating Plot Axis



6. Return the MODE knob to REAL/MAGNITUDE position and execute the following command to reset the MODE switch information for the next execution of 5800 and to facilitate the next setup.

   USER PROG 5809 ENTER

7. Enter the following command to create log magnitude data.

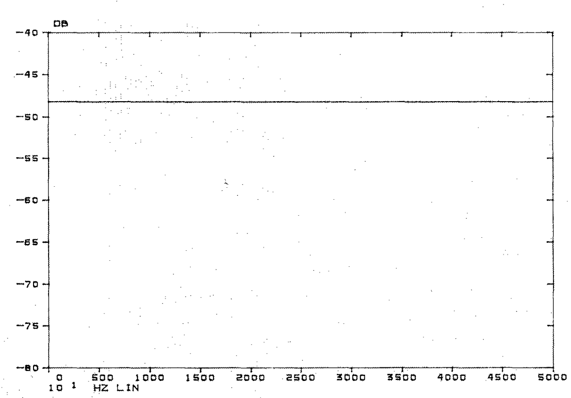
   LOG MAG ENTER

8. Enter the following commands to plot data in dB.

   USER PROG 5814 ENTER
   USER PROG 5865 SPACE 40 ENTER
   USER PROG 5800 SPACE −1 ENTER

   The following plot example should result.

Figure 6-23. Example 4 of Annotating Plot Axis



9. To see additional examples of annotating the plot axis, continue to change the MODE knob and plot additional figures.

5818 SPECIFY GRID LINES TO BE DRAWN

User Program 5818 is used to specify whether or not grid lines are to be drawn on the plot. Also, you can specify whether all the grid lines are to be drawn or whether just the major grid lines are to be drawn. *The grid lines are actually drawn when User Program 5817 is executed. Therefore, in order to have grid lines included in your plot, they must be specified in advance of executing User Program 5817.*

Command Format

USER PROG 5818 SPACE n1 SPACE n2 ENTER

n1 = 0 or default, no grid lines are drawn

n1 = 1, draw grid lines of the type specified by n2

n2 = 0 or default, all grid lines are drawn

n2 = 1, only major grid lines are drawn

When all grid lines are specified the log decade major and minor tic marks are drawn as grid lines.

When major grid lines are specified the log decade major and linear tic marks are drawn as grid lines.

Error Conditions

None.

Familiarization Procedure

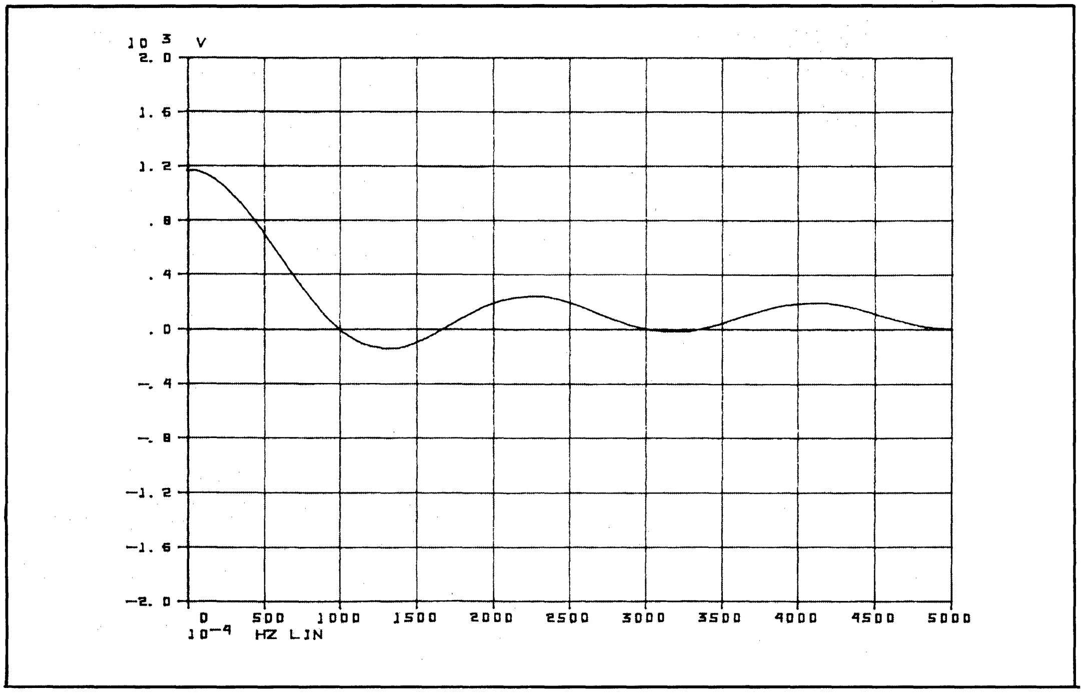
Use the following procedure to become more familiar with the 5818 command.

1. To obtain a plot of non-log data with all grid lines, enter the following commands.

   BLOCK SIZE 512 ENTER
   CLEAR ENTER
   KEYBOARD 0 SPACE 0 SPACE 5 ENTER
   KEYBOARD 5 SPACE 0 SPACE 9 ENTER
   1 ENTER
   F ENTER
   USER PROG 5813 SPACE 1 ENTER
   USER PROG 5814 ENTER
   USER PROG 5818 SPACE 1 ENTER
   USER PROG 5821 SPACE n1 ENTER (where n1 is 6, 10, or 35)

   Figure 6-24 is an example of the plot that should result.

Figure 6-24. Non-Log Plot with All Grid Lines Drawn

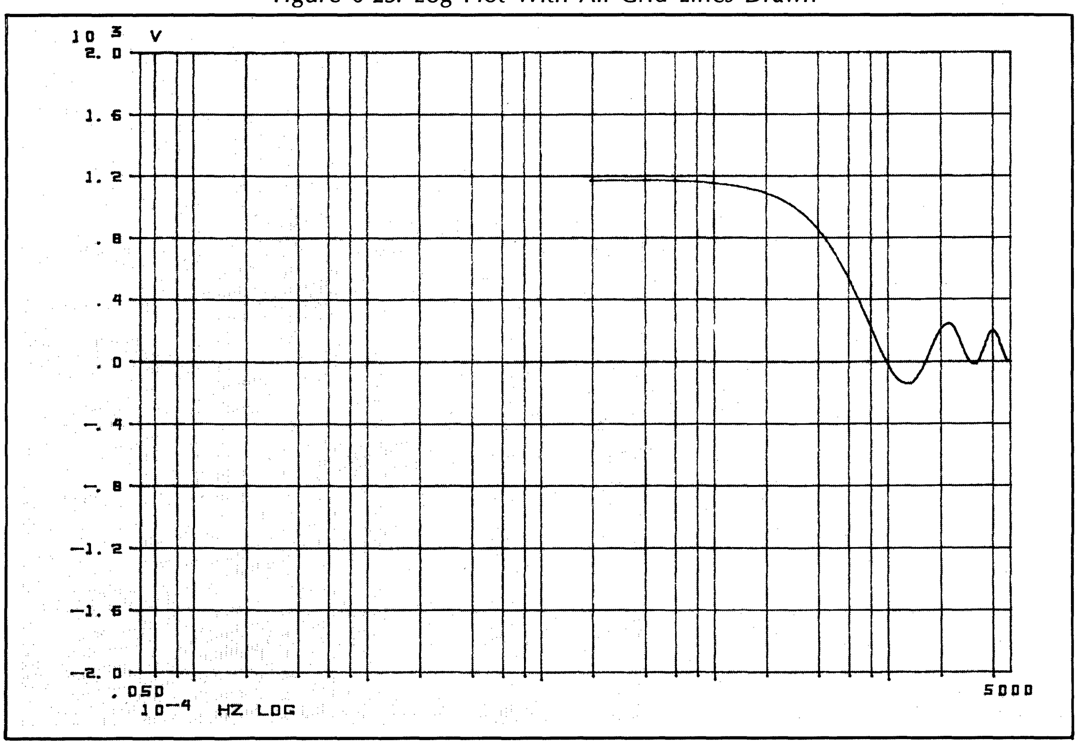


2. Enter the following commands to obtain a plot with all grid lines for log data.

   USER PROG 5809 SPACE 0 SPACE -5 SPACE 0 ENTER
   USER PROG 5818 SPACE 1 ENTER
   USER PROG 5814 ENTER
   USER PROG 5800 SPACE -1 ENTER

   Figure 6-25 is an example of the plot that should result. Note that .050 and 5000 annotate the end vertical lines.

Figure 6-25. Log Plot With All Grid Lines Drawn

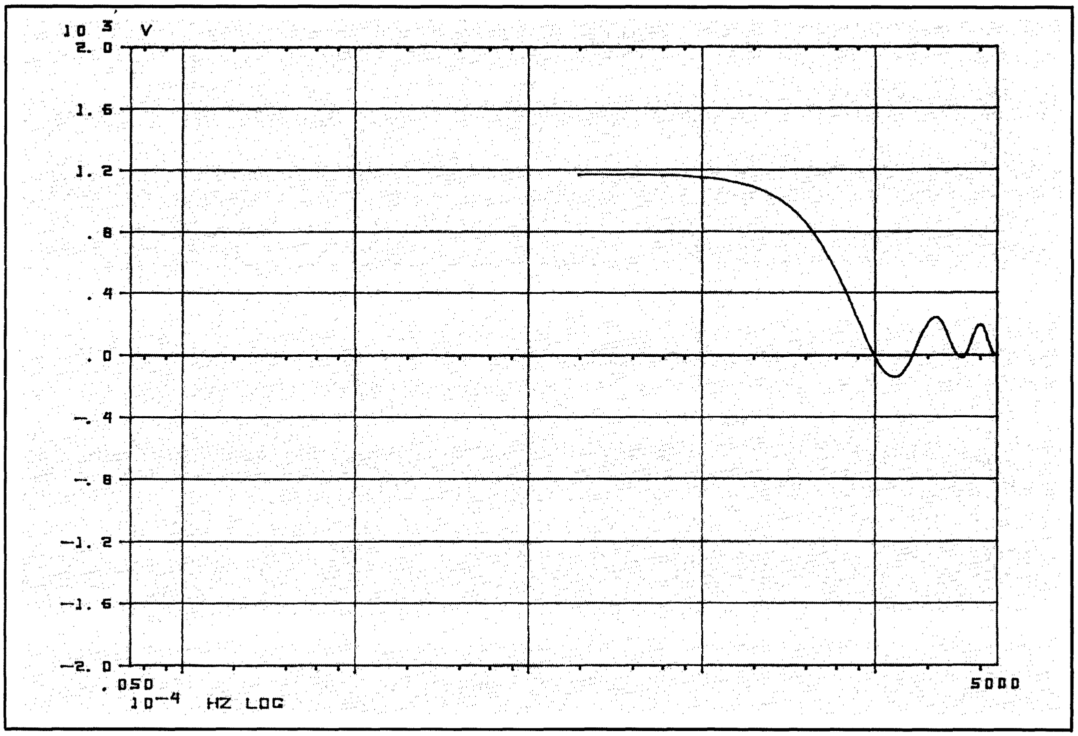


3. Now plot log data with only major grid lines. Use the following commands.

   USER PROG 5818 SPACE 1 SPACE 1 ENTER
   USER PROG 5814 ENTER
   USER PROG 5800 SPACE -1 ENTER

   Figure 6-26 is an example of the plot that should result. Note that .050 and 5000 annotate the end vertical lines.

Figure 6-26. Log Plot With Only Major Grid Lines Drawn



5819 WRITE A LINE OF TEXT

User Program 5819 is used to write a line or lines of text on the Terminal screen or on a paper on the Plotter. The position of the text is determined by executing User Program 5808. The text must have been previously created via the use of 5803 or it can be created on line with this command.

Text that is previously created must be in the core-resident text buffer. You can do this by executing 5803 previous to this command or you can have already stored text on Disc and it can be recalled via the use of User Program 5838 (see description in this section).

Command Format

USER PROG 5819 SPACE n1 SPACE n2 ENTER

n1 = message ID number (must be core-resident).

n1 = 0 or default, the same message as the previous call is written.

n1 < 0, write on-line text (see note).

n2 = 1, write the text to the system terminal (see note).

NOTE

When n1 is <0, the system allows you to type text on the system terminal. After each character is typed, the cursor (Terminal) or pen (Plotter) is moved. Once a CRLF is entered, the system writes the line on the plotting device. On the Graphics Terminal, more than one line of text can be entered and written as long as a slash (/) is not entered as the first character of a line (i.e., the slash terminates the on-line text entry).

The n2 parameter allows you to output text (whether in the core-resident text buffer or entered on-line) to the system terminal. Also, if button 4 is set on the DISPLAY REGISTER of the Processor, the text will be punched on paper tape. If button 6 is set, the text is printed on an optional line printer. The n2 parameter is useful in Keyboard Programs to output messages to the system operator.

Error Conditions

B4 WHAT?	n1 is greater than 99. Message ID ranges from 01 to 99. Repeat the command with the correct message ID.
B5 WHAT?	Message number doesn't exist in the core-resident buffer. Verify that the message you want does exist by executing USER PROG 5803 and then using the LIST command to see what is present.

Familiarization Procedure

Use the following procedure to become more familiar with the 5819 command.

1. Create some text using the following commands.

   USER PROG 5803 SPACE 1 ENTER (text will be stored in text buffer 1 when you are through)
   INSRT ENTER
   01 CR
   THE MESSAGE IS PLACED CR
   IN THE UPPER LEFT CORNER CR
   /* CR
   02 CR
   THE MESSAGE IS NOW PLACED CR
   IN THE LOWER LEFT QUADRANT CR
   03 CR
   THE MESSAGE IS PLACED CR
   IN THE UPPER RIGHT QUADRANT
   /* CR
   TERM ENTER

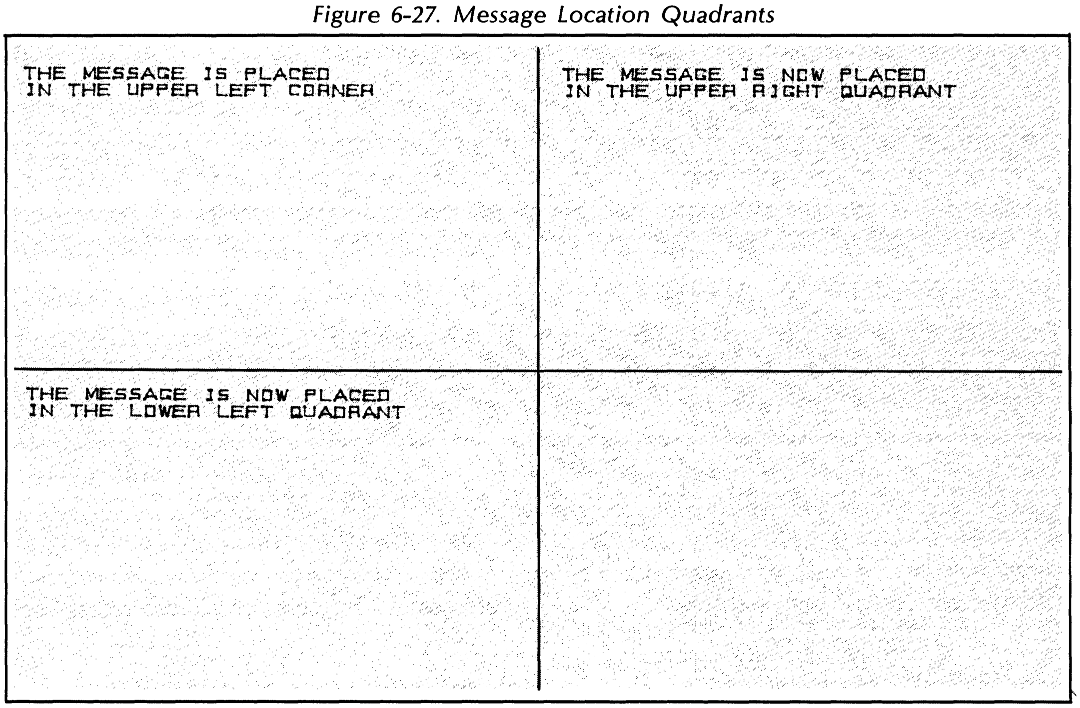
   The text is now stored on disc text buffer 1 and is also in the core-resident text buffer.

2. Next imagine the screen or the page on the Plotter is divided into quandrants. The following commands place the message 01 in the upper left quadrant, message 2 in the lower left quadrant, and message 03 in the upper right quadrant.

   USER PROG 5800 SPACE n1 ENTER (where n1 is 6, 10, or 35)
   USER PROG 5813 SPACE 1 ENTER
   USER PROG 5814 ENTER
   USER PROG 5808 ENTER
   USER PROG 5819 SPACE 1 ENTER (upper left quadrant)
   USER PROG 5808 SPACE 500 ENTER
   USER PROG 5819 SPACE 2 ENTER (lower left quadrant)
   USER PROG 5808 SPACE 980 SPACE 500 ENTER
   USER PROG 5819 SPACE 3 ENTER (upper right quadrant)

   Figure 6-27 is an example of the result of executing the above commands

Figure 6-27. Message Location Quadrants



3. Now, write some on-line text by using the 5819 command as follows.

   USER PROG 5808 SPACE 500 SPACE 500 ENTER (places the text about to be
   USER PROG 5819 SPACE -1 SPACE entered in the lower right quadrant)

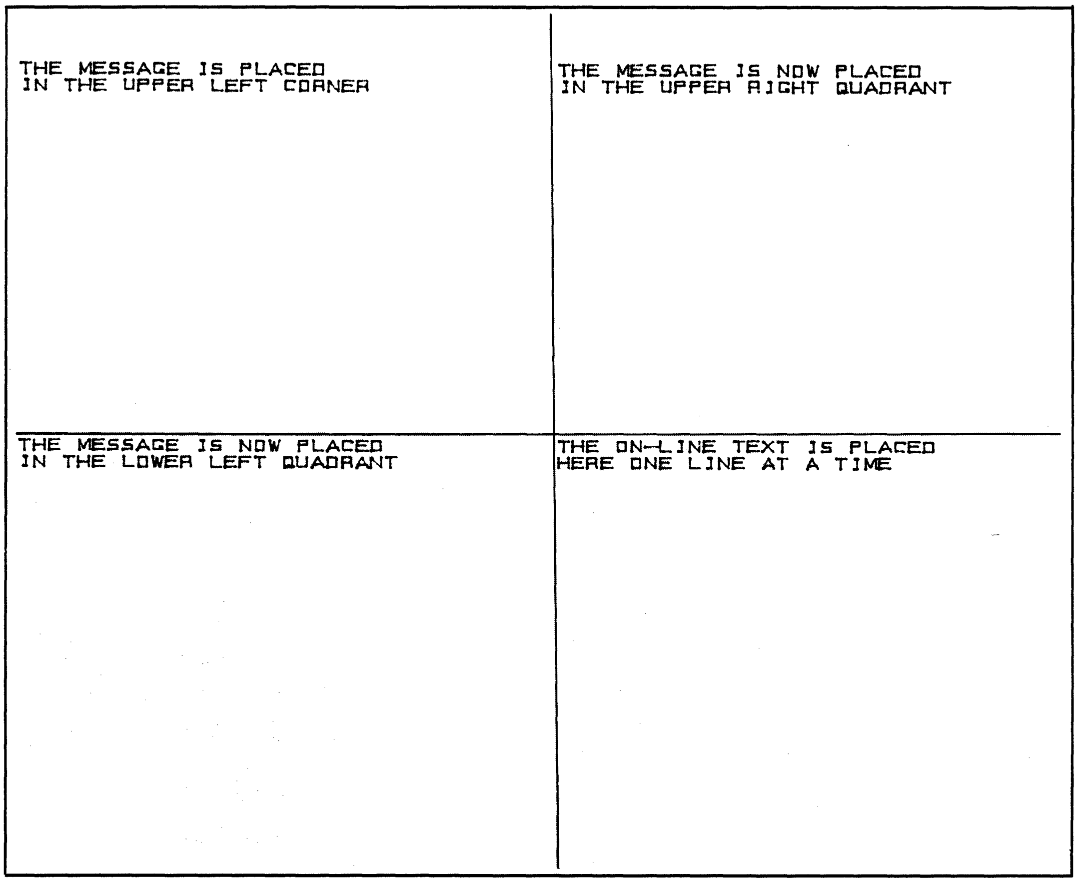
   The system is now waiting for you to enter a line(s) of text. Your text entry will be terminated when you enter the slash (/) as the first character of a line.

   Enter:

   THE ON-LINE TEXT IS PLACED CR
   HERE ONE LINE AT A TIME CR
   / CR

   Figure 6-28 is the final result if you did not change the paper on the Plotter or erase the screen on the Terminal.

Figure 6-28. Adding On-Line Text



5820 MAKE HARD COPY

This command allows you to use an optional hard copy device to make hard copies of the Terminal screen. If the plotting device is not a Terminal, then this command will cause no action to be taken by the system.

Command Format

USER PROG 5820 ENTER

Error Conditions

None.

Familiarization Procedure

Take any one of the previous procedures to create some data to be plotted. Ready the hard copy unit. Execute the above command. The Terminal screen should be copied.

5821 INITIALIZE THE SYSTEM FOR PLOTTING

The execution of User Program 5821 tells the system which plotting device is to be used (i.e., Terminal or Plotter). Once 5821 is executed, subsequent plots are drawn on the device indicated. User Program 5821 also executes the following sequence of commands to reset all the internal plot parameters for the device specified.

Command Format

USER PROG 5821 SPACE n1 SPACE n2 ENTER

n1 = 6 for the Terminal (the cursor on the Terminal should reposition to the upper left-most position of the screen.

n1 = 10 for the 7210 digital plotter (the pen should move to the upper left corner)

n1 = 35 for the 9872 or 7245 plotter (after initialization, pen should move to upper left corner of plotting area as defined by P1 and P2; these scaling points will be set such that a subsequent plot will fit on an 8-1/2 x 11 sheet of paper. Refer to User Program 5883 for changing P1 and P2).

n2 = SUBUNIT NUMBER (if n2 is Defaulted, then the first device specified in the EQT of type n2 is used).

NOTE

Parameter n1 cannot be defaulted.

Error Conditions

A1 WHAT?	Device type n1 is not present. Choose another device and repeat the command.
A2 WHAT?	Unit number n2 is not present. Choose another subunit number and repeat the command.

5829 SET HORIZONTAL RANGE USING CHANNEL NUMBERS

User Program 5829 is used before 5815, 5816, and/or 5817 are called in order to let the system know the horizontal range that is to be plotted.

In the process of making plots, 5815, 5816, or 5817 look at the frequency code and coordinate code (stored with the data being displayed), and the parameters entered using 5829, to determine the appropriate labels and to determine the horizontal range.

Normally 5829 is executed without parameters in order to define the total horizontal range, However, using the n1, n2, and n3 parameters, you can indicate that only a portion of the total horizontal range is to be plotted and then labels are appropriately applied when 5817 is executed.

NOTE

> User Program 5829 allows you to specify the horizontal range in terms of channel numbers (i.e., from some starting channel number to some ending channel number), whereas User Program 5810 allows you to specify the horizontal range in terms of frequency or time units. Refer to the discussion on 5810 for more information.
>
> Note that the horizontal minimum is set to n2 times Δt or Δf, and the horizontal maximum is set to n3 times Δt or Δf. Only channels whose frequency (or time) lies between n2 times Δt (or Δf) and n3 times Δt (or Δf) SEC (or Hz) are plotted.

Command Format

USER PROG 5829 SPACE n1 SPACE n2 SPACE n3 ENTER

n1 = core-resident data block number

n2 = first data channel

n3 = last data channel

If n3 is defaulted, the last channel of the data block is plotted which is equal to TOTAL TIME if the data is in the time domain, or the MAX FREQ divided by 2 if the data is in the frequency domain.

NOTE

> If n2 and n3 are defaulted, the data to be plotted will run from 0 to MAX FREQ or TOTAL TIME.
>
> MAX FREQ = SIZE $\div$ 2 $\times$ Δf (for frequency data)
>
> TOTAL TIME = SIZE $\times$ Δt (for time data)

Error Conditions

A9 WHAT?	An illegal number of parameters was entered. Repeat the command.
B0 WHAT?	n2 and/or n3 is less than 0, or n2 is greater than n3, or illegal block number was entered. Repeat the command with correct parameters.

Familiarization Procedure

1. Enter the following data to become more familiar with the 5829 command.

   BLOCK SIZE 512 ENTER
   CLEAR ENTER
   KEYBOARD 0 SPACE 0 SPACE 50 ENTER
   KEYBOARD 1 SPACE 0 SPACE 39 ENTER
   1 ENTER
   F ENTER
   POLAR ENTER

2. Ensure the system is setup properly by entering the following commands.

   USER PROG 5809 ENTER
   USER PROG 5810 ENTER
   USER PROG 5829 ENTER
   USER PROG 5813 SPACE 1 ENTER

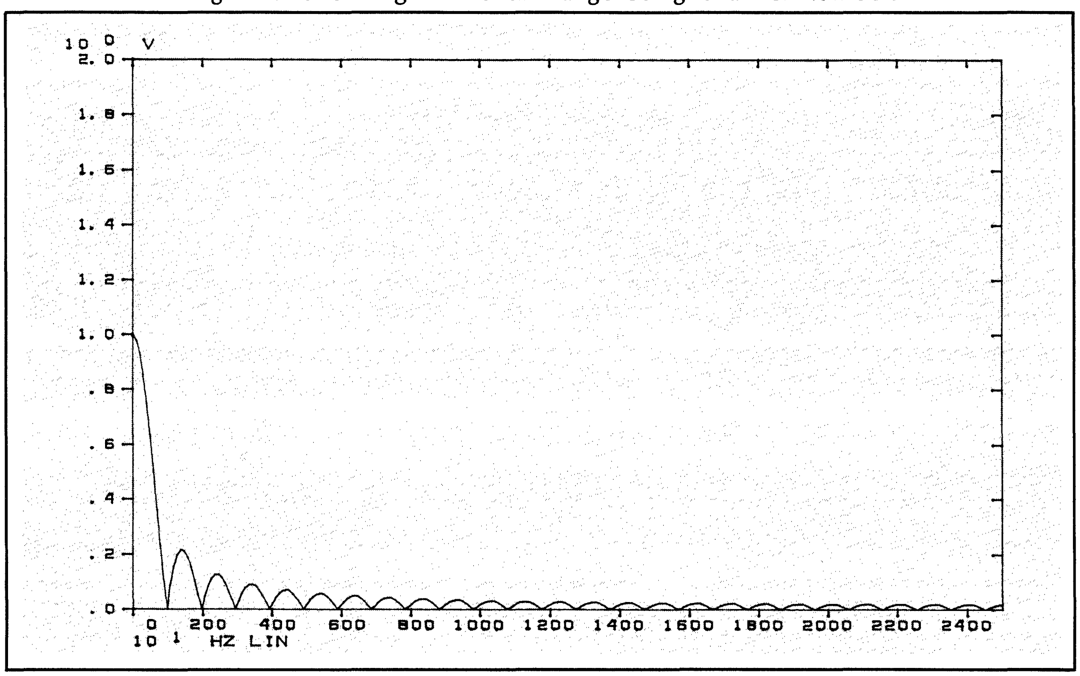
3. Clear the screen by entering USER PROG 5814 ENTER.

4. Enter the following commands to plot the data being displayed.

   USER PROG 5815 ENTER
   USER PROG 5816 ENTER
   USER PROG 5817 ENTER

   Figure 6-29 is an example of the plot that should result. Notice that the horizontal range of the data is full scale and was indicated by defaulting n2 and n3 in 5815 and 5829.

Figure 6-29. Setting Horizontal Range Using Channel Numbers

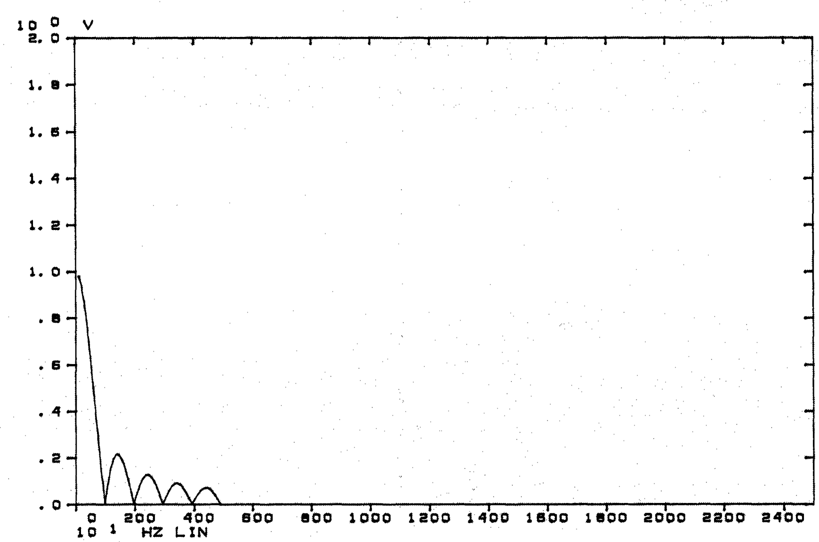


5. Now specify that channels 1 through 50 are to be plotted in a full scale data block.

   USER PROG 5814 ENTER
   USER PROG 5815 SPACE 0 SPACE 1 SPACE 50 ENTER
   USER PROG 5816 ENTER
   USER PROG 5817 ENTER

Figure 6-30 is an example of the plot that should result.

Figure 6-30. Changing Horizontal Range Using Channel Number

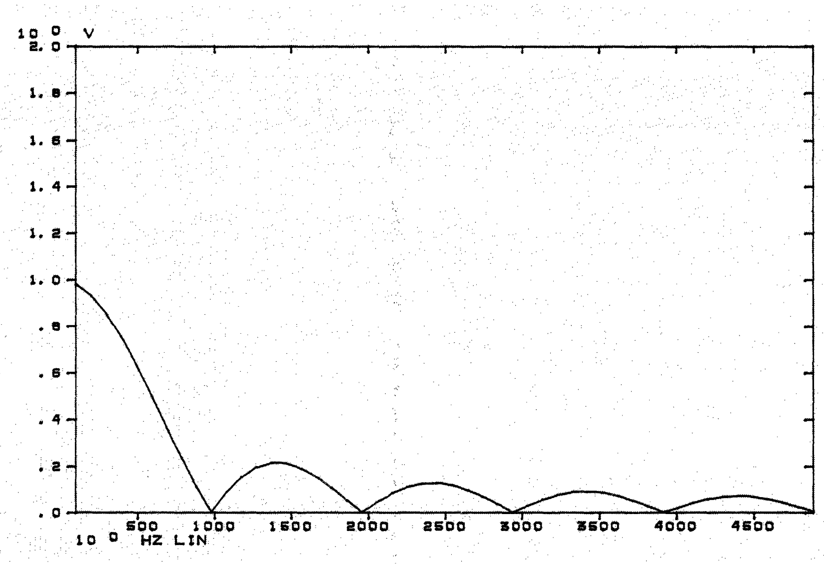


6. Now expand the scale of the plot to fit the number of channels you wish to observe.

   USER PROG 5829 SPACE 0 SPACE 1 SPACE 50 ENTER
   USER PROG 5815 ENTER
   USER PROG 5816 ENTER
   USER PROG 5817 ENTER

   Figure 6-31 is an example of the plot that should result. Note that horizontal axis labeling has been changed.

Figure 6-31. Expanding Scale Using Channel Numbers



5838 READ DISC TEXT BUFFER INTO CORE-RESIDENT TEXT BUFFER

User Program 5838 reads the contents of the specified disc text buffer into the core-resident text buffer (this text buffer having been previously stored on the Disc by User Program 5803 or 5839). Note that text messages are not verified in any way.

Command Format

USER PROG 5838 SPACE n1 ENTER

n1 = the disc text buffer to be read (range depends on number of text buffers available — see Section 2)

Error Conditions

B8 WHAT?	Disc I/O is not configured.
B9 WHAT?	Disc text buffer number specified is illegal. Repeat the command with correct parameter.

Familiarization Procedure

Use the following procedure to become more familiar with the 5838 command.

1. Using the 5803 command, store two different messages in two different disc text buffers.

   USER PROG 5803 SPACE 1 ENTER
   CLEAR ENTER
   INSRT ENTER
   01 CR
   THIS MESSAGE IS STORED IN CR
   DISC TEXT BUFFER NUMBER 1 CR
   /* CR
   TERM ENTER

   USER PROG 5803 SPACE 2 ENTER
   CLEAR ENTER
   INSRT ENTER
   01 CR
   THIS MESSAGE IS STORED IN CR
   DISC TEXT BUFFER NUMBER 2 CR
   /* CR
   TERM ENTER

   The text messages just entered using 5803, should be stored in disc text buffers 1 and 2 respectively.

2. Recall disc text buffer number 1 and write message number 01.

   USER PROG 5838 SPACE 1 ENTER
   USER PROG 5808 ENTER
   USER PROG 5813 SPACE 1 ENTER
   USER PROG 5814 ENTER
   USER PROG 5819 SPACE 1 ENTER

   The following message should be placed in the upper left corner:

   "This message is stored in disc text buffer number 1"

3. Recall disc text buffer number 2 and write message number 01.

   USER PROG 5838 SPACE 2 ENTER
   USER PROG 5808 SPACE 800 ENTER (reposition so as not to write over previous message if you are using a Plotter)
   USER PROG 5819 SPACE 1 ENTER

   The following message should be placed just below the upper left corner.

   "This message is stored in disc text buffer number 2"

5839 WRITE CORE-RESIDENT TEXT BUFFER ONTO DISC TEXT BUFFER

User Program 5839 writes the contents of the core-resident text buffer onto the disc text buffer. Note that the system does not verify if text messages are present in the core-resident text buffer, or in the disc text buffer being written to. Text buffers 51 through 55 are used by the automated power spectrum and transfer function programs described in Section 2 and Appendix D.

CAUTION

Be careful — This command can destroy desired Text Buffers on Disc.

Command Format

USER PROG 5839 SPACE n1 ENTER

n1 = the number of disc text buffer to be written to (range depends number of text buffers available — see Section 2)

Error Conditions

D6 WHAT? Illegal parameter; n1 is ≤0 or it has been defaulted. Repeat the command.

Familiarization Procedure

Use the following procedure to become more familiar with the 5839 command.

1. Using the 5803 command, create a text message.

   USER PROG 5803 ENTER
   CLEAR ENTER
   INSRT ENTER
   01 CR
   THIS MESSAGE IS STORED IN CR
   DISC TEXT BUFFER CR
   /* CR
   TERM ENTER

   The message just entered is stored in the core-resident text buffer and can be written over or cleared unless it is saved on the Disc.

2. Now, using 5839, write the text message 01 to disc text buffers 4, 5, and 6.

   USER PROG 5839 SPACE 4 ENTER
   USER PROG 5839 SPACE 5 ENTER
   USER PROG 5839 SPACE 6 ENTER

3. Clear the core-resident text buffer using the following commands:

   USER PROG 5803 ENTER
   CLEAR ENTER (Erase what was in the core-resident text buffer)
   LIST ENTER (LIST command — nothing should be listed)
   TERM ENTER

4. Recall disc text buffer numbers 4, 5, and 6.

   USER PROG 5821 SPACE n1 ENTER
   USER PROG 5813 SPACE 1 ENTER
   USER PROG 5814 ENTER
   USER PROG 5808 ENTER
   USER PROG 5838 SPACE 4 ENTER
   USER PROG 5819 SPACE 1 ENTER

Message 01 should be written to the plotting device specified via the 5821 SPACE n1 command.

USER PROG 5838 SPACE 5 ENTER
USER PROG 5808 SPACE 500 ENTER
USER PROG 5819 SPACE 1 ENTER

Message 01 should be written to the plotting device specified.

USER PROG 5838 SPACE 6 ENTER
USER PROG 5808 SPACE 200 ENTER
USER PROG 5819 SPACE 1 ENTER

Message 01 should be written to the plotting device specified.

5864 SPECIFY VERTICAL HALF-PLANE

User Program 5864 allows you to specify that only half the Y-plane is to be plotted. However, please note that when you are plotting single precision, frequency, linear, or polar data and the MODE switch is set to MAGNITUDE, the system will appear to give you a positive vertical half-plane whether you have specified it or not (using 5864).

Using the 5864 command you can specify the positive half of the Y-plane or 5864 can be defaulted to specify the entire Y-plane.

Command Format

USER PROG 5864 SPACE n1 ENTER

n1 = -1, the negative half of the Y-plane is plotted.

n1 = 0, the entire Y-plane is plotted.

n1 = +1 or 1, the positive half of the Y-plane is plotted.

An exception to this command is when log vertical data is specified. In this case, the 5864 command is ignored. Once 5864 has been specified, you must execute it again with different parameters in order to change the Y-plane specification of the previous call to 5864.

Error Conditions

X0 WHAT?	More than 1 parameter has been specified. Repeat the command with only 1.
X1 WHAT?	n1 is not 1, 0, or -1. Repeat the command.

Familiarization Procedure

Use the following procedure to become more familiar with the 5864 command. Note that the data that is manually entered is for demonstration purposes and is not intended to depict any normal sequence of keystrokes.

1. Enter the following commands to create some data.

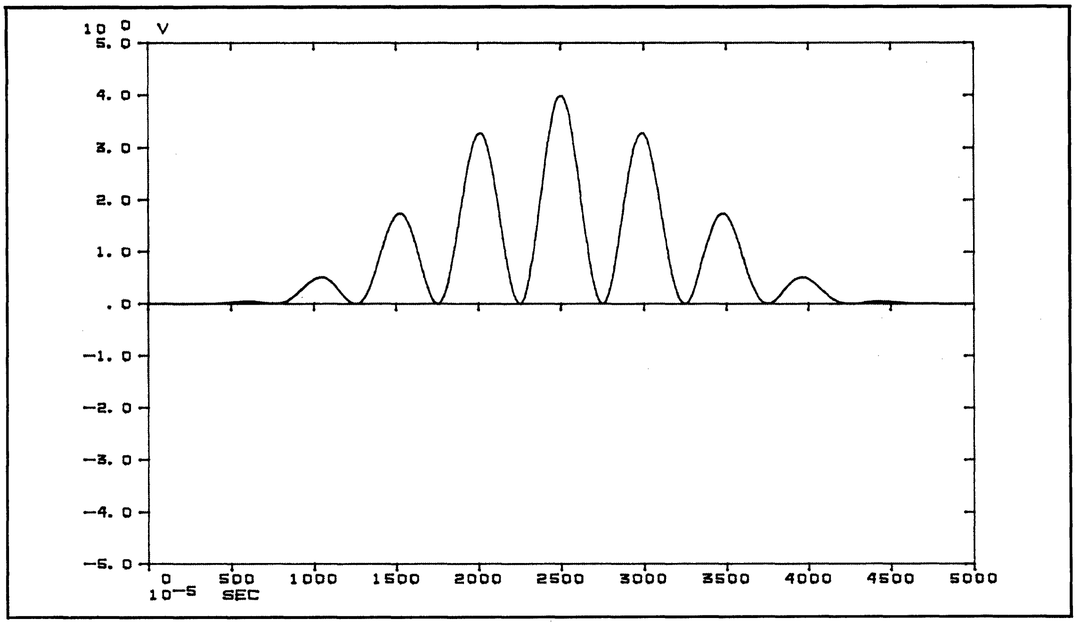
   BLOCK SIZE 512 ENTER
   CLEAR 0 ENTER
   KEYBOARD 0 SPACE 5 SPACE 5 ENTER
   KEYBOARD -4 SPACE 4 SPACE 62 ENTER
   10000 ENTER
   F ENTER
   HANN ENTER
   *MULT ENTER

2. Plot the data being displayed by entering:

   USER PROG 5800 SPACE n1 ENTER (where n1 is 6, 10, or 35)

   Figure 6-32 is an example of the plot that should result.

Figure 6-32. Initial Plot for Specifying Vertical Half-Plane



3. Now enter the 5864 command specifying the positive half of the Y-plane.

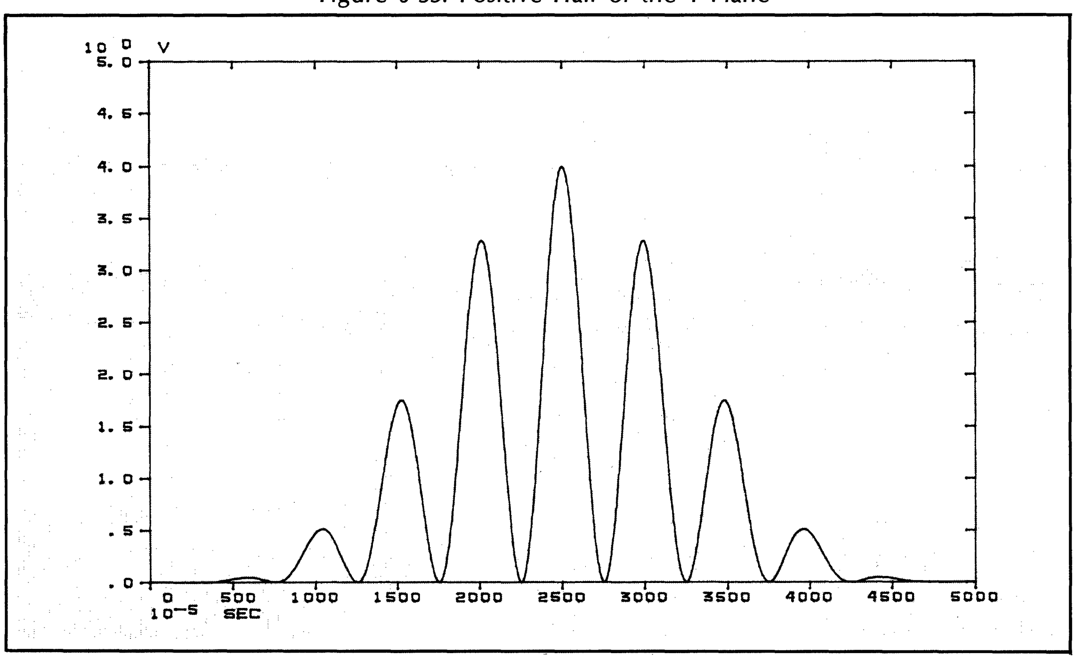
   USER PROG 5864 SPACE 1 ENTER

4. Now plot the data being displayed.

   USER PROG 5800 SPACE n1 ENTER

   Figure 6-33 is the result of the above commands.

Figure 6-33. Positive Half of the Y-Plane



5865 SPECIFY VERTICAL SCALE

User Program 5865 allows you to select the vertical range for log vertical plots and whether that range is to specify dB or decades. As in the case of 5810 and 5829 which expand the horizontal scale, 5865 expands the vertical scale when plotting (i.e., a portion of the data you are looking at can be expanded vertically to a full page plot). Refer to the Familiarization Procedure in this discussion.

Command Format

USER PROG 5865 SPACE n1 ENTER

n1 = 1 through 8, decades (enter a number between 1 and 8)

n1 = 10 through 80, dB (enter a number between 10 and 80 in multiples of 10)

NOTE

The default condition (no call to 5865 or 5865 called with no parameter) is 80 dB. When 5865 is called to set a range other than 80 dB, the data is taken starting at the top of the display screen for plotting. For example: when 10 dB is specified, the data seen in the top division of the display screen will be plotted. If the data of interest is not in the top division, the SCALE switch on the Display Unit, when used with User Program 5809, will raise (expand vertically) the data to the top division so that it will be plotted. When 5865 is called to specify decades, the equivalent scale is 20 dB/decade for voltage data, and 10 dB/decade for power data. Refer to Section 2 for additional information.

Error Conditions

X2 WHAT?	More than one parameter was entered. Repeat the command with only one parameter.
X3 WHAT?	Invalid parameter passed. Repeat the command with an integer between 1 and 8 or 10 and 80.

Familiarization Procedure

User the following procedure to become more familiar with the 5865 command.

1. Enter some single precision data using the following keyboard commands.

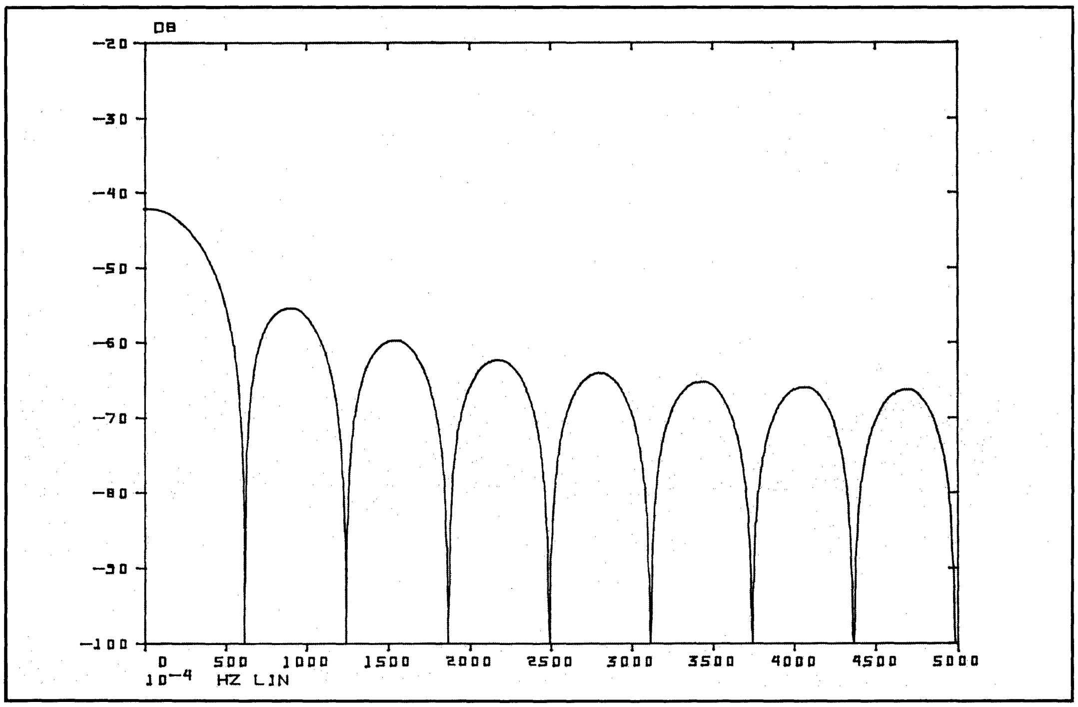
   BLOCK SIZE 2048 ENTER
   CLEAR ENTER
   KEYBOARD 0 SPACE 0 SPACE 15 ENTER
   KEYBOARD 0 SPACE 0 SPACE 9 ENTER
   1 ENTER
   F ENTER
   LOG MAG ENTER

2. Using the default condition of 5865, plot 80 dB of the data.

   USER PROG 5865 ENTER
   USER PROG 5800 SPACE n1 ENTER (where n1 is 6, 10, or 35)

Figure 6-34 is an example of the plot that should result.

Figure 6-34. Initial Plot for Specifying Vertical Scale

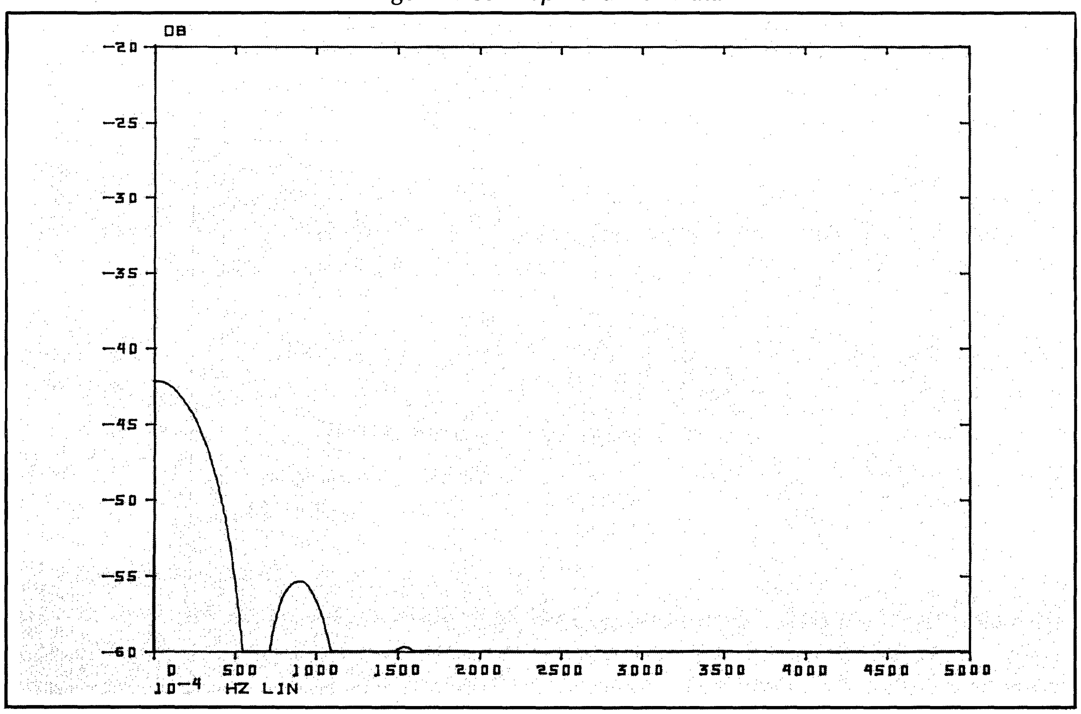


3. Now, plot 40 dB of the data (starting at the top of the display screen).

   USER PROG 5865 SPACE 40 ENTER
   USER PROG 5800 SPACE n1 ENTER (where n1 is 6, 10, or 35)

   The data should be plotted as shown in Figure 6-35.

Figure 6-35. Top 40 dB of Data

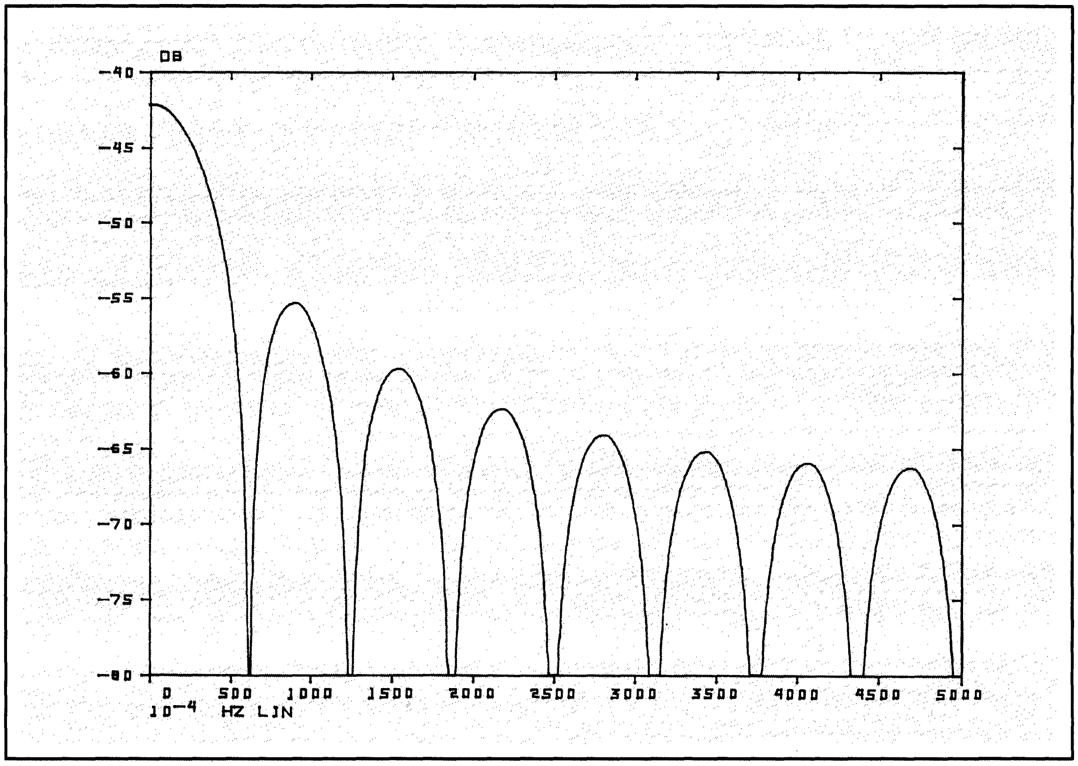


4. Now, using the SCALE switch, move the data into the top 40 dB of the display screen. First move the SCALE switch so that most of the data is situated in the top 40 dB of the display screen without overflowing, then execute the following commands.

   USER PROG 5865 SPACE 40 ENTER
   USER PROG 5800 SPACE n1 ENTER

   Figure 6-36 is an example of the plot that should result.

Figure 6-36. Moving Data Using SCALE Switch

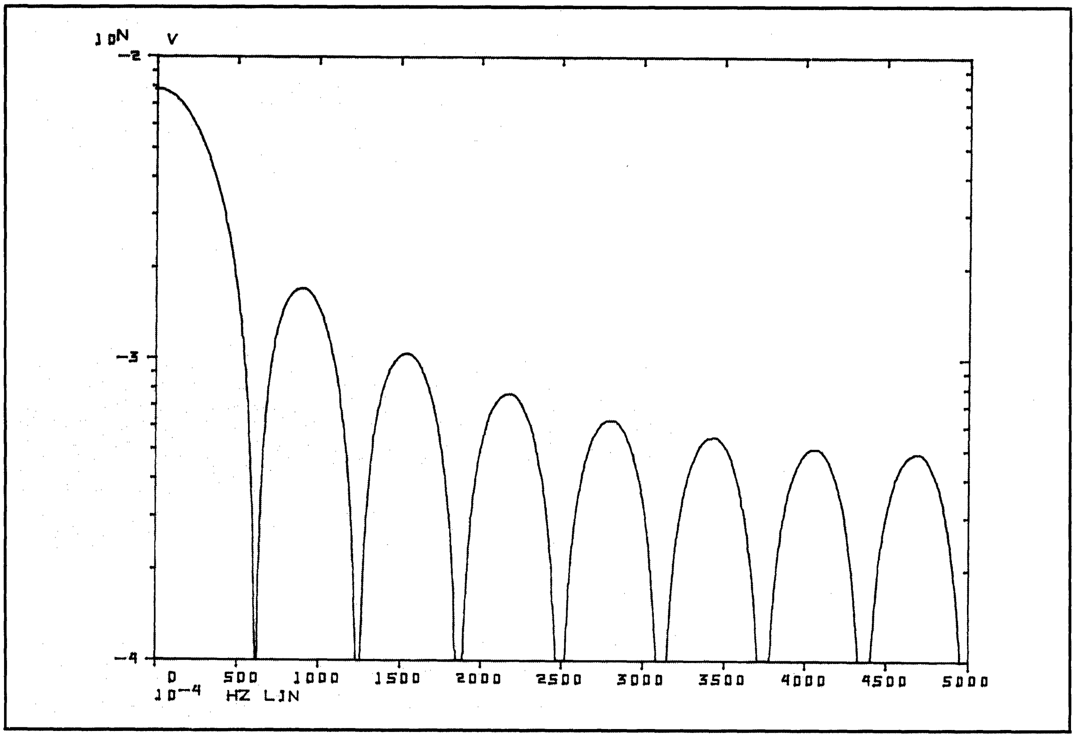


5. If you want to plot the same data in decades, you must specify 2 decades with 5865 since there are 20 dB/decade for single precision (voltage) data.

   USER PROG 5865 SPACE 2 ENTER
   USER PROG 5800 SPACE n1 ENTER

Figure 6-37 is an example of the plot that should result.

Figure 6-37. Data Plotted in Two Decades

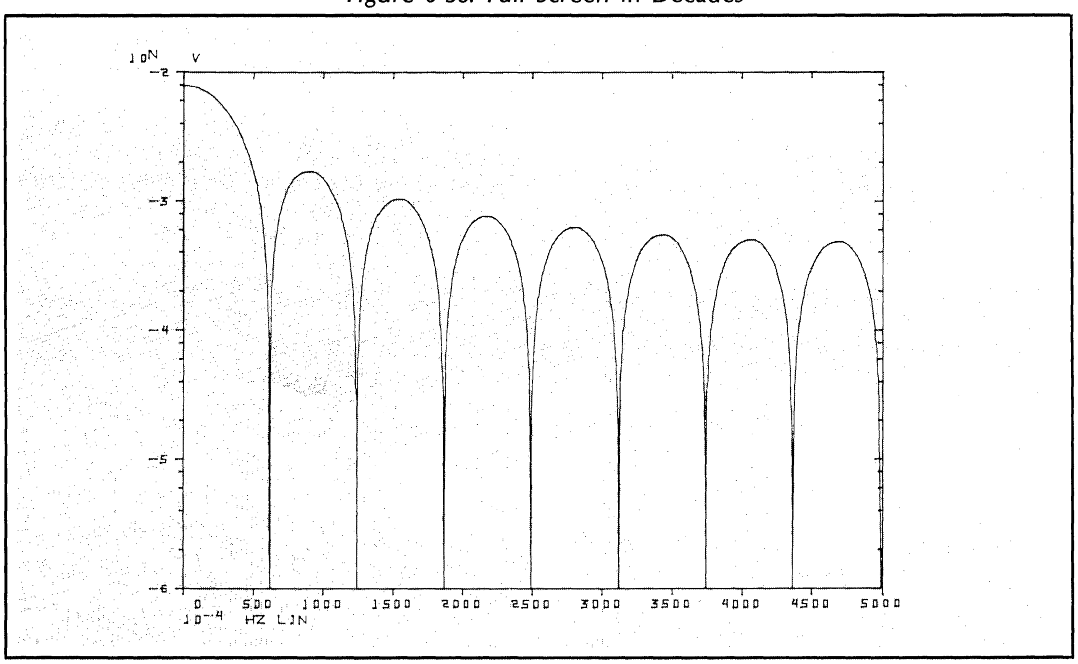


6. To plot the data in 4 decades, enter the following commands.

   USER PROG 5865 SPACE 4 ENTER
   USER PROG 5800 SPACE n1 ENTER

Figure 6-38 is a plot of the full screen in decades (i.e., 80 dB equals 4 decades).

Figure 6-38. Full Screen in Decades



5881 PEN SELECTION (HP 9872)

User Program 5881 allows you to select different pens on the HP 9872 Plotter for plotting data, axes and labels, and text.

Command Format

USER PROG 5881 SPACE n1 SPACE n2 SPACE n3 ENTER

n1 = pen number for data (or vectors).

n2 = pen number for the axes and labels.

n3 = pen number for text.

Range for n1, n2, and n3 is 0 to 4. If 0 is entered for any parameter, the corresponding pen remains unchanged.

Parameter n1 selects the pen to be used in subsequent calls to USER PROG 5815 (plot data) and USER PROG 5812 (draw vector).

Parameter n2 selects the pen to be used in subsequent calls to USER PROG 5816 (draw axes) and USER PROG 5817 (draw labels).

Parameter n3 selects the pen to be used in subsequent calls to USER PROG 5819 (plot text).

If any parameters are defaulted, the corresponding pens remain unchanged. If USER PROG 5881 is never called, the system defaults to:

n1 = 1 pen #1 for data

n2 = 2 pen #2 for axes/labels

n3 = 3 pen #3 for text

Error Conditions

K2 WHAT?	Incorrect number of parameters or illegal parameter value.

Familiarization Procedure

To select pen #4 for data, pen #1 for axes/labels, and pen #3 for text, press:

USER PROG 5881 SPACE 4 SPACE 1 SPACE 3 ENTER

To change the text pen (#3) to pen #2, press:

USER PROG 5881 SPACE 0 SPACE 0 SPACE 2 ENTER

To change the data pen (#4) to pen #1, press:

USER PROG 5881 SPACE 1 ENTER

5882 LINE TYPE SELECTION (HP 7245 & 9872)

User Program 5882 allows you to select one of seven different line types for plotting data (User Program 5815) and drawing vectors (User Program 5812).

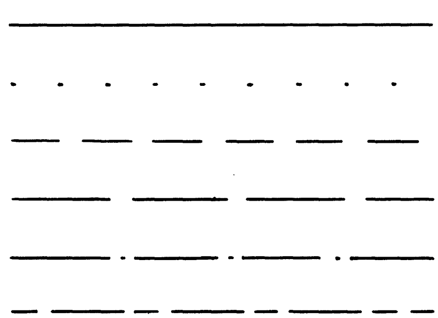
Command Format

USER PROG 5882 SPACE n1 ENTER

Parameter n1 selects the line type to be used in subsequent calls to USER PROG 5815 or 5812 according to the table below:

n1 = **Line Type**

0
1*
2
3
4
5



Default = 0.

*Dot at each data point.

Error Conditions

K3 WHAT? Incorrect number of parameters or illegal parameter value.

Familiarization Procedure

To select dots at each data point, press:

USER PROG 5882 SPACE 1 ENTER

To select a solid line, press:

USER PROG 5882 ENTER

If User Program 5882 is never called, the system defaults to n1 = 0 (solid line).

5883 SET SCALING POINTS P1 & P2 (HP 7245 & 9872)

The scaling points P1 and P2 define the plotting area on the platen surface. P1 and P2 can be entered manually by using the controls on the plotter or they can be entered from the Analyzer. User Program 5883 allows you to set P1 and P2 to be the total plotting area including labels or excluding labels.

Command Format

USER PROG 5883 SPACE n1 SPACE n2 ENTER

n1 = 0 or default, P1 and P2 define the plotting area including labels.

n1 ≠ 0, P1 and P2 define the plotting area excluding labels.

n2 = a positive number indicating the first of four consecutive integer variable parameter locations into which the X and Y coordinates of the current P1 and P2 settings are stored. These coordinates are stored as follows:

V.P.*	Contents
n2	P1 X
n2 + 1	P1 Y
n2 + 2	P2 X
n2 + 3	P2 Y

*Variable Parameter locations n2 through n2 + 3 must be type integer.

n2 = a negative number indicating the first of four consecutive variable parameters which contain the P1 and P2 values which are to be output to the plotter. The desired coordinates of P1 and P2 must be previously stored in V.P. locations n2 to n2+3 according to the table above.

The range of allowable values for P1 and P2 depends on the particular HP-IB plot device. Refer to the Operating Manual shipped with your HP-IB plotter.

Calling User Program 5883 with n1 ≠ 0 is useful for aligning the plot axes origin on pre-printed graph paper. When n1 ≠ 0, be sure to leave enough room outside the area defined by P1 and P2 for the labels to be plotted.

Error Conditions

K4 WHAT?	Incorrect number of parameters or illegal parameter value.
K5 WHAT?	Not enough room left for vertical/horizontal labels on graph.
K7 WHAT?	Invalid V.P. given for n2 (must be type integer).

Familiarization Procedure

1. For this example P1 and P2 are set manually, using the controls on the plotter, so that P1 and P2 define the total plot area including labels. Then the values of P1 and P2 are saved in V.P. locations 5 through 8.

   Using the directional arrows on the plotter, move the pen to the desired location for P1 and press ENTER P1 on the plotter. Move the pen to the desired lcoation for P2 and press ENTER P2 on the plotter.

   On the system Keyboard, press:

   USER PROG 5883 SPACE 0 SPACE 5 ENTER

V.P. locations 5 through 8 will be set as follows:

V.P.	**Contents**
5	P1 X
6	P1 Y
7	P2 X
8	P2 Y

Any subsequent plot will be scaled such that the entire plot, including labels, will be within the boundaries defined by P1 and P2.

To perform the above example without saving P1 and P2 in variable parameters, simply enter:

USER PROG 5883 ENTER

2. For this example, values are stored in V.P. locations 10 through 13 defining the plot area excluding labels. These values are:

   P1 X = 1000
   P1 Y = 2000
   P2 X = 10000
   P2 Y = 8000

   To set the V.P. values, enter the following gold key commands on the Keyboard:

   USER PROG SET 10 SPACE 1000 ENTER
   USER PROG SET 11 SPACE 2000 ENTER
   USER PROG SET 12 SPACE 10000 ENTER
   USER PROG SET 13 SPACE 8000 ENTER
   USER PROG 5883 SPACE 1 SPACE -10 ENTER

   In the last command, the second parameter (-10) specifies that the values in V.P. locations 10 ;tghrough 13 are to be output to the plotter setting P1 and P2. The first parameter (1) indicates that P1 and P2 define the plot area excluding labels. The labels will be plotted outside of the boundaries defined by P1 and P2.

5884 SELECT DRAWN OR DOT MATRIX CHARACTERS AND PAGING (HP 7245)

User Program 5884 allows you to select either drawn characters (similar to the HP 9872) or high-speed dot matrix characters of fixed size (plotted at 38 characters/second). Also, this program allows you to automatically set the 7245 to the next page.

Command Format

USER PROG 5884 SPACE n1 ENTER

n1 = 0 or default, selects high-speed, fixed size, dot matrix characters.

n1 > 0, selects drawn characters.

n1 < 0, sets 7245 to the next page.

If 5884 is never called, the system defaults to the high-speed dot matrix characters.

Error Conditions

K6 WHAT?	Incorrect number of parameters.

GRAPHICS COMMANDS

Table 6-1 is a summary of the graphics commands.

Table 6-1. Graphics Commands

Command	Function
USER PROG 5800	AUTOMATIC PLOTTING n1 = 0 or default for standard plot n1 ≠ 0, execute currently specified plot parameters except 5815, 5816, and 5817 which are always default
USER PROG 5803	TEXT BUFFER EDITOR n1 = 0 or default, core-resident text buffer is used for editing n1 = disc text buffer number n2 = 01 through 99 text ID number of text about to be entered n2 = default, use text editing commands
USER PROG 5804	SET PLOT SIZE n1 = 1 to 1000 for vertical dimension n1 = default, full scale (1000 = 100% from top) n2 = 1 to 1000 for horizontal dimension n2 = default, full scale (1000 = 100% from right of page) n3 = 0 or default, include space for labels n3 ≠ 0, exclude space for labels
USER PROG 5805	SET PLOT LOCATION ON PAGE n1 = 1 to 1000 (vertical distance from bottom in .1% of page) n2 = 1 to 1000 (horizontal distance from left in .1% of page) n1 and n2 = default (start at lower left corner)
USER PROG 5806	SET PLOT POSITION INCREMENT n1 = 0 to 2000 (vertical axis increment in .1% of full scale) n2 = 0 to 2000 (horizontal axis increment in .1% of full scale) n1 and n2 = 0 (no incrementing will take place)
USER PROG 5807	SET PLOT ORIENTATION n1 = 0 or default, orientation from left to right n1 = <0, rotate plot 90° clockwise n1 = >0, rotate plot 90° counterclockwise
USER PROG 5808	SET MESSAGE POSITION n1 = 1 to 986 (vertical distance from bottom in .1% of full scale) n1 = default, top left side of page/screen n2 = 1 to 999 (horizontal distance from left in .1% of full scale) n2 = default, left side of page/screen
USER PROG 5809	SET MODE, ORIGIN, AND SCALE PARAMETERS n1 = 0 or default, Real/Magnitude plotted n1 = 1, Imaginary/Phase plotted n1 = –1, Complex plotted n2 = 0 or default, Linear Horizontal plotted n2 = 1, origin center plotted n2 = –1 to –5, log horizontal plotted with 1 to 5 decades, respectively n2 = default with ORIGIN switch in LOG, 5 decades are plotted n3 = 0 or default, scale switch 12 o-clock n3 = >0 up to 8, plot data to scale indicated by the parameter entered All Defaulted = copy the Display Unit switch settings for mode, origin, and scale

Table 6-1. Graphics Commands (cont'd)

Command	**Function**
USER PROG 5810	SET HORIZONTAL RANGE USING FREQUENCY OR TIME UNITS n1,n2 = End Point = $n1 \times 10^{n2}$ n3,n4 = Start Point = $n3 \times 10^{n4}$ Default all = automatic scaling of range Default n2, n3, n4 = 0
USER PROG 5811	PLOT BOUNDARY ALIGNMENT n1 = 0, move pen to left center of plot area n1 = >0, move pen to upper right of plot area n1 = <0, move pen to lower right of plot area n1 = default, move to the next pen position in sequence
USER PROG 5812	DRAW VECTOR n1,n2 = End Point X,Y n3,n4 = Start Point X,Y *Range* for X and Y (start and end point) on Plotter is 0 to 9999. On Terminals, *range* is 0 to 1023 for X and 0 to 779 for Y. Default n2, n3, n4 = current position
USER PROG 5813 (Terminal only)	ECHO CONTROL n1 = 0 or default, entries and system response are displayed n1 ≠ 0, entries are not displayed
USER PROG 5814 (Terminal only)	ERASE SCREEN Erases Terminal screen when entered
USER PROG 5815	PLOT DISPLAY SCREEN n1 = core-resident data block number n1 = default or 0, data block plotted from core-resident data block 0 n2 = first channel to be plotted n3 = last channel to be plotted n2, n3 = default, plot full block
USER PROG 5816	DRAW AXIS n1 = axis type: 1 ├── Freq or Time rect. 2 ─┼─ Complex 3 ┌── Log. Vert. 4 ─┬─ Neg. Y-half plane 5 └── Power Spectrum 6 ─┴─ Pos. Y-half plane 0 (default to type 1)

USER PROG 5817	ANNOTATE PLOT AXES
	n1 = text message ID for vertical axis unit or 0 for default labeling
	n2 = text message ID for horizontal axis unit or 0 for default labeling
	n3 = <0, do not label vertical axis
	n3 = >0, do not label horizontal axis
	n3 = 0, label both axes
	n4 = 0 or default, use default labeling
	n4 = 1, vertical units may be DB RE G, G SQR/HZ, DB RE MSG # n1
	n4 = 2, horizontal units are SEC
	n4 = 3, horizontal units are RPM

USER PROG 5818	SPECIFY GRID LINES TO BE DRAWN n1, n2 = 1, default or 0 (all grid lines are drawn) n1, n2 = 1, 1 (major grid lines are drawn) n1, n2 = default or zeroes (no grid lines)

Table 6-1. Graphics Commands (cont'd)

Command	Function
USER PROG 5819	WRITE A LINE OF TEXT n1 = message ID number (01 to 99) n1 = 0 or default, write same message as previous call n1 = <0, write on-line text n2 = 1, write the text to the Terminal
USER PROG 5820 (Terminal only)	MAKE HARD COPY No parameters are necessary. When command is entered the plot being displayed on the Terminal is copied on the hard copy unit.
USER PROG 5821	INITIALIZE THE SYSTEM FOR PLOTTING n1 = 6 (for plotting on the Terminal) n1 = 10 (for plotting on the 7210A Plotter) n1 = 35 (for plotting on HP-IB Plotter) n2 = subunit number or default to indicate n1
USER PROG 5829	SET HORIZONTAL RANGE USING CHANNEL NUMBERS n1 = core-resident data block number n2 = first data channel to be plotted n3 = last data channel to be plotted n2,n3 = defaulted, entire data block is plotted
USER PROG 5838	READ DISC TEXT BUFFER INTO CORE-RESIDENT TEXT BUFFER n1 = disc text buffer to be read (0-118)
USER PROG 5839	WRITE CORE-RESIDENT TEXT BUFFER ONTO DISC TEXT BUFFER n1 = disc text buffer number (0-118)
USER PROG 5864	SPECIFY VERTICAL HALF-PLANE n1 = –1, negative half of Y-Plane is plotted n1 = 0, the entire Y-Plane is plotted n1 = 1, positive half of Y-Plane is plotted 5864 is ignored for log vertical data
USER PROG 5865	SPECIFY VERTICAL SCALE n1 = 1 to 8, decades are plotted for vertical axis n1 = 10 to 80, dB's are plotted for vertical axis n1 = default (80 dB is plotted for vertical axis)
USER PROG 5881	PEN SELECTION (HP 9872) n1 = pen number for data (or vectors) n2 = pen number for axes and labels n3 = pen number for text
USER PROG 5882	LINE TYPE SELECTION (HP 7245 & 9872) n1 = 0 thru 6 for line type, e.g., 0 = solid lines, 1 = dots, 2 = small dashes, etc.
USER PROG 5883	SET SCALING POINTS P1 & P2 (HP 7245 & 9872) n1 = 0, include labels within P1, P2 n1 ≠ 0, exclude labels within P1, P2 n2 = variable parameter locations for P1, P2 coordinates
USER PROG 5884	SELECT CHARACTERS & PAGING (HP 7245) n1 = 0, selects dot matrix characters n1 >0, selects drawn characters n1 <0, sets to next page

ERROR MESSAGES

Table 6-2 is a summary of the possible error messages that can occur, the user program that generated the message, and the meaning of the error message.

Table 6-2. Error Messages

Message	USER PROG	Meaning
A0 WHAT?	5816	An illegal axis type has been entered (i.e., n<1 or >6)
A1 WHAT?	5821	An illegal device type has been entered (i.e., n1 not 6, 7, or 10)
A2 WHAT?	5821	The subunit number entered does not exist.
A3 WHAT?	5812	Insufficient number of parameters or no parameters were passed.
A4 WHAT?	5812	Vector is out of screen/plot limits.
A6 WHAT?	5810	n1, n3 <0 or ≥10000
A7 WHAT?	5810	n4 is illegal
A8 WHAT?	5810	$n3 \times 10^{n4} \geq n1 \times 10^{n2}$
A9 WHAT?	5829	Illegal number of parameters
B0 WHAT?	5829	n1, n2 ≤0, illegal block #, or n2 > n3
B1 WHAT?	5803	Disc text buffer # (n1) is <0
B2 WHAT?	5803	Text ID n2 ≤0 or >99
B3 WHAT?	5803	Error detected during special edit
B4 WHAT?	5829	Illegal message identification (<01 or >99)
B5 WHAT?	5819	Message does not exist in the core-resident text buffer
B6 WHAT?	5815	Illegal frequency code
B7 WHAT?	5804, 5805, 5806, 5808	n1, n2 = 0 or n1, n2 = 2000
B8 WHAT?	5803, 5838	Disc I/O is not configured
B9 WHAT?	5803, 5838	Illegal disc buffer. Not allowed access.
C0 WHAT?	5809	Number of parameters is >4
C1 WHAT?	5809	Parameter is out of range
C2 WHAT?	5815	2 or 4 or more parameters are not allowed
C4 WHAT?	5803	Disc I/O error
C5 WHAT?	5817	n1 or n2 <0, n4 >0, n4 >3
C7 WHAT?	5815	Illegal core-resident block number
C8 WHAT?	5815	Frequency Domain ORIGIN CENTER plot attempted
D6 WHAT?	5839	n1 ≤0 or it has been defaulted
K2 WHAT?	5881	Illegal number or value of parameters
K3 WHAT?	5882	Illegal number or value of parameters
K4 WHAT?	5883	Illegal number or value of parameters
K5 WHAT?	5883	Insufficient space for labels on graph
K6 WHAT?	5884	Illegal number of parameters
K7 WHAT?	5883	Illegal V.P. given for n2

Table 6-2. Error Messages (cont'd)

Message	USER PROG	Meaning
L6 WHAT?	5800	n1 isn't a valid device code
L7 WHAT?	5800	n2 is an invalid block number
X0 WHAT?	5864	More than one parameter has been entered
X1 WHAT?	5864	n1 is not 1, 0, or −1
X2 WHAT?	5865	More than one parameter has been entered
X3 WHAT?	5865	Invalid parameter passed
???	5803	5803 editing commands not understood by system
ILLEGAL BUFFER	5803	Data in the text buffer just read is not valid
WARNING — CLEAR NEW DISC BUFFERS	5803	Extraneous data may be present in the disc text buffer just accessed
HALT 00 (102000_8) in "S" register	5808, 5811, 5816, 5817	Y5821 has not been executed to specify the plot device

SECTION 7
USER PROGRAMS

INTRODUCTION

This section provides information on the User Programs used in the Fourier Analyzer System. Table 7-1 provides an overall view of the structure of the system software. Table 7-2 lists the individual User Programs and indicates where each is located in the software. It also lists the section of the manual in which each of the User Programs are described. All User Programs (both resident and non-resident) that have not been described previously are covered in this section. In addition, some overlay swapping considerations are explained.

Table 7-1. Overall Structure of System Software

Drivers		Programs
SYSTEM	D.00 D.34 (Disc) D.53 (FDB) D.33 (Mag Tape opt.)	Operating software for all keyboard functions except: ANALG OUT, gold PLOT, MASS STORE 58
OVERLAY 0 (Baseband)		DAC software, Filter software (Y 100), Y 99 (transient capture), Y 5, Y 6, Y 999, MASS STORE 58
OVERLAY 1 (Software BSFA)		Filter software (Y 100), Y 5, Y 6, Standard (offline software) BSFA: Y 41, Y 44, Y 45, Y 141
OVERLAY 2 (Hardware BSFA)		Filter software (Y 100), Y 5, Y 6, Hardware BSFA: Y 40, Y 43, Y 44, Y 45, Y 141
OVERLAY 3 (Graphics and text buffers)	D.06 (Terminal) D.10	Graphics, Text buffer (except system-resident function Y 5819, Y 5838), gold PLOT (Y 5800)

Table 7-2. Resident User Program Locations and Section Index

System

User Program	Section Index
1800	2
1801	
1802	
1803	
1804	
1805	
1806	
1807	
1808	
1809	
1810	
1817	
1821	
1822	
1823	
1827	
1828	
1829	
1830	
1850	
1851	
1852	
1853	
1854	2
3010	7
3011	
3012	
3016	
3020	
3022	
3024	
3025	7
3026	7
5819	6
5838	6

Overlay 00

User Program	Section Index
0005	7
0006	
0099	
0100	
0999	
3021	
3023	7

Overlay 01

User Program	Section Index
0005	7
0006	7
0041	5
0044	5
0045	5
0100	7
0141	5
3017	7

Overlay 02

User Program	Section Index
0005	7
0006	7
0040	5
0043	5
0044	5
0045	5
0100	7
0141	5
3017	7

Overlay 03

User Program	Section Index
5800	6
5803	
5804	
5805	
5806	
5807	
5808	
5809	
5810	
5811	
5812	
5813	
5814	
5815	
5816	
5817	
5818	
5820	
5821	
5829	
5839	
5864	
5865	6

OVERLAY SWAPPING CONSIDERATIONS

When an overlay is read into core from the Disc, any programs previously residing in that portion of core are destroyed. Normally this is of no concern. However, in certain instances, a program may set up parameters for itself in core to be used in subsequent operations. Consequently, if these parameters are inadvertently lost by reading in another overlay, the program could operate incorrectly if called again. This danger is particularly present when an "invisible" overlay swap occurs, since it then may not be apparent that an overlay has been read into core. In order to minimize the danger of destroying program parameters by overlay swapping, it is good practice to include all the User Programs you will need for the measurement you wish to perform in the same overlay.

Consult the User Program locations of Table 7-2. Notice that some of the User Programs in overlay 0 also appear in other overlays. This duplication would be unnecessary (that is, only one overlay would have to contain a given User Program) except that a lot of overlay swapping would be necesary to perform measurements. With the User Programs distributed as shown in Table 7-2, however, you are able to make a baseband measurement using only overlay 0, a software BSFA measurement using only overlay 1, and so forth. When adding your own overlays to the system, you should attempt to load all the programs you will need into that overlay to perform the desired measurement. Keep in mind that the following devices are affected by overlay swapping:

1. If the DAC is running at the time of an overlay swap, it is turned off before the new overlay is brought into core.
2. If the Low Pass Filters are set to auto-track the ADC setting, this is disabled if an overlay swap occurs. Filters will stay at their last settings but will no longer auto-track unless set up again to do so.

RESIDENT USER PROGRAMS

The resident User Programs described in the following paragraphs are those programs that have not been described in other sections. (Table 7-2 is a list of all resident User Programs and the section in which each is described.)

USER PROGRAM 5

User Program 5 checks ADC data for overload. If an overload is detected, the program branches in the keyboard program stack.

Command Format

USER PROG 5 SPACE n1 SPACE n2 ENTER

n1 is the first block of analog input (default = block 0).

n2 is the number of steps in the keyboard stack program to to skip if an overload is detected in any of the data blocks (n2 can be negative. Default n2 = –1, i.e., skip back one step).

User Program 5 checks the ADC data starting in block n1 for overload. If no overload occurs, the program exits and the next keyboard instruction is executed. If an overload did occur, the overloaded channel is displayed and the number of instructions specified by n2 are skipped.

The following Keyboard Program listing shows several examples of the instruction that would be executed following the call to User Program 5 if an overload is detected, based on parameter n2.

KEYBOARD INSTRUCTION (goes here if n2 = -2)

KEYBOARD INSTRUCTION (goes here if n2 = -1 or is defaulted)

Y 5 SPACE n1 SPACE n2 (never goes here)

KEYBOARD PROGRAM (goes here if n2 = 0, or if no overload is detected)

KEYBOARD INSTRUCTION (goes here if n2 = 1,or if no overload is detected)

A typical use of User Program 5 could be:

Y 99 X Y Z (capture transient in block 0)

Y 5 (re-execute Y 99 if an overload occurs)

Error Messages

None. You should take care that n2 is small enough so that User Program 5 will not attempt to skip outside the Keyboard Program should an overload be detected. This condition is not checked by the program.

USER PROGRAM 6

User Program 6 checks that the ADC INPUT SELECTOR switch is set to a specified number of channels. If it is, the program skips the number of steps specified. If it is not, the program goes to the next instruction.

Command Format

USER PROG 6 SPACE n1 SPACE n2 ENTER

n1 is the number of ADC input channels desired. It must be 1, 2, 3, or 4 with no default allowed.

n2 is the number of instructions to skip forward in the program stack (n2 must be zero or positive). Default value is 1.

Error Messages

Z4 WHAT?	no parameters
Z5 WHAT?	n1 not 1,2,3, or 4. n2 negative.

Also, note that you must guard against n2 being greater than the number of remaining program instructions as this condition is not checked by the program.

USER PROGRAM 99 (Single Channel Only)

User Program 99 is used to capture transients and the incidents that immediately precede them. It can only be used with single channel input.

Command Format

USER PROG 99 SPACE n1 SPACE n2 SPACE n3 SPACE n4 ENTER

n1 is the trigger level in millivolts.

n2 is the trigger delay time. It can be positive or negative.

n3 is the trigger delay time units (0 = seconds, -3 = milliseconds, -6 = microseconds, and 99 = channels).

n4 specifies the type of trigger signal (+1 indicates to trigger on a rising signal, 0 indicates the unit is to free run with a + delay allowable, and -1 indicates triggering on a falling signal).

If any parameters are defaulted, their previous values are used. The initial values are "Y99 0 0 -3 1" corresponding to zero (0) milliseconds, (-3) delay following the zero (0) millivolt-and-rising (+1) trigger.

When User Program 99 is used, the ADC is started and the data is monitored for a "trigger condition". Before, during, or after (as selected with the parameters) the event, a block of data is input. The digital trigger is defined via a level (between -10000 mV and +10000 mV inclusive), and a sense (i.e., "rising" or "falling" signal). Freerun (i.e., don't wait for trigger) is also provided. Note the trigger channel is always channel A.

The digital ± delay ranges from -1 blocktime (this allows the capture of events preceding the trigger event) to greater than 12 hours following the event. Thus, delay = +10 waits 10 "units" after the trigger, then gathers 1 block of data. Similarly, delay = -10 gathers 1 block of data but begins 10 units of time before the trigger. The "units" of time may be seconds, milliseconds, microseconds, or channels. The maximum sample rate which may be used with this program is 100 kHz.

If the sample rate is greater than 2 kHz, all devices (except the device on the I/O channel assigned to the dummy driver D.76 in the EQT table) are turned off; an interrupt on the D.76 channel (this is typically the Keyboard I/O channel) is considered a restart, and Y 99 jumps to 28 if an interrupt (e.g., RESTART) occurs. If the sample rate is slower than 2 kHz, any device can interrupt properly.

Error Messages

Y 99 : TOO FEW BLOCKS FOR ADC

Y 99 requires 1 data block for each ADC channel plus 1 block to unravel the data. This message occurs if there is not enough data space for the operation.

Y 99 : POS DELAY TOO LARGE

Y 99 uses a double-precision counter for the positive delay. This message occurs if this counter is too large based on current conditions (parameters n2, n3, and the ADC sample rate).

Y 99 : NEG DELAY LARGER THAN BLOCKSIZE

A '-' delay cannot exceed the total sample time for a data block. The above message occurs if the negative delay exceeds the total time for one data block.

Y 99 : ADC CALIBRATION TOO LOW

This message occurs if the trigger level is greater than the overload voltage for channel A of the ADC.

Y 99 : ADC DELTA T TOO SMALL

This message will occur if the sample rate exceeds 100 kHz.

Y 99 : NO D.76 IN EQT TABLE

While running faster than 2 kHz, Y 99 allows interrupts on only the ADC I/O channel, and one other I/O channel for restart capability. This channel is defined by assigning it to D.76 in the EQT Table at system generation time. If this was not done, the above message occurs.

USER PROGRAM 999

User Program 999 reads program DCOPY from its location on track 0, sector 10_8 and then jumps to it (this may be handier than accessing DCOPY from the switch register if your Fourier system is running).

Command Format

USER PROG 9 9 9 ! ENTER

Error Messages

L8 WHAT?

Select code is less than 10_8 or greater than 77_8.

NOTE

The program DCOPY must be properly installed on the system softrware disc for this program to operate sucessfully. Although this User Program is self-configuring, the program DCOPY is configured with the Disc in the standard I/O channels.

USER PROGRAM 3010

Under normal conditions a data block is upscaled (maximized) before and after execution of a command to maintain full accuracy during calculations. This process entails doubling each data word in the data block until further doubling would cause overflow. Calibration of the data block is maintained by adjusting the block calibrator and scale factor each time the data word is doubled. User program 3010 allows you to disable or enable the maximization process before execution of a command.

Command Format

USER PROG 3010 SPACE n1 ENTER

n1 = 1 to disable maximization. Default of n1 enables maximization.

NOTE

Pressing RESTART will automatically cause maximization to be enabled.

Error Messages

None.

USER PROGRAM 3011

This command enables or disables double precision when performing a self-conjugate multiply on a data block.

Command Format

USER PROG 3011 SPACE n1 ENTER

where:

n1 is defaulted to enable double precision results for self-conjugate multiply.

If n1 is entered (any number) double precision is disabled and the results will be the same data type as a normal (or block) conjugate multiply.

USER PROGRAM 3012

This command allows you to specify the maximum number of scaledowns that could occur in block division. In the worst case, this means that the quotient could be scaled down (divided by 2) as many times as specified by the maximum number of scaledowns. To specify a maximum scaledown number, enter this command before performing block division.

Command Format

USER PROG 3012 SPACE n1 ENTER

n1 is the number of scaledowns ranging from 1 to 16. Default of n1 is not allowed.

NOTE

When RESTART is pressed, number of scaledowns (n1) is set to 8.

Error Messages

None.

USER PROGRAM 3016

This command allows you to change the stable averaging of the POWER SPEC key to an exponentially weighted average, as described in Section 3.

Command Format

USER PROG 3016 SPACE n1 ENTER

n1 = the time constant K for the average.

To reset the system to a stable average mode use the command:

USER PROG 3016 ENTER

NOTE

Pressing RESTART automatically enables the stable averaging mode.

Error Messages

None.

USER PROGRAM 3020

This command enables or disables the invisible (automatic) overlay swapping capability of the system (normally this capability is enabled).

If invisible overlay swapping is enabled and a USER PROGRAM is executed that is not resident in the system and the current overlay, the system will search all the disc-resident overlays for the desired program. If an overlay containing the program is found, that overlay is read into core and the program is executed.

Command Format

USER PROG 3020 SPACE n1 ENTER

n1 = 0 (or default) to enable invisible overlay swapping.

n1 ≠ 0 to disable invisible overlay swapping.

Error Messages

Y WHAT?

If invisible overlay swapping is enabled, this message is displayed if no overlay containing the required User Program is found.

If invisible overlay swapping is disabled (via Y 3020) and a User Program is executed that is not resident in the system and the current overlay, the message Y WHAT? is displayed immediately and no searching is done.

USER PROGRAM 3021

This is a dummy User Program number, assigned to the program F8LST, which executes the MASS STORE 58 commands to list User Programs in the system and/or the overlays. F8LST is itself in a system overlay, so assigning it a user program number enables this overlay to be automatically read in when the MS 58 command is used.

Command Format

None. This is a dummy User Program and should never be called via this User Program number.

Software generation note: the entry "@,3021" must appear in the User Program table of the overlay containing the program F8LST.

USER PROGRAM 3022 (Gold Key LOG MAG function)

User Program 3022 alters the number of dB displayed full scale when the data is in LOG coordinates (see Section 3, LOG MAG key). The normal LOG VERT display is 80 dB full scale. This may be changed to anywhere from 10 to 80 dB in 10 dB steps.

Command Format

USER PROG 3022 SPACE n1 ENTER

n1 = number of dB full scale (from 10 to 80 in steps of 10). Default = 80 dB.

If a number less than 10 is entered for n1, a value of 10 is assumed. If a number greater than 80 is entered for n1, a value of 80 is assumed. All entries are truncated so that the quantity (number of dB full scale/10) is an integer.

NOTE

If less than 80 dB full scale is displayed, the display refresh speed will decrease slightly.

USER PROGRAM 3023

This is a dummy User Program number assigned to the program HPDAC, which executes the ANALOG OUT command (on systems with the appropriate option). It enables the overlay containing HPDAC to be automatically read in when the ANALOG OUT command is used.

Command Format

None. This is a dummy User Program and should never be called via this User Program number.

Software generation note: the entry "@, 3023" must appear in the User Program table of any overlay containing the program HPDAC.

USER PROGRAM 3024

User Program 3024 modifies the number of decades displayed in a LOG HORIZ display.

Command Format

USER PROG 3024 SPACE n1 ENTER

n1 = the number of decades to display in LOG HORIZ from 1 to 5. Default equals 5 (normal LOG HORIZ display).

Error Messages

Y WHAT?

This message is displayed if n1 equals 0 or |n1|>5. (If n1 is between -1 and -5, the absolute value of n1 is used as the entry.)

USER PROGRAM 3025

User Program 3025 initializes a specified software driver.

A software 'driver' is a program which interfaces the system to a particular hardware device, such as the Terminal or the Disc. Usually these drivers are resident in memory at all times. However, if a driver is resident in a system overlay, the driver wil be read into memory in its unitialized form (that is, unconfigured for the correct I/O channel for its device) if the overlay is read from the Disc. (Initialization of all drivers is performed only when the system is first started or when RESTART is pushed). If this driver is subsequently used without being initialized, errors or incorrect device operation may result.

To avoid this problem, User Program 3025 should be used to initialize any driver in any overlay just read from the Disc. If the driver is system-resident, then User Program 3025 need not be used.

In the standard system, only the drivers D.06 (for Terminal graphics plotting) and D.10 (for Plotter graphics plotting) reside in a system overlay. However, User Program 3025 need not be used for these drivers as long as User Program 5821 is invoked instead (refer to Section 6 for additional information). If any user-written drivers are added to system overlays, then User Program 3025 must be used in conjunction with them.

Command Format

USER PROG 3025 SPACE n1 SPACE n2 ENTER

n1 equals the Driver Number, D.n1 (no default allowed).

n2 equals the Unit Number, default equals 0.

Examples

Assume the system Equipment Table (refer to the System Software Manual for further details on the Equipment Table and software drivers) contains an entry

23,D.44

and that driver D.44 resides in a system overlay. After reading this overlay into memory, this driver should be initialized as follows:

USER PROG 3025 SPACE 44 ENTER

When a Unit Number is supplied (parameter n2) in the calling sequence, the program will search the Equipment Table for an occurrence of driver D.n1, unit n2. It then configures the driver for the I/O channel of the Equipment Table entry found.

Assume the system Equipment Table contains an entry

22,D.15,U01

The correct entry to configure driver D.15 is

USER PROG 3025 SPACE 15 SPACE 1 ENTER

Error Messages

Z6 WHAT?	No parameters
Z7 WHAT?	Parameter not in range 0-77 decimal.
Z8 WHAT?	Digit in parameter not in range 0-7.
Z9 WHAT?	Driver or unit not found in Equipment Table.

USER PROGRAM 3026

User Program 3026 allows ADC throughput at high (>300 kHz) ADC data rates (= sample frequency × number of throughput channels).

In normal ADC throughput operations (i.e., data rates of <300 kHz) the ADC is programmed NOT to turn off (i.e., keep sampling) between throughput records, as this allows the highest real time throughput speed to be obtained. However, data rates above 300 kHz cause overflow of the ADC interface card data buffer, and therefore data loss, to occur during the throughput record. (Usually this data loss will be within the first few data records only.)

User Program 3026 prevents loss of data at ADC data rates above 300 kHz when entered as shown below. Remember that the throughput operation for this data rate will not be real time; that is, some data will be lost *between* successive records. This command prevents data loss from occurring *during* the records. In addition, the real time throughput speed of the system may be decreased when User Program 3026 is invoked since the ADC will be programmed to turn off between data records.

Command Format

USER PROG 3026 SPACE n1 ENTER

n1 = 1 to perform ADC throughput at data rates exceeding 300 kHz.

n1 = 0 (or default) to return the ADC throughput to its normal mode of operation (data rate of less than 300 kHz).

This program must be entered before the Write ADC Throughput command.

Error Messages

None.

NON-RESIDENT USER PROGRAMS

The following User Programs and software modules are not resident in the base system. However, they have been provided in relocatable format on the Relocatable Software Disc provided with your system. In order to include them in your system, you must either generate a new system overlay containing the programs, or in some cases, regenerate both the system and the overlays. These procedures are documented in the System Software Manual.

A brief description of each program is provided below that includes the following:

1. User Program Number.
2. 'NAM' of the program.
3. Name of the file or library where the program may be found on the Relocatable Software Disc. (The revision character is not supplied, but this may be found from a Relocatable Software Disc directory listing.)
4. Additional information, such as where the program should be loaded (in the system overlay), required PCS input entries (if any), etc.

USER PROGRAM 117

NAM : 'Y 117'

FILE NAME: T3210# ('#' stands for revision level character, which is an alphabet character.

This program maintains a set of data block qualifiers and scales a requested data block to them for display purposes. It may reside in an overlay.

Command Format

USER PROG 117 SPACE n1 SPACE n2 ENTER

n1 = the block where data is currently stored (no default allowed).

n2 = the block for saving scaled data (no default allowed).

If n1 and n2 are not specified, Y 117 is initialized.

Error Messages

Y WHAT? Exactly 2 parameters were not supplied.

When Y 117 is called, it checks to see if block n1 qualifiers are larger than the largest qualifiers previously detected since Y 117 was last initialized. If they are, they are saved for subsequent blocks, and block n1 is moved to block n2 without rescaling. If they are less, the data is moved to block n2 and scaled down so that its qualifiers equal the largest found. As a result, block n2 can be displayed without oscillating since its qualifiers are "locked".

USER PROGRAM 998

NAM : 'Y 998'

FILE NAME: T3221#

This program boots up the FSDS (Fourier System Development Software) system from the lower disc. It may reside in an overlay.

Command Format

USER PROG 998 ENTER

Error Messages

The FSDS sysem must be resident on the lower disc for this program to operate successfully. If it is not, the results of this program are unpredictable.

L7 WHAT? Select Code is less than 10_8 or greater than 77_8.

USER PROGRAM 3201

NAM : 'DSQRT'

FILE NAME: 'A3030#' (Library)

Additional Programs Required: NAM 'SQRT' from Library 'N3501#'

This program takes the square root of the data block specified. It may reside in an overlay.

Command Format

USER PROG 3201 SPACE n1 ENTER

n1 = the block number. Default = 0.

The calculation for various data types is as follows:

Time Domain

Square root is computed by setting any negative values to positive, taking square root, then seting sign back to negative on appropriate points.

Frequency Domain

If single precision, the calculation is done in polar domain (on the magnitude information), then converted back to original domain.

Double precision rectangular data is simply calculated. Double precision log mag data is converted to rectangular domain, calculated, then converted back to log.

USER PROGRAM 1818

NAM : 'VPASC'

FILE NAME: T3049#

User Program 1818 inputs or outputs ASCII from the Terminal to Variable Parameters. It is recommended that integer Variable Parameters be used with this program. It may reside in an overlay.

Command Format

USER PROG 1818 SPACE n1 SPACE n2 SPACE n3 ENTER

n1 = the starting Variable Parameter to contain characters, 2 characters per parameter.

n2 = the number of characters to input or output, limited to 72. Default equals 2.

n3 = 0 or default for entering characters from the Terminal into Variable Parameters.

n3 ≠ 0 for outputting characters from Variable Parameters to Terminal.

User Program 1818 stores ASCII characters in specified (integer) Variable Parameters, 2 characters per parameter. This is useful since you may input the characters into the Variable Parameters, operate on or test them using standard Variable Parameter commands, and then output the characters back to the Terminal.

When a character input is requested, the system will enter the BUSY state until the desired characters have been typed on the Terminal and the RETURN key has been pressed to end the entry.

If character output is requested and the value of the last Variable Parameter to be output is 20137 octal = 8278 decimal, then a CRLF is not done following the character output.

Error Messages

Y WHAT?	No parameters, or more than 72 characters requested.

USER PROGRAM 5840

NAM : 'MBLOK'

File Name: A3030# (Library)

User Program 5840 multiplies or divides data block 0 by a multiplier and an exponent of 10. It may reside in an overlay.

Command Format

USER PROG 5840 SPACE n1 SPACE n2 SPACE n3 ENTER

n1 = multiplier or divisor

n2 = exponent of 10.

n3 = mode (defaulted means divide, undefaulted means multiply).

Error Messages

40 WHAT?	No parameters.

USER PROGRAMS RELATED TO 5477A SYSTEM CONTROL UNIT OPTION

The relocatable Software Disc provided with your system contains programs to operate the 5477A System Control Unit and to implement Remote ADC Programming in systems containing a 5477A. The 5477A and remote programming cable (P/N 05477-60010) are supplied with the Modal Analysis, Signature Analysis, and Vibration Control options to the 5451 system. This section applies only to systems containing a 5477A, in which case the following distinct functions apply:

1. System Control — The pushbuttons on the 5477A may be used to execute a JUMP instruction to a specific label in the keyboard program in Record 0, File 3. In addition, the pushbutton lights and indicator lights may be turned on or off via specific User Program commands.

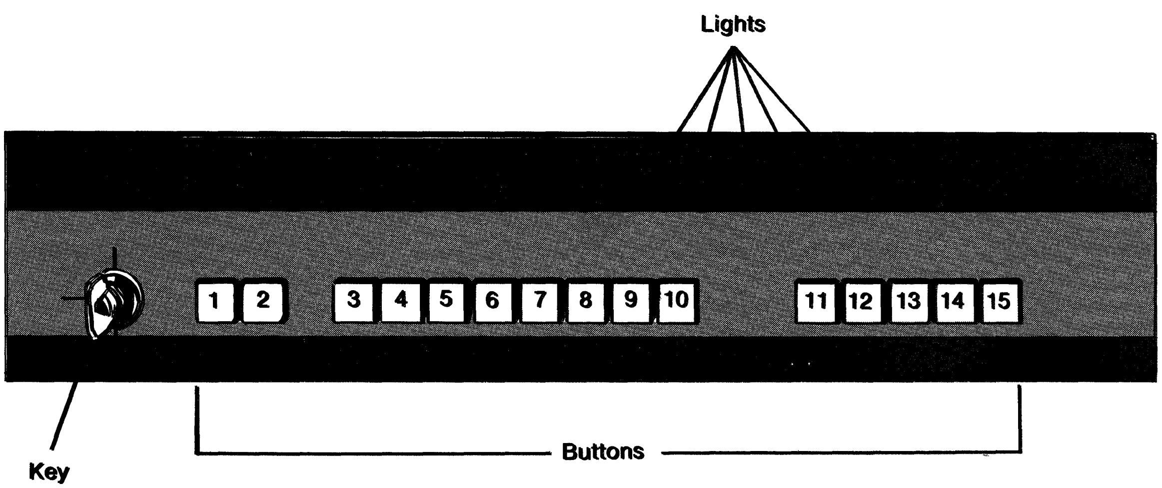
2. Remote ADC Programming — The ADC frequency range and input attenuator settings can be programmed by a User Program command when the ADC SAMPLE MODE switch is set to REMOTE. In addition, auto-scaling of the input attenuator settings may be invoked by another User Program.

OPERATION

Refer to the 5451 System Software Manual for regenerating the software to include the 5477A System Control Unit option. The following paragraphs provide operating information for the User Programs involved.

5477A System Control

The 5477A contains a turnkey, 5 indicator lights, and 15 pushbuttons; the positions of these are shown below:



When the key is in position 1, the unit is turned off; in position 2 the unit is on and the buttons are enabled; in position 3 the unit is on but the buttons are disabled.

In position 2 whenever a button is pressed, the system executes a JUMP instruction to a specific label in the keyboard program in Record 0, File 3. Exceptions to this are buttons 1 and 2. Button 1 executes a RESTART on the Fourier system. Button 2 is like a shift key in that it toggles the other pushbuttons between Mode 1 and Mode 2. Mode 1 is indicated by Light 1 being on; Mode 2 is indicated by Light 2 being on. Pressing Button 1 causes a RESTART in either case and places the unit in Mode 1.

Having two modes enables each button (3 through 15) to execute two JUMP commands. These jumps, using the stack-to-stack jump capability, are defined as follows.

Button	**Mode 1**	**Mode 2**
1	RESTART	RESTART
2	MODE SHIFT	MODE SHIFT
3	J 1003 0	J 2003 0
4	J 1004 0	J 2004 0
5	J 1005 0	J 2005 0
6	J 1006 0	J 2006 0
7	J 1007 0	J 2007 0
8	J 1008 0	J 2008 0
9	J 1009 0	J 2009 0
10	J 1010 0	J 2010 0
11	J 1011 0	J 2011 0
12	J 1012 0	J 2012 0
13	J 1013 0	J 2013 0
14	J 1014 0	J 2014 0
15	J 1015 0	J 2015 0

For example, pressing Button 10 in Mode 2 will execute a jump to Label 2010 in Record 0, File 3.

The User Programs associated with the 5477A are executed as follows.

USER PROG 5842 SPACE n1 ENTER (Enable Mode 1 or Mode 2)

where:

n1 = 1 for Mode 1 (or default)

n1 = 2 for Mode 2

USER PROG 5843 SPACE n1 ENTER (Enable/Disable Pushbuttons for Current Mode)

where:

n1 = 1 through 15 to enable single pushbuttons

n1 = –1 through –15 to disable single pushbuttons

n1 = 0 (default) to enable all buttons

USER PROG 5844 SPACE n1 ENTER (Turn On/Turn Off Lights)

where:

n1 = 1 through 15 to turn on single pushbutton lights or 16 through 20 for indicator lights 1 through 5 respectively

n1 = –1 through –15 to turn off single pushbutton lights or –16 through –20 to turn off indicator lights 1 through 5 respectively

n1 = 0 (default) to turn off all lights

Remote ADC Programming

The following User Programs exercise the Remote ADC Programming capability. Set the 5477A key-operated control to either one of its non-OFF positions (as required by the system in which it is operated). Set the ADC switches to the following positions:

SAMPLE MODE to REMOTE
OVERLOAD VOLTAGE (all) to 8

USER PROGRAM 1 — sets input range and sample rate control.

USER PROG 1 SPACE n1 SPACE n2 SPACE n3 ENTER

where:

n1 selects the channel to be programmed:

0 = channel A
1 = channel B
2 = channel C
3 = channel D
4 = all channels

n2 selects the input range setting (OVERLOAD VOLTAGE switches):

8 = 8V
4 = 4V
2 = 2V
1 = 1V
5 = 0.5V
25 = 0.25V
125 = 0.125V

n3 selects the frequency code (combined SAMPLE MODE & MULTIPLIER setting). Defaulting n3 maintains the code last entered. Note: See page 3-8 for table of frequency codes; 99 is not a valid code.

To disable remote programming and return to direct control, enter:

USER PROG 1 ENTER

USER PROGRAM 2 — rescales amplitude of sampled data between ½ and ¼ of full scale.

USER PROG 2 SPACE n1 ENTER

where:

n1 is the data block to be scaled. No default allowed.

This program checks the contents of data block n1 for points greater than ½ full scale. If found, the previous command (should be Analog In or Buffered Analog) is repeated with the input range (OVERLOAD VOLTAGE) set on position higher. This program simultaneously checks if all points are less than ¼ of full scale, in which case the previous command is repeated with the input range set one position lower.

A Y WHAT? error message will occur if the SAMPLE MODE switch is not set to REMOTE, or conversely, if the switch is in REMOTE but the user programs are not entered.

SECTION 8
DISC MISCELLANEOUS

INTRODUCTION

This section provides descriptions and operating instructions for the programs "NIBBL" and "DCOPY". These programs are used to read or write coreloads or partial cylinders to or from the Disc (NIBBL) or for copying the contents of one disc platter to another (DCOPY). Both are absolute programs that reside on every system software and relocatable software Disc. Note, however, that they are not contained on the Fourier Software Development System (FSDS) Disc.

Figure 8-1 is a flow diagram showing the sequence of operations for reading these programs from the Disc, and subsequently using them to access other software on the Disc. The Processor disc boot loader is used to read the program NIBBL into core. NIBBL in turn may be used to read the system software, DCOPY, or system diagnostics into core. Other operations besides these are possible but are normally not performed.

CAUTION

NIBBL and (especially) DCOPY are dangerous programs to use, in that both programs may, with certain inputs, write over system software and thereby destroy it. Make sure you have a thorough understanding of these programs before using them!

NIBBL

When run, NIBBL will read or write a full 32K coreload, or part of a cylinder, to or from a Disc.

CALLING SEQUENCE

Load NIBBL using the Processor disc boot loader. (Per Section 2, Loading System Software From the Disc.)

Set P-REG = 77600_8.

Set A-REG = 00DDSS octal

where:

DD = 00 for HEAD Ø

DD = 04 for HEAD 1

DD = 10 for HEAD 2

DD = 14 for HEAD 3

SS = Starting sector number (0-58)

Set B-REG = X00CCC octal

where:

X = 0 for read.

X = 1 for write.
(CAUTION: This mode should never be used in normal operation.)

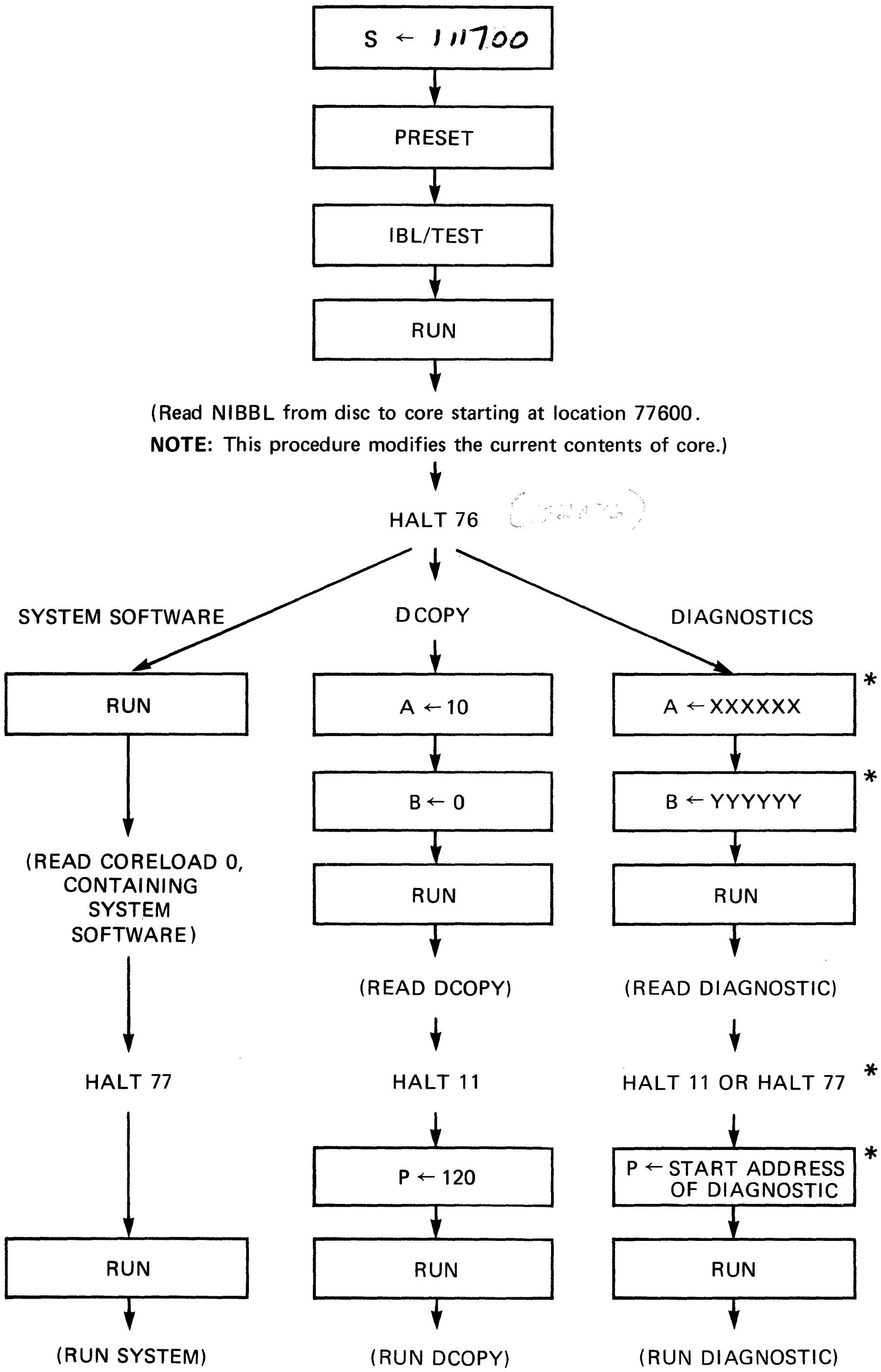
CCC = Starting cylinder number (0 - 620)

Press PRESET'S and RUN.

Figure 8-1. 5451C Disc Reading Procedure

The following items are accessed using this procedure:

NIBBL (On System and Relocatable Software Discs)
DCOPY (On System and Relocatable Software Discs)
Operating Software (On System Software Discs)
Diagnostics (On 5451C System Software Disc Only)



* Refer to System Service Manual for values of these quantities for various diagnostics.

If the operation was started at sector 0 of the designated cylinder (A-REG = 0 or 00DD00), then NIBBL will read or write an entire coreload (6 cylinders for 32K), and stop with a HALT 77_8. If a HALT 11_8 occurs, or nothing happens at all, an error has occurred, and must be fixed to proceed (at the HALT 11_8, the Disc status is in the A-REG). However, if the operation was begun at a non-zero sector (A-REG = 00DDSS, where SS is not equal to 0), then NIBBL will read the rest of the designated cylinder into core, starting at sector SS, and execute a HALT 11_8. Note that a HALT 11_8 is a normal end for this mode of use.

When the HALT 77_8 is encountered after reading a full coreload, pushing RUN will start the system at P = 2.

DCOPY

DCOPY is a stand-alone, system-independent utility which is used to perform disc to disc copies of 7900 series disc packs. You communicate with the program by answering questions on the Terminal. The following options are available:

CO COpy disc to disc with automatic verify.

BT Boot FSDS on lower (fixed) disc.

TE Terminate operations (HALT 77_8).

Several error halts are included in this program to insure proper hardware operation. Disc with cylinders flagged defective cannot be copied or dumped. Protected cylinders are copied as protected. DCOPY has the following capabilities:

1. Copy verification.
2. Disc time out loops occur (if no flag returned in set time period, a halt occurs).
3. Discs with cylinders flagged defective cannot be copied or dumped. (REFER TO DINIT INSTRUCTIONS AT END OF SECTION (ADDED BY THIS CHANGE).
4. Protected cylinders are copied as protected, but data protected packs are not copied.
5. All 400 cylinders are copied.
6. During disc copying, the destination disc is always initialized.
7. On the Processor, the switch register displays the octal track number of the present disc track being read (620 is last track).
8. The Boot ("BT") function allows an FSDS system to be copied to the lower platter (via the 'CO' function) and then booted up without having to use the Processor switch register.

CALLING SEQUENCE

To run DCOPY, perform the following steps and respond to the message output on the Terminal.

1. Ready all equipment. PUT THE 7906 FORMAT SWITCH IN ITS FORMAT POSITION.
2. Load and start DCOPY via the procedure of Figure 8-1.

The following messages can occur depending upon the operation chosen. The format of the message presentation shows the message in capital letters, its meaning, and the required response.

SELECT CO (COPY), C2 (COPY FILE2), BT (BOOT), OR TE (TERM.

Meaning:

Select the operation you want to perform.

Response:

The possible responses are:

CO Copy disc to disc (with auto verify).

BT Boot FSDS on lower (fixed) disc (note: this command will yield unpredictable results if the FSDS system is not on the fixed disc.

TE TErminate (halt 77_8).

SOURCE SUBCHANNEL # (0 = FIXED, 1 = REMOV)?

Copy from ? copy to

Meaning:

The program needs to know what disc subchannel is the source disc (i.e., the disc to be copied from).

Response:

Enter subchannel 0 or 1 (one ASCII number, press RETURN). The removable platter is 1 and the fixed platter is 0. (Typically, a response of '1' would be given to specify the removable disc as the source disc.)

DESTINATION SUBCHANNEL # ?

Meaning:

The program needs to know the destination channel for the disc to disc copy (i.e., the disc to be copied to).

Response:

Enter subchannel 0 or 1 (one ASCII number, press RETURN). The fixed platter is 0 and the removable platter is 1. (Typically, a response of 0 would be given to specify the fixed disc as the destination disc.)

CAUTION

The disc residing at the DESTINATION SUBCHANNEL will be altered after this response is entered.

Under normal operation, the Display Register will count to ~~Y~~ 620 If not, refer to Operational Halts following.

Unloading, loading, or opening the Disc door during a COPY operation may alter the Disc status, which will halt the Processor (see Operational Halts below). Avoid such changes except while the program is waiting for a response on the Terminal.

"The Fourier system manages defective tracks on the disc in two different ways. One method is used for the ADC Throughput file (File 2); the other method is used for the other files. When copying a disc, you should use the "C2" command to copy the Throughput area of the disc, and the "CO" command to copy other data. If there is a defective track on either the source or destination disc, use of the wrong command will result in an invalid copy.

EXAMPLES

The following examples copy a removable disc to a fixed disc.

SELECT 'CO' (COPY), 'BT' (BOOT), or 'TE' (TERM)

CO (your response)

SOURCE SUBCHANNEL (0 = FIXED, 1 = REMOV)?

1 (your response)

DESTINATION SUBCHANNEL # ?

0 (your response)

The removable disc is now copied to the fixed disc.

The following example is a printout resulting from the 'BT' function

SELECT 'CO' (COPY), 'BT' (BOOT), or 'TE' (TERM)

BT (your response)

(The following message or one similar to it is printed by the FSDS system):

```
:SV,1
TE
TE
TE
TE WELCOME TO THE FOURIER SOFTWARE DEVELOPMENT SYSTEM (REV XXXX)
TE
TE
TE
::
:
```

The FSDS system should now be operational.

OPERATIONAL HALTS

If a bad track is detected on the destination disc during a copying operation, the message:

```
BAD TRACK ON DESTINATION DISC
TRACK NUMBER IS IN A REGISTER
HEAD NUMBER IS IN B REGISTER
USE THE DISC DIAGNOSTIC AND DINIT TO INITIALIZE
PRESS RUN TO CONTINUE COPY
```

will be given, via the Terminal, followed by HALT11 (octal) (DISPLAY REGISTER bits 15, 10, 3, and 0 on, all others off). The Disc Diagnostic program and the program DINIT are described at the end of this manual section.

A HALT11 (octal) (bits 15, 10, 3, and 0 on) **without** the message shown above indicates that a hardware failure may have occurred. In this case:

- The A-Register contains the disc controller's status word 1, and —
- the B-Register contains the controller's status word 2.

The meanings of these status words are given in the Disc Controller Installation and Service Manual.

HALT	Meaning
HLT 00_8	Disc not ready.
HLT 03_8	Time-out on Disc Data Flag; Press RUN to get the status in the A-register (status 31_8 = defective cylinder).
HLT 05_8	Time-out on Disc Command Flag. Push RUN to get Disc Unit status in the A-register.
HLT 07_8	Disc SEEK error or data PROTECT error. Press RUN to get Disc status in the A-register.
HLT 10_8	Any Disc status bit set except 3 and 4 after a read or write (check OVERRIDE/PROTECT switch). Press RUN to obtain the status in the A-register.
HLT 13_8	Illegal status from Disc.
HLT 16_8	Bad cylinder status.
HLT 21_8	Bad data verify.
HLT 35_8	Cylinder count unequal to 203.
HLT 36_8	Incomplete data transfer for the current Disc cylinder being read or written.
HLT 55_8	The current cylinder is flagged as defective.
HLT 77_8	Orderly halt. Press RUN to continue.

A-Register Bit	Status	A-Register Bit	Status
0	Any Error	8	Seek Check
1	Data Error	9	Not Used
2	Drive Busy	10	Data Protect
3	Flagged Cylinder	11	Drive Unsafe
4	Address Error	12	Not Used
5	End of Cylinder	13	Overrun
6	Not Ready	14	First Status
7	Not Used	15	Not Used

DISC DIAGNOSTIC AND DINIT

If you purchase a blank 7906 disc cartridge from Hewlett-Packard, it may have a small number of defective tracks. The Fourier Analyzer can use this disc cartridge if it has been properly formatted and initialized. You should use the 7906 Disc Diagnostic Program to format and the program DINIT to initialize any disc cartridge that you plan to use in the 5451C other than those that came from the factory with software.

Diagnostic Calling Sequence

The 7906 Disc Diagnostic Program is on the Diagnostic Library Cartridge (Part Number 24396-13101). It can be loaded into the computer as follows.

Insert the Diagnostic Library Cartridge into the disc drive. Set the S register to 111710 (octal). Press STORE, PRESET, IBL/TEST, and RUN. The computer will halt and display 102077 (octal). Set the S register to 131711, the A register to 151403, and the B register to 0. Press PRESET and RUN. The computer will halt and display 102077. Set the P register to 100 and the S register to 17. Press RUN. The computer will halt and display 102074. Set the S register to 40011. Press RUN. The terminal will display:

```
H0  79XX/13037 DISC MEMORY DIAGNOSTIC
H37 UNIT TABLE: 01 DRIVE(S);0
H25 WISH TO CHANGE?
```

Type: NO

```
H55 ENTER INSTRUCTIONS (CURRENT UNIT = 0)
```

Remove the Diagnostic Library Cartridge from the disc drive.

Diagnostic Operation

1. Insert the cartridge to be formatted into the disc drive, ready disc and place the FORMAT switch in the up position (disc protect switches should always be down).

2. Type

   ```
   F0
   F1
   EN
   ```

   The program will format the cartridge and then look for bad tracks. If a bad track is found, the terminal will display:

   ```
   H22 VERIFY IN STEP 90
   H135  ....  (the text here depends on the nature of
   E64   ..... the error caused by the bad track
   H137  ..... and can be ignored when formatting)
   START xxx/yy/00 - . . . .
   ```

   (where xxx is the number of the bad track and yy is the number of the surface on which the track is located).

   When the cartridge has been checked, the terminal will display:

   ```
   H55 ENTER INSTRUCTIONS (CURRENT UNIT = 0)
   ```

3. Type

   ```
   SK, xxx, yy
   ID, , D
   EN
   ```

   (where xxx is the number of a bad track and yy is the number of the surface on which it is located).

   The program will mark this trace as defective then the terminal will display:

   ```
   H55 ENTER INSTRUCTIONS (CURRENT UNIT = 0)
   ```

   Repeat this step once for each bad track discovered in Step 2.

4. When all defective tracks have been flagged, remove the disc.

DINIT Calling Sequence

DINIT is on the Fourier Analyzer system software disc (part number 54451-10101 or 54451-10102 if the system has a Mag Tape unit). It can be loaded into the computer using NIBBL. Read NIBBL from the disc as shown in Figure 8-1. When the computer does a HALT 76, set the B register to 46 (octal) and the A register to 1. Press RUN. When the computer does a HALT 11, set the P register to 2. Press RUN. The terminal will display:

THIS PROGRAM INITIALIZES A DISC SURFACE
FOR USE IN THE FOURIER SYSTEM.

TYPE THE NUMBER OF A SURFACE
TO BE INITIALIZED (0, 1, 2, OR 3)

Remove the Fourier System cartridge from the disc drive.

DINIT Operation

Insert the cartridge to be initialized into the disc drive.

Type

0

The program will initialize the top surface of the cartridge. This means that it will find the tracks marked as defective by the Diagnostic and replace them with spare tracks. When the program is finished, the terminal will display:

TYPE THE NUMBER OF A SURFACE
TO BE INITIALIZED (0, 1, 2, OR 3)

Type

1

The program will initialize the bottom surface of the cartridge. When the above message is printed the initialization procedure is finished. Put the FORMAT switch in the down position.

APPENDIX A
ERROR MESSAGES

The WHAT? messages in Table A-1 are in response to the WHAT? light on the Keyboard panel and a corresponding code on the Terminal.

The system error halt codes in Table A-2 are the result of a premature Processor halt. The halt code represents the right-most octal digits in the S register ($1020xx_8$). Note that some halt codes are further defined in the A register as well.

The text messages in Table A-3 are printed on the Terminal in response to miscellaneous program errors.

Table A-1. System WHAT? Messages

Message	Source	Meaning
AD WHAT?	Y 40	ADC codeword error (check ADC settings)
A0 WHAT?	Y 5816	An illegal axis type has been entered (i.e., n < 1 or > 6).
A1 WHAT?	Y 5821	An illegal device type has been entered (i.e., n1 not 6, 10 or 35).
A2 WHAT?	Y 5821	The subunit number entered does not exist.
A3 WHAT?	Y 5812	Insufficient number of parameters or no parameters were passed.
A4 WHAT?	Y 5812	Vector is out of screen/plot limits.
A5 WHAT?	Y 5838	n1 is an illegal buffer number.
A6 WHAT?	Y 5810	n1, n3 < 0 or ≥ 10000.
A7 WHAT?	Y 5810	n4 is illegal.
A8 WHAT?	Y 5810	$n3 \times 10_{n4} \geq n1 \times 10_{n2}$.
A9 WHAT?	Y 5829	Illegal number of parameters.
BS WHAT?	System	n1 is an illegal blocksize.
BK WHAT?	Analog Out	Block number out of range.
BL WHAT?	Y 45	Block required for BSFA doesn't exist. Needs n1, n1 – 1 for single channel. n1, n1 + 1, n1 – 1, n1 – 2 for dual channel.
BW WHAT?	Y 40, Y 43	Bandwidth too large.
BZ WHAT?	Y 45	Current blocksize is larger than maximum standard BSFA blocksize (1024).
B0 WHAT?	Y 5829	n1, n2 ≤ 0, illegal block number, or n2 > n3.
B1 WHAT?	Y 5803	Disc text buffer number (n1) is < 0.
B2 WHAT?	Y 5803	Text ID n2 ≤ 0 or > 99.
B3 WHAT?	Y 5803	Error detected during special edit.
B4 WHAT?	Y 5829	Illegal message identification (< 01 or > 99).
B5 WHAT?	Y 5819	Message does not exist in the core-resident text buffer.
B6 WHAT?	Y 5815	Illegal frequency code.
B7 WHAT?	Y 5804, Y 5805, Y 5806, Y 5808	n1, n2 < 0; n1, n2 > 2000.
B8 WHAT?	Y 5803, Y 5838	Disc I/O is not configured.

Table A-1. System WHAT? Messages (cont'd)

Message	Source	Meaning
B9 WHAT?	Y 5803, Y 5838	Illegal disc buffer. Not allowed access.
CF WHAT?	Y 40, Y 43	If message occurs before printout of setup parameters, then center frequency is above F_{max}. If after, then CF and BW combine to put upper band edge above F_{max} — check filter setting.
CH WHAT?	Transfer Function	Blocks n1, n1 + 1, n1 + 2, n1 + 3, n1 + 4, n1 + 5 do not exist; n2 = 0 or not given.
C0 WHAT?	Y 5809	Number of parameters is > 4.
C1 WHAT?	Y 5809	Parameter is out of range.
C2 WHAT?	Y 5815	2 or 4 or more parameters are not allowed.
C3 WHAT?	Y 40	ADC set to 3 channels. Should be 1, 2, or 4.
C3 WHAT?	Y 5815	Start or end channel is out of range.
C4 WHAT?	Y 40	ADC set to 4 channels when 2-channel is max.
C4 WHAT?	Y 5803	Disc I/O error.
C5 WHAT?	Y 5817	n1 or n2 < 0, n4 > 0, n4 > 3.
C7 WHAT?	Y 5815	Illegal core-resident block number.
C8 WHAT?	Y 5815	Frequency Domain ORIGIN CENTER plot attempted.
C9 WHAT?	Gold EXP, Y 1827	Command does not have 3 parameters.
C9 WHAT?	Gold GET or PUT, Y 1821, Y 1822	0 parameters, invalid block or channel number.
C9 WHAT?	Gold QUALS, Y 1823	0 parameters, invalid block number.
C9 WHAT?	Y 1828	Not enough parameters, invalid block number.
DL WHAT?	Y 40	FPP data word rate too high. Reduce sample rate or use narrower bandwidth, or perform operation with no display.
DL WHAT?	Analog Out	Data output rate too fast. Could be caused by incorrect ADC setting, sample clock rate, or DAC A5 or A6 assemblies.
D6 WHAT?	Y 5839	n1 ≤ 0 or has been defaulted.
E3 WHAT?	Y 180n	Illegal VP number.
E4 WHAT?	Y 180n	No second parameter in VP call.
E5 WHAT?	Y 5842	Illegal shift mode.
E6 WHAT?	Y 5843	Illegal command number.
E7 WHAT?	Y 5844	Illegal lamp number.
E8 WHAT?	Y 1810	Four parameters must be entered.
FL WHAT?	Y 100	Incorrect filter range.
FL WHAT?	Analog Out	Incorrect filter range.
FQ WHAT?	Analog Out	Data block is not in time-domain-rectangular coordinates.
FS WHAT?	Y 45	The quantity: $2 \times F_{max} \times$ #ADC CHANNELS must be equal to or less than 250 kHz. Either reduce F_{max} or # of input channels.
F0 WHAT?	Jump	A jump command was made to a non-existent label in a disc-resident keyboard program.
F1 WHAT?	Subreturn	Nothing to return to (subroutine stack empty).

Table A-1. System WHAT? Messages (cont'd)

Message	Source	Meaning
G8 WHAT?	VP	Error in a command involving variable parameters (try the same command without variable parameters to determine the error).
H WHAT?	Y 141	MS header encountered with blocksize different from that found in last execution.
H1 WHAT?	Hanning	n1 does not exist or n1 is in the frequency domain.
H3 WHAT?	Y 1828	Block not in the frequency domain.
H5 WHAT?	Y 1828	Channel number too large for block.
HW WHAT?	Analog Out	Incorrect code word returned from the DAC; could be caused by incorrect ADC setting, the sample rate, or the DAC A5, A6 or A7 assemblies.
IF WHAT?	Skip	n1 does not exist; n2 is greater than blocksize for time domain data or 1/2 blocksize for frequency domain; command would skip over the last storage location available in the program stack.
IO WHAT?	Y 41, Y 43, Y 45	.IOC. status error on Mass Store throughput file.
J WHAT?	Jump	n1 is a label which does not exist or n1 is not a number.
L WHAT?	Label	Command given when not in programming mode.
L0 WHAT?	Count	n3 in a COUNT statement is a VP error.
L1 WHAT?	Y 1850	Command does not have 2 parameters.
L2 WHAT?	Y 1851	Command does not have 2 parameters.
L3 WHAT?	Y 1852	Command does not have 2 parameters.
L4 WHAT?	Y 1853	Command does not have 3 parameters.
L5 WHAT?	Y 1854	Command does not have 3 parameters.
L6 WHAT?	Gold Plot, Y 5800	n1 not valid.
L7 WHAT?	Gold Plot, Y 5800	n2 is invalid block number.
L7 WHAT?	Y 998	n1 (I/O select code) is less than 10_8 or greater than 77_8.
L8 WHAT?	Y 999	n1 (I/O select code) is less than 10_8 or greater than 77_8.
MS WHAT?	Mass Store	File number not in range 1-8; buffer size for MS file in FMTXX table = 0; illegal command; MS 26 is illegal number; Read/Write Data Block number too large; Write ADC Throughput ≠ 3 parameters or 0 throughput channels or 0 records; Read ADC Throughput record 0 or channel 0 requested; Position command record number less than 0; Search for Key not on files 1-4; ADC Transcribe need (NCHAN + 1) × SIZE data space; file 8 invalid command. Software not loaded?
M0 WHAT?	Y 45	Illegal Mass Store channel number (n3).
M1 WHAT?	Y 45	More than 6 different Mass Store channels were requested.
M2 WHAT?	Y 45	Next file header or end-of-session encountered.
M6 WHAT?	Y 142	Invalid floating point VP # given for input parameter n4.
M7 WHAT?	Y 142	Invalid integer VP # given for input parameter n2 or n3.
M8 WHAT?	Y 142	A value greater than 100 was given for input parameter n1.

SEE PAGE 6-68 FOR K ERRORS.

Table A-1. System WHAT? Messages (cont'd)

Message	Source	Meaning
M9 WHAT?	Y 45	Y 45 called with 5 parameters for processing 3 channels of throughput data. Only 1, 2 or 4 channels can be processed simultaneously with the optional hardware BSFA.
NB WHAT?	Y 43	Bandwidth 0 or too small overflow in bandwidth computation.
NG WHAT?	Y 43	Illegal negative parameter.
OV WHAT?	Y 41	Overflow during BSFA. Increase value for n4 or ADC attenuator setting.
OV WHAT?	System bootup	Invalid file 8 directory.
OV WHAT?	Analog Out	Max voltage value of block greater than 10V.
RA WHAT?	Analog In	SAMPLE MODE incorrectly set to REMOTE; blocksize too large for command; data block n1 does not exist or n1 + 1 does not exist in dual channel mode; data block n2 does not exist; number of ADC channels has changed in REPEAT mode.
RB WHAT?	Buffered Analog	n1 does not exist or n1 + 1 does not exist in multi-channel mode; not enough blocks for buffer blocks (one extra per channel); loop counter missing in program stack; n2 does not exist; SAMPLE MODE incorrectly set to REMOTE.
RH WHAT?	Histogram	Block 0 not in time domain; n1 does not exist; n1 = 0; channel n2 greater than blocksize.
RS WHAT?	RNTRY (Software Routine)	No more room in routine table (see System Software Manual for RNTRY).
SH WHAT?	Y 41	Shift parameter value (n4) illegal.
SK WHAT?	Power Spectrum	Microcode stack error — data blocks to be filled are not initialized or are altered during measurement.
SP WHAT?	Power Spectrum	n1 and n1 + 1 do not exist in single-channel command; n1, n1 + 1, n1 + 2, n1 + 3, n1 + 4 do not exist in dual-channel command; n1 = 0 in dual-channel command; n2 = 0 in single-channel command or $\neq$ 0 in dual-channel command.
T0 WHAT?	Y 40	The number of ADC input channels was changed during a buffered/overlapped BSFA measurement keyboard program.
T1 WHAT?	Y 45	The BSFA blocksize is too big. Must have blocksize $\leq$ 1024 for software BSFA (Y 41), or blocksize $\leq$ 2048 for hardware BSFA (Y 40 or Y 43).
X0 WHAT?	Y 5864	More than one parameter has been entered.
X1 WHAT?	Y 5864	n1 is not 1, 0, or -1.
X2 WHAT?	Y 5865	More than one parameter has been entered.
X3 WHAT?	Y 5865	Invalid parameter passed.
Y WHAT?	User Programs	n1 not given or program named n1 does not exist in system.
Y WHAT?	Y 1	Command $\neq$ 0, 2, or 3 parameters; invalid overload voltage code; more than 4 channels; frequency code not returned by Keyboard when programmed; ADC not in REMOTE.
Y WHAT?	Y 2	Command $\neq$ 1 parameter.
Y WHAT?	Y 40, Y 41	Incorrect Keyboard entry; center frequency out of range; previously displayed block no longer available.

Table A-1. System WHAT? Messages (cont'd)

Message	Source	Meaning
Y WHAT?	Y 44	Incorrect Keyboard entry or n2, n3 n4 result in request to display non-existent portion of block n1 (illegal horizontal calibration).
Y WHAT?	Y 45	Incorrect Keyboard entry or program not properly initialized.
Y WHAT?	Y 100	More than 3 parameters entered or n2 greater than 15.
Y WHAT?	Y 117	Command $\neq$ 2 parameters.
Y WHAT?	Y 141	Incorrect Keyboard entry.
Y WHAT?	Y 1818	No parameters or more than 72 characters requested.
Y WHAT?	Y 3017	Incorrect Keyboard entry; illegal block n1; blocksize greater than 2048.
Y WHAT?	Y 3024	Number of decades $\neq$ 1 through 5.
Y1 WHAT?	Y 1829, Y 1830	Incorrect command entry for n1, n2; n2, n3 too large (outside header); illegal VP transferred.
Z0 WHAT?	Y 1817	Less than 2 parameters entered.
Z1 WHAT?	Y 1817	Invalid parameter; invalid VP n1; n2 $<$ 0 or $>$ 15; n3 $<$ 1 or $>$ 16; n4 $<$ 0 or $>$ 1; n5 $<$ 0 or $>$ 1.
Z2 WHAT?	Y 1817	Cannot write to display word.
Z4 WHAT?	Y 6	No parameters entered.
Z5 WHAT?	Y 6	n1 $\neq$ 1 through 4 or n2 is negative.
Z6 WHAT?	Y 3025	No parameters entered.
Z7 WHAT?	Y 3025	Parameter not 0-77.
Z8 WHAT?	Y 3025	Digit in parameter not 0-7.
Z9 WHAT?	Y 3025	Driver, unit not found in EQT.
40 WHAT?	Y 5840	No parameters entered.
.WHAT?	End	Command entered when system not in BUSY mode or CONTINUE pressed at end of a keyboard program.
/ WHAT?	Terminate	Command entered when system not in BUSY mode.
/D WHAT?	Delete	n1 or n2 greater than max stack size.
/L WHAT?	List	n1 or n2 greater than max stack size.
/I WHAT?	Insert	n1 too large for max program stack size or is last command but not an END command.
/R WHAT?	Replace	n1 or n2 greater than max stack size.
? WHAT?	Pointer	n1 greater than max stack size.
/. WHAT?	Cursor	No D.41 in EQT or invalid block n2.
# WHAT?	Count	Label n1 does not exist.
÷ WHAT?	Divide	n1 does not exist or n2 = 0 and not given from inside a count loop.
@ WHAT?	Y 45	Next file header or end-of-session encountered.

Table A-2. System Error Halt Codes

Octal Code	Program	Meaning
00	Graphics	Y 5821 has not been executed to specify the plot device.
00	DCOPY	Disc not ready.
00	I/O instruction	Check A register bits: 0 = .IOC. request code error 1 = .IOC. ordinal error 2 = .IOC. SQT entry error
01	System startup	Word written to memory not read back intact (run memory diagnostic).
02	System startup	Less than 36 contiguous 1K memory pages, either too little memory or incorrect memory board jumpering.
03	DCOPY	Time-out on Disc data flag — see Section 8 for more detail.
05	DCOPY	Time-out on Disc command flag — see Section 8 for more detail.
06	ADC	DMA 6 unavailable.
07	MIOB	No DMA channel available.
07	MS 36, MS 16	Driver D.34 for Disc Drive not in system.
07	DCOPY	Disc seek or data protect error — see Section 8 for more detail.
10	DCOPY	Any Disc status bit set except 3 and 4 after a read or write command (check OVERRIDE/PROTECT switch). See Section 8 for more detail.
10	MIOB	Unable to switch MIOB device to command mode.
11	System startup	Power-up to an MIOB device did not work.
11	NIBBL	OK if reading one track ($A \neq 0$); otherwise bad disc status — see Section 8 for more detail.
11	Core dump	Check A register bits: 1 = read/write parity error 3 = bad track, cylinder cannot be used 6 = drive not ready 10 = DATA PROTECT lighted 14 = repeat dump procedure
13	MIOB	Illegal codeword (in A register) for non-existent MIOB device.
13	DCOPY	Illegal status from Disc.
16	DCOPY	Bad cylinder status.
21	DCOPY	Bad data verify.
35	DCOPY	Cylinder count $\neq$ 203.
36	DCOPY	Incomplete data transfer for the current disc cylinder being read or written.
55	DCOPY	Current cylinder is flagged as defective.
77	DCOPY, NIBBL	Orderly halt, press RUN to continue.

Table A-3. Text Messages

Message	Meaning/Remedy
ADC DMA BUSY	Press RESTART.
BLOCKSIZE TOO BIG	Reduce blocksize (throughput requires $2 \times BS \times$ no. of channels ≤ 4096).
END OF TAPE OR FILE	Record number larger than max allowable in FMTXX table.
FILE PROTECTED	If Disc, turn off protect switch for upper or lower unit; if Mag Tape, install write enable ring for writing operation.
ILLEGAL BUFFER (Y 5803)	Data in the text buffer just read is not valid.
I/O ERROR	Repeat, run diagnostics (check that device is in READY mode).
NO HEADER	Incorrect positioning in ADC Throughput file.
NOT REAL TIME	ADC sample rate too fast for real-time (no points lost) throughput.
SIZE NOT =	Set blocksize = blocksize during ADC throughput.
UNIT OFF LINE	Place Disc or Mag Tape on line.
WARNING — CLEAR NEW DISC BUFFERS (Y 5803)	Extraneous data may be present in the disc text buffer just accessed.
??? (Y 5803)	Editing commands not understood by system.

APPENDIX B
ALPHABETICAL INDEX OF KEY DESCRIPTIONS

This appendix contains an alphabetical index of the key descriptions contained in Section 3. Table B-1 lists each key alphabetically by function followed by an illustration showing the key exactly as it appears on the front panel of the 5475A Control Unit. In addition, the symbol that is printed out for each key is shown. When a key has more than one function (e.g., KEYBOARD/PLOT), each is listed separately. The page in Section 3 that contains the complete key description is listed in the final column.

Table B-1. Alphabetical Index

Function	Key	Symbol	Page	Function	Key	Symbol	Page
Add*	ADD +	Y A+	3-29	Block Size	BLOCK SIZE	BS	3-77
Analog In	ANALG IN	RA	3-2	Block Size*	BS BLOCK SIZE	Y BS	3-77
Analog Output	ANALG OUT	B	3-20	Block Subtraction	−	A−	3-29
Block Addition	+	A+	3-29	Buffered Analog	BUFFD ANALG	RB	3-3
Block Conjugate Multiply	* MULT	*_	3-32	Clear	CLEAR	CL	3-25
Block Division	÷	:	3-34	Complex Multiply	MULT	*	3-31
Block Multiply	MULT	*	3-30	Conjugate	* MULT	*_	3-33
Block Shift	←	—	3-28	Continue	CONTINUE		3-83

*Variable Parameter or User Program

Table B-1. Alphabetical Index (cont'd)

Function	Key	Symbol	Page	Function	Key	Symbol	Page
Convolution	CONV	CV	3-50	F1*	F1 CORR	Y CR	3-50
Correlation	CORR	CR	3-42	F2*	F2 TRANS FCN	Y CH	3-63
Count	COUNT	#	3-74	F3*	F3 F	Y F	3-42
Cursor	CURSR	/.	3-80	F4*	F4 CONV	Y CV	3-55
Delete	DELET	/D	3-67	F5*	F5 POWER SPECT	Y SP	3-59
Differentiation	d/dx	%	3-39	F6*	F6 HISTO GRAM	Y RH	3-57
Display	DSPLY	D	3-17	Get*	GET LOAD	Y X<	3-26
Division*	÷	Y :	3-37	Histogram	HISTO GRAM	RH	3-55
End	DIV END	•	3-76	Input*	INPUT PHOTO READR	Y R	3-12
Exponentiate*	EXP * MULT	Y *_	3-33	Insert	INSRT	/I	3-67
Fourier Transform	F	F	3-40	Integer Divide	÷	:	3-34

**Variable Parameter or User Program*

Table B-1. Alphabetical Index (cont'd)

Function	Key	Symbol	Page
Integer Multiply	MULT	*	3-31
Integration	∫	$	3-38
Interchange	INTER CHNG	X	3-27
Interval-centered Hanning	HANN	H1	3-27
Jump	JUMP	J	3-70
Keyboard	KEY BOARD	K	3-7
Label	LABEL	L	3-70
List	LIST	/L	3-68
List*	LIST LIST	Y /L	3-69
Load	LOAD	X<	3-25
Log Magnitude	LOG MAG	TL	3-22
Log*	LOG MAG LOG	Y TL	3-24
Loop Counter Divide	÷	:	3-33
Loop Counter Multiply	MULT	*	3-30
Multiply*	MULT MULT	Y *	3-32
Mass Store	MASS STORE	MS	3-19
Output*	PUNCH OUTPUT	Y P	3-17
Photoreader	PHOTO READR	R	3-12
Plot*	PLOT KEY BOARD	Y K	3-11
Pointer	POINT	?	3-69
Polar Coordinates	POLAR	TP	3-21
Power Spectrum	POWER SPECT	SP	3-57

* *Variable Parameter or User Program*

Table B-1. Alphabetical Index (cont'd)

Function	Key	Symbol	Page
Print	PRINT	W	3-13
Punch	PUNCH	P	3-16
Put*	PUT STORE	Y X>	3-26
Qualifiers*	QUALS RECT	Y TR	3-22
Rectangular Coordinates	RECT	TR	3-21
Replace	RPLAC	/R	3-65
Restart	RE START		3-84
Rotate	SET SET ←	—	3-28
Skip	SKIP	IF	3-75
Skip*	SKIP SKIP	Y IF	3-76
Stop	STOP		3-83
Store	STORE	X>	3-26
Subtract*	SUB –	Y A–	3-30
Subroutine Return	SUB RTRN	<	3-76
Terminate	TERM	/	3-69
Text*	TEXT PRINT	Y W	3-16
Transfer Function	TRANS FCN	CH	3-59
User Program (Gold Key)	USER PROG	Y	3-78,79

* Variable Parameter or User Program

APPENDIX C
SAMPLE PROGRAMS

INTRODUCTION

This Appendix provides a number of typical sample programs, plus a discussion of the considerations that went into the development of each.

For details on how programs are entered into the Fourier Analyzer, see Section 2.

SPECTRUM AVERAGING PROGRAMS

The following remarks apply to the first four sample programs. These are the auto spectrum summation average program, cross power spectrum summation average, auto spectrum stable average, and auto spectrum exponentially decaying average. All these programs consist of power spectra (or linear spectra) averaged in different ways.

POWER SPECTRUM

To compute a power spectrum of a random process, it is necessary to generate a positive quantity which can be averaged. The resulting average will then be a measure of the power in each frequency of the spectrum band.

The typical random process yields a Fourier transform which will have positive and negative real, and positive and negative imaginary values, randomly distributed across the spectrum. Thus, if the transform is averaged without a time synchronization, the results will average out to zero. Therefore, to obtain the required positive quantities, the Fourier transform must be conjugate multiplied, yielding a positive quantity at each frequency which is the power at that frequency. If we sum a number of such spectra, and then divide each frequency channel by that number, we achieve the desired average.

The Processor uses 16-bit data words (15 bits plus sign) thus providing a range of 0 to 65,534 numbers (−32,768 to +32,767). 65,534 is about 4.8 decades. If we compute the linear spectrum, we have 20 dB/decade times 4.8 or 96 dB dynamic range. With the power spectrum we have 10 dB/decade times 4.8 or only 48 dB dynamic range. However, since the result of a power spectrum is in double precision, the same dynamic range is maintained.

If the function is a Gaussian process, then the real and imaginary components of the Fourier transform are a Gaussian random quantity, each frequency component having a value expressed by: a+ib, where a is the value of the real component, and b the value of the imaginary. When we square this expression by conjugate multiplication, we arrive at an expression of the form $a^2 + b^2$, whose statistics are Chi-squared with a degree of freedom for each squared term, or in other words, with two degrees of freedom. Thus, each time record yields one estimate of a power spectrum with two degrees of freedom. For the first estimate, the variance on the expected value of each spectral line is equal to the mean value, which is a very large statistical variation. However, the variation decreases as the number of estimates increases. That is:

$$\sigma = \frac{1}{\sqrt{K}} \times 100$$

where:

σ is one standard deviation in percent.

K is the number of estimates.

The same programming strategy is employed for all four spectrum averaging programs. This strategy consists in reading data into block 0, forming the power spectrum there, and reserving block 1 to store the accumulating sums of the estimates. Then the contents of block 1, which are 0 the first time, are added to block 0. Thus, the current record is always added to the sum of the past records. The result is stored back into block 1, and a new record obtained in block 0. The current record is worked on in block 0 because in all arithmetic operations involving two blocks, one of them must be block 0.

A feature of the ANALOG INPUT command is that it permits you to take in data in one block (or two in the case of dual channel input) and display a different block. In the case of spectrum averaging programs, this permits you to observe the sum accumulating in the storage block, which is of more interest than observing the input block, since this would merely show each record of a random process.

When running the summation average programs, you will observe the sums accumulating in the storage block. If the program is stopped in mid-run, a calibrated average up to that point cannot be read, since the division takes place at the end. However, you can compute the calibrated average by first listing the COUNT line of the program which will also give the number of sums up to that point. Then the block can be manually divided by that number to give a calibrated average.

The stable and the decaying average programs provide a calibrated average at every repetition, and thus no additional operations are required when the program is stopped in mid-run.

LINEAR SPECTRUM

To accumulate a linear spectrum rather than a power spectrum, use a POLAR COORDINATE command in place of the CONJUGATE MULTIPLY command. The result will be the magnitude values in the real part of the data block, and the phase in the imaginary. The magnitudes then sum as repeated spectral estimates are taken and stored. The phase values will randomly sum and so must be disregarded. The major disadvantage of a linear spectrum (as opposed to a power spectrum) is that it takes longer to form. This is because the polar coordinate operation takes longer than conjugate multiplication.

SUMMATION AVERAGE

The summation average is the simplest and fastest average, because it consists solely of summations, followed by a division at the end. When a stable or exponentially decaying power spectrum average is performed using the POWER SPECTRUM key, the averaging is done in microcode and is, therefore, actually faster than summation averaging. The following programs are presented only as examples of how data can be manipulated with keyboard programs.

$$A_n = \frac{\sum_{1}^{n} I_i}{n}$$

where:

A_n = the average after n estimates

I_i = the i^{th} estimate

n = the number of estimates

However, as mentioned above, it has the disadvantage of not being calibrated at all times. Also, the display changes very rapidly as the summation grows. The stable average on the other hand, is always calibrated, and the display does not change so rapidly. But it is also slower than the summation average.

SPECTRUM STABLE AVERAGE

The spectrum stable average is computed from the following algorithm:

$$A_n = A_{n-1} + \frac{I_n - A_{n-1}}{n}$$

where:

A_n = the average after n estimates

A_{n-1} = the average after n–1 estimates

I_n = the n^{th} estimate

n = the number of estimates

Although the sample programs which employ this algorithm are for spectrum averaging, the algorithm can be applied to any averaging process, for example time averaging. The algorithm is derived as follows. The previous, i.e., n–1 average, A_{n-1} is:

$$A_{n-1} = \frac{\sum_{1}^{n-1} I_i}{n-1}$$

Then, multiplying:

$$(A_{n-1})(n-1) = \sum_{1}^{n-1} I_i$$

If we now add the nth estimate, I_n, to both sides, we get:

$$(A_{n-1})(n-1) + I_n = \sum_{1}^{n-1} I_i + I_n$$

Dividing both sides by n:

$$\frac{(A_{n-1})(n-1) + I_n}{n} = \frac{\sum_{1}^{n-1} I_i + I_n}{n}$$

But the right hand side is A_n, the average after n estimates. And, rewriting the left hand side, we get:

$$\frac{(A_{n-1})\cancel{n}}{\cancel{n}} - \frac{A_{n-1}}{n} + \frac{I_n}{n} = A_n$$

Further rewriting:

$$A_n = A_{n-1} + \frac{I_n - A_{n-1}}{n}$$

The program can thus be stopped at any time, and the average in block 1 will always be calibrated. The price of this convenience is the additional processor time required to do the subtraction, division, and addition after each estimate.

EXPONENTIALLY DECAYING AVERAGE

In a stable average, all ensemble estimates contribute equally to the final average. This is satisfactory if the signal as a whole is not changing with time. But if it is changing compared to one sample record, we may want to see what the average is over a short period: in other words, we may want to look at an average that approaches the final value exponentially as $(1-e^{-n/k})$. That is:

$$A_n = A_f (1 - e^{-n/k})$$

where:

A_n = the average after n estimates

A_f = the final average

n = the number of estimates

k = the weighting factor

It is clear that A_n is a close approximation to A_f only as n grows large compared to k.

The derivation of the above equation from the running average algorithm is as follows:

$$A_n = A_{n-1} + \frac{I_n - A_{n-1}}{k}$$

Therefore

$$A_1 = A_0 + \frac{I_1 - A_0}{k} = \left(\frac{k-1}{k}\right) A_0 + \frac{I_1}{k}$$

And

$$A_2 = \left(\frac{k-1}{k}\right) A_1 + \frac{I_2}{k} = \left(\frac{k-1}{k}\right)^2 A_0 + \frac{k-1}{k^2} I_1 + \frac{I_2}{k}$$

But A_0, the average after 0 estimates, is 0. Therefore, by induction,

$$E\left[A_n\right] = E\left[\frac{1}{k} \sum_{i=1}^{n} I_i \left(\frac{k-1}{k}\right)^{n-i}\right]$$

where:

E means "the mean value of . . ."

I_i = the i^{th} estimate

$$E\left[A_n\right] = \frac{1}{k} \sum_{i=1}^{n} E\left[I_i \left(\frac{k-1}{k}\right)^{n-i}\right]$$

$$E\left[A_n\right] = \frac{1}{k} \sum_{i=1}^{n} \left(\frac{k-1}{k}\right)^{n-i} E\left[I_i\right]$$

$$E\left[A_n\right] = E\left[I_i\right] \left(\frac{k-1}{k}\right)^{n} \frac{1}{k} \sum_{i=1}^{n} \left(\frac{k-1}{k}\right)^{-i}$$

But since

$$\sum_{i=0}^{n-1} ar^{i} = a\left(\frac{r^{n} - 1}{r - 1}\right)$$

and letting

$$a = 1,\ r = \left(\frac{k-1}{k}\right)^{-1} = \frac{k}{k-1},$$

$$\sum_{i=1}^{n} \left(\frac{k-1}{k}\right)^{-i} = \frac{\left(\frac{k}{k-1}\right)^{n+1} - \frac{k}{k-1}}{\frac{k}{k-1} - 1}$$

$$E\left[A_n\right] = E\left[I_i\right] \frac{1}{k} \left(\frac{k-1}{k}\right)^{n} \left[\frac{\left(\frac{k}{k-1}\right)^{n+1} - \frac{k}{k-1}}{\frac{k}{k-1} - 1}\right]$$

$$E\left[A_n\right] = E\left[I_i\right] \frac{1}{k} \left[\frac{1 - \left(\frac{k-1}{k}\right)^{n}}{1 - \frac{k-1}{k}}\right]$$

$$E\left[A_n\right] = E\left[I_i\right] \left[1 - \frac{k-1}{k}^{\,n}\right]$$

For k very large,

$$E\left[A_n\right] = E\left[I_i\right]\left(1 - e^{-n/k}\right)$$

For sufficiently large n, E (A_n) approaches A_f. Thus:

$$A_n = A_f\left(1 - e^{-n/k}\right)$$

The exponentially decaying average like the stable average will always produce a calibrated result if the program is stopped in mid-run.

The effective number of averages contained in the ensemble is k. Thus, each spectrum in an exponentially decaying average has 2k degrees of freedom.

Auto Spectrum Summation Average Program

Program Commands	Contents of Block 0	Contents of Block 1	Purpose of Command
LABEL 0 ENTER			Establishes initial label point
CLEAR 1 ENTER		Cleared	Initializes block 1 (removes old data)
LABEL 1 ENTER		Sum of past power spectra $G_{xx}(f)$. 0 first time.	Establishes target point for ensemble average
ANALOG IN 0 SPACE 1 ENTER	Current time record	"	Reads current record into block 0, displays block 1, where sums are accumulating
F ENTER	Fourier transform of current record	"	Fourier transforms block 0
M$\overset{*}{\text{U}}$LT ENTER	$G_{xx}(f)$ of current record	"	Forms $G_{xx}(f)$ of current record
+1 ENTER	Sum of current plus past $G_{xx}(f)$'s	"	Adds current $G_{xx}(f)$ to sum of past $G_{xx}(f)$'s
STORE 1 ENTER	"	Sum of current plus past $G_{xx}(f)$'s	Stores sum of current plus past $G_{xx}(f)$'s in block 1 for next pass
COUNT 1 SPACE n2 ENTER	"	"	Loop back to target label 1 n2 times, for n2 sums
÷1 SPACE n2 ENTER	"	Average of n2 $G_{xx}(f)$'s	Forms average of n2 $G_{xx}(f)$ sums
END ENTER	"	"	Ends program

Linear Spectrum Summation Average Program

Program Commands	Contents of Block 0	Contents of Block 1	Purpose of Command
LABEL 0 ENTER			Establishes initial label point
CLEAR 1 ENTER		Cleared	Initializes block 1 (removes old data)
LABEL 1 ENTER		Sum of past linear spectra. 0 first time.	Establishes target point for ensemble average
ANALOG IN 0 SPACE 1 ENTER	Current time record	"	Reads current record into block 0, displays block 1, where sums are accumulating
F ENTER	Fourier transform of current record	"	Fourier transforms block 0
POLAR ENTER	Current linear spectrum in polar coordinates	"	Convert linear spectrum to polar coordinates (magnitude and phase)
+1 ENTER	Sum of current plus past linear spectra	"	Forms sum of current plus past linear spectra in block 0
STORE 1 ENTER	"	Sum of current plus past linear spectra	Stores sum of current plus past linear spectra in block 1 for next pass
COUNT 1 SPACE n2 ENTER	"	"	Loop back to target label 1 n2 times, for n2 sums
÷0 SPACE n2 ENTER	Average of n2 linear spectra sums	Sum of current plus past linear spectra	Forms average of n2 linear spectra
END ENTER	Average of n2 linear spectra sums	"	Ends program

Cross Spectrum Summation Average Program

Program Commands	Contents of Block 0	Contents of Block 1	Contents of Block 2	Purpose of Command
LABEL 0 ENTER				Establishes initial label point
CLEAR 2 ENTER			Cleared	Initializes block 2 (removes old data)
LABEL 1 ENTER			"	Establishes target point for cross power spectrum $G_{yx}(f)$ sums
ANALOG IN 0 SPACE 1 ENTER	Current time record from channel A	Current time record from channel B	Sum of past $G_{yx}(f)$'s. 0 first time.	Data input
F 0 SPACE 1 ENTER	Fourier transform of channel A time record	Fourier transform of channel B time record	"	Obtain Fourier transform of data
INTERCHANGE 1 ENTER	Fourier transform of channel B time record	Fourier transform of channel A time record	"	Interchange channels A and B
MULT* 1 ENTER	$G_{yx}(f)$ of channel A and channel B records	"	"	Obtain cross power spectrum
+ 2 ENTER	Sum of current plus past $G_{yx}(f)$'s	"	"	Adds current $G_{yx}(f)$ to sum of past $G_{yx}(f)$'s
STORE 2 ENTER	"	"	"	Stores result of current pass for next pass
COUNT 1 SPACE n2 ENTER	"	"	"	Loop back to target label 1 n2 times for n2 sums
÷ 0 SPACE n2 ENTER	Average of n2 $G_{yx}(f)$'s	Fourier transform of channel B time record	Sum of past $G_{yx}(f)$'s. 0 first time.	Form average of $G_{yx}(f)$'s
END ENTER	"	"	"	Ends program

Auto Spectrum Stable Average Program

Program Commands	Contents of Block 0	Contents of Block 1	Purpose of Command
LABEL 0 ENTER			Establishes initial label point
CLEAR 1 ENTER		Cleared	Initializes block 1 (removes old data)
LABEL 1 ENTER		True average of past power spectra $G_{xx}(f)$'s. 0 first time.	Establishes target point for stable average
ANALOG IN 0 SPACE 1 ENTER	Current time record	"	Reads current record into block 0, displays block 1, where average is accumulating
F ENTER	Fourier transform of current record	"	Fourier transforms block 0
$\overset{*}{\text{MULT}}$ ENTER[1,2]	$G_{xx}(f)$ of current record	"	Forms $G_{xx}(f)$ of current record
− 1 ENTER[2]	Difference between current $G_{xx}(f)$ and average of past $G_{xx}(f)$'s	"	Subtract average of past $G_{xx}(f)$'s (in block 1) from current $G_{xx}(f)$ in block 0
÷0 SPACE 0 ENTER[2]	Result of dividing above difference by number of $G_{xx}(f)$'s averaged so far.	"	Divides the variance between current $G_{xx}(f)$ and past $G_{xx}(f)$'s by the loop counter, i.e., number of $G_{xx}(f)$'s averaged so far
+ 1 ENTER[2]	Sum of above division result and average of past $G_{xx}(f)$'s	True average of past power spectrums $G_{xx}(f)$'s. 0 first time.	Adds division result to average of past $G_{xx}(f)$'s to obtain new average
STORE 1 ENTER[2]	Sum of above division result and average of past $G_{xx}(f)$'s (i.e., new average)	Same as block 0 (i.e., new average)	Stores result of current pass for next pass
COUNT 1 SPACE n2 ENTER	"	"	Loop back to label 1 n2 times, for n2 averages
END ENTER	"	"	Ends program

[1] Replace this command with POLAR ENTER if you want to accumulate the average of voltage rather than power spectra.
[2] The series of steps identified by the symbol at left performs the same operation as a single POWER SPECT ENTER command.

Auto Spectrum Exponentially Decaying Average Program

Program Commands	Contents of Block 0	Contents of Block 1	Purpose of Command
LABEL 0 ENTER			Establishes initial label point
CLEAR 1 ENTER		Cleared	Initializes block 1 (removes old data)
LABEL 1 ENTER		True average of past power spectrums $G_{xx}(f)$'s. 0 first time.	Establishes target point for ensemble average
ANALOG IN 0 SPACE 1 ENTER	Current time record	"	Reads current record into block 0, displays block 1, where average is accumulating
F ENTER	Fourier transform of current record	"	Fourier transforms block 0
*MULT ENTER[1,2]	$G_{xx}(f)$ of current record	"	Forms $G_{xx}(f)$ of current record
- 1 ENTER[2]	Difference between current $G_{xx}(f)$ and average of past $G_{xx}(f)$'s	"	Subtract average of past $G_{xx}(f)$'s (in block 1) from current $G_{xx}(f)$ in block 0
÷ 0 SPACE K ENTER[2]	Result of dividing above difference by time constant, K	"	Divides the variance between current $G_{xx}(f)$ and past $G_{xx}(f)$'s by time constant, K
+ 1 ENTER[2]	Sum of above division result and average of past $G_{xx}(f)$'s	True average of past power spectrums $G_{xx}(f)$'s. 0 first time.	Adds division result to average of past $G_{xx}(f)$'s to obtain new average
STORE 1 ENTER[2]	Sum of above division result and average of past $G_{xx}(f)$'s (i.e., new average)	Same as block 0 (i.e., new average)	Stores result of current pass for next pass
COUNT 1 SPACE n2 ENTER	"	"	Loop back to Label 1 n2 times, for n2 averages.
END ENTER	"	"	Ends program

[2] NOTE

[1] Replace this command with POLAR ENTER if you want to accumulate the average of voltage rather than power spectra.

[2] The series of Program Commands identified by this symbol performs the same operation as a single POWER SPECT ENTER command.

NOTE

When the POWER SPECT ENTER command is used, your program should include a USER PROG 3016 SPACE n1 ENTER command, where n1 specifies "K", the time constant for the average.

TIME ENSEMBLE AVERAGE

If a signal is buried in noise, and a time synchronization is available so that you know the periodicity of the signal, the time averaging program is an efficient means of recovering the desired signal either in the time or spectrum domain. This is because the Fourier transform and subsequent conjugate multiply (or conversion to polar coordinates in the case of a linear spectrum) is only done once, namely after the time average is completed. During the time average the signal always has the same value in each ensemble, but the noise having random plus and minus values will average out.

Time Ensemble Average Program
(Note: A time synchronization is required for this program.)

Program Commands	Contents of Block 0	Contents of Block 1	Purpose of Command
LABEL 0 ENTER			Establishes initial label point
CLEAR 1 ENTER		Cleared	Initializes block 1 (removes old data)
LABEL 1 ENTER		Past sum. 0 first time.	Establishes target point for summation
ANALOG IN 0 SPACE 1 ENTER	Current time record	"	Reads current record into block 0, displays block 1, where sums are accumulating
+1 ENTER	Sum of current record plus past sums	"	Forms sum of current record plus past sum in block 0
STORE 1 ENTER	"	Sum of current record plus past sum	Stores new sum in block 1 for next pass
COUNT 1 SPACE n2 ENTER	New sum	New sum	Loop back to target label 1 n2 times for n2 sums
÷0 SPACE n2 ENTER	Average of n2 sums	"	Forms average of n2 sums
END ENTER	"	"	Ends program

NORMALIZING A HISTOGRAM TO UNIT AREA

In the program to normalize a histogram to unity, we see how imaginative use of the keys can create very flexible operations. The program runs as follows: The histogram is first integrated so that the total number of counts, n, will be in the last channel. Then the entire block is shifted so that the last channel is in the first channel and the remainder of the block is then cleared out. Then the entire block is again integrated, resulting in a function which is flat all the way across the block. Thus, we have the number n in all channels of the block. We then divide the original histogram, which has remained in another block, by this block which means that the number of counts in each channel, n_i, are divided by n.

$$\text{Normalized histogram} = \sum_{0}^{N} \frac{n_i}{n} = \frac{1}{n} \sum_{0}^{N} n_i$$

This technique of integrating, shifting, and clearing can be used to normalize a spectrum or any other function to the unity value for its integral. Alternatively, the gold BS, gold GET, and integer division commands may be used to obtain the number of counts in the last channel of the block, after integration, and then divide the histogram by this value. A program to do this is given after the first normalization program. This program illustrates how Variable Parameter operations may be used in data operations to reduce the number of block operations required.

Normalizing A Histogram To Unit Area (Probability Density Function)
(Note: This program assumes histogram is already in block 0.)

Program Commands	Contents of Block 0	Contents of Block 1	Purpose of Command
STORE 1 ENTER	Histogram	Histogram	Stores histogram in block 1
∫ 1 ENTER	"	Integral of histogram	To obtain, in last channel of block 1, the value for the total no. of counts in the histogram
←1 SPACE n–1 ENTER, where n is the block size	"	Integral of histogram with last channel now in first channel	
CL 1 SPACE 1 SPACE n–1 ENTER	"	Total no. of counts in channel 0, rest of block cleared	Prepare to divide histogram by total no. of counts
∫ 1 ENTER	Histogram	Total no. of counts is in all channels	Prepare to divide histogram by total no. of counts
÷1 ENTER	Normalized histogram	Total no. of counts is in all channels	Obtain normalized histogram in block 0
END ENTER	Normalized histogram	Total no. of counts is in all channels	Ends program

Alternate Program to Normalize a Histogram to Unit Area

Program Commands	Contents of Block 1	Contents of Block 2	Purpose of Command
STORE 1 ENTER	Histogram	Histogram	Stores histogram in block 1
∫ 1 ENTER	"	Integral of histogram	To obtain, in last channel of block 1, the value for the total number of counts in the histogram
USER PROG BLOCKSIZE 0 ENTER	"	"	Get system blocksize in VP 0
USER PROG — 0 ENTER	"	"	Determine number of last channel (BS-1)
USER PROG LOAD 1 SPACE 1 SPACE 0D ENTER	"	"	Get number of counts from last channel of block 1 into VP 1
÷ 0 SPACE 1D ENTER	Normalized histogram	"	Obtain normalized histogram in block 0
END ENTER	Normalized histogram	"	Ends program

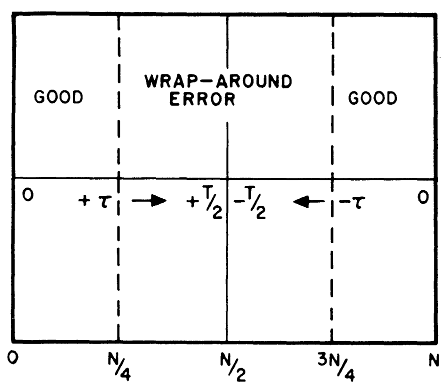
Note: This program assumes histogram is already in block 0.

AUTO AND CROSS CORRELATION

The auto and cross correlation sample subroutine provides for the elimination of wrap-around error. This error is discussed under the CORRELATION command in Section 3.

The subroutine is designed as follows: For a cross correlation, the data is read into block 0 and block 1. By the nature of the STORE command the initial set of data remains also in block 0.

As the first step in prevention of wrap-around error, the first and last quarters of block 1 are cleared out. Then block 0 and 1 are cross-correlated, providing a cross correlation if the blocks are different, and an auto correlation if the blocks are the same. The result if displayed would be set up as follows:



It can be seen that there is still wrap-around error in the middle of the block from +N/4 to 3N/4 (N is the block size), and so as the final step in the subroutine, we clear these channels. Now the correlation is valid for 0 to ±T/4. You can set the ORIGIN switch to CENTER to review the results more conveniently.

To achieve a higher degree of certainty, a number of correlations can be averaged via one of the averaging algorithms mentioned under the discussion of the spectrum averaging programs. The greater certainty appears because the averaging effectively makes the length of the integral in the function greater than T.

Auto[1] and Cross-Correlation Subroutine

Program Commands	Contents of Block 0	Contents of Block 1	Purpose of Command
LABEL 1 ENTER			Establishes initial label point
ANALOG IN ENTER[1]	Current time record from channel A	Current time record from channel B	Input data
CLEAR 1 SPACE 0 SPACE (n/4)–1 ENTER (where n = block size)	"	Current time record from channel B with first 1/4 block cleared	Clear first 1/4 block of time record from channel B
CLEAR 1 SPACE 3/4(n) SPACE n–1 ENTER	"	Current time record from channel B with first and last 1/4's of block cleared	Clear last 1/4 block of time record from channel B
CORR 1 ENTER	Auto or cross correlation	Current time record from channel B with first and last 1/4's of block cleared	Obtain correlation function
MULT 0 SPACE n ENTER	Auto or cross correlation corrected for block size	"	Correct correlation for block size
[1]For auto-correlation, substitute the following two commands for the analog input step.			
ANALOG IN ENTER	Current time record from channel A		Input data
STORE 1 ENTER	"	Current time record from channel A	Store block 0 in block 1 for later CORR command
CLEAR 0 SPACE n/4 SPACE (3/4n)–1 ENTER	Auto or cross correlation with channels n/4 to 3/4(n) cleared	Current time record from channel B with first and last 1/4's of block cleared	Clear wrap-around error from final function

Note: By placing the ORIGIN switch on CENTER, the display will show the + lags to the right of the vertical axis, and the – lags to the left. The last quarter block on each side was cleared by the above command.

TRANSFER FUNCTION AND COHERENCE FUNCTION

Transfer Function

A sample program is provided to measure the transfer and coherence functions between two sets of signals, X and Y. The ensemble estimates are summed as in the summation average programs, although if you desire, you can use other averaging techniques. In this way the average of the input auto spectrum ($\overline{G_{xx}}$), the output auto spectrum ($\overline{G_{yy}}$), and the cross spectrum ($\overline{G_{yx}}$) are formed. Then the transfer function is calculated:

$$H(f) = \frac{\overline{G_{YX}}}{\overline{G_{XX}}}$$

The reason why we average the individual lower spectra first, rather than averaging the entire function is that if a random noise process is used for excitation, not all frequencies may be present in any one estimate of the spectrum, and hence large computational errors may result, causing overflows in the division process. However, if we average (sum) the spectra first, then divide, there is little chance of these errors occurring.

Coherence Function

The coherence function tells the degree of similarity between input and output. Its equation is:

$$\gamma^2 = \frac{|\overline{G_{YX}}|^2}{\overline{G_{XX}} \cdot \overline{G_{YY}}}$$

It is computed by taking summation averages of the cross spectrum ($\overline{G_{YX}}$), then forming the square of its magnitude ($|\overline{G_{YX}}|^2$). This is done by multiplying $\overline{G_{YX}}$ by its self complex conjugate. Then $|\overline{G_{YX}}|^2$ is divided by the input auto spectrum average ($\overline{G_{XX}}$), then by the output auto spectrum average ($\overline{G_{YY}}$).

Combined Transfer and Coherence Function Program

Program Commands	Contents of Block 0	Contents of Block 1	Contents of Block 2	Contents of Block 3	Contents of Block 4	Contents of Block 5	Purpose of Command
LABEL 0 ENTER							Establishes initial point for start of program
CLEAR 3				Cleared			Initializes block 3 (Removes old data)
CLEAR 4 ENTER					Cleared		Initializes block 4 (Removes old data)
CLEAR 5 ENTER						Cleared	Initializes block 5 (Removes old data)
LABEL 1 ENTER				Sum of past $G_{XX}(f)$'s. 0 first time.	Sum of past $G_{YY}(f)$'s. 0 first time.	Sum of past G_{yx}(Real)'s 0 first time.	Establishes target point for power spectrum summation.
ANALOG IN 1 SPACE 3 ENTER		Current time record from Channel A (input, X)	Current time record from Channel B (output, Y)	"	"	"	Data input
HANN 1 ENTER (optional)		Current Channel A record multiplied by sine bell window	"	"	"	"	To make input A time record more periodic in sampling window *optional
HANN 2 ENTER (optional)		"	Current Channel B record multiplied by sine bell window	"	"	"	To make input B time record more periodic in sampling window
F1 SPACE 2 ENTER		Fourier transform of Channel A record	Fourier transform of Channel B record	"	"	"	Obtain Fourier transform of data
CLEAR 1 SPACE 0 ENTER (optional)		Fourier transform of Channel A record minus dc value	Fourier transform of Channel B record	Sum of past $G_{XX}(f)$'s. 0 first time.	Sum of past $G_{YY}(f)$'s. 0 first time.	Sum of past G_{yx}(Real)'s 0 first time.	Clears dc value from Channel A record to improve spectrum dynamic range
CLEAR 2 SPACE 0 ENTER (optional)		Fourier transform of Channel A record minus dc value	Fourier transform of Channel B record minus dc value	Sum of past $G_{XX}(f)$'s. 0 first time.	Sum of past $G_{YY}(f)$'s. 0 first time.	Sum of past G_{yx}(Real)'s 0 first time.	Clears dc value from Channel B record to improve spectrum dynamic range
POWER SPECT 1 SPACE 2 SPACE 2 ENTER	Sum of current plus past $G_{YX}(f)$'s.	"	"	Sum of current plus past $G_{XX}(f)$'s.	Sum of current plus past $G_{YY}(f)$'s.	Sum of current plus past G_{yx}(Real)'s.	Calculates current power spectra (input, output, and cross) adds past spectra and stores results for next pass
COUNT 1 SPACE n1 ENTER	"	"	"	"	"	"	Loop back to target label 1 n1 times
TRANS FCN 1 SPACE 2 SPACE 2 ENTER	Coherence function $\gamma^2 = \frac{\lvert\overline{G_{YX}(f)}\rvert^2}{\overline{G_{XX}(f)}\cdot\overline{G_{YY}(f)}}$	Transfer function $H(f) = \frac{\overline{G_{YX}(f)}}{\overline{G_{XX}(f)}}$	Coherence function $\gamma^2 = \frac{\lvert\overline{G_{YX}(f)}\rvert^2}{\overline{G_{XX}(f)}\cdot\overline{G_{YY}(f)}}$	"	"	"	Calculates transfer and coherence functions and stores results in block 1 and 2.
END ENTER	"	"	"	"	"	"	End program

APPENDIX D
AUTOMATIC MEASUREMENTS

This appendix contains flow charts and listings of the pre-written soft key programs shipped on the 5451C operating disc pack. These programs are for the Gold Key F2 (Transfer Function) and F5 (Power Spectrum) soft keys. The purposes for including these flow charts and listings are as follows:

1. For those who wish to modify the programs to better fit specific applications.
2. As an example for those who wish to write their own soft key programs.
3. As a model for re-entering portions that may have been accidentally written over on the disc (programs reside in unprotected areas).

As an aid toward rapid understanding of these programs, a certain programming "style" has been used. As a result of this style, the programs are longer than they would otherwise need to be, but are easier to comprehend. Some elements of the style are described below.

BLOCK STRUCTURE THROUGH USE OF LABELS

The code in the programs is organized in "blocks" — functional segments which are delimited by LABEL instructions. A convention is followed in the choice of label numbers. The beginning of each block is designated by a label number ending in a multiple of 50, e.g. L1050. The block ends with the label 9 higher, e.g. L1059. Within the block, label numbers are between these limits, e.g. L105l, L1052 etc. One block may contain others — each will use the above delimiting convention.

Although the use of these labels lengthen the programs, it makes them much easier to understand, and to correlate with the flow diagrams. For example, when several branches of a program go to the same point, each branch will jump to its own appropriate label. Since these labels all appear together at one point in the program, one can tell that multiple branches converge at that point. An example is in the Transfer Function program, stack 54, where labels 1209, 1259 and 1609 all appear together. This is just one example of the way in which complex internal linkages in the program are made more visible.

BRANCHING THROUGH USE OF "COMPUTED GOTO's"

In most complex programs, branching is common. One means of branching is to use an IF statement, provided in Keyboard language by the "GOLD KEY" "SKIP" instruction. When there are more than two possible branches, however, use of IF branching tends to get complicated, involving multiple decision points. The "Computed GoTo" or "switch" type of branching statement is more suitable in such cases for simplicity of understanding. It has been used extensively in these keyboard programs, even for simple two way branching. By standardizing on it, the code becomes recognizable and easier to read.

In Keyboard language, the "Computed GoTo" is implemented by computing the number of a label and jumping to it. The following is an example.

```
L 1050                      Start of branch block

Y A+   0   1050   11D       Set variable 0 to 1050
                            + (the value of variable
                            11)

J 0D                        Jump to the label number
                            in variable 0

L 1051                      Code to be executed if
  .                         variable 11 = 1
  .
  .

J 1059                      Go to end of block

L 1052                      Code to be executed if
  .                         variable 11 = 2
  .
  .
  .
  .

L 1059                      End of branch block
```

USE OF SUBROUTINES

In these programs there are several functions which have been set up as subroutines. These include the parameter entry routine (see below) and the measurement routines. The measurement routines are handled this way to simplify the flow of the main program and allow easy replacement of the measurement code in case you modify it in some way.

PARAMETER ENTRY ROUTINE

Since these programs ask the operator for many input parameters, a single subroutine, LABEL 100 in STACK 0, handles all the parameter entries. The routine is called with variable parameters 1 and 2 equal to the lower and upper limits on the range of allowable operator inputs. The routine reads your input, checks it against the range limits, and, if it is valid, passes it back to the calling program in (floating point) variable parameter 2000. If the input is out of range, this routine notifies you and waits for the new input. The routine will not return to the calling program until a valid input has been received.

PRECAUTIONARY NOTES

The following precautions apply to the operation of the preprogrammed measurements:

1. When using the standard software zoom (BSFA), the measurement blocksize can be no larger than 1024. When using the Option 670 Fourier Preprocessor for BSFA measurements, the maximum blocksize is 2048.

2. The maximum center frequency you may specify for a BSFA measurement is 32767 Hz.

3. The messages CF WHAT? or BW WHAT? may result if the center frequency and/or bandwidth you have chosen for your measurement are such that the BSFA analysis band is either less than 0 or greater than the ADC Fmax setting. Specifying different parameters should remove this problem.

4. The message DL WHAT? may occur when performing the on-line BSFA measurment. This is because the display is active during the on-line measurement. To remove this problem, either reduce the measurement bandwidth (thereby increasing the zoom power and lowering the data rate into the computer), or edit the appropriate keyboard stacks (stack 56 for transfer function, stack 61 for power spectrum) to remove parameter n3 from the calls to User Prog 45 for the on-line measurement (refer to commented program listings which follow).

5. When performing an off-line BSFA measurement with an optional Mag Tape unit, perform the following steps before making the measurement.

   Set:

   ADC SAMPLE MODE to INTERNAL KHz/μs
   MULTIPLIER to 100/10/5
   INPUT SELECTOR to A
   TRIGGERING to FREE RUN
   OVERLOAD VOLTAGE A to CHECK

   Enter:

   BLOCKSIZE 4096 ENTER
   MASS STORE 32 ENTER
   MASS STORE 22 SPACE 1 SPACE 150 ENTER
   MASS STORE 32 ENTER

   This writes 150 records of data on the magtape so that the magtape will be able to position to record 135 on the tape when the ADC throughput is performed. It will position by looking for the interrecord gaps written by the WRITE ADC throughput command.

6. After completing a BSFA measurement, be sure that all data space declared by the zoom programs is released by pressing RESTART.

As you go through the flow charts and commented listings, remember that these are only examples of programming the soft keys F1 through F6 on the Keyboard. It is up to you to determine which, if any, portions of these programs should be maintained. Because these programs are stored in unprotected areas of the Disc, there is the possibility they can be written over. If this should happen, you should enter the program stacks from the listing, substitute your own program, or copy from your back-up disc.

The soft key programs and the associated ASCII text and variable parameters were originally stored on the system disc pack in Files 3,4, and 7. The records used are as follows:

File 3 (Keyboard Programs)

Record 0
Records 51 through 62

File 4 (Text Buffers)

Text buffers 51 through 55
ASCII records ∧ through 3449*
3245

File 7 (Common)

Common) Record 0

3700
*This assumes that there are V records in File 4. If not, the first and last ASCII text records may be computed as follows:

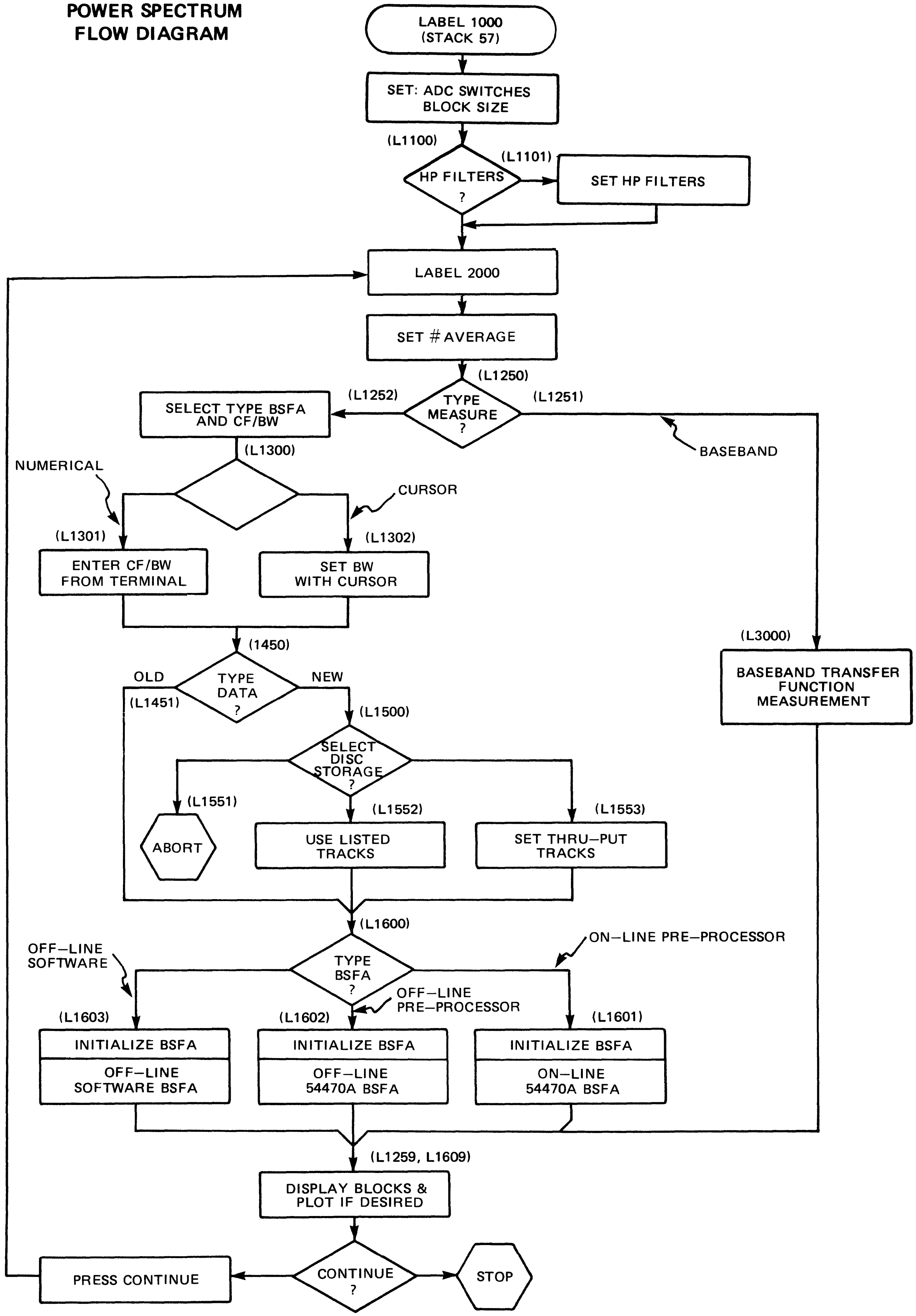
First record number = NR − (5 × last text buffer number)

Last record number = NR − (5 × first text buffer number) + 4

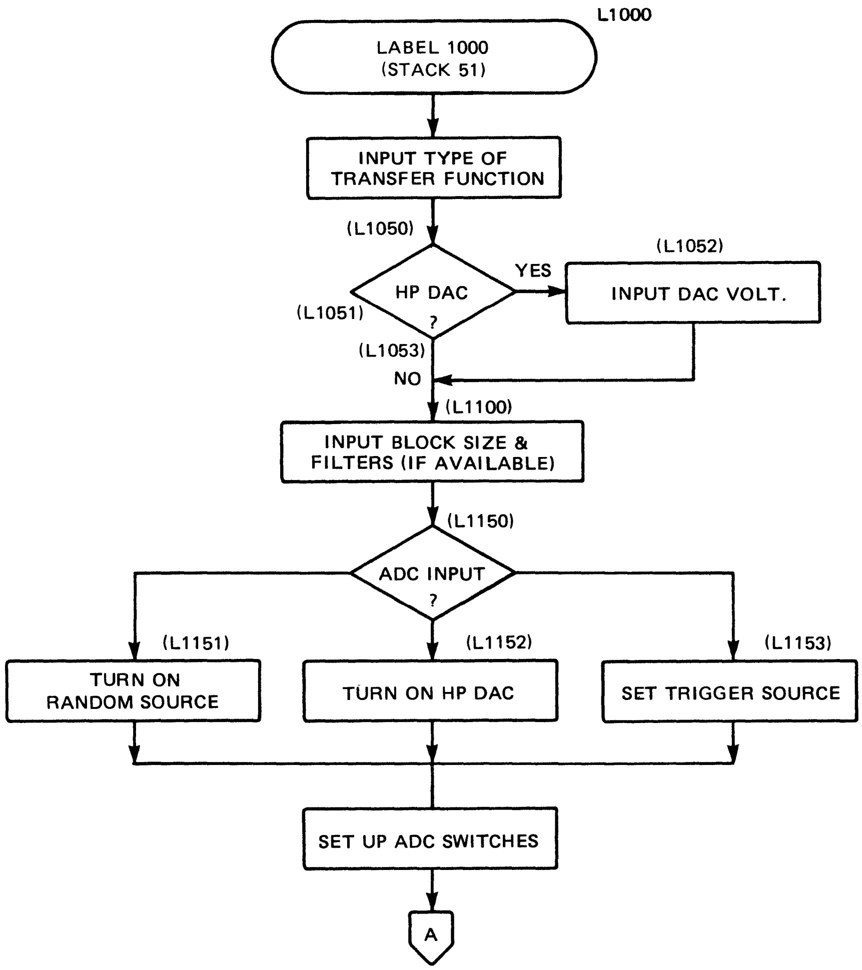
where NR = number of records in File 4.

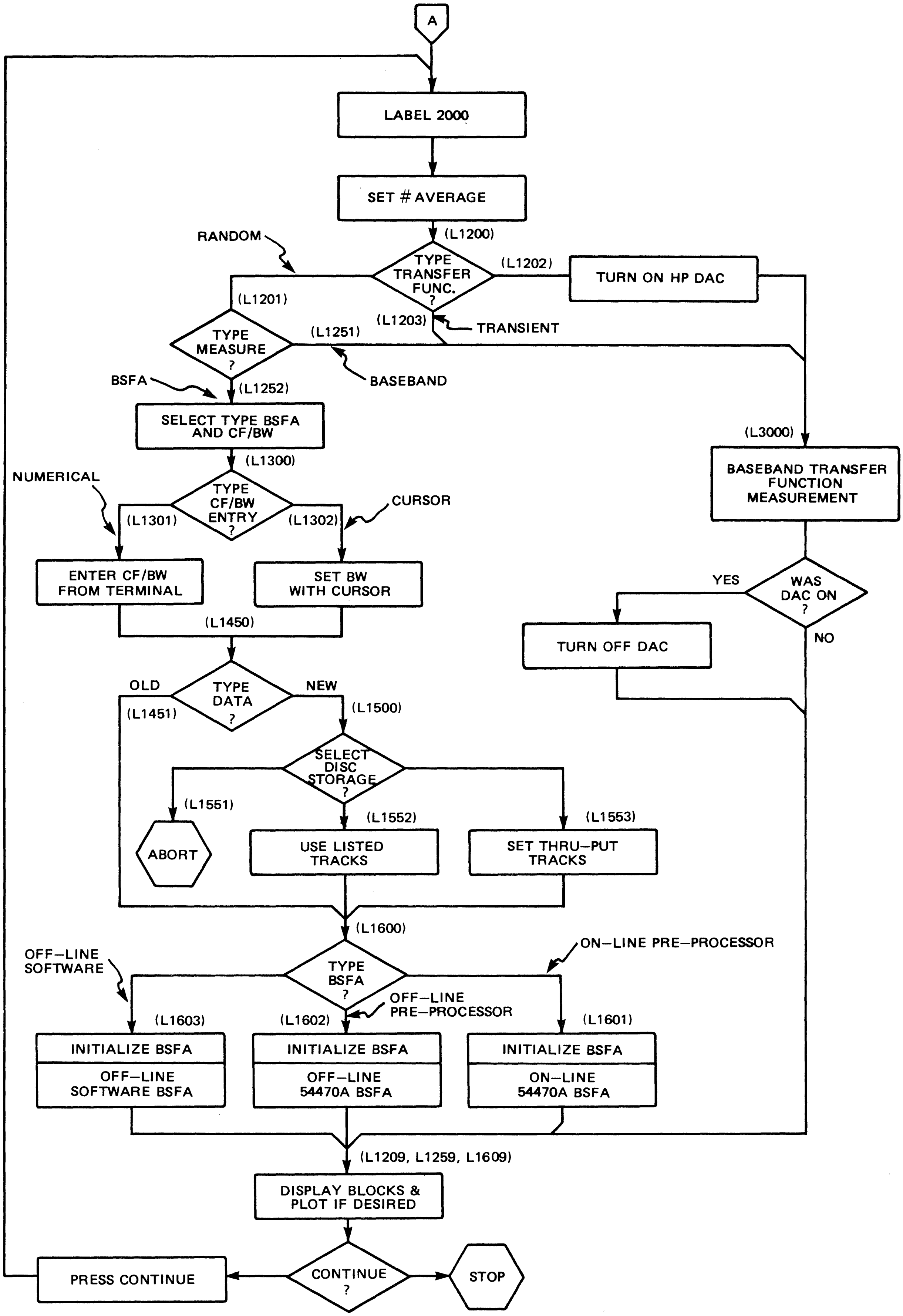
The allocations above should be kept in mind so the above records are not inadvertently altered or destroyed when using the Fourier system. Should you wish to alter the allocations, you will also have to modify the keyboard programs to reflect such changes.

POWER SPECTRUM FLOW DIAGRAM



TRANSFER FUNCTION
FLOW DIAGRAM




A
LABEL 2000
SET # AVERAGE
(L1200)
TYPE TRANSFER FUNC. ?
RANDOM
(L1202)
TURN ON HP DAC
(L1201)
(L1203)
TRANSIENT
TYPE MEASURE ?
(L1251)
BASEBAND
BSFA
(L1252)
SELECT TYPE BSFA AND CF/BW
(L3000)
BASEBAND TRANSFER FUNCTION MEASUREMENT
(L1300)
NUMERICAL
TYPE CF/BW ENTRY ?
CURSOR
(L1301)
(L1302)
ENTER CF/BW FROM TERMINAL
SET BW WITH CURSOR
YES
WAS DAC ON ?
NO
TURN OFF DAC
(L1450)
OLD
TYPE DATA ?
NEW
(L1451)
(L1500)
SELECT DISC STORAGE ?
(L1551)
(L1552)
(L1553)
ABORT
USE LISTED TRACKS
SET THRU–PUT TRACKS
(L1600)
OFF–LINE SOFTWARE
ON–LINE PRE–PROCESSOR
TYPE BSFA ?
OFF–LINE PRE–PROCESSOR
(L1603)
(L1602)
(L1601)
INITIALIZE BSFA
OFF–LINE SOFTWARE BSFA
INITIALIZE BSFA
OFF–LINE 54470A BSFA
INITIALIZE BSFA
ON–LINE 54470A BSFA
(L1209, L1259, L1609)
DISPLAY BLOCKS & PLOT IF DESIRED
PRESS CONTINUE
CONTINUE ?
STOP

```
L     0                  STACK 0
                          Soft key jump directory
                          "GOLD KEY" "Fn" causes jump to LABEL n
                          in this stack. User can program the stack
                          anywhere from there.
                          Transfer Function and Power Spectrum
                          are preprogrammed in. These may also
                          be changed.

L     2                   F2   TRANSFER FUNCTION
J  1000    51                Label 1000 in stack 51
L     5                   F5   POWER SPECTRUM
J  1000    57                Label 1000 in stack 57
L     1                   F1   USER ASSIGNABLE
L     3                   F3   USER ASSIGNABLE
L     4                   F4   USER ASSIGNABLE
L     6                   F6   USER ASSIGNABLE
D                         Pause here if unassigned function called
J     6                     Loop if "CONTINUE" pressed
                         ============================================
L    100                  PARAMETER ENTRY ROUTINE
Y R   2000                 Read user entry into variable 2000
Y IF  2000   1D  5   1     Check if entry >= lower limit
Y W     99    1              Too small. Cue operator with range
Y P      1
Y W     98    1
Y P      2
J    100    -1                and loop for new entry
L    101
Y IF  2000   2D  5  -1     Check if entry <= upper limit
Y W     99    1              Too large. Cue operator with range
Y P      1
Y W     98    1
Y P      2
J    100    -1                and loop for new entry
(                          Return to calling program--entry in
                             variable 2000, within specified range

L    109
.
                         ============================================
```

```
L     -51                               STACK 51
L    1000                               TRANSFER FUNCTION PROGRAM
MS     37                                Recall previous variable parameters
MS     17
Y    5838      51                        Read text buffer #51
Y W       1       1                     Print messages 1 & 2
Y W       2       1
Y _       1       1                     Set allowable range for user entry
Y _       2       3
J    100       0                        Call user entry routine  L100, stack 0
Y _      11    2000D
L    1050                               BLOCK 1050
Y A+      0    1050     11D              Branch per value of variable 11
J        0D                                to L1051, 1052 or 1053
L    1051
L    1053
J    1059
L    1052
Y W       3       1
Y _       1       3                      Parameter entry
Y _       2    9999                        Range: 0 - 1
J    100       0                           Variable: 12
Y _      12    2000D
L    1059                               END OF BLOCK 1050
Y W       5       1
BS    512                               Block size 512--user may change after pause
D                                       Pause for operator action
Y W       6       1
Y _       1       0                     Parameter entry
Y _       2       1                       Range: 0 - 1
J    100       0                          Variable: 14
Y _      14    2000D
L    1100                               BLOCK 1100
Y A+      0    1101     14D
J        0D
L    1101
J    1109
L    1102
Y     100      0       -2                Set 2 filters--auto select mode
L    1109                               END OF BLOCK 1100
Y W       7       1
D                                       Pause for operator action
Y    5838      52
L    1150                               BLOCK 1150
Y A+      0    1150     11D              Branch per variable 11
J        0D                                to L1151, 1152 or 1153
L    1151
Y W       8       1
J    1159
L    1152
B         6    12D                       Fill DAC buffer and
B         6                                turn on DAC
J    1159
L    1153
Y W       9       1
L    1159                               END OF BLOCK 1150
Y W      10       1
RA                                      Analog input in "REPEAT" to monitor inputs
F
TL                                      Power spectrum (log) for user reference
J    2000      1       1                Jump to next stack and continue
.
```

```
L      -52                    STACK 52
L     2000                    LOOP POINT FOR TRANSFER FUNCTION
Y     5838      52
Y W      11      1
Y        1       1             Parameter entry
Y -      2   32767               Range: 1 - 32767
J -    100       0               Variable: 15
Y        15   2000D
Y -    1200                    BLOCK 1200 (ends in stack 54)
L A+      0    1200     11D     Branch per variable 11
Y        0D                       to L1201, 1202 or 1203
J     1202
L W      22       1
Y                               Pause for operator action
D         6                     Turn on DAC
B     3000       3       1      Subroutine call - baseband measurement
J                               Turn off DAC
B     1209       2       1        then go to end of block 1200
J     1203
L W      22       1
Y                               Pause for operator action
D     3000       3       1      Subroutine call - baseband measurement
J     1209       2       1        then go to end of block 1200
J     1201
L W      12       1
Y        1        1             Parameter entry
Y -      2        2               Range: 1 - 2
Y -    100        0               Variable: 16
J        16   2000D
Y -    1250                     BLOCK 1250 (ends in stack 54)
L A+      0    1250     16D      Branch per variable 16
Y        0D                        to L1251 or 1252
J     1251
L W      22       1
Y                                Pause for operator action
D     3000       3       1       Subroutine call - baseband measurement
J     1259       2       1         then go to end of block 1250
J     1252
L W      13       1
Y        1        1              Parameter entry
Y -      2        3                Range: 1 - 3
Y -    100        0                Variable: 17
J        17   2000D
Y -   5838       53
Y W      15       1
Y        1        1              Parameter entry
Y -      2        2                Range: 1 - 2
Y -    100        0                Variable: 18
J        18   2000D
Y -

L     1300                       BLOCK 1300 (ends in stack 53)
Y A+      0    1300     18D       Branch per variable 18
J        0D                         to L1301 or 1302
L     1301
Y W      16       1
Y        1        1               Parameter entry
Y -      2    32767                 Range: 1 - 32767
Y -    100        0                 Variable: 19
J        19   2000D
Y W      17       1
Y        1        1               Parameter entry
Y A-      2      19D      1          Range: 1 - (value of var 19 - 1)
J      100        0                  Variable: 20
Y        20   2000D
J -   1309        1       1
L     1302
J     1303        1       1
```

```
L     -53                       STACK 53
L    1303
Y W      18       1
/.   2000                          Cursor on. Cursor parameters to
                                     variables 2000 - 2002 (2001=frequency)
Y      2003   2001D                Variable 2003 gets 1st cursor frequency
Y W      19       1
/.   2000                          Cursor on, parameters to 2000 - 2002
/.     -1                          Cursor off
Y _    2004   2001D                Variable 2004 gets 2nd cursor frequency
L    1350                          BLOCK 1350
Y IF   2003   2004D   2  -2         If 1st cursor frequency > 2nd
Y _    2004   2003D                   then swap them
Y _    2003   2001D
L    1359                          END OF BLOCK 1350
Y A-   2000   2004D   2003D
Y        20   2000D                Variable 20 gets zoom bandwidth
Y T    2000
Y A+   2001   2003D   2000D
Y _      19   2001D                Variable 19 gets zoom center frequency
L    1400                          BLOCK 1400
Y IF     20     19D   1  -2         If bandwidth not < ctr freq
Y A-     20     19D   1               then bandwidth = ctr freq - 1
L    1409                          END OF BLOCK 1400
L    1309                         END OF BLOCK 1300 (from stack 52)
L    1450                         BLOCK 1450 (ends in stack 54)
Y A+      0   1450     17D         Branch per variable 17
J        0D                          to L1451, 1452 or 1453
L    1451
Y    5838      54
Y W      22       1
D                                  Pause for operator action
J    1459       1       1
L    1452
L    1453
Y W      20       1
Y _       1       1                Parameter entry
Y _       2       2                  Range: 1 - 2
J     100       0                    Variable: 24
Y _      24   2000D
L    1500                          BLOCK 1500 (ends in stack 54)
Y    5838      54
Y A+      0   1500     24D          Branch per variable 24
J        0D                           to L1501 or 1502
L    1501
J    1509       1       1
L    1502
Y W      28       1
Y _       1       1                 Parameter entry
Y _       2       3                   Range: 1 - 3
J     100       0                     Variable: 2000 (temporary)
L    1550                           BLOCK 1550 (ends in stack 54)
Y A+      0   1550   2000D           Branch per variable 2000
J        0D                            to L1551, 1552 or 1553
L    1551
D                                    Pause & loop to here--abort
J    1551
L    1552
J    1554       1       1
L    1553
J    1555       1       1
.
```

```
L    -54                    STACK 54
L    1554
Y _     22    134                Variable 22 gets default start track
Y _     21     64                Variable 21 gets default # of records
J -  1559
L    1555
Y W     27      1
Y _      1      1                Parameter entry
Y _      2    197                  Range: 1 - 197
J -   100      0                   Variable: 22
Y       22   2000D               Variable 22 gets user's start track
Y Ā-    21    198   2000D        variable 21 gets # of records left
L    1559                       END OF BLOCK 1550 (from stack 3)
MS     32     22D               Position throughput file to start track
Y W     22      1
D                               Pause for operator action
Y BS     13                     Variable 13 gets current block size
BS   2048                       Set block size to max for throughput
MS     22       2     21D       Perform ADC throughput
BS     13D                      Restore block size
Y W     21      1
L    1509                       END OF BLOCK 1500 (from stack 53)
L    1459                      END OF BLOCK 1450 (from stack 53)
L    1600                      BLOCK 1600
Y A+      0   1600     17D      Branch per variable 17
J        0D                       to L1601, 1602 or 1603
L    1601
Y      40     19D     20D       Initialize zoom--on line preprocessor
J    4000      2       1        Subroutine call--zoom measurement
Y      40      0                Reset zoom to baseband
J    1609                       Go to end of block 1601
L    1602
Y      43   19D   20D   22D     Initialize zoom--off line preprocessor
J    4500      2       1        Subroutine call--zoom measurement
Y      43      0                Reset zoom to baseband
J    1609                       Go to end of block 1601
L    1603
Y      41   19D   20D   22D     Initialize zoom--off line software
J    4500      2       1        Subroutine call--zoom measurement
Y      41      0                Reset zoom to baseband
L    1609                      END OF BLOCK 1600 (from stack 60)
L    1259                     END OF BLOCK 1259 (from stack 52)
L    1209                    END OF BLOCK 1209 (from stack 52)
Y    5838     54
Y W     23      1                Transfer function complete
Y W     24      1                  Print instructions to operator
Y W     25      1
Y W     26      1
MS     37                        Save variable parameters
MS     27
D                                Pause for operator action
J    2000     -2       1         Loop to repeat transfer function
.                              END OF TRANSFER FUNCTION (except subroutines)
```

```
L      -55                       STACK 55
L     3000                       BASEBAND TRANSFER FUNCTION SUBROUTINE
Y     5838    55
Y        6     2     3            Check ADC input selector. If not 2 channel
Y W       31     1                  inform operator
D                                   pause for operator to correct it and
J     3000    -1                    loop to check again.
CL       2                        Clear blocks needed for averaging
CL       3
CL       4
CL       5
L     3050                        BLOCK 3050
Y A+        0  3050     11D        Branch per variable 11
J       0D    -1                     to L3051, 3052 or 3053
L     3051                         L3051 & 3052 equivalent--non-transient
L     3052
RA       0     3                   Analog input
H1       0                         Hann
H1       1
F        0     1                   Transform
CL       0     0                   Clear DC
CL       1     0
SP       0     2     2             Average spectra
#     3051   15D     0             Loop for specified # of averages
J     9050     7     1             Subroutine call - clear 2nd half block
CH       0     2     2             Compute transfer function
*        2    16                   Correct amplitude for Hanning
:        2     3
*        3    16
:        3     3
J     3059    -1
L     3053                         L3053 transient signals
Y W       29     1                 Instruct operator to impact on cue
L     3054
Y W       30     1                 Cue with "beep"
RA       0     3                   Analog input
Y        5     0    -2             Check for overload--if so, cue again
F        0     1                   Transform
CL       0     0                   Clear DC
CL       1     0
SP       0     2     2             Average spectra
#     3054   15D     0             Loop for specified # of averages
J     9050     7     1             Subroutine call - clear 2nd half block
CH       0     2     2             Compute transfer function
L     3059                        END OF BLOCK 3050
TL       0                        Take logs of spectra
TL       2
TL       3
TL       4
<                                 Return to calling program
.
```

```
L     -56                       STACK 56
L    4000                       SUBROUTINE FOR ON LINE ZOOM TRANSFER FCN
Y    5838      55
Y       6       2       3        Check ADC input selector If not 2 channels
Y W      31       1                inform operator
D                                  pause for operator to correct it and
J    4000     -1                   loop to check again
CL      7                        Clear blocks needed for averaging
CL      8
CL      9
CL     10
L    4010
Y      45    5    1     8        Zoom, displaying Channel B power spectrum
SP      5       2       2        Average spectra
#    4010     15D       0        Loop for specified # of averages
CH      5       2       2        Compute transfer function
J    4200                        Subroutine call to move data
<                                Return to calling program
                                ===========================================
L    4200                       SUBROUTINE TO MOVE DATA BLOCKS
X<      9                        Move data blocks from zoom area
X>      4                          to area for operator viewing
X<      8
X>      3
X<      7
X>      2
X<      6
X>      1
X<      5
TL      0                        Take logs of spectra
TL      2
TL      3
TL      4
<                                Return to calling program
                                ===========================================
L    4500                       SUBROUTINE FOR OFF LINE ZOOM TRANSFER FCN
CL      7                        Clear blocks needed for averaging
CL      8
CL      9
CL     10
L    4510
Y      45    6    1     2        Zoom on Channel B
Y      45    5    1     1        Zoom on Channel A
D       8       4                Display Channel B power spectrum
SP      5       2       2        Average spectra
#    4510     15D       0        Loop for specified # of averages
CH      5       2       2        Compute transfer function
J    4200                        Subroutine call to move data
<                                Return to calling program
                                ===========================================
```

```
L    -57                              STACK 57
-   1000                              POWER SPECTRUM PROGRAM
                                       Recall previous variable parameters
MS    37
MS    17                               Read text buffer #51
Y   5838      51                       Print messages 4 & 5
Y W      4       1
Y W      5       1                     Block size 512--user may change after pause
BS   512                               Pause for operator action
D
Y W      6       1                     Parameter entry
Y        1       0                       Range: 0 - 1
Y -      2       1                       Variable: 14
J -  100       0
Y       14   2000D                     BLOCK 1100
L -  1100
Y A+      0   1101      14D
J        0D
L    1101
J    1109
L    1102                              Set 2 filters--auto select mode
Y     100        0      -2             END OF BLOCK 1100
L    1109
Y W       7       1                    Pause for operator action
D
Y    5838       52
Y W      10       1                    Analog input in "REPEAT" to monitor inputs
RA
F                                      Power spectrum (log) for user reference
TL                                     Jump to next stack and continue
J    2000        1       1
.
```

```
L     -58                        STACK 58
L    2000                        LOOP POINT FOR POWER SPECTRUM
Y    5838      52
Y W      11     1
Y _       1     1                 Parameter entry
Y _       2 32767                   Range: 1 - 32767
J _  100        0                   Variable: 15
Y        15  2000D
Y W      12     1
Y _       1     1                 Parameter entry
Y _       2     2                   Range: 1 - 2
J    100        0                   Variable: 16
Y _      16  2000D
L _  1250                         BLOCK 1250 (ends in stack 60)
Y A+      0  1250  16D             Branch per variable 16
J        0D                          to L1251 or 1252
L    1251
Y W      22     1
D                                  Pause for operator action
J    3000       3     1            Subroutine call - baseband measurement
J    1259       2     1              then go to end of block 1250
L    1252
Y W      13     1
Y _       1     1                  Parameter entry
Y _       2     3                    Range: 1 - 3
J _  100        0                    Variable: 17
Y _      17  2000D
Y _  5838      53
Y W      15     1
Y _       1     1                  Parameter entry
Y _       2     2                    Range: 1 - 2
J _  100        0                    Variable: 18
Y _      18  2000D
L _  1300                          BLOCK 1300 (ends in stack 59)
Y A+      0  1300  18D              Branch per variable 18
J        0D                           to L1301 or 1302
L    1301
Y W      16     1
Y _       1     1                   Parameter entry
Y _       2 32767                     Range: 1 - 32767
J _  100        0                     Variable: 19
Y        19  2000D
Y W      17     1
Y         1     1                   Parameter entry
Y Ā-      2   19D     1               Range: 1 - (value of var 19 - 1)
J    100        0                     Variable: 20
Y _      20  2000D
J _  1309       1     1
L    1302
J    1303       1     1
.
```

```
L     -59                       STACK 59
L    1303
Y W      18      1
/.   2000                         Cursor on. Cursor parameters to
                                    variables 2000 - 2002 (2001=frequency)
Y      2003   2001D               Variable 2003 gets 1st cursor frequency
Y Q      19      1
/.   2000                         Cursor on, parameters to 2000 - 2002
/.     -1                         Cursor off
Y _    2004   2001D               Variable 2004 gets 2nd cursor frequency
L    1350                         BLOCK 1350
Y IF   2003   2004D   2  -2        If 1st cursor frequency > 2nd
Y _    2004   2003D                  then swap them
Y _    2003   2001D               END OF BLOCK 1350
L    1359
Y A-   2000   2004D   2003D
Y _      20   2000D               Variable 20 gets zoom bandwidth
Y T    2000
Y A+   2001   2003D   2000D
Y _      19   2001D               Variable 19 gets zoom center frequency
L    1400                         BLOCK 1400
Y IF     20     19D   1  -2        If bandwidth not < ctr freq
Y A-     20     19D   1              then bandwidth = ctr freq - 1
L    1409                         END OF BLOCK 1400
L    1309                        END OF BLOCK 1300 (from stack 58)
L    1450                        BLOCK 1450 (ends in stack 60)
Y A+      0   1450     17D        Branch per variable 17
J        0D                         to L1451, 1452 or 1453
L    1451
Y    5838      54
Y W      22      1
D                                 Pause for operator action
J    1459      1      1
L    1452
L    1453
Y W      20      1
Y _       1      1                Parameter entry
Y _       2      2                  Range: 1 - 2
J     100      0                    Variable: 24
Y _      24   2000D
L    1500                         BLOCK 1500 (ends in stack 60)
Y    5838      54
Y A+      0   1500     24D         Branch per variable 24
J        0D                          to L1501 or 1502
L    1501
J    1509      1      1
L    1502
Y W      28      1
Y _       1      1                 Parameter entry
Y _       2      3                   Range: 1 - 3
J     100      0                     Variable: 2000 (temporary)
L    1550                          BLOCK 1550 (ends in stack 60)
Y A+      0   1550   2000D          Branch per variable 2000
J        0D                           to L1551, 1552 or 1553
L    1551
D                                   Pause & loop to here--abort
J    1551
L    1552
J    1554      1      1
L    1553
J    1555      1      1
.
```

```
L     -60                        STACK 60
L    1554
Y _     22    134                   Variable 22 gets default start track
Y -     21     64                   variable 21 gets default # of records
J -  1559
L    1555
Y W     27      1
Y       1       1                   Parameter entry
Y -     2     197                     Range: 1 - 197
J -   100       0                     Variable: 22
Y       22   2000D                  Variable 22 gets user's start track
Y A-    21    198    2000D          Variable 21 gets # of records left
L    1559                          END OF BLOCK 1550 (from stack 59)
MS     32     22D                  Position throughput file to start track
Y W     22      1
D                                  Pause for operator action
Y BS    13                         Variable 13 gets current block size
BS   4096                          Set block size to max for throughput
MS     22       1     21D          Perform ADC throughput
BS     13D                         Restore block size
Y W     21      1
L    1509                           END OF BLOCK 1500 (from stack 59)
L    1459                          END OF BLOCK 1450 (from stack 59)
L    1600                          BLOCK 1600
Y A+     0   1600     17D           Branch per variable 17
J       0D                            to L1601, 1602 or 1603
L    1601
Y      40    19D   20D              Initialize zoom--on line preprocessor
J    4000      1      1             Subroutine call--zoom measurement
Y      40      0                    Reset zoom to baseband
J    1609                           Go to end of block 1600
L    1602
Y      43    19D   20D   22D        Initialize zoom--off line preprocessor
J    4500      1      1             Subroutine call--zoom measurement
Y      43      0                    Reset zoom to baseband
J    1609                           Go to end of block 1600
L    1603
Y      41    19D   20D   22D        Initialize zoom--off line software
J    4500      1      1             Subroutine call--zoom measurement
Y      41      0                    Reset zoom to baseband
L    1609                          END OF BLOCK 1600
L    1259                          END OF BLOCK 1250 (from stack 58)
Y    5838     54
Y W     23      1                  Power spectrum complete
Y W     25      1                    Print messages to operator
Y W     26      1
MS     37                          Save variable parameters
MS     27
D                                  Pause for operator action
J    2000     -2      1            Loop to repeat power spectrum
.                                 END OF POWER SPECTRUM (except subroutines)
```

```
L     -61                     STACK 61
L     3000                    BASEBAND POWER SPECTRUM SUBROUTINE
Y     5838     55
Y        6      1      3       Check ADC input selector. If not 1 channel
Y W       32      1              inform operator
D                                pause for operator to correct it and
J     3000     -1                loop to check again
CL       1                     Clear block needed for averaging
L     3010
RA       0      1              Analog input, displaying average
H1       0                     Hann
F                              Transform
CL       0      0              Clear DC
SP       0                     Average spectrum
#     3010     15D     0       Loop for specified # of averages
J     9055      1      1       Subroutine call - clear 2nd half block
*        1     16              Hanning correction for broadband noise
:        1      3
X(       1                     Load average to block 0
TL       0                     Take log
(                              Return to calling program
L     4000                    ============================================
Y     5838     55              SUBROUTINE FOR ON LINE ZOOM POWER SPECTRUM
Y        6      1      3
Y W       32      1            Check ADC selector. If not 2 channels
D                                inform operator
J     4000     -1                pause for operator to correct it and
CL       2                       loop to check again
L     4010                     Clear block needed for averaging
Y       45      1      1         2
SP       1                     Zoom, displaying power spectrum
#     4010     15D     0       Average spectrum
X(       2                     Loop for specified # of averages
TL                             Load average to block 0
(                              Take log
L     4500                     Return to calling program
CL       2                    ============================================
L     4510                    SUBROUTINE FOR OFF LINE ZOOM POWER SPECTRUM
Y       45      1      1       Clear block needed for averaging
D        2      4
SP       1                     Zoom
#     4510     15D     0       Display average (1 sweep)
X(       2                     Average
TL                             Loop for specified # of averages
(                              Load average to block 0
.                              Take log
                               Return to calling program
                              ============================================
```

```
L     -62                          STACK 62
L    9050                          CLEAR 2ND HALF BLOCK SUBROUTINE
Y _      29   9057                  Set return label value
J ¯  9056     -1
L    9057                           Return label
CL       4    25D     26D   Clear last half block 4
CL       5    25D     26D   Clear last half block 5
J    9059     -1
L    9055                           Entry pt. Pwr Spec pgm
Y _      29   9058                  Set return label value
J ¯  9056     -1
L    9058                           Return label
CL       1    25D     26D   Clear last half block 1
J    9059     -1
L    9056
Y A+      0   9051     14D   Was HP filter selected?
J        0D    -1            No, Gto 9051
L    9052                    Yes
Y BS     13                  Get current blocksize
Y :       26    13D          Store Bs/2 in variable 26
Y :       25    26D          Store Bs/4 in variable 25
J       29D    -1            Indirect return
L    9051
L    9059                    End of pgm 9050&9055
<                            Return to calling program
.
```

Text buffer messages for Transfer Function and Power Spectrum programs

```
 BUFFER #   MESSAGE #


       51       1
HP TRANSFER FUNCTION PROGRAM


       51       2
SELECT EXCITATION TYPE
  51=RANDOM - BASEBAND OR ZOOM
  2=HP DAC - BASEBAND ONLY
  3=TRANSIENT - BASEBAND ONLY


       51       3
INPUT DESIRED DAC OUTPUT IN MV


       51       4
HP POWER SPECTRUM PROGRAM


       51       5
SET ADC FREQUENCY RANGE AS DESIRED
   (SAMPLE MODE & MULTIPLIER)
SET ADC TRIGGER TO "FREE RUN"
CHANGE BLOCK SIZE IF DESIRED
PRESS "CONTINUE" WHEN READY


       51       6
ARE HP FILTERS INSTALLED?
  0=NO, 51=YES
```

```
      51       7
SET KEYBOARD REPEAT/SINGLE SWITCH
 TO "REPEAT"
PRESS "CONTINUE" WHEN READY


      51       98
  UPPER LIMIT =


      51       99
ENTRY OUT OF RANGE-PLEASE REENTER
  LOWER LIMIT =
  PRESS CONTINUE WHEN READY


      52       8
TURN ON RANDOM EXCITATION SOURCE


      52       9
SET TRIGGER SOURCE AS DESIRED
IMPACT STRUCTURE REPEATEDLY


      52       10
SET OVERLOAD VOLTAGES AND TRIGGER
 LEVELS FOR SIGNAL AMPLITUDES
MOVE REPEAT/SINGLE SWITCH TO
 "SINGLE" WHEN READY.  IF SOURCE
 IS NOT IN FREE RUN, TRIGGER THE
 SYSTEM AGAIN TO CONTINUE.


      52       11
ENTER NUMBER OF AVERAGES DESIRED


      52       12
ENTER MEASUREMENT TYPE
  1=BASEBAND
  52=ZOOM


      52       13
ENTER ZOOM MEASUREMENT MODE
  1=ON LINE, PREPROCESSOR
  52=OFF LINE, PREPROCESSOR
  3=OFF LINE, SOFTWARE
```

```
       52       14
ZOOM NOT APPROPRIATE WITH HP DAC          Beeps to cue operator
 BASEBAND MEASUREMENT WILL BE MADE


       53       15
HOW WILL YOU SPECIFY ZOOM BANDWIDTH?
  1=NUMERIC ENTRY - CTR FREQ & BW
  2=CURSOR - ON PRIOR MEASUREMENT


       53       16
ENTER CENTER FREQUENCY


       53       17
ENTER BANDWIDTH


       53       18
MOVE CURSOR TO START FREQUENCY
 PRESS "VALUE" (SWITCH REGISTER 11)


       53       19
MOVE CURSOR TO END FREQUENCY
 PRESS "VALUE"


       53       20
ANALYZE OLD OR NEW DATA?
  1=OLD (FROM THROUGHPUT FILE)
  2=NEW


       54       21
THROUGHPUT COMPLETED


       54       22
PRESS "CONTINUE" FOR MEASUREMENT


       54       23
MEASUREMENT COMPLETE


       54       24
TO DISPLAY RESULTS, PRESS:
  "DISPLAY" "0"  LOG TRANSFER FCN
  "DISPLAY" "1"  COHERENCE
  "DISPLAY" "2"  INPUT POWER SPECT
```

```
 "DISPLAY" "3"  OUTPUT POWER SPECT
 "DISPLAY" "54"  CROSS POWER SPECT


         54         25
TO COPY DISPLAY ON TERMINAL:
 PUT TERMINAL IN GRAPHICS MODE
 PRESS "GOLD KEY" "PLOT"


         54         26
TO MAKE ANOTHER MEASUREMENT:
 PUT TERMINAL IN ASCII MODE
 PRESS "CONTINUE"


         54         27
ENTER STARTING TRACK FOR THROUGHPUT


         54         28
THROUGHPUT WILL USE TRACKS
 135 THROUGH 198 ON THE LOWER
 (FSDS) DISC. IS THIS OK?
  1=NO - ABORT
  2=YES - PROCEED
  3=NO - ASK ME FOR TRACK #


         55         29
IMPACT STRUCTURE ON CUE (BEEP)
 FOR EACH AVERAGE


         55         30
                                              Beeps to cue operator
         55         31
SET ADC INPUT SELECTOR TO "AB"                Beeps to cue operator
 PRESS "CONTINUE" WHEN READY


         55         32
SET ADC INPUT SELECTOR TO "A"                 Beeps to cue operator
 PRESS "CONTINUE" WHEN READY
```

APPENDIX E
INPUT/OUTPUT OPTIONS

INTRODUCTION

This section provides operating information for the optional Low Pass Filter and DAC. It also contains brief descriptions of how to use other peripheral devices to input or output information to the system (e.g., data, text, or programs). The operating information for the optional Pre-Processor is contained in Section 5. An explanation of how data is represented for input or output is contained at the end of this Appendix.

LOW PASS FILTER

The Low Pass Filter provides signal conditioning of an analog input signal through one of ten selectable filter ranges.

The Low Pass Filter has no front panel controls; all filter settings are programmed from the system Keyboard or through software. Ten indicator lights on the front panel indicate the frequency range of the active filter. Two additional lights indicate whether the filter is being bypassed and if the input signal is ac-coupled. Two front panel BNC's provide one connection for the analog signal to be filtered (INPUT) and a second connection for the filtered output signal (OUTPUT).

SETUP

The software for the Low Pass Filter is resident in several of the system overlays.

When power is first applied and the filters are in the bypassed mode, the FILTER BYPD lamp should light. When a filter range is activated, the appropriate range lamp and the AC CPLG lamps will light, and the FILTER BYPD lamp will go out. If no light appears, check the position of the rear panel LINE ON switch.

KEYBOARD COMMAND FORMAT

The filter is operated from the Keyboard through variations of the following command:

USER PROG 100 SPACE n1 SPACE n2 SPACE n3 ENTER

where:

n1 = filter range.

50 = 50 Hz, 125 = 125 Hz, 250 = 250 Hz, 1250 = 1250 Hz.
2500 = 2500 Hz, 5000 = 5 kHz, 12500 = 12.5 kHz,
25000 = 25 kHz, 5 = 50 kHz. 500 = 500Hz.

0 = auto select mode: if in F_{max} setting the filter range = $1/2_{max}$ (where F_{max} is from 100 Hz to 100 kHz); for Δf sampling mode, a filter in the range of .38 = F_c = .8 of F_{max} or filter bypass is selected. See Table E-1 for filter range automatically selected for Δf sample mode and block size.

99 = filter is set to bypassed mode.

n2 = select code, specifying the unit (or units) which are to respond to the command.

The interface channel may be shared with other units. Every unit using the channel is identified to the Processor (or software) by a select code, set by switches inside the device. The select code set by these switches is the address by which the device is known for communication with the Processor. See later paragraph, Select Code Switches.

Where n2 in the command is a positive number from 1 through 15; only the unit having the select code specified by that number will respond to the command.

Where n2 in the command is a negative number from -1 through -15*; all units whose select code is in the range 1 through |n2| will respond to the command.

*When addressing all filters, do not use a number higher than the highest numbered filter. For example, if you have 4 filter modules with select codes 1 through 4, then n2 would be -4 (not -15) to set all filters to the cutoff frequency of n1.

Where n2 is defaulted, only the unit having select code 1 will respond to the command.

n3 = ac/dc coupling: 1 = ac coupling, 0 (or default) = dc coupling.

NOTE

Each time a command is issued to change the filter range, a variable time delay occurs during which the filter output is automatically grounded to allow switching transients to settle. The time delay = 10 ($1/F_c$) + 1 ms = time required for reed relay to make proper contact.

USER PROGRAM 100

When User Program Y100 is used with n1 = 0, the filter will track the ADC sample rate each time an ANALOG IN command is used.

Example:

Set ADC SAMPLE MODE to MAX FREQ, kHz/us and MULTIPLIER to 100/200/5
Press USER PROG 100 SPACE 0 SPACE 1 ENTER
Filter module 1 will be set to 50 kHz.
Set MULTIPLIER to 50/100/10
Press ANALG IN ENTER
Filter will be set to 25 kHz.

If the filters are tracking the ADC sample rate and an overlay swap occurs, the filters will remain at their last settings but will stop tracking the ADC.

The following command retains the current filter settings, but stops the filter from tracking the ADC sample rate:

USER PROG 100 ENTER

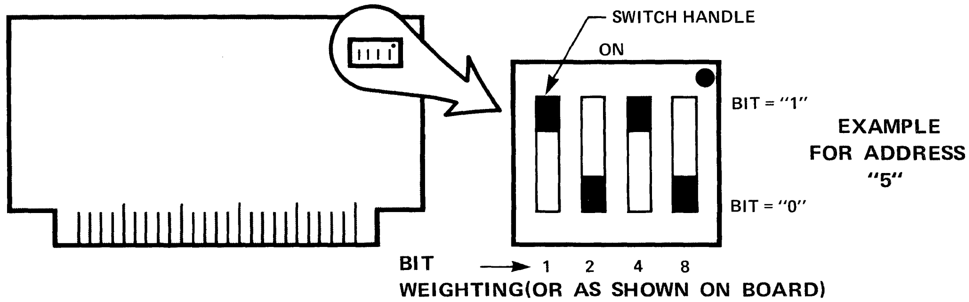
SELECT CODE SWITCHES

The select code is set by four switches on the Decoder board (A8), similar to those shown in the illustration below. When any of the switches is set toward the black dot (on the switch assembly body) or the "ON" label (adjacent to the switch position on some boards), the corresponding bit will be interpreted as a 1 by the Processor.

Select codes for the filter(s) are set at the factory in consecutive order beginning with 1. If a DAC is part of the system, its select code is normally 15 and should not be duplicated in the filter codes.

For convenience, front panel stickers are supplied to show the select code assigned to each filter.

Select Code Switches



DIGITAL TO ANALOG CONVERTER

The DAC module produces an analog waveform from the contents of a Fourier time-domain data block. The time and frequency scale of the signal produced at the DAC output is determined by the ADC sample rate; the DAC output rate is synchronized with the ADC sample rate. There are two DAC output modes: continuous and single shot. The continuous mode produces a periodic repetition of the stored waveform. The single shot mode produces one record of the data block as a single transient. In addition, a special form of the ANALOG OUT command on the system Keyboard creates a random time domain signal in a data block which can be used to produce a pseudo-random stimulus signal.

SETUP

The software for the DAC is resident in one of the system overlays. The select code for the DAC is set to 15 at the factory.

When power is first applied to the DAC, the GND lamp should light. When the DAC is operated, other lamps will light as described later in this section. If no lights appear, check the position of the rear panel LINE ON switch.

KEYBOARD COMMAND FORMAT

The following command format activates the DAC output from the Keyboard.

ANALG OUT n1 SPACE n2 SPACE n3 ENTER

where:

n1 specifies the data block used by the ANALOG OUT command. It cannot be the same data block used by the ANALOG IN command.

n2 determines the nature of the DAC command. If n2 = 1, the time-domain data is read out once (after the ADC triggering conditions are met). If n2 = 0, or defaulted, a continuous analog output is selected and the analog signal is read out repeatedly until an ANALOG OUT stop command or a RESTART is executed. If n2 ≥ 2 then the data block n1 is filled with random data with a peak value equal to n2 mV.

n3 sets the DAC filters to one of nine cutoff frequencies. If n3 = 0 the DAC filter is bypassed, that is, set to the throughput state. If n3 is defaulted, the DAC software automatically sets the filter cutoff to the optimum filter range. ($.38\ F_{max} \leq F_c \leq .8\ F_{max}$ where F_c = the cutoff frequency of the filter.)

The following command turns off continuous DAC output (analog stop command):

ANALG OUT ENTER

The analog stop command establishes a zero output value in the DAC and stops the output. When the DAC is started, the ON and RUN lights turn on. When the analog stop command is entered, the ON and RUN lights turn off and the amplifier output is set to ground. The filter is set to the throughput state (filters are bypassed) and the DAC is cleared and ready for the next command.

NOTE

If a 5477A System Control is part of the system be sure it is turned OFF whenever the ADC is not in remote.

Table E-1 shows the DAC filter settings for each blocksize versus ADC sample rates.

Table E-1. Low Pass or DAC Filter Setting vs. Block Size

ΔF	Total Time	4096	2048	1024	512	256	128	64
1 kHz	1 ms	N/A*	N/A	N/A	N/A	N/A	N/A	25 kHz
500 Hz	2 ms	N/A	N/A	N/A	N/A	N/A	25 kHz	12.5 kHz
200Hz	5 ms	N/A	N/A	N/A	25 kHz	12.5 kHz	5 kHz	5 kHz
100Hz	10 ms	N/A	N/A	25 kHz	12.5 kHz	5 kHz	5 kHz	2.5 kHz
50 Hz	20 ms	N/A	25 kHz	12.5 kHz	5 kHz	5 kHz	2.5 kHz	1.25 kHz
20Hz	50 ms	25 kHz	12.5 kHz	5 kHz	2.5 kHz	1.25 kHz	500 Hz	500 Hz
10 Hz	100 ms	12.5 kHz	5 kHz	2.5 kHz	1.25 kHz	500 Hz	500 Hz	250 Hz
5 Hz	200 ms	5 kHz	2.5 kHz	1.25 kHz	500 Hz	500 Hz	250 Hz	125 Hz
2 Hz	500 ms	2.5 kHz	1.25 kHz	500 Hz	250 Hz	125 Hz	50 Hz	50 Hz
1 Hz	1 second	1.25 kHz	500 Hz	250 Hz	125 Hz	50 Hz	50 Hz	50 Hz
.5 Hz	2 second	500 Hz	250 Hz	125 Hz	50 Hz	50 Hz	50 Hz	50 Hz
.2 Hz	5 second	250 Hz	125 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz
.1 Hz	10 second	125 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz
50 mHz	20 second	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz
20 mHz	50 second	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz
10 mHz	100 second	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz
5 mHz	200 second	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz
2 mHz	500 second	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz
1 mHz	1 ks	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz
.5 mHz	2 ks	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz
.2 mHz	5 ks	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz	50 Hz

*N/A indicates sample rate settings that are out of the range of the DAC.

Continuous Output Mode

The command format to produce a continuous periodic output from a time-domain data block is as follows:

ANALG OUT n1 ENTER

where:

n1 is the number of the time-domain data block to be output. No default of n1 is allowed.

Before the continuous analog output command is entered, the DAC output is zero. When the command is executed, the DAC starts to output block n1 and continues to do so in a periodic manner until the analog stop command is issued. The period is determined by the ADC sample rate.

At the time the DAC output is started, the level of the output is determined from the calibration factors of block n1. These factors are used to determine the max output level of the DAC. This results in a full-scale point in a data block being a full-scale value of the DAC output. If the full-scale value of the data block exceeds the max output level of the DAC (10V peak) a OV WHAT? error message is generated. (Error messages and halts are discussed in a following paragraph.) This error message will occur when the full-scale level of the data block exceeds 10 volts, even if no point in a data block exceeds 10 volts.

The output anti-aliasing filters are also set when the ANALOG OUT command is first executed. After the continuous analog output command is entered, the ADC continues to output the current block n1 without regard to new scale factors, filter settings, or changes in F_{max}. To change the output, an analog stop command must be given and a new ANALOG OUT command entered.

When a continuous output is running, an ANALOG IN command may be executed. The analog input samples are synchronous with the DAC output samples and both input and output are periodic in the same interval. However, the input starts when the ANALOG IN command is executed and is not locked to the start of the analog output record.

IMPORTANT

Before entering a continuous ANALOG OUT command, be certain the ADC TRIGGER SOURCE switch is in the FREE RUN position. This allows a continuous string of sample pulses to be generated by the ADC clock.

If an overlay swap takes place while the DAC is running, the DAC will be turned off before the overlay swap takes place.

One-Shot Output Mode

The command format to produce a one-shot output from a data block is as follows:

ANALG OUT n1 SPACE 1 ENTER

where:

n1 is the time-domain data block to output, and 1 is the code number for a single sweep output. If this code is defaulted, or 0, a continuous output is generated.

When the one-shot command is entered, a single sweep output is executed. At the end of the sweep, the last data sent to the DAC is set to zero volts so that the DAC output ends up at zero. The last data point is then restored in the data block to its original value. The sample out signal is held off until an ANALOG IN command is executed and the triggering conditions are met.

To synchronize one-shot output with the analog input, the LINE or EXT trigger modes are recommended. The one-shot command is affected in the following ways by various trigger modes.

ADC TRIGGER POSITIONS

FREE RUN:

In the free run trigger mode the DAC output generates one record immediately upon execution of an ANALOG OUT command. This mode does not allow the ANALOG IN and ANALOG OUT commands to be started simultaneously.

LINE:

The line trigger mode holds off the SAMP OUT signal from the ADC until the triggering conditions are met. Thus, the one-shot ouput will not occur until the ADC is commanded to input a record and the next line trigger occurs.

With the ADC TRIGGER set to LINE, the command sequence is as follows:

ANALG OUT n1 SPACE 1 ENTER

followed by:

ANALG IN n1 ENTER

The analog one-shot output and analog input will occur together when the triggering conditions are met (within 20 ms).

EXT:

The external trigger mode disables the sample out signal from the ADC until an ANALOG IN command is executed and the selected trigger condition for external occurs. This mode synchronizes the analog input and analog output with an external trigger. With the TRIGGER SOURCE switch in the EXT position, the command sequence is as follows;

ANALG OUT n1 SPACE 1 ENTER

followed by:

ANALG IN n1 ENTER

followed by an external trigger to the ADC. (n1 cannot be the same since analog output from a block cannot be input into the same block.)

INTERNAL (A):

The internal triggering mode relies on the signal entering INPUT CHANNEL A of the ADC. The DAC is held off until the triggering requirements are met at Input A, therefore this setting is not recommended.

HALT CODES AND ERROR MESSAGES

The system may indicate hardware failure or operator error by coming up with a HALT code in the Processor display register or a WHAT statement on the system Terminal. These are interpreted as follows:

Halt Codes

102010	No flag from Module Input/Output Bus (MIOB). This indicates a failure of the DAC to respond to a control command (no flag response to a second mode device enable from the interface card).
102012	An attempted data transmission to the DAC has failed to occur (no flag response to a first mode device command from the interface card).
102011	The power up command was not accepted by the DAC.
102007	DMA is not available. This indicates that another device has requested DMA7. Either the operator or the software has executed a command requiring an illegal combination of I/O devices.

Error Messages

Error messages are the result of incorrect commands or data block content. If any of these occur during the procedure, check for the indicated operational problem and try the procedure again.	
DL WHAT	The DAC output rate exceeds the I/O rate.
HW WHAT	An incorrect code word has been returned from the DAC.
BK WHAT	Block number out of range.
OV WHAT	Max voltage value of block greater than 10V.
FL WHAT	Incorrect filter range.
FQ WHAT	Data block is not in time-domain rectangular coordinates.

OUTPUT FILTERS

The DAC contains output filters to eliminate images, or aliasing products, due to the sample and hold nature of the DAC output. These filters have 9 fixed cutoff frequencies as listed in Table E-2. The Output filter can be set in the ANALOG OUT command by the n3 value described earlier in this section. If n3 is defaulted, a filter cutoff frequency is selected by the software according to: $.38\ F_{max} \leq F_c \leq .8\ F_{max}$.

For values of F_{max} from dc to 50 kHz, the filter cutoff is as shown in the table. If the sample rate is changed while an analog output command is executing, the filter range will not change. The filter range will only change when a new ANALOG OUT command is issued after the prior output is stopped. At the end of a one-shot output, the DAC output is set to zero and the filter set to the throughput mode.

Table E-2. DAC Output Filter Cutoff Frequencies

Range **R (n3)**	**Filter Cutoff** F_c
0	Throughput
50	50 Hz
125	125 Hz
250	250 Hz
500	500 Hz
1250	1.25 kHz
2500	2.5 kHz
5000	5 kHz
12500	12.5 kHz
25000	25 kHz
Default	$.38\ F_{max} < F_c < .8\ F_{max}$

GENERATING A RANDOM STIMULUS

A form of the ANALOG OUT command may also be used to create a random signal in a data block. This random stimulus is uniformly distributed over a peak value selected between 2 millivolts and 10 volts. This special form of the ANALOG OUT command only generates a new block of data and must be followed by the standard ANALOG OUT command to output the random stimulus. The command to generate a random signal in a data block is as follows:

ANALG OUT n1 SPACE n2 ENTER

where:

n1 is the data block in which the signal is to be created.

n2 is the peak level of the signal in millivolts. n2 must be greater than 2 and less than 10000.

When this signal is output through the DAC with the filter set to the throughput mode (filter out), the signal will have an approximately uniform amplitude distribution. If the output filter is enabled and set close to 1/2 F_{max}, the output signal will be approximately Gaussian in distribution.

Since a random number generator of very long chain length is used, each data block is random relative to other data blocks that may be created. In addition, the spectrum of each data block is also random and uncorrelated with every other spectrum so created. The expected value of the spectrum is flat within +/-1.0 dB over the range from dc to 1/2 F_{max}. The random signal created is designed for use as a stimulus in the transfer function or impedance function testing. Typical application programs for the continuous output mode are given in the following paragraphs.

NOTE

When the DAC output is running in the continuous mode, additional commands to generate new random blocks can be given into the block being output. These commands will change the random signal being output without the need to issue a new ANALOG OUT command. However, the output scale and amplitude of the random signal will remain unchanged until a new continuous or one-shot output command is given.

GENERATING A SINEWAVE

This setup and program demonstrates how the DAC can be used to generate a signal from a data block. The first part of the procedure shows how to set up a sinewave in a data block, and the second part demonstrates how to output the signal.

Using the Keyboard for data entry in the frequency domain, the sinewave is created as a single line in a data block and then is transformed to the time domain to form the sinewave. To create a sinewave, a sample rate at least four times (and an F_{max} at least two times) the sinewave frequency must be selected. For example, to create a sinewave of approximately 1 kHz, a sample rate of 5 kHz and an F_{max} of 2.5 kHz could be used. With a blocksize of 512, each frequency line would be 2500/256 Hz or 9.765625 Hz. Thus, to generate the 1 kHz sinewave, a spectral component in the frequency domain is entered in channel 102. The exact frequency then would be 102 x 9.765625 = 996.09 Hz.

To set the amplitude of the sinewave it must be remembered that the amplitude of a component in the frequency domain is 1/2 that of the time domain (to account for the symmetry of the frequency domain presentation of the Fourier transform). Thus, to achieve a 1-volt peak sinewave at the output, a 0.5 peak spectral component is entered in the frequency domain. The procedure for entering the signal described above is as follows:

BLOCK SIZE 512 ENTER
CLEAR 2 ENTER
KEYBOARD 2 SPACE 102 SPACE 102 ENTER
KEYBOARD -4 SPACE 4 ENTER
5000 SPACE 0 ENTER

The display will now show a single line of 0.5V amplitude. To transform the signal from the frequency to the time domain, press:

F 2 ENTER

Now the display will show a time-domain sinewave which is sampled at slightly more than four samples per cycle. For this reason, the pattern on the screen will not appear to be a simple sinusoid, but will, nevertheless, create a good sinewave.

The sinewave is now ready for output through the DAC. Set the ADC SAMPLE CONTROLS as follows:

SAMPLE MODE TO INT, MAX FREQ, kHz/μs
MULTIPLIER to 200/5/2.5
TRIGGERING to FREE RUN

On the system Keyboard, press:

ANALG OUT 2 ENTER

A 996.1 Hz sinewave of 2V p-p amplitude (actually .936V) will appear at the DAC output; this signal can be verified with an oscilloscope at the DAC output. The signal can be read back into the Fourier Analyzer by connecting the DAC output to the ADC input and executing an ANALOG IN command to any block except the output block (block 2). To stop the output, enter the analog stop command as follows:

ANALG OUT ENTER

GENERATING A PERIODIC RANDOM STIMULUS (To Measure Transfer and Impedance Functions)

The continuous DAC output can be combined with the random time-domain block generator to create a periodic random stimulus for transfer and impedance function testing.

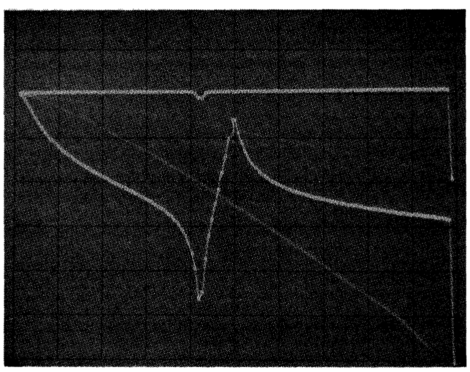
The random data block generator capability of the DAC allows you to have the speed and measurement advantages of a random noise source without the disadvantages of a pure random noise as a signal source. Random noise as a system stimulus has the advantage of reducing or minimizing the effects of non-linearities in the unit under test. The major disadvantage of a pure random signal source is that, by its nature, noise sources are not periodic in time and therefore generate leakage. The random data clocked out of the DAC is periodic in the measurement window and therefore is a leakage-free random noise signal.

A random stimulus is generated and applied to the unit under test until the transient response dies out and its output signal becomes periodic. By waiting for the transient response of the under unit test to dampen out before a data record is saved, leakage from this source is effectively eliminated. Using the Hanning or other window on every input/output data record is no longer required and will, in fact, degrade the measurement.

Typically, any transients will die out in the time required to output the random record two and, at the most, three times. After the device response has settled into periodicity an ANALOG IN command is executed and the input and output saved. The response measured after the transient response has settled out is used to form the cross-and auto-spectrum averages used in the transfer function calculation.

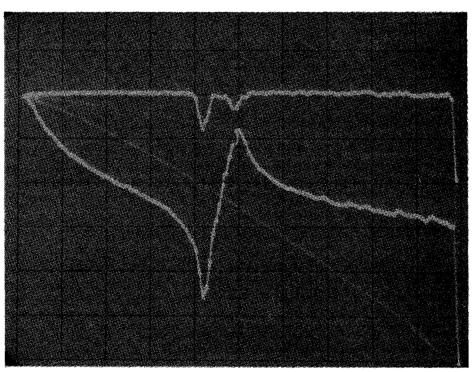
After one ensemble average of the spectra has been calculated, a new random time record is generated and entered into a data block. This new stimulus is uncorrelated with the prior stimulus and is output until the device has again settled before a new average is formed. After a sufficient number of averages are taken, the transfer and coherence functions are computed. The two photographs following show a comparison of a typical transfer and coherence function measurement made using pure random noise and "periodic" random noise.

The result of this technique is a measurement that converges faster, with fewer data averages, and with less variance on the measurement than can be obtained with a conventional random noise signal source.



Periodic Random Stimulus

This is a transfer function measured with the Keyboard Program above and using the DAC as a periodic random noise source. Note the improvement in the coherence function as a result of leakage-free measurements.



Pure Random Stimulus

This is the same transfer function measured with a random noise source, and the input and output records Hanned once. All other setup and test programs are the same. Even with the Hanning function, note that the leakage causes the coherence to drop at critical frequencies.

KEYBOARD PROGRAM

A 1000mV peak random data block is first generated in block 6 (line 21, of the program shown below). The data block is then output using the continuous ANALOG OUT command, line 26. This random data is output for three measurement periods by looping back to line 30. This technique minimizes leakages due to the induced transient response of the unit under test. Typically, the response to the startup transient dies out in the time required to output the random record two, and at the most three, times.

After the system response record has settled into periodicity, the last ANALOG IN command of the three is executed and this input and output data is used to form the cross and auto spectrum averages used in the transfer function calculation.

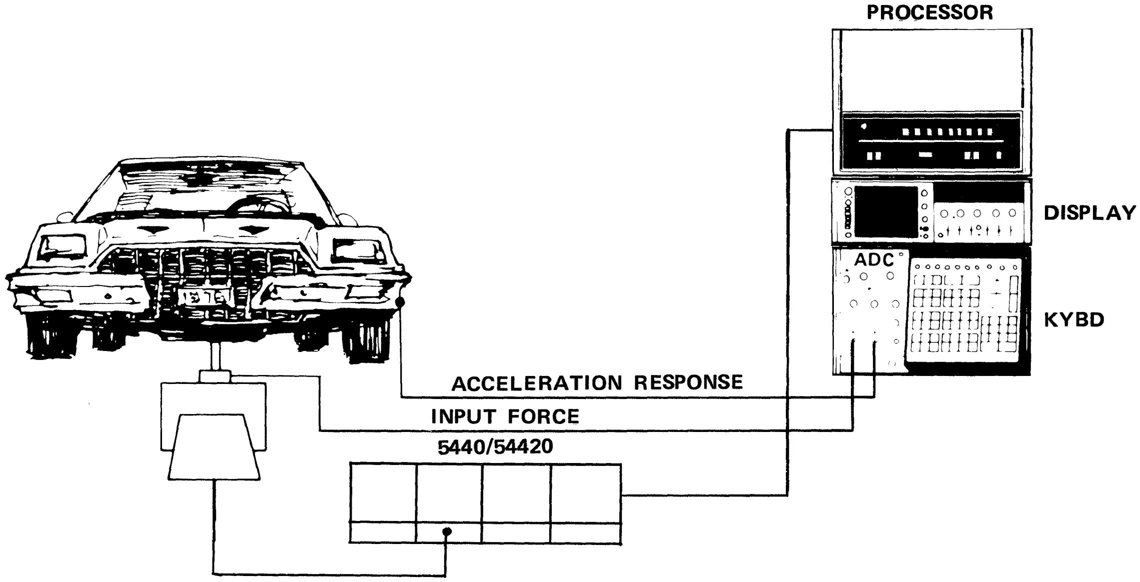
After one ensemble average of the spectra has been accumulated, a new random time record is generated and entered into data block 6 by jumping back to label 1 from line 60. This new stimulus is uncorrelated with the prior stimulus and is output until the test unit output has settled to a periodic output before new measurements are made. After 16 measurement records are taken, the transfer and coherence functions are computed. The program to execute this measurement method is as follows:

	1	L	0			LABEL 0 ENTER
	5	CL	2			CLEAR 2 ENTER
	9	CL	3			CLEAR 3 ENTER
	13	CL	4			CLEAR 4 ENTER
	17	CL	5			CLEAR 5 ENTER
Generate Random data in block 6 at 1V peak	21	B	6	1000		ANALOG OUT 6 SPACE 1000 ENTER
Start DAC output	26	B	6			ANALOG OUT 6 ENTER
A three record wait loop to allow test system impulse response to settle	30	L	1			LABEL 1 ENTER
	34	RA	0	4		ANALOG IN 0 SPACE 4 ENTER
	39	#	1	3		COUNT 1 SPACE 3 ENTER
Generate new time domain random data record	45	B	6	1000		ANALOG OUT 6 SPACE 1000 ENTER
	50	F	0	1		F 0 1 ENTER
	54	SP	0	2	2	POWER SPECTRUM 0 SPACE 2 SPACE 2 ENTER
	60	#	1	16	0	COUNT 1 SPACE 16 SPACE 0 ENTER
Turns off continuous output from DAC	66	B				ANALOG OUT ENTER
Clears out upper half of the data blocks 4 and 5 to eliminate aliasing signal components and low level signals above $1/2F_{max}$	69	CL	4	128	256	CLEAR 4 SPACE 128 SPACE 256 ENTER
	75	CL	5	128	256	CLEAR 5 SPACE 128 SPACE 256 ENTER
	81	CH	0	2	2	TRANSFER FUNCTION 0 SPACE 2 SPACE 2 ENTER
	87	•				END ENTER

This program shown operates on a blocksize of 512. It is a conventional transfer function/coherence function program as described in Section 3 with the following exceptions. In step 21, a random record of amplitude 1 volt is created in block 6. In step 26, a continuous analog output is started. The loop to label 1 around the RA command in step 39, assures that three output and input records exist before the last input/output records are kept for processing. Immediately after the analog input (step 45) is completed, a different random record of the same amplitude is created in block 6. The stimulus and response records are now Fourier transformed and the spectrums averaged. Step 60 repeats the process for 16 averages. On each new average the repeat loop (step 39) allows the new random record to become periodic. Because each random record is different from each previous record the stimulus can be considered random. And because each output record is allowed to become periodic in the window period, the response can also be considered periodic in the measurement window and no Hanning window is required.

To run this program, set the ADC sample rate to an F_{max} value that is twice the bandwidth required. Steps 69 and 70 will clear the data above 1/2 F_{max} where the output filter on the DAC does not provide sufficient excitation energy and aliasing spectral components contaminate the measurement. For a blocksize other than 512, steps 69 and 70 should be changed accordingly.

The output of the DAC is connected to drive the device under test. Channel A of the ADC is connected to the forcing point on the unit under test (see diagram below) and channel B to the response point.



INPUT/OUTPUT MODES

The following paragraphs give a brief description of the input/output modes (for data, text, and Keyboard Programs) that can be used with various optional peripheral devices (e.g., Printer, High-Speed Photo-reader, or High-Speed Punch).

These peripherals can be connected directly to the standard system with no software modifications. Refer to your System Configuration Notice for the proper I/O channels for these devices. See Section 3 PHOTO READR, PRINT, and PUNCH keys for command formats.

DEVICE SELECTION

Switch register bits on the Processor front panel may be used to select the input/output device to be used for various operations as follows:

Input Device Selection

The PHOTO READER command assumes that data will be entered via photoreader. If this device is not part of the System, the PHOTO READER command will apply to the Terminal, but only if bit 3 of the Processor S-Register is lighted. Other Keyboard input commands (such as KEY BOARD data or command entry and RPLAC or INSRT key entry) are also affected by the state of this bit. The following table shows the effect of bit 3 on input device selection for these commands:

Command	Bit 3 = 0	Bit 3 = 1
PHOTO READR	Photoreader	Terminal
KEY BOARD	Terminal	Photoreader
RPLAC or INSRT	Terminal	Photoreader
"GOLD" INPUT	Terminal	Photoreader

Note: Bit 3 also affects the REPLACE and INSERT commands as shown above when the Text Buffer Editor program (USER PROG 5803) is used.

Output Device Selection

The PUNCH command assumes that data will be punched on the high speed punch. If this device is not part of the system, the PUNCH command will apply to the Terminal or to the Printer if bits 4 and/or 6 of the Processor S-Register are set as shown below. Various other commands are also affected by the state of bits 4 and 6; the following table shows the effect of these bits on the output device selection for these commands:

Command	Bits 4 and 6 = 0	Bit 6 = 1	Bit 4 = 1
PUNCH	High Speed Punch	H.S. Punch	Terminal
LIST	Terminal	Printer	High Speed Punch
PRINT	Terminal	Printer	High Speed Punch
"GOLD" LIST	Terminal	Printer	High Speed Punch
"GOLD" OUTPUT	Terminal	Printer	High Speed Punch
"GOLD" TEXT	Terminal	Printer	High Speed Punch

Note: If both bits 4 and 6 are set, the Printer will be selected as the output device.

Bits 4 and 6 also affect the LIST command as shown above, when the Text Buffer Editor program (USER PROG 5803) is used. in addition, most system messages (except for WHAT? and other error messages) may be directed to an output device other than the Terminal by using bits 4 or 6.

KEYBOARD PROGRAM I/O WITH PAPER TAPE

Punching a Program on Paper Tape

1. Turn on high-speed punch.
2. Generate adequate tape leader.
3. Enter the LIST command for the type of listing desired (see previous pages) but do not press ENTER.
4. Set bit 4 on the Processor S-Register.
5. Now press the ENTER key. The LIST command will punch the program or portion of program on the paper tape.
6. Press the FEED HOLES switch to generate tape trailer.
7. Clear bit 4 on the Processor S-Register.

Reading a Program on Paper Tape

1. Place paper tape in reader and press READ.
2. Set bit 3 on the Processor S-Register and then press STORE.
3. Give the appropriate REPLACE command on the Keyboard to enter the program into program memory.

   RPLAC LINE n1 ENTER

   where:

   n1 is the starting line at which the program is to be entered. Remember if you are entering this program immediately behind another program, then you must immediately re-enter the last line of that program after giving the REPLACE command, or it will be deleted.
4. The tape should now read in.
5. When the tape stops, bit 3 on the Processor will be cleared automatically.

DATA BLOCK I/O WITH PAPER TAPES

Loading Tape via Photoreader

1. Place data tape roll in photoreader tape holder, feed holes toward the instrument.
2. Press POWER pushbutton.
3. Press LOAD pushbutton.
4. Run the tape leader underneath the wire guide and through the pair of feed rollers.
5. Press READ pushbutton.
6. Give photoreader input command.

Data Format of Punched Tape

The data format on any tape produced by the PUNCH command is show below.

Date format on punched tape output

```
P
SF      -4      0      9   8192
(        0)  32767  16384      0
(        8)      0      0      0
```

SF stands for "scale factors"

–4 is "k" in the expression "10^k". All data words are multiplied by 10^k giving the number system called "floating point on a block basis". Thus, in the above example, all data words are multiplied by $10^{-4} = 0.0001$.

0 is coordinate code from the table at the end of this Appendix. In the above example, data is in time domain, rectangular coordinates.

9 frequency code from the table at the end of this Appendix, which expresses data sampling parameters in terms of SAMPLE MODE AND MULTIPLIER switch settings on the ADC. In the above example, pertinent data sampling parameters were F_{max} of 0.5 Hz, Δt of 1000 ms.

8192 is block calibrator which multiplies all data words in block.

32,767
16,384 are data words. Data word system is as follows:
0 stands for 0; 32767 stands for 1.

To convert data words into actual physical values:

Use the following formula:

$$\frac{\text{data word}}{32{,}767} \times \frac{\text{block calibrator}}{32{,}767} \times 10^k$$

Thus, in the above example, actual physical value of first data word, 32767 is:

$$\frac{32767 \times 8192}{(32767)^2} \times 10^{-4} = .25 \times 10^{-4}$$

Actual physical value of second data word is:

$$\frac{16384 \times 8192}{(32767)^2} \times 10^{-4} = .125 \times 10^{-4}$$

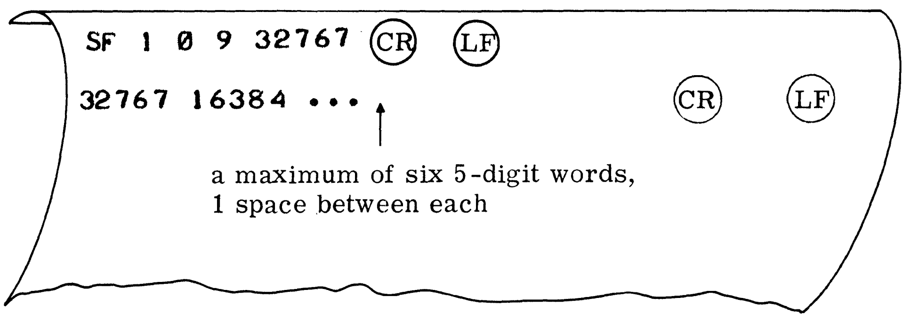
DATA FORMAT REQUIRED FOR PUNCHED TAPE INPUT

All data tapes generated by the PUNCH command on the Fourier Analyzer can be re-entered by PHOTO READER command. A tape generated by the PRINT command cannot be entered by PHOTO READER command because data formats are different. Tapes from other computers can be entered if they have the correct data format, as spelled out below.

Important Note: When the tape contains frequency domain data, there must be a zero for the imaginary or phase (dc) term and for the imaginary or phase value of the highest frequency (F_n) when these points are part of the data. For example:

dc, 0, 1st real, 1st imag, 2nd real, 2nd imag, ---, last real, 0 (last imag.)

Data format on punched tape input



CR = carriage return

LF = line feed

SF stands for "scale factor" (may be any two ASCII characters).

1 is "k" in the expression "10^k".

0 coordinate code from the table at the end of this Appendix. In the above example, data is in time domain, rectangular coordinates.

9 any arbitrary number from 0 through 16,383 or a frequency code from the table at the end of this Appendix. The table expresses data sampling parameters in terms of SAMPLE MODE and MULTIPLIER switch settings on the ADC. All tapes produced by the Fourier Analyzer thus contain a frequency code. But it is not required in tapes from other computers so long as any number from 0 through 16,383 is inserted in its place. (Data entered with invalid frequency codes may not be plotted or examined via the cursor, however.)

In the example above, pertinent data sampling parameters were F_{max} of 0.5 Hz, Δt of 1000 ms.

32,767 is the data block calibrator associated with every data point in the block. The calibrator word system is as follows: 0 = 0; 32,767 = 1. So, in the above example the block calibrator = 1. If this number is defaulted, the value 32,767 is automatically inserted. You will not need to be concerned with it other than to know it is present and will be punched out when the PUNCH command is given.

32,767
16,384 are data words. Data word system is as follows: 0 = 0; 32,767 = 1. So, in above example where block calibrator = 1, the first data word = $32{,}767 \times 10^1 = 32{,}767 \times 10 = 327{,}670 = 10$. Second data word is $16{,}384 \times 10^1 = 16{,}384 \times 10 = 16{,}3840 = 5$. No data words can have a magnitude greater than 32,767.

To convert any set of data into this number system (block calibrator = 1, i.e., 32,767):

1. Choose a k that will make 10^k slightly larger than the largest number in the data.

   For example, if data is 0.5, 1, 2, 9, choose k = 1, because $10^1 = 10$, which is one higher than 9.

   Second example: if data is 0.0005, 0.001, 0.002, 0.009 choose k = -2 because $10^{-2} = 0.01$.

2. Then convert data into Fourier Analyzer data words per the following formula:

   $$\frac{\text{data value}}{10^k} \times 32{,}767 = \text{Fourier Analyzer data word}$$

   For example, if data value = 2, as in the first example above, then the data word is:

   $$\frac{2}{10} \times 32{,}767 = 6{,}553$$

   And, from second example, taking data value = 0.001, the data word is:

   $$\frac{0.001}{0.01} \times 32{,}767 = 3{,}277$$

DATA REPRESENTATION

Data values in the system are represented by a combination of scale factor, calibrator, and data word.

The routine that serves KEYBOARD data entry and PRINTED data output does not allow access to, or provide information about, the value of the calibrator. PRINT output operations print data words that already have the calibrator compensation included, and require only correct application of the scale factor value to obtain the actual data value.

The routine that serves PHOTOREADER data entry and PUNCHED data output allows direct access to the calibrator, in addition to the scale factor and data word values. PHOTOREADER input operations allow the calibrator, as well as the scale factor or data word value to be changed. PUNCH output operations punch (or print) the calibrator value, and data words that represent (in decimal) the actual values of the memory contents of those locations; these outputs require correct application of both the calibrator and scale factor values to the data word value in order to obtain the actual data value.

The correct application of the scale factor, and of the calibrator, depends on the kind of data you are working with, as indicated by the coordinate code associated with the data. Some examples of PRINTED and PUNCHED data outputs are presented below, along with their calibration factors. The printed representations of PUNCHED data were obtained by setting bit 4 of the Processor S-Register, to select the alternate output device.

By inference, KEYBOARD data inputs should have the same format as the PRINTED data outputs, and PHOTOREADER data inputs should have the same format as the PUNCHED data outputs. Note, however, that input operations ignore numbers enclosed in parentheses (the channel numbers in the output examples).

Each of the examples (Figures E-1 through E-7) in this Appendix includes a "header" indicating how the data was put out, and the locations from which it was obtained. For example:

W 0 6 26 means that the data was PRINTED (W), and came from block 0, channels 6 through 26.

P 0 6 26 means that the data was PUNCHED (P), and came from block 0, channels 6 through 26.

The first channel in the data block is channel 0. The last word in a data block of size n is n-1 in the time domain, or n/2 in the frequency domain.

In each of the examples presented here, the PRINTED and PUNCHED examples represent the same data. Note that the PRINTED data provides calibrated data values (directly), whereas the PUNCHED output consists of raw data values which must be multiplied by a calibrator value to get the actual data value.

SF indicates the line containing the Scale Factors required to fully calibrate the data.

The number in parentheses at the beginning of each line following SF indicates the channel number of the first location whose data is presented on that line. For example:

(14) indicates that the first data on that line came from channel 14 in the selected data block.

Figure E-1. Coordinate Code 0 — Data Representation

Coordinate Code Ø = Single Precision, Time, Linear, Rectangular Data

1 = Scale Factor. See examples below for use.

2 = Coordinate Code.

3 = Frequency Code. **Meaningless, in this case, since data was entered manually.**

4 = Calibrator. See PUNCHed data example below.

PRINTed Output

```
                    1         2         3
W Ø  6  26
SF          1        Ø        15
(           6)       5        1Ø       2Ø       5Ø      [1ØØ]     2ØØ      5ØØ     1ØØØ
(          14)    2ØØØ         Ø        Ø        Ø        -1       -1       -2       -5
(          22)     -1Ø       -5Ø     -1ØØ     -2ØØ      -5ØØ
```

Data (one word per channel)

Calibration: data x $1Ø^{\text{scale factor}}$ Channel 1Ø = 1ØØ x $1Ø^1$ = 1ØØØ

PUNCHed Output

```
                    1         2         3         4
P Ø  6  26
SF          5        Ø        15      6554
(           6)      81       163       327      818     [1637]    3275     8189    16378
(          14)   32767         Ø         Ø        Ø        -4      -14      -29      -79
(          22)    -159      -814     -1634    -3274     -8189
```

Data (one word per channel)

Calibration: $\frac{\text{data}}{32767} \times \frac{\text{calibrator}}{32767} \times 1Ø^{\text{scale factor}}$

Channel 1Ø = $\frac{1637}{32767} \times \frac{6554}{32767} \times 1Ø^5 = 999.26$

Figure E-2. Coordinate Code 2 — Data Representation

Coordinate Code 2 = Single Precision, Time, Log, Rectangular Data

① = Scale Factor. See examples below for use.

② = Coordinate Code.

③ = Frequency Code. Meaningless, in this case, since data was entered manually.

④ = Calibrator. See PUNCHed data example below.

PRINTed Output

```
                    ①          ②          ③
W Ø 6 26
SF       1ØØ          2          15
(          6)   -6611    -6ØØ3    -5399   [-46Ø3]   -4ØØØ    -3399    -26Ø2    -2ØØ1
(         14)   -1399   -1ØØØØ   -1ØØØØ   -1ØØØØ    -9222    -8135    -75Ø2    -6633
(         22)   -6Ø25    -46Ø7    -4ØØ2    -3398    -26Ø2
```

Data (one word per channel)

Calibration: $\frac{\text{data}}{100}$ + scale factor Channel 9 = $\frac{-4603}{100} + 100 = 53.97$ dB

PUNCHed Output

```
                    ①          ②          ③        ④
P Ø 6 26
SF         5          2          15     -458Ø
(          6)  -17Ø8Ø   -15Ø9Ø   -1311Ø  [-1Ø5ØØ]   -8526    -6556    -3946    -1974
(         14)      -2   -32767   -32767   -32767   -25638   -22Ø74   -2ØØØ2   -17152
(         22)  -15162   -1Ø514    -8532    -6554    -3946
```

Data (one word per channel)

$$\text{Calibration: } 100\left[\frac{\text{data}}{32767} + \frac{\text{calibrator}}{32767}\right] + 20 \times \text{scale factor}$$

$$\text{Channel 9} = 100\left[\frac{-10500}{32767} + \frac{(-4580)}{32767}\right] + 20 \times 5 = 53.98 \text{ dB}$$

Figure E-3. Coordinate Code 4 — Data Representation

Coordinate Code 4 = Single Precision, Frequency, Linear, Rectangular Data

1 = Scale Factor. See examples below for use.

2 = Coordinate Code. See description above, or Table E-1.

3 = Frequency Code. Meaningless, in this case, since data was entered manually. See Table E-2 for codes for data entered via ADC.

4 = Calibrator. See PUNCHed data example below.

PRINTed Output

Channel 6, imaginary part

Channel 6, real part

```
W Ø 3 13
SF      1      4      15
(      3)      5      9      2Ø     49     1ØØ    199    5ØØ    999
(      7)   2ØØØ      Ø      Ø      Ø      -1     -1     -2     -5
(     11)    -1Ø    -5Ø   -1ØØ   -2ØØ    -5ØØ  -1ØØØ
```

Data (two words per channel)

Calibration (real or imaginary): data x $1Ø^{\text{scale factor}}$

Channel 1Ø imaginary = 49 x $1Ø^{1}$ = 49Ø volts

PUNCHed Output

```
P Ø 3 13
SF      5      4      15    6554
(      3)     81     163    327    818    1637    3275    8189   16378
(      7)  32767      Ø      Ø      Ø      -4     -14     -29     -79
(     11)   -159    -814  -1634  -3274   -8189  -16379
```

Data (two words per channel)

Calibration (real or imaginary): $\frac{\text{data}}{32767} \times \frac{\text{calibrator}}{32767} \times 1Ø^{\text{scale factor}}$

Channel 1Ø imaginary = $\frac{818}{32767} \times \frac{6554}{32767} \times 1Ø^{5}$ = 499.33 volts

Figure E-4. Coordinate Code 5 — Data Representation

Coordinate Code 5 = **Single Precision, Frequency, Linear, Polar Data**

① = **Scale Factor. See examples below for use.**

② = **Coordinate Code.**

③ = **Frequency Code. Meaningless, in this case, since data was entered manually.**

④ = **Calibrator. See PUNCHed data example below.**

PRINTed Output

Channel 6, magnitude

Channel 6, phase

```
N Ø 3 13
SF     -1      5      15
(       3)   149Ø     489   1911  -2334   2271  -5159   2559  -7989
(       7)   2767  -1Ø821   2888 -13655   2921 -16493   2866  16665
(      11)   2726   1382Ø   25Ø6  1Ø973   2216   8121
```

Data (two words per channel)

Calibration (magnitude): data x $10^{\text{scale factor}}$

Channel 6 magnitude = 2559×10^{-1} = 255.9 volts

Calibration (phase): data x .Ø1

Channel 6 phase = -7989 x .Ø1 = -79.89 degrees

NOTE: **In all cases, the phase term is multiplied by .Ø1 to get the actual phase. The scale factor is never used to obtain phase.**

PUNCHed Output

```
P Ø 3 13
SF      3      5      15  1Ø24Ø
(       3)  15616     892  2ØØ34  -4248  238Ø6  -9391  2683Ø -14543
(       7)  29ØØ8  -19698  3Ø282 -24858  3Ø628 -3ØØ24  3ØØ46  3Ø338
(      11)  28574   25168  26272  19976  23234  14784
```

Data (two words per channel)

Calibration (magnitude): $\frac{\text{data}}{32767} \times \frac{\text{calibrator}}{32767} \times 10^{\text{scale factor}}$

Channel 6 magnitude = $\frac{26830}{32767} \times \frac{10240}{32767} \times 10^{3}$ = 255.89 volts

Calibration (phase): $\frac{\text{data}}{32767} \times 180$

Channel 6 phase = $\frac{-14543}{32767} \times 180$ = -79.89 degrees

Coordinate Code 7 = Single Precision, Frequency, Log, Polar Data

1 = Scale Factor. See examples below for use.

2 = Coordinate Code.

3 = Frequency Code. Meaningless, in this case, since data was entered manually.

4 = Calibrator. See PUNCHed data example below.

PRINTed Output

Channel 6, magnitude

Channel 6, phase

```
4 Ø 3 13
SF      6Ø     7      15
(        3)  -1655    489   -1439  -2334   -1289  -5159   -1186   -7989
(        7)  -1117 -1Ø821   -1Ø8Ø -13655   -1Ø7Ø -16493   -1Ø87   16665
(       11)  -113Ø  1382Ø   -12Ø4  1Ø973   -131Ø   8121
```

Data (two words per channel)

Calibration (magnitude): $\frac{\text{data}}{100}$ + scale factor Channel 6 magnitude = $\frac{-1186}{100} + 60 = 48.14$ dB

Calibration (phase): $\frac{\text{data}}{100}$ Channel 6 phase = $\frac{-7989}{100} = -79.89$ degrees

PUNCHed Output

```
P 4 3 13
SF       3      7      15   -3312
(        3)  -211Ø    892   -14Ø2   -4248    -91Ø   -9391    -572  -14543
(        7)   -348 -19698    -226  -24858    -194  -3ØØ24    -248   3Ø338
(       11)   -39Ø  2516Ø    -63Ø   19976    -98Ø   14784
```

Data (two words per channel)

Calibration (magnitude): $100\left(\frac{\text{data}}{32767} + \frac{\text{calibrator}}{32767}\right) + (20 \times \text{scale factor})$

Channel 6 magnitude = $100\left(\frac{-572}{32767} + \frac{-3312}{32767}\right) + (20 \times 3) = 48.15$ **dB**

Calibration (phase): $\frac{\text{data}}{32767} \times 180$ Channel 6 phase = $\frac{-14543}{32767} \times 180 = -79.89$ **degrees**

Figure E-6. Coordinate Code 12 — Data Representation

Coordinate Code 12 = Double Precision, Frequency, Linear, Rectangular Data

1 = Scale Factor. See examples below for use.

2 = Coordinate Code.

3 = Frequency Code. Meaningless, in this case, since data was entered manually.

4 = Calibrator. See PUNCHed data example below.

PRINTed Output

Channel 8, LSB

Channel 8, MSB

1 2 3

```
W Ø 5 16
SF     -6    12   1737
(       5)  1582     Ø   1477     Ø   1360     Ø   1233     Ø
(       9)  11Ø2     Ø    968     Ø    835     Ø    7Ø6     Ø
(      13)   584     Ø    47Ø     Ø    368     Ø    277     Ø
```

Data (two words per channel)

Calibration: (MSB + LSB) x $1Ø^{\text{scale factor}}$ Channel 8 = (1233 + Ø) x $1Ø^{-6}$ = .ØØ1233 V^2

PUNCHed Output

1 2 3 4

```
P Ø 5 16
SF     -2     12    1737    6396
(       5)  26556       Ø  24788       Ø  22814       Ø  2Ø691       Ø
(       9)  1849Ø       Ø  16246       Ø  14ØØ6  -32767  11848  -32767
(      13)   9793  -32767   7886  -16384   6166  -32767   4645   16384
```

Data (two words per channel)

Calibration (LSB positive):

$$\left(\frac{\text{MSB}}{32767} + \frac{\text{LSB}}{32767 \times 65536}\right) \times \left(\frac{\text{calibrator}}{32767}\right) \times 1Ø^{\text{scale factor}}$$

$$\text{Channel 16} = \left(\frac{4645}{32767} + \frac{16384}{32767 \times 65536}\right) \times \left(\frac{6396}{32767}\right) \times 1Ø^{-2} = 2.767223165 \times 10^{-4}\ V^2$$

Calibration (LSB negative)

$$\left(\frac{\text{MSB}}{32767} + \frac{\text{LSB} + 65536}{32767 \times 65536}\right) \times \left(\frac{\text{calibrator}}{32767}\right) \times 1Ø^{\text{scale factor}}$$

$$\text{Channel 15} = \left[\frac{6166}{32767} + \frac{(-32768) + 65536}{32767 \times 65536}\right] \times \frac{6396}{32767} \times 1Ø^{-2} = 3.673447426 \times 1Ø^{-4}\ V^2$$

Figure E-7. Coordinate Code 14 — Data Representation

Coordinate Code 14 = Double Precision, Frequency, Log, Rectangular Data

1 = Scale Factor. See examples below for use.

2 = Coordinate Code.

3 = Frequency Code. Meaningless, in this case, since data was entered manually.

4 = Calibrator. See PUNCHed data example below.

PRINTed Output

Channel 8, MSB

Channel 8, LSB

```
                 (1)        (2)      (3)
4 Ø  5  16
SF      -2Ø        14      1737
(        5)    -8Ø2        Ø    -831        Ø    -868        Ø    -91Ø        Ø
(        9)    -959        Ø   -1Ø15        Ø   -1Ø79        Ø   -1152        Ø
(       13)   -1235        Ø   -1329        Ø   -1435        Ø   -1558        Ø
```

Data (two words per channel)

Calibration: $\frac{\text{data}}{100}$ + scale factor   Channel 16 = -35.58 dB

PUNCHed Output

```
               (1)        (2)      (3)      (4)
P Ø  5  16
SF      -2         14      1737    -2325
(        5)    -33Ø        Ø    -398        Ø    -516        Ø    -655        Ø
(        9)    -815        Ø    -998        Ø   -121Ø        Ø   -1448        Ø
(       13)   -1719        Ø   -2Ø27        Ø   -2377        Ø   -278Ø        Ø
```

Data (two words per channel)

$$\text{Calibration: } 100\left(\frac{\text{data}}{32767} + \frac{\text{calibrator}}{32767}\right) + (10 \times \text{scale factor})$$

$$\text{Channel 16} = 100\left(\frac{-2780}{32767} + \frac{-2325}{32767}\right) + [10 \times (-2)] = -35.5769909 \text{ dB}$$

NOTE ON COORDINATE CODES

The 5451 system software assumes that any data having coordinate code "12" (rectangular, linear, double-precision) was obtained by performing a conjugate multiply operation on data that had coordinate code "4" (rectangular, linear, single-precision), and assumes that this data is calibrated in terms of "V^2" (instead of "V") regardless of how this coordinate code "12" data was obtained. This can cause unexpected results when data with coordinate code "12" is entered manually and converted to log magnitude.

An example, comparing conversion of manually-entered single-precision data and manually-entered double-precision data to log magnitude is presented below.

If the data values 1,10,100,1000, and 10000 are entered with coordinate code "4", and a log magnitude operation is then performed, the coordinate code will be changed to "7" (polar, log, single-precision), and the vertical spacing between the dots (in a MAGNITUDE display) will be 2 cm.

If the same data values are entered with coordinate code "12", and a log magnitude operation is then performed, the coordinate code will be changed to "14" (rectangular, log, double-precision), and the vertical spacing between the dots (in a REAL display) will be 1 cm.

In reality, these two data displays represent the same difference (in dB) between points, since the coordinate code "7" vertical calibration is —

$dB_7 = 20 \log \frac{V_n}{V_{ref}}$ where V_n is the original data value (in volts) V_{ref} is 1 volt peak

and the coordinate code "14" vertical calibration is —

$dB_{14} = 10 \log \frac{(V_n)^2}{(V_{ref})^2}$ where these terms have the same meaning as given above.

Consider the examples below—

- Using the data values "1" and "100", entered with coordinate code "4", LOG MAG gives the following deflections (and coordinate code "7")—

  for "1": $\frac{(20 \log \frac{1}{1})}{10} = 0$ cm for "100": $\frac{(20 \log \frac{100}{1})}{10} = 4$ cm

- Using the data values "1" and "100", entered with coordinate code "4", conjugate multiplication gives the values "1" and "10000" (and coordinate code "12"); a LOG MAG operation on this data gives the following deflections (and coordinate code "14")—

  for "1": $\frac{(10 \log \frac{1}{1})}{10} = 0$ cm for "100^2": $\frac{(10 \log \frac{10000}{1})}{10} = 4$ cm

- Using the data values "1" and "100", entered with coordinate code "12", LOG MAG gives the following deflections (and coordinate code "14")—

  for "1": $\frac{10 (\log \frac{1}{1})}{10} = 0$ cm for "100": $\frac{10 (\log \frac{100}{1})}{10} = 2$ cm

Coordinate Codes

Coordinate Code	Time	Frequency	Rectangular	Polar	Log	Linear	Single Precision	Double Precision
0	x		x			x	x	
2*	x		x		x		x	
4		x	x			x	x	
5		x		x		x	x	
7*		x		x	x		x	
12†		x	x			x		x
14*		x	x		x			x

*Log data cannot be entered via KEYBOARD data entry mode.

†Double-precision data entered via KEYBOARD data entry mode must be between +32767 and -32768.

Frequency Codes

Freq. Code	F_{max} (MAX FREQ)	Δt (Δ TIME)
47	100 kHz	5 μs
43	50 kHz	10 μs
39	25 kHz	20 μs
46	10 kHz	50 μs
42	5 kHz	100 μs
38	2.5 kHz	200 μs
45	1 kHz	500 μs
41,8	.5 kHz	1000 μs
37,4	.25 kHz	2000 μs
33,15	0.1 kHz	5000 μs
11	50 Hz	10 ms
7	25 Hz	20 ms
14	10 Hz	50 ms
10	5 Hz	100 ms
6	2.5 Hz	200 ms
13	1 Hz	500 ms
9	0.5 Hz	1000 ms
5	0.25 Hz	2000 ms
1	0.1 Hz	5000 ms

Freq. Code	Δt (Δ FREQ)	T (TOTAL TIME)
56	1000 Hz	1 ms
52	500 Hz	2 ms
63	200 Hz	5 ms
59	100 Hz	10 ms
55	50 Hz	20 ms
62	20 Hz	50 ms
58	10 Hz	100 ms
54	5 Hz	200 ms
61	2 Hz	500 ms
57,24	1 Hz	1000 ms
53,20	.5 Hz	2000 ms
49,31	.2 Hz	5000 ms
27	100 mHz	10 s
23	50 mHz	20 s
30	20 mHz	50 s
26	10 mHz	100 s
22	5 mHz	200 s
29	2 mHz	500 s
25	1 mHz	1000 s
21	0.5 mHz	2000 s
17	0.2 mHz	5000 s

0 = general code, 99 = BSFA data, see Section 5

APPENDIX F

USING A MULTIPLEXER IN HP 5451 FOURIER ANALYZER SYSTEMS

GENERAL

Several customers use a multiplexer with the 5451C system. The intent of this appendix is to give a guideline on the installation and operation of a multiplexer based on a Preston Multiplexer. The standard HP 5451C system software allows for the operation of the Preston Model Series GM910 Multiplexer. The normal simultaneous sample and hold, sequential scan mode is used for acquiring data from multiple channels simultaneously.

NOTE

There is a part number and serial number tag on the rear panel of multiplexer. Units with a part number of "59310-" only operate with 200 kHz digitizers in the HP 5466 ADC. Units with a part number of "54550-" are compatible with both 100 kHz and 200 kHz digitizers in the HP 5466 ADC.

INSTALLATION

1. Check to be sure that the system ADC A9 board is HP P/N 05466-60009 (or 05466-60016).

2. On the above mentioned A9 board, check to be sure A9 (5, E) is connected to P15 (48) and J25 and that the "MULTX/ADC/INT" jumper is in the "MULTX" position.

3. Place the multiplexer unit in a suitable place in the rack, if a rack is used.

   Note: The multiplexer and system ADC should use a common power line ground.

4. Connect ADC input BNC for channel A to BNC J39 on rear panel of multiplexer.

5. Connect the multiplexer to the back of the 5475A as follows:

System ADC Mainframe	to	Multiplexer Rear Panel
MULTX SAMPE		J34
$\overline{\text{DIGITIZE}}$		J38
$\overline{\text{DAR}}$		J37

6. Connect the multiplexer input BNC's as discussed below.

NOTE

All input channels of multiplexer must be connected to a device or terminated in less than 1K ohm.

If you want to operate system without multiplexer as input device, connect MULTX SAMPLE and $\overline{\text{DIGITIZE}}$ lines together using a BNC-to-BNC cable.

If single channel operation of ADC input A is all that is required, then the multiplexer J34 (MULTX SAMPLE) and J38 ($\overline{\text{DIGITIZE}}$) lines can be left connected. The first and last MUX address switches must be set to same number.

Rear panel multiplexer input BNC's have numbers corresponding to input channel numbers (i.e., J1 is 1st input; J2 is 2nd input; J12 is 12th input; etc.).

7. Connect a ground strap from GRD Lug on rear of multiplexer to the 5475 chassis.

OPERATION

GENERAL PRECAUTIONS

When the MUX is in use, each of the channels that are to be used must be connected to a signal source of 1K, or less, output impedance. If this is not done, the input will float and the channel output may drift to ±12 volts.

Do not turn off the MUX while there are signals connected to its inputs.

CHANNEL ADDRESSING

The rear panel MUX channel (J1 - BNC) is the first input channel which is referred to as channel "Ø". The second channel (J2) is the second input channel, or channel "1".

To set MUX for operation between the first MUX input channel and the tenth MUX input channel:

FIRST SET MUX ADDRESS SWITCHES ALL DOWN (OR TO Ø);
NEXT SET LAST MUX ADDRESS SWITCHES TO 9 (SWITCH 8 AND 1 UP).

The 5466 ADC input control must be set to "A" for single channel operation. When multiplexer is disconnected, the ANALG IN command works as usual. When used with the multiplexer, the command is:

ANALOG IN N1 SPACE N2 SPACE N3

N1 = The first data block data is to be inputted.

N2 = Block to be displayed (default not allowed).

N3 = Number of channels to be inputted.

NOTE

Normal operation requires one more data block available than number of channels being acquired; thus, for 8 channels of data, 9 blocks need be available.

The command will not execute (gives "WHAT?" indication) if N1, N2, or N3 are not valid data blocks for the memory data space available.

CHOOSING SAMPLE PARAMETERS

When the multiplexer is operated with a 5466 ADC, the multiplexer samples all its input channels simultaneously, at a rate determined by the ADC's sample rate controls. The ADC, when it digitizes the data stored in the multiplexer, digitizes each channel independently, in sequence, at its maximum digitizing rate, independent of the sample rate control settings.

The multiplexer requires 12 μsec for its first input operation (regardless of the number of channels used). The ADC requires 7 μsec for each additional channel it digitizes. Therefore, the fastest sampling rate you can choose is one that allows enough time between sample commands for the multiplexer to sample, and the ADC to digitize, the selected number of channels.

For any specified number of multiplexer channels to be used ("N"), the fastest sampling rate (i.e., minimum time between samples) that can be set on the ADC is —

$12\ \mu\text{sec} + (N - 1)\ (7\ \mu\text{sec})$

Examples

1. What is the fastest sample rate that can be sent when eight channels are to be multiplexed?

   $\Delta t = 12\ \mu sec + (8 - 1) \times 7\ \mu sec = 61\ \mu sec$

   This (61 μsec) is the minimum time that is allowable when eight channels are multiplexed. The fastest ADC sampling rate that is slower than this is $\Delta t = 100\ \mu sec$, which corresponds to F_{max} of 5 kHz.

   Note: If external sampling signal is available then a signal of 16.393 kHz (1/61 μsec) as sampling rate would allow an F_{max} of 8.196 kHz.

2. What is the maximum number of channels that can be multiplexed, when the maximum frequency to be analyzed (F_{max}) is 2.5 kHz?

   Δt, the time between samples, must be $\leq \dfrac{1}{2\ F_{max}}$

   which, in this case, is $\dfrac{1}{2 \times 2500}$

   so, Δt must be $\leq 200\ \mu sec$.

   $N = \dfrac{\Delta t - 5\ \mu sec}{7\ \mu sec}$

   which, in this case, is $\dfrac{200\ \mu sec - 5\ sec}{7\ \mu sec} = 27$

   N, the number of channels that can be multiplexed can be 27 or less (assuming Δt or 200 μsec for the time between samples).

   Note: When the multiplexer is being used with a system that includes a Mass Storage device, the data storage space required in the computer's memory for the ADC throughput operation is —

   2 × block size × number of MUX channels being used.

   ADC Throughput Example:

   To do an ADC throughput operation for 8 multiplexer channels, using a block size of 256 words requires

   2 × 256 × 8, or 4096

   words in the computer's memory.

   The rate at which data can be transferred through the multiplexer and ADC to the ADC Throughput File in the mass storage device may be limited by the transfer rate of the mass storage device as follows:

   2 × number of multiplexer channels used × $F_{max} \leq$ max. throughput

PERFORMANCE CHECK

Subtract Test

1. Connect the same sinewave signal from an audio oscillator to channel 0 and channel 1 of the multiplexer. (Terminate all other channels.)

2. Set the multiplexer switches for the largest number of channels in the multiplexer.

3. Set the 5451 keyboard BLOCK SIZE to the largest number that is allowed by the particular memory configuration and number of channels (see OPERATION above).

4. Give the Fourier Analyzer the following command:

   ANALG IN   0   SPACE   0   SPACE   N3   ENTER

   where N3 is the maximum number of channels in the multiplexer.

5. The frequency of the audio oscillator should be adjusted so that several cycles of the sinewave appear in channel 0, and the level should be adjusted to give 5 to 8 volts peak-to-peak output.

6. Give the Fourier Analyzer the following command:

   -    | ENTER

7. The result (in block 0) should be less than 1% of the original input level.
   - Repeat the measurement for each channel.

Zero Offset Test

1. Terminate the active inputs of the multiplexer. Clear data blocks.

2. Give the Fourier Analyzer the following command:

   ANALG IN   0   SPACE   0   SPACE   N3   ENTER

   where N2 is the channel being tested and N3 is the maximum number of channels in the multiplexer.

3. The result in block N1, which is displayed, should be less than .1 volts peak-to-peak. The result in data blocks 0 through (N3-1) should be the same.

To use mass store (ADC Throughput File 2) the following command sequence is used:

MASS STORE   3   2   ENTER

MASS STORE   2   2   SPACE   N1   SPACE   N2   ENTER

N1 = Number multiplexer channels
N2 = Number of records

Note: ADC must be set for single channel input.

The following table provides some examples of maximum frequency of analysis as limited by the ADC, number of multiplexer channels, and/or throughput operation.

# Multiplexer Channels	Max Block Size	Fmax			
			Throughput With		
		ADC Alone	7900A	7970B	7970E
1	4096	25 kHz	10 kHz	5 kHz	10 kHz
2 7900 only	2048		5 kHz		
2	4096	25 kHz		2.5 kHz	5 kHz
4 7900A only	1024		2.5 kHz		
4	2048	10 kHz		500 Hz	2.5 kHz
8 7900A only	512		500 Hz		
8	1024	5 kHz		250 Hz	1 kHz

To set the multiplexer for operation from the first input channel to the sixth input channel:

a. Set FIRST MUX ADDRESS switches to zero (all down).

b. Set LAST MUX ADDRESS switches to five (set switch "1" and "4" up and all others down).

c. Set MODE switch to NORM.

Note: Other switch positions of MODE switch result in non-automatic operation. This "non-automatic" operation is without any system software control. The most useful other position for the MODE switch is SEQUENTIAL-MAN which allows the MANUAL button to step each input channel to the output. (Useful for input circuit check-out.)

The FIRST MUX ADDRESS switches may be set to a channel other than zero. However, the FIRST MUX ADDRESS must be equal to or less than LAST MUX ADDRESS.

AFRICA, ASIA, AUSTRALIA

ANGOLA
Telectra
Empresa Técnica de Equipamentos
Eléctricos, S.A.R.L.
R. Barbosa Rodrigues, 41-I°DT.°
Caixa Postal, 6487
Luanda
Tel: 35515/6
Cable: TELECTRA Luanda

AUSTRALIA
Hewlett-Packard Australia Pty. Ltd.
31-41 Joseph Street
Blackburn, Victoria 3130
P.O. Box 36
Doncaster East,
Victoria 3109
Tel: 896351
Telex: 31-024
Cable: HEWPARD Melbourne

Hewlett-Packard Australia Pty. Ltd.
31 Bridge Street
Pymble
New South Wales, 2073
Tel: 4496566
Telex: 21561
Cable: HEWPARD Sydney

Hewlett-Packard Australia Pty. Ltd.
153 Greenhill Road
Parkside, S.A., 5063
Tel: 2725911
Telex: 82536
Cable: HEWPARD Adelaide

Hewlett-Packard Australia Pty. Ltd.
141 Stirling Highway
Nedlands, W.A. 6009
Tel: 3865455
Telex: 93859
Cable: HEWPARD Perth

Hewlett-Packard Australia Pty. Ltd.
121 Wollongong Street
Fyshwick, A.C.T. 2609
Tel: 804244
Telex: 62650
Cable: HEWPARD Canberra

Hewlett Packard Australia Pty. Ltd.
5th Floor
Teachers Union Building
495-499 Boundary Street
Spring Hill, Queensland
4000
Tel: 2291544
Cable: HEWPARD Brisbane

BANGLADESH
The General Electric Co. of
Bangladesh Ltd.
Magnet House 72
Dilkusha Commercial Area
Motijhell, Dacca 2
Tel: 252415, 252419
Telex: 734
Cable: GECDAC Dacca

ETHIOPIA
Abdella Abdulmalik
P.O. Box 2635
Addis Ababa
Tel: 11 93 40

GUAM
Guam Medical Supply, Inc.
Suite C, Airport Plaza
P.O. Box 8947
Tamuning 96911
Tel: 646-4513
Cable: EARMED Guam

HONG KONG
Hewlett-Packard Hong Kong Ltd.
Room 105, Austin Centre
1st Floor
21 Austin Avenue
TST P.O. Box 98524
Kowloon
Tel: 3-697446 (5 lines)
Telex: 36678 HX
Cable: HEWPACK Hong Kong

Medical/Analytical Only
Schmidt & Co. (Hong Kong) Ltd.
Wing On Centre, 28th Floor
Connaught Road, C.
Hong Kong
Tel: 5-455644
Telex: 74766 SCHMX HX

INDIA
Blue Star Ltd.
Sahas
414/2 Vir Savarkar Marg
Prabhadevi
Bombay 400 025
Tel: 45 78 87
Telex: 011-4093
Cable: FROSTBLUE

Blue Star Ltd.
Band Box House
Prabhadevi
Bombay 400 025
Tel: 45 73 01
Telex: 011-3751
Cable: BLUESTAR

Blue Star Ltd.
Bhavdeep
Stadium Road
Ahmedabad 380 014
Tel: 43922
Telex: 012-234
Cable: BLUEFROST

Blue Star Ltd.
7 Hare Street
Calcutta 700 001
Tel: 23-0131
Telex: 021-7655
Cable: BLUESTAR

Blue Star Ltd.
Bhandari House
91 Nehru Place
New Delhi 110 024
Tel: 682547
Telex: 031-2463
Cable: BLUESTAR

Blue Star Ltd.
T.C. 7/603 'Poornima'
Maruthankuzhi
Trivandrum 695 013
Tel: 65799
Telex: 0884-259
Cable: BLUESTAR

Blue Star Ltd.
11 Magarath Road
Bangalore 560 025
Tel: 55668
Telex: 0845-430
Cable: BLUESTAR

Blue Star Ltd.
Meeakshi Mandiram
XXXXV/1379-2 Mahatma
Gandhi Rd.
Cochin 682 016
Tel: 32069
Telex: 085-514
Cable: BLUESTAR

Blue Star Ltd.
1-1-117/1 Sarojini Devi Road
Secunderabad 500 033
Tel: 70126
Telex: 0155-459
Cable: BLUESTAR

Blue Star Ltd.
133 Kodambakkam High Road
Madras 600 034
Tel: 82057
Telex: 041-379
Cable: BLUESTAR

INDONESIA
BERCA Indonesia P.T.
P.O. Box 496/Jkt.
Jln. Abdul Muis 62
Jakarta
Tel: 349255, 349886
Telex: 46748 BERSIL IA
Cable: BERSAL

BERCA Indonesia P.T.
P.O. Box 174/Sby.
23 Jln. Jimerto
Surabaya
Tel: 42027
Cable: BERCACON

JAPAN
Yokogawa-Hewlett-Packard Ltd.
29-21, Takaido-Higashi 3-chome
Suginami-ku, **Tokyo** 168
Tel: 03-331-6111
Telex: 232-2024 YHP-Tokyo
Cable: YHPMARKET TOK 23 724

Yokogawa-Hewlett-Packard Ltd.
Chuo Bldg., 4th Floor
4-20, Nishinakajima 5-chome
Yodogawa-ku, Osaka-shi
Osaka, 532
Tel: 06-304-6021
Telex: 523-3624

Yokogawa-Hewlett-Packard Ltd.
Sunitomo Seimei Nagaya Bldg.
11-2 Shimosasajima-cho,
Nakamura-ku, **Nagoya,** 450
Tel: 052 571-5171

Yokogawa-Hewlett-Packard Ltd.
Tanigawa Building
2-24-1 Tsuruya-cho
Kanagawa-ku
Yokohama, 221
Tel: 045-312-1252
Telex: 382-3204 YHP YOK

Yokogawa-Hewlett-Packard Ltd.
Mito Mitsui Building
105, 1-chome, San-no-maru
Mito, Ibaragi 310
Tel: 0292-25-7470

Yokogawa-Hewlett-Packard Ltd.
Inoue Building
1348-3, Asahi-cho, 1-chome
Atsugi, Kanagawa 243
Tel: 0462-24-0452

Yokogawa-Hewlett-Packard Ltd.
Kumagaya Asahi
Hachijuni Building
4th Floor
3-4, Tsukuba
Kumagaya, Saitama 360
Tel: 0485-24-6563

KENYA
ADCOM Ltd., Inc.
P.O. Box 30070
Nairobi
Tel: 331955
Telex: 22639

Medical Only
International Aeradio (E.A.) Ltd.
P.O. Box 19012
Nairobi Airport
Nairobi
Tel: 336055/56
Telex: 22201/22301
Cable: INTAERIO Nairobi

Medical Only
International Aeradio (E.A.) Ltd.
P.O. Box 95221
Mombasa

KOREA
Samsung Electronics Co., Ltd.
22nd Floor Dongbang Bldg.,
250, 2-KA, Taepyung-Ro
Chung-Ku,
Seoul
Tel: 777-4886
Telex: SAMSAN 27364

MALAYSIA
Hewlett-Packard Sales
(Malaysia) Sdn. Bhd.
Suite 2.21/2.22
Bangunan Angkasa Raya
Jalan Ampang
Kuala Lumpur
Tel: 483680, 485653

Protel Engineering
P.O. Box 1917
Lot 259, Satok Road
Kuching, **Sarawak**
Tel: 53544
Cable: PROTELENG

MOZAMBIQUE
A.N. Goncalves, Ltd.
162, 1° Apt. 14 Av. D. Luis
Caixa Postal 107
Maputo
Tel: 27091, 27114
Telex: 6-203 NEGON Mo
Cable: NEGON

NEW GUINEA
Hewlett-Packard Australia Pty. Ltd.
Development Bank Building
Ground Floor
Ward Strip
Port Moresby, Papua
Tel: 258933

NEW ZEALAND
Hewlett-Packard (N.Z.) Ltd.
4-12 Cruickshank Street
Kilbirnie, Wellington 3
P.O. Box 9443
Courtney Place
Wellington
Tel: 877-199
Cable: HEWPACK Wellington

Hewlett-Packard (N.Z.) Ltd.
P.O. Box 26-189
169 Manukau Road
Epsom, **Auckland**
Tel: 687-159
Cable: HEWPACK Auckland

Analytical/Medical Only
Northrop Instruments &
Systems Ltd.,
Sturdee House
85-87 Ghuznee Street
P.O. Box 2406
Wellington
Tel: 850-091
Telex: NZ 31291

Northrup Instruments &
Systems Ltd.
Eden House, 44 Khyber Pass Rd.
P.O. Box 9682, Newmarket
Auckland 1
Tel: 794-091

Northrup Instruments &
Systems Ltd.
Terrace House, 4 Oxford Terrace
P.O. Box 8388
Christchurch
Tel: 64-165

NIGERIA
The Electronics
Instrumentations Ltd.
N6B/770 Oyo Road
Oluseun House
P.M.B. 5402
Ibadan
Tel: 461577
Telex: 31231 TEIL NG
Cable: THETIEL Ibadan

The Electronics
Instrumentations Ltd.
144 Agege Motor Road, Mushin
P.O. Box 481
Mushin, **Lagos**
Cable: THETEIL Lagos

PAKISTAN
Mushko & Company Ltd.
Oosman Chambers
Abdullah Haroon Road
Karachi-3
Tel: 511027, 512927
Telex: 2894
Cable: COOPERATOR Karachi

Mushko & Company, Ltd.
10, Bazar Rd.
Sector G-6/4
Islamabad
Tel: 28264
Cable: FEMUS Rawalpindi

PHILIPPINES
The Online Advanced Systems
Corporation
Rico House
Amorsolo cor. Herrera Str.
Legaspi Village, Makati
P.O. Box 1510
Metro **Manila**
Tel: 85-35-81, 85-34-91, 85-32-21
Telex: 3274 ONLINE

RHODESIA
Field Technical Sales
45 Kelvin Road North
P.O. Box 3458
Salisbury
Tel: 705231 (5 lines)
Telex: RH 4122

SINGAPORE
Hewlett-Packard Singapore
(Pte.) Ltd.
1150 Depot Road
P.O. Box 58
Alexandra Post Office
Singapore 3
Tel: 270-2355
Telex: HPSG RS 21486
Cable: HEWPACK, Singapore

SOUTH AFRICA
Hewlett-Packard South Africa
(Pty.), Ltd.
Private Bag Wendywood,
Sandton, Transvaal, 2144
Hewlett-Packard Centre
Daphne Street, Wendywood,
Sandton, 2144
Tel: 802-5111/25
Telex: 8-4782
Cable: HEWPACK Johannesburg

Hewlett-Packard South Africa
(Pty.), Ltd.
P.O. Box 120
Howard Place,
Cape Province, 7450
Pine Park Centre, Forest Drive,
Pinelands,
Cape Province, 7405
Tel: 53-7955 thru 9
Telex: 57-0006

SRI LANKA
Metropolitan Agencies Ltd.
209/9 Union Place
Colombo 2
Tel: 35947
Telex: 1377METROLTD CE
Cable: METROLTD

SUDAN
Radison Trade
P.O. Box 921
Khartoum
Tel: 44048
Telex: 375

TAIWAN
Hewlett-Packard Far East Ltd.
Taiwan Branch
Bank Tower, 5th Floor
205 Tun Hau North Road
Taipei
Tel: (02) 751-0404 (15 lines)
Cable: HEWPACK TAIPEI

Hewlett-Packard Far East Ltd.
Taiwan Branch
68-2, Chung Cheng 3rd. Road
Kaohsiung
Tel: (07) 242318-Kaohsiung

Analytical Only
San Kwang Instruments Co., Ltd.
20 Yung Sui Road
Taipei
Tel: 3615446-9 (4 lines)
Telex: 22894 SANKWANG
Cable: SANKWANG Taipei

TANZANIA
Medical Only
International Aeradio (E.A.), Ltd.
P.O. Box 861
Dar es Salaam
Tel: 21251 Ext. 265
Telex: 41030

THAILAND
UNIMESA Co. Ltd.
Elcom Research Building
2538 Sukumvit Ave.
Bangchak, Bangkok
Tel: 3932387, 3930338
Cable: UNIMESA Bangkok

ZAMBIA
R.J. Tilbury (Zambia) Ltd.
P.O. Box 2792
Lusaka
Tel: 73793
Cable: ARJAYTEE, Lusaka

OTHER AREAS NOT LISTED, CONTACT:
Hewlett-Packard Intercontinental
3495 Deer Creek Road
Palo Alto, California 94304
Tel: (415) 856-1501
TWX: 910-373-1267
Cable: HEWPACK Palo Alto
Telex: 034-8300, 034-8493

SALES OFFICES (cont.)

CANADA

ALBERTA
Hewlett-Packard (Canada) Ltd.
11620A - 168th Street
Edmonton T5M 3T9
Tel: (403) 452-3670
TWX: 610-831-2431

Hewlett-Packard (Canada) Ltd.
210, 7220 Fisher St. S.E.
Calgary T2H 2H8
Tel: (403) 253-2713
TWX: 610-821-6141

BRITISH COLUMBIA
Hewlett-Packard (Canada) Ltd.
10691 Shellbridge Way
Richmond V6X 2W7
Tel: (604) 270-2277
TWX: 610-925-5059

MANITOBA
Hewlett-Packard (Canada) Ltd.
380-550 Century St.
St. James,
Winnipeg R3H 0Y1
Tel: (204) 786-6701
TWX: 610-671-3531

NOVA SCOTIA
Hewlett-Packard (Canada) Ltd.
P.O. Box 931
800 Windmill Road
Dartmouth B3B 1L1
Tel: (902) 469-7820
TWX: 610-271-4482

ONTARIO
Hewlett-Packard (Canada) Ltd.
1020 Morrison Dr.
Ottawa K2H 8K7
Tel: (613) 820-6483
TWX: 610-563-1636

Hewlett-Packard (Canada) Ltd.
6877 Goreway Drive
Mississauga L4V 1M8
Tel: (416) 678-9430
TWX: 610-492-4246

Hewlett-Packard (Canada) Ltd.
552 Newbold Street
London N6E 2S5
Tel: (519) 686-9181
TWX: 610-352-1201

QUEBEC
Hewlett-Packard (Canada) Ltd.
275 Hymus Blvd.
Pointe Claire H9R 1G7
Tel: (514) 697-4232
TWX: 610-422-3022

FOR CANADIAN AREAS NOT LISTED:
Contact Hewlett-Packard (Canada) Ltd. in Mississauga.

CENTRAL, SOUTH AMERICA

ARGENTINA
Hewlett-Packard Argentina S.A.
Santa Fe 2035, Martinez
6140 **Buenos Aires**
Tel: 792-1239, 798-6086
Telex: 122443 AR CIGY
Cable: HEWPACKARG

Biotron S.A.C.I.y M.
Avda. Paseo Colon 221
9 piso
1399 **Buenos Aires**
Tel: 30-4846/1851/8384
34-9356/0460/4551
Telex: (33) 17595 BIO AR
Cable: BIOTRON Argentina

BRAZIL
Hewlett-Packard do Brasil
I.e.C. Ltda.
Alameda Rio Negro, 750
Alphaville
06400 **Barueri** SP
Tel: 429-3222
Cable: HEWPACK São Paulo

Hewlett-Packard do Brasil
I.e.C. Ltda.
Rua Padre Chagas, 32
90000-**Pôrto Alegre**-RS
Tel: 22-2998, 22-5621
Cable: HEWPACK Pôrto Alegre

Hewlett-Packard do Brasil
I.e.C. Ltda.
Av. Epitacio Pessoa, 4664
22471-**Rio de Janeiro**-RJ
Tel: 286-0237
Telex: 021-21905 HPBR-BR
Cable: HEWPACK Rio de Janeiro

CHILE
Jorge Calcagni y Cia. Ltda.
Arturo Burhle 065
Casilla 16475
Correo 9, **Santiago**
Tel: 220222
Telex: JCALCAGNI

COLOMBIA
Instrumentación
Henrik A. Langebaek & Kier S.A.
Carrera 7 No. 48-75
Apartado Aéreo 6287
Bogotá, 1 D.E.
Tel: 269-8877
Telex: 44400
Cable: AARIS Bogotá

Instrumentación
H.A. Langebaek & Kier S.A.
Carrera 63 No. 49-A-31
Apartado 54098
Medellin
Tel: 304475

COSTA RICA
Cientifica Costarricense S.A.
Avenida 2, Calle 5
San Pedro de Montes de Oca
Apartado 10159
San José
Tel: 24-38-20, 24-08-19
Telex: 2367 GALGUR CR
Cable: GALGUR

ECUADOR
CYEDE Cia. Ltda.
P.O. Box 6423 CCI
Av. Eloy Alfaro 1749
Quito
Tel: 450-975, 243-052
Telex: 2548 CYEDE ED
Cable: CYEDE-Quito

Medical Only
Hospitalar S.A.
Casilla 3590
Robles 625
Quito
Tel: 545-250
Cable: HOSPITALAR-Quito

EL SALVADOR
IPESA
Bulevar de los Heroes 11-48
Edificio Sarah 1148
San Salvador
Tel: 252787

GUATEMALA
IPESA
Avenida Reforma 3-48
Zona 9
Guatemala City
Tel: 316627, 314786, 66471-5, ext. 9
Telex: 4192 Teletro Gu

MEXICO
Hewlett-Packard Mexicana, S.A. de C.V.
Av. Periférico Sur No. 6501
Tepepan, Xochimilco
Mexico 23, D.F.
Tel: 905-676-4600
Telex: 017-74-507

Hewlett-Packard Mexicana, S.A. de C.V.
Ave. Constitución No. 2184
Monterrey, N.L.
Tel: 48-71-32, 48-71-84
Telex: 038-410

PANAMA
Electrónico Balboa, S.A.
Aparatado 4929
Panama 5
Calle Samuel Lewis
Edificio "Alfa," No. 2
Ciudad de Panama
Tel: 64-2700
Telex: 3483103 Curundu, Canal Zone
Cable: ELECTRON Panama

PERU
Compañía Electro Médica S.A.
Los Flamencos 145
San Isidro Casilla 1030
Lima 1
Tel: 41-4325
Telex: Pub. Booth 25424 SISIDRO
Cable: ELMED Lima

SURINAM
Surtel Radio Holland N.V.
Grote Hofstr. 3-5
P.O. Box 155
Paramaribo
Tel: 72118, 77880
Cable: Surtel

TRINIDAD & TOBAGO
CARTEL
Caribbean Telecoms Ltd.
P.O. Box 732
69 Frederick Street
Port-of-Spain
Tel: 62-53068

URUGUAY
Pablo Ferrando S.A.C.el.
Avenida Italia 2877
Casilla de Correo 370
Montevideo
Tel: 40-3102
Telex: 702 Public Booth
Para Pablo Ferrando
Cable: RADIUM Montevideo

VENEZUELA
Hewlett-Packard de Venezuela C.A.
P.O. Box 50933
Caracas 105
Los Ruices Norte
3a Transversal
Edificio Segre
Caracas 107
Tel: 239-4133 (20 lines)
Telex: 25146 HEWPACK
Cable: HEWPACK Caracas

FOR AREAS NOT LISTED, CONTACT:
Hewlett-Packard Intercontinental
3495 Deer Creek Road
Palo Alto, California 94304
Tel: (415) 856-1501
TWX: 910-373-1260
Cable: HEWPACK Palo Alto
Telex: 034-8300, 034-8493

EUROPE, NORTH AFRICA, MIDDLE EAST

AUSTRIA
Hewlett-Packard Ges.m.b.H.
Wehlistrasse 29
P.O. Box 7
A-1205 **Vienna**
Tel: 35-16-21-0
Cable: HEWPACK Vienna
Telex: 13582/135066

Hewlett-Packard Ges.m.b.H.
Wehlistrasse, 29
A-1205 **Wien**
Tel: 35-16-21
Telex: 135066

BAHRAIN
Medical Only
Wael Pharmacy
P.O. Box 648
Bahrain
Tel: 54886, 56123
Telex: 8550 WAEL GJ
Cable: WAELPHARM

Al Hamidiya Trading and Contracting
P.O. Box 20074
Manama
Tel: 259978, 259958
Telex: 8895 KALDIA GJ

BELGIUM
Hewlett-Packard Benelux S.A./N.V.
Avenue du Col-Vert, 1, (Groenkraaglaan)
B-1170 **Brussels**
Tel: (02) 660 50 50
Cable: PALOBEN Brussels
Telex: 23-494 paloben bru

CYPRUS
Kypronics
19 Gregorios Xenopoulos Street
P.O. Box 1152
Nicosia
Tel: 45628/29
Cable: Kypronics Pandehis
Telex: 3018

CZECHOSLOVAKIA
Hewlett-Packard
Obchodni zastupitelstvi v CSSR
Pisemny styk
Post. schranka 27
CS 118 01 **Praha** 011
CSSR

Vyvojova a Provozni Zakladna
Vyzkumnych Ustavu v Bechovicich
CSSR-25097 **Bechovice u Prahy**
Tel: 89 93 41
Telex: 12133

Institute of Medical Bionics
Vyskumny Ustav Lekarskej Bioniky
Jedlova 6
CS-88346 **Bratislava-Kramare**
Tel: 44-551
Telex: 93229

DENMARK
Hewlett-Packard A/S
Datavej 52
DK-3460 **Birkerod**
Tel: (02) 81 66 40
Cable: HEWPACK AS
Telex: 37409 hpas dk

Hewlett-Packard A/S
Navervej 1
DK-8600 **Silkeborg**
Tel: (06) 82 71 66
Telex: 37409 hpas dk
Cable: HEWPACK AS

EGYPT
I.E.A.
International Engineering Associates
24 Hussein Hegazi Street
Kasr-el-Aini
Cairo
Tel: 23 829
Telex: 93830
Cable: INTENGASSO

SAMITRO
Sami Amin Trading Office
18 Abdel Aziz Gawish
Abdine-Cairo
Tel: 24932
Cable: SAMITRO CAIRO

FINLAND
Hewlett-Packard Oy
Revontulentie, 7
SF-02100 **Espoo** 10
Tel: (90) 455 0211
Cable: HEWPACKOY
Telex: 121563 hewpa sf

FRANCE
Hewlett-Packard France
Zone d'activites de Courtaboeuf
Avenue des Tropiques
Boite Postale 6
91401 **Orsay**-Cédex
Tel: (1) 907 78 25
TWX: 600048F

Hewlett-Packard France
Chemin des Mouilles
B.P. 162
69130 **Ecully**
Tel: (78) 33 81 25
TWX: 310617F

Hewlett-Packard France
20, Chemin de La Cépière
31081 **Toulouse Le Mirail**-Cédex
Tel: (61) 40 11 12

Hewlett-Packard France
Le Ligoures
Place Romée de Villeneuve
13100 **Aix-en-Provence**
Tel: (42) 59 41 02
TWX: 410770F

Hewlett-Packard France
2, Allee de la Bourgonette
35100 **Rennes**
Tel: (99) 51 42 44
TWX: 740912F

Hewlett-Packard France
18, rue du Canal de la Marne
67300 **Schiltigheim**
Tel: (88) 83 08 10
TWX: 890141F

Hewlett-Packard France
Immeuble péricentre
rue van Gogh
59650 **Villeneuve** D'Ascq
Tel: (20) 91 41 25
TWX: 160124F

Hewlett-Packard France
Bâtiment Ampère
Rue de la Commune de Paris
B.P. 300
93153 **Le Blanc Mesnil**-Cédex
Tel: (01) 931 88 50
Telex: 211032F

Hewlett-Packard France
Av. du Pdt. Kennedy
33700 **Merignac**
Tel: (56) 97 01 81

Hewlett-Packard France
Immeuble Lorraine
Boulevard de France
91035 **Evry**-Cédex
Tel: 077 96 60
Telex: 692315F

Hewlett-Packard France
23 Rue Lothaire
57000 **Metz**
Tel: (87) 65 53 50

GERMAN FEDERAL REPUBLIC
Hewlett-Packard GmbH
Vertriebszentrale Frankfurt
Berner Strasse 117
Postfach 560 140
D-6000 **Frankfurt** 56
Tel: (06011) 50041
Cable: HEWPACKSA Frankfurt
Telex: 04 13249 hpffm d

Hewlett-Packard GmbH
Technisches Büro Böblingen
Herrenberger Strasse 110
D-7030 **Böblingen,** Württemberg
Tel: (07031) 667-1
Cable: HEWPACK Böblingen
Telex: 07265739 bbn

Hewlett-Packard GmbH
Technisches Büro Düsseldorf
Emanuel-Leutze-Str. 1 (Seestern)
D-4000 **Düsseldorf**
Tel: (0211) 5971-1
Telex: 085/86 533 hpdd d

Hewlett-Packard GmbH
Technisches Büro Hamburg
Kapstadtring 5
D-2000 **Hamburg** 60
Tel: (040) 63804-1
Cable: HEWPACKSA Hamburg
Telex: 21 63 032 hphh d

Hewlett-Packard GmbH
Technisches Büro Hannover
Am Grossmarkt 6
D-3000 **Hannover** 91
Tel: (0511) 46 60 01
Telex: 092 3259

Hewlett-Packard GmbH
Technisches Büro Nürnberg
Neumeyerstrasse 90
D-8500 **Nürnberg**
Tel: (0911) 52 20 83
Telex: 0623 860

EUROPE, NORTH AFRICA, MIDDLE EAST

Hewlett-Packard GmbH
Technisches Büro München
Eschenstrasse 5
D-8021 **Taufkirchen**
Tel: (089) 6117-1
Telex: 0524985

Hewlett-Packard GmbH
Technisches Büro Berlin
Kaithstrasse 2-4
D-1000 **Berlin** 30
Tel: (030) 24 90 86
Telex: 018 3405 hpbln d

GREECE
Kostas Karayannis
8 Omirou Street
Athens 133
Tel: 32 30 303/32/37 731
Telex: 21 59 62 RKAR GR
Cable: RAKAR ATHENS

ICELAND
Medical Only
Elding Trading Company Inc.
Hafnarnvoli - Tryggvagötu
P.O. Box 895
IS-**Reykjavik**
Tel: 1 58 20/1 63 03
Cable: ELDING Reykjavik

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnersh, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London

Hewlett-Packard Ltd.
Kestrel House
Clanwilliam Place
Lower Mount Street
Dublin 2, Eire

Hewlett-Packard Ltd.
2C Avongberg Ind. Est.
Long Mile Road
Dublin 12
Tel: 514322/514224
Telex: 30439

Medical Only
Cardiac Services (Ireland) Ltd.
Kilmore Road
Artane
Dublin 5, Eire
Tel: (01) 315820

Medical Only
Cardiac Services Co.
95A Finaghy Rd. South
Belfast BT10 0BY
GB-Northern Ireland
Tel: (0232) 625566
Telex: 747626

ISRAEL
Electronics Engineering Div.
of Motorola Israel Ltd.
16, Kremenetski Street
P.O. Box 25016
Tel-Aviv
Tel: 38973
Telex: 33569, 34164
Cable: BASTEL Tel-Aviv

ITALY
Hewlett-Packard Italiana S.p.A.
Via G. Di Vittorio, 9
20063 **Cernusco Sul Naviglio** (MI)
Tel: (2) 903691
Telex: 334632 HEWPACKIT

Hewlett-Packard Italiana S.p.A.
Via Turazza, 14
35100 **Padova**
Tel: (49) 664888
Telex: 430315 HEWPACKI

Hewlett-Packard Italiana S.p.A.
Via G. Armellini 10
1-00143 **Roma**
Tel: (06) 54 69 61
Telex: 610514
Cable: HEWPACKIT Roma

Hewlett-Packard Italiana S.p.A.
Corso Giovanni Lanza 94
I-10133 **Torino**
Tel: (011) 659308
Telex: 221079

Hewlett-Packard Italiana S.p.A.
Via Principe Nicola 43 G/C
I-95126 **Catania**
Tel: (095) 37 05 04
Telex: 970291

Hewlett-Packard Italiana S.p.A.
Via Nuova san Rocco A
Capadimonte, 62A
80131 **Napoli**
Tel: (081) 710698

Hewlett-Packard Italiana S.p.A.
Via Martin Luther King, 38/111
I-40132 **Bologna**
Tel: (051) 402394
Telex: 511630

JORDAN
Mouasher Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1456
Cable: MOUASHERCO

KUWAIT
Al-Khaldiya Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 4910/41 1726
Telex: 2481 Areeg kt
Cable: VISCOUNT

LUXEMBURG
Hewlett-Packard Beneluz S.A./N.V.
Avenue du Col-Vert, 1
(Groenkraaglaan)
B-1170 **Brussels**
Tel: (02) 660 5050
Cable: PALOBEN Brussels
Telex: 23 494

MOROCCO
Dolbeau
81 rue Karatchi
Casablanca
Tel: 3041 82
Telex: 23051/22822
Cable: MATERIO

Gerep
2, rue d'Agadir
Boite Postal 156
Casablanca
Tel: 272093/5
Telex: 23 739
Cable: GEREP-CASA

NETHERLANDS
Hewlett-Packard Benelux N.V.
Van Heuven Goedhartlaan 121
P.O. Box 667
1181KK **Amstelveen**
Tel: (20) 47 20 21
Cable: PALOBEN Amsterdam
Telex: 13 216

NORWAY
Hewlett-Packard Norge A/S
Ostendalen 18
P.O. Box 34
1345 **Osteraas**
Tel: (02) 1711 80
Telex: 16621 hpnas n

Hewlett-Packard Norge A/S
Nygaardsgaten 114
P.O. Box 4210
5013 Nygaardsgaten,
Bergen
Tel: (05) 21 97 33

POLAND
Biuro Informacji Technicznej
Hewlett-Packard
Ul Stawki 2, 6P
PL00-950 **Warszawa**
Tel: 39 59 62, 39 51 87
Telex: 81 24 53

PORTUGAL
Telectra-Empresa Técnica de
Equipamentos Eléctricos S.a.r.l.
Rua Rodrigo da Fonseca 103
P.O. Box 2531
P-**Lisbon** 1
Tel: (19) 68 60 72
Cable: TELECTRA Lisbon
Telex: 12598

Medical Only
Mundinter
Intercambio Mundial de Comércio
S.a.r.l.
P.O. Box 2761
Avenida Antonio Augusto
de Aguiar 138
P-**Lisbon**
Tel: (19) 53 21 31/7
Telex: 16691 munter p
Cable: INTERCAMBIO Lisbon

QATAR
Nasser Trading & Contracting
P.O. Box 1563
Doha
Tel: 22170
Telex: 4439 NASSER
Cable: NASSER

ROMANIA
Hewlett-Packard Reprezentanta
Bd.n. Balcescu 16
Bucuresti
Tel: 15 80 23/13 88 85
Telex: 10440

SAUDI ARABIA
Modern Electronic
Establishment (Head Office)
P.O. Box 1228, Baghdadiah Street
Jeddah
Tel: 27 798
Telex: 40035
Cable: ELECTA JEDDAH

Modern Electronic Establishment
(Branch)
P.O. Box 2728
Riyadh
Tel: 62596/66232
Telex: 202049

Modern Electronic Establishment
(Branch)
P.O. Box 193
Al-Khobar
Tel: 44678-44813
Telex: 670136
Cable: ELECTA AL-KHOBAR

SPAIN
Hewlett-Packard Española, S.A.
Calle Jerez 3
E-**Madrid** 16
Tel: (1) 458 26 00 (10 lines)
Telex: 23515 hpe

Hewlett-Packard Española S.A.
Colonia Mirasierra
Edificio Juban
c/o Costa Brava, 13
Madrid 34

Hewlett-Packard Española, S.A.
Milanesado 21-23
E-**Barcelona** 17
Tel: (3) 203 6200 (5 lines)
Telex: 52603 hpbe e

Hewlett-Packard Española, S.A.
Av Ramón y Cajal, 1
Edificio Sevilla, planta 9°
E-**Sevilla** 5
Tel: 64 44 54/58

Hewlett-Packard Española S.A.
Edificio Albia II 7° B
E-**Bilbao** 1
Tel: 23 83 06/23 82 06

Hewlett-Packard Española S.A.
C/Ramon Gordillo 1
(Entlo.)
E-**Valencia** 10
Tel: 96-361.13.54/361.13.58

SWEDEN
Hewlett-Packard Sverige AB
Enighetsvägen 3, Fack
S-161 **Bromma** 20
Tel: (08) 730 05 50
Telex: 10721
Cable: MEASUREMENTS
Stockholm

Hewlett-Packard Sverige AB
Frötallsgatan 30
S-421 32 **Västra Frölunda**
Tel: (031) 49 09 50
Telex: 10721 via Bromma office

SWITZERLAND
Hewlett-Packard (Schweiz) AG
Zürcherstrasse 20
P.O. Box 307
CH-8952 **Schlieren-Zürich**
Tel: (01) 7305240
Telex: 53933 hpag ch
Cable: HPAG CH

Hewlett-Packard (Schweiz) AG
Château Bloc 19
CH-1219 **Le Lignon-Geneva**
Tel: (022) 96 03 22
Telex: 27333 hpag ch
Cable: HEWPACKAG Geneva

SYRIA
General Electronic Inc.
Nuri Basha-Ahnaf Ebn Kays Street
P.O. Box 5781
Damascus
Tel: 33 24 87
Telex: 11215 ITIKAL
Cable: ELECTROBOR DAMASCUS

Medical only
Sawah & Co.
Place Azmé
B.P. 2308
Damascus
Tel: 16 367-19 697-14 268
Telex: 11304 SATACO SY
Cable: SAWAH, DAMASCUS

Suleiman Hilal El Mlawi
P.O. Box 2528
Mamoun Bitar Street, 56-58
Damascus
Tel: 11 46 63
Telex: 11270
Cable: HILAL DAMASCUS

TUNISIA
Tunisie Electronique
31 Avenue de la Liberte
Tunis
Tel: 280 144

Corema
1 ter. Av. de Carthage
Tunis
Tel: 253 821
Telex: 12319 CABAM TN

TURKEY
TEKNIM Company Ltd.
Riza Sah Pehlevi
Caddesi No. 7
Kavaklidere, **Ankara**
Tel: 275800
Telex: 42155

Teknim Com., Ltd.
Barbaros Bulvari 55/12
Besikyas, **Istanbul**
Tel: 613 546
Telex: 23540

E.M.A.
Muhendislik Kollektif Sirketi
Mediha Eldem Sokak 41/6
Yüksel Caddesi
Ankara
Tel: 17 56 22
Cable: EMATRADE/Ankara

Yilmaz Ozyurek
Milli Mudafaa Cad 16/6
Kizilay
Ankara
Tel: 25 03 09 - 17 80 26
Telex: 42576 OZEK TR
Cable: OZYUREK ANKARA

UNITED ARAB EMIRATES
Emitac Ltd. (Head Office)
P.O. Box 1641
Sharjah
Tel: 354121/3
Telex: 8136

Emitac Ltd. (Branch Office)
P.O. Box 2711
Abu Dhabi
Tel: 331370/1

UNITED KINGDOM
Hewlett-Packard Ltd.
King Street Lane
Winnersh, Wokingham
Berkshire RG11 5AR
GB-England
Tel: (0734) 784774
Telex: 84 71 78/9

Hewlett-Packard Ltd.
Fourier House,
257-263 High Street
London Colney
St. Albans, Herts
GB-England
Tel: (0727) 24400
Telex: 1-8952716

Hewlett-Packard Ltd.
Trafalgar House
Navigation Road
Altrincham
Cheshire WA14 1NU
GB-England
Tel: (061) 928 6422
Telex: 668068

Hewlett-Packard Ltd.
Lygon Court
Hereward Rise
Dudley Road
Halesowen,
West Midlands, B62 8SD
GB-England
Tel: (021) 501 1221
Telex: 339105

Hewlett-Packard Ltd.
Wedge House
799, London Road
Thornton Heath
Surrey, CR4 6XL
GB-England
Tel: (01) 684-0103/8
Telex: 946825

Hewlett-Packard Ltd.
14 Wesley St
Castleford
Yorks WF10 1AE
Tel: (0977) 550016
TWX: 5557335

Hewlett-Packard Ltd.
Tradax House
St. Mary's Walk
Maidenhead
Berkshire, SL6 1ST
GB-England

Hewlett-Packard Ltd.
Morley Road
Staplehill
Bristol, BS16 4QT
GB-England

Hewlett-Packard Ltd.
South Queensferry
West Lothian, EH30 9TG
GB-Scotland
Tel: (031) 331 1188
Telex: 72682

Hewlett-Packard Ltd.
Kestrel House
Clanwilliam Place
Lower Mount Street
Dublin 2, Eire

Hewlett-Packard Ltd.
2C Avonberg Ind. Est.
Long Mile Road
Dublin 12
Tel: 514322/514224
Telex: 30439

USSR
Hewlett-Packard
Representative Office
USSR
Pokrovsky Boulevard 4/17-kw 12
Moscow 101000
Tel: 294.20.24
Telex: 7825 hewpak su

YUGOSLAVIA
Iskra Commerce, n.sol.o.
Zastopstvo Hewlett-Packard
Obilicev Venac 26
YU 11000 **Beograd**
Tel: 636-955
Telex: 11530

Iskra Commerce, n.sol.o.
Zastopstvo Hewlett-Packard
Miklosiceva 38/VII
YU-61000 **Ljubljana**
Tel: 321-674, 315-879
Telex: 31583

SALES OFFICES (cont.)

EUROPE, NORTH AFRICA, MIDDLE EAST

SOCIALIST COUNTRIES NOT SHOWN, PLEASE CONTACT:
Hewlett-Packard Ges.m.b.H.
Handelskai 52
P.O. Box 7
A-1205 **Vienna,** Austria
Tel: (0222) 35 16 21 to 27
Cable: HEWPAK Vienna
Telex: 75923 hewpak a

MEDITERRANEAN AND MIDDLE EAST COUNTRIES NOT SHOWN, PLEASE CONTACT:
Hewlett-Packard S.A.
Mediterranean and Middle East Operations
35, Kolokotroni Street
Platia Kefallariou
GR-Kifissia-**Athens,** Greece
Tel: 8080359/429
Telex: 21-6588
Cable: HEWPACKSA Athens

FOR OTHER AREAS NOT LISTED, CONTACT:
Hewlett-Packard S.A.
7, rue du Bois-du-Lan
P.O. Box
CH-1217 Meyrin 2 - **Geneva**
Switzerland
Tel: (022) 82 70 00
Cable: HEWPACKSA Geneva
Telex: 2 24 86

UNITED STATES

ALABAMA
P.O. Box 4207
8290 Whitesburg Dr.
Huntsville 35802
Tel: (205) 881-4592

8933 E. Roebuck Blvd.
Birmingham 35206
Tel: (205) 836-2203/2

ARIZONA
2336 E. Magnolia St.
Phoenix 85034
Tel: (602) 244-1361

2424 East Aragon Rd.
Tucson 85706
Tel: (602) 889-4661

***ARKANSAS**
Medical Service Only
P.O. Box 5646
Brady Station
Little Rock 72215
Tel: (501) 376-1844

CALIFORNIA
1579 W. Shaw Ave.
Fresno 93771
Tel: (209) 224-0582

1430 East Orangethorpe Ave.
Fullerton 92631
Tel: (714) 870-1000

3939 Lankershim Boulevard
North Hollywood 91604
Tel: (213) 877-1282
TWX: 910-499-2671

5400 West Rosecrans Blvd.
P.O. Box 92105
World Way Postal Center
Los Angeles 90009
Tel: (213) 776-7500
TWX: 910-325-6608

***Los Angeles**
Tel: (213) 776-7500

3200 Hillview Av
Palo Alto, CA 94304
Tel: (408) 988-7000

3003 Scott Boulevard
Santa Clara 95050
Tel: (408) 988-7000
TWX: 910-338-0518

***Ridgecrest**
Tel: (714) 446-6165

646 W. North Market Blvd.
Sacramento 95834
Tel: (916) 929-7222

9606 Aero Drive
P.O. Box 23333
San Diego 92123
Tel: (714) 279-3200

***Tarzana**
Tel: (213) 705-3344

COLORADO
5600 DTC Parkway
Englewood 80110
Tel: (303) 771-3455

CONNECTICUT
47 Barnes Industrial Road
Barnes Park South
Wallingford 06492
Tel: (203) 265-7801

FLORIDA
P.O. Box 24210
2727 N.W. 62nd Street
Ft. Lauderdale 33309
Tel: (305) 973-2600

4080 Woodcock Drive #132
Brownett Building
Jacksonville 32207
Tel: (904) 398-0663

P.O. Box 13910
6177 Lake Ellenor Dr.
Orlando 32809
Tel: (305) 859-2900

P.O. Box 12826
Suite 5, Bldg. 1
Office Park North
Pensacola 32575
Tel: (904) 476-8422

Computer Systems Only
110 South Hoover Blvd.
Suite 120
Tampa 33609
Tel: (813) 872-0900

GEORGIA
P.O. Box 105005
450 Interstate North Parkway
Atlanta 30348
Tel: (404) 955-1500
TWX: 810-766-4890

Medical Service Only
***Augusta** 30903
Tel: (404) 736-0592

P.O. Box 2103
1172 N. Davis Drive
Warner Robins 31098
Tel: (912) 922-0449

HAWAII
2875 So. King Street
Honolulu 96826
Tel: (808) 955-4455

ILLINOIS
5201 Tollview Dr.
Rolling Meadows 60008
Tel: (312) 255-9800
TWX: 910-687-2260

INDIANA
7301 North Shadeland Ave.
Indianapolis 46250
Tel: (317) 842-1000
TWX: 810-260-1797

IOWA
2415 Heinz Road
Iowa City 52240
Tel: (319) 351-1020

KENTUCKY
10170 Linn Station Road
Suite 525
Louisville 40223
Tel: (502) 426-0100

LOUISIANA
P.O. Box 1449
3229-39 Williams Boulevard
Kenner 70063
Tel: (504) 443-6201

MARYLAND
7121 Standard Drive
Parkway Industrial Center
Hanover 21076
Tel: (301) 796-7700
TWX: 710-862-1943

2 Choke Cherry Road
Rockville 20850
Tel: (301) 948-6370
TWX: 710-828-9684

MASSACHUSETTS
32 Hartwell Ave.
Lexington 02173
Tel: (617) 861-8960
TWX: 710-326-6904

MICHIGAN
23855 Research Drive
Farmington Hills 48024
Tel: (313) 476-6400

724 West Centre Ave.
Kalamazoo 49002
Tel: (616) 323-8362

MINNESOTA
2400 N. Prior Ave.
St. Paul 55113
Tel: (612) 636-0700

MISSISSIPPI
322 N. Mart Plaza
Jackson 39206
Tel: (601) 982-9363

MISSOURI
11131 Colorado Ave.
Kansas City 64137
Tel: (816) 763-8000
TWX: 910-771-2087

1024 Executive Parkway
St. Louis 63141
Tel: (314) 878-0200

NEBRASKA
Medical Only
7101 Mercy Road
Suite 101
Omaha 68106
Tel: (402) 392-0948

NEVADA
***Las Vegas**
Tel: (702) 736-6610

NEW JERSEY
W. 120 Century Rd.
Paramus 07652
Tel: (201) 265-5000
TWX: 710-990-4951

Crystal Brook Professional Building
Route 35
Eatontown 07724
Tel: (201) 542-1384

NEW MEXICO
P.O. Box 11634
Station E
11300 Lomas Blvd., N.E.
Albuquerque 87123
Tel: (505) 292-1330
TWX: 910-989-1185

156 Wyatt Drive
Las Cruces 88001
Tel: (505) 526-2484
TWX: 910-9983-0550

NEW YORK
6 Automation Lane
Computer Park
Albany 12205
Tel: (518) 458-1550
TWX: 710-444-4961

650 Perinton Hill Office Park
Fairport 14450
Tel: (716) 223-9950
TWX: 510-253-0092

No. 1 Pennsylvania Plaza
55th Floor
34th Street & 8th Avenue
New York 10001
Tel: (212) 971-0800

5858 East Molloy Road
Syracuse 13211
Tel: (315) 455-2486

1 Crossways Park West
Woodbury 11797
Tel: (516) 921-0300
TWX: 510-221-2183
Tel: (513) 671-7400

NORTH CAROLINA
5605 Roanne Way
Greensboro 27405
Tel: (919) 852-1800

OHIO
Medical/Computer Only
Bldg. 300
1313 E. Kemper Rd.
Cincinnati 45426
16500 Sprague Road
Cleveland 44130
Tel: (216) 243-7300
TWX: 810-423-9430

330 Progress Rd.
Dayton 45449
Tel: (513) 859-8202

1041 Kingsmill Parkway
Columbus 43229
Tel: (614) 436-1041

OKLAHOMA
P.O. Box 32008
6301 N. Meridan Avenue
Oklahoma City 73112
Tel: (405) 721-0200

9920 E. 42nd Street
Suite 121
Tulsa 74145
Tel: (918) 665-3300

OREGON
17890 S.W. Lower Boones Ferry Road
Tualatin 97062
Tel: (503) 620-3350

PENNSYLVANIA
111 Zeta Drive
Pittsburgh 15238
Tel: (412) 782-0400

1021 8th Avenue
King of Prussia Industrial Park
King of Prussia 19406
Tel: (215) 265-7000
TWX: 510-660-2670

PUERTO RICO
Hewlett-Packard Inter-Americas
Puerto Rico Branch Office
Calle 272,
Edif. 203 Urg. Country Club
Carolina 00924
Tel: (809) 762-7255
Telex: 345 0514

SOUTH CAROLINA
P.O. Box 6442
6941-0 N. Trenholm Road
Columbia 29260
Tel: (803) 782-6493

TENNESSEE
8914 Kingston Pike
Knoxville 37922
Tel: (615) 523-0522

3027 Vanguard Dr.
Director's Plaza
Memphis 38131
Tel: (901) 346-8370

***Nashville**
Medical Service Only
Tel: (615) 244-5448

TEXAS
4171 North Mesa
Suite C110
El Paso 79902
Tel: (915) 533-3555

P.O. Box 1270
201 E. Arapaho Rd.
Richardson 75080
Tel: (214) 231-6101

P.O. Box 42816
10535 Harwin Dr.
Houston 77036
Tel: (713) 776-6400

***Lubbock**
Medical Service Only
Tel: (806) 799-4472

205 Billy Mitchell Road
San Antonio 78226
Tel: (512) 434-8241

UTAH
2160 South 3270 West Street
Salt Lake City 84119
Tel: (801) 972-4711

VIRGINIA
P.O. Box 9669
2914 Hungry Springs Road
Richmond 23228
Tel: (804) 285-3431

Computer Systems/Medical Only
Airport Executive Center
Suite 111
5700 Thurston Avenue
Virginia Beach 23455
Tel: (804) 460-2471

WASHINGTON
Bellefield Office Pk.
1203 - 114th Ave. S.E.
Bellevue 98004
Tel: (206) 454-3971
TWX: 910-443-2446

P.O. Box 4010
Spokane 99202
Tel: (509) 535-0864

***WEST VIRGINIA**
Medical/Analytical Only
4604 Mac Corkle Ave., S.E.
Charleston 25304
Tel: (304) 925-0492

WISCONSIN
150 South Sunny Slope Road
Brookfield 53005
Tel: (414) 784-8800

FOR U.S. AREAS NOT LISTED:
Contact the regional office nearest you:
Atlanta, Georgia. . .North Hollywood, California. . .Rockville, Maryland. . .Rolling Meadows, Illinois. Their complete addresses are listed above.

***Service Only**

HEWLETT
PACKARD

Manual Part No. 05451-90529
Microfiche No. 05451-90530

Printed in U.S.A.