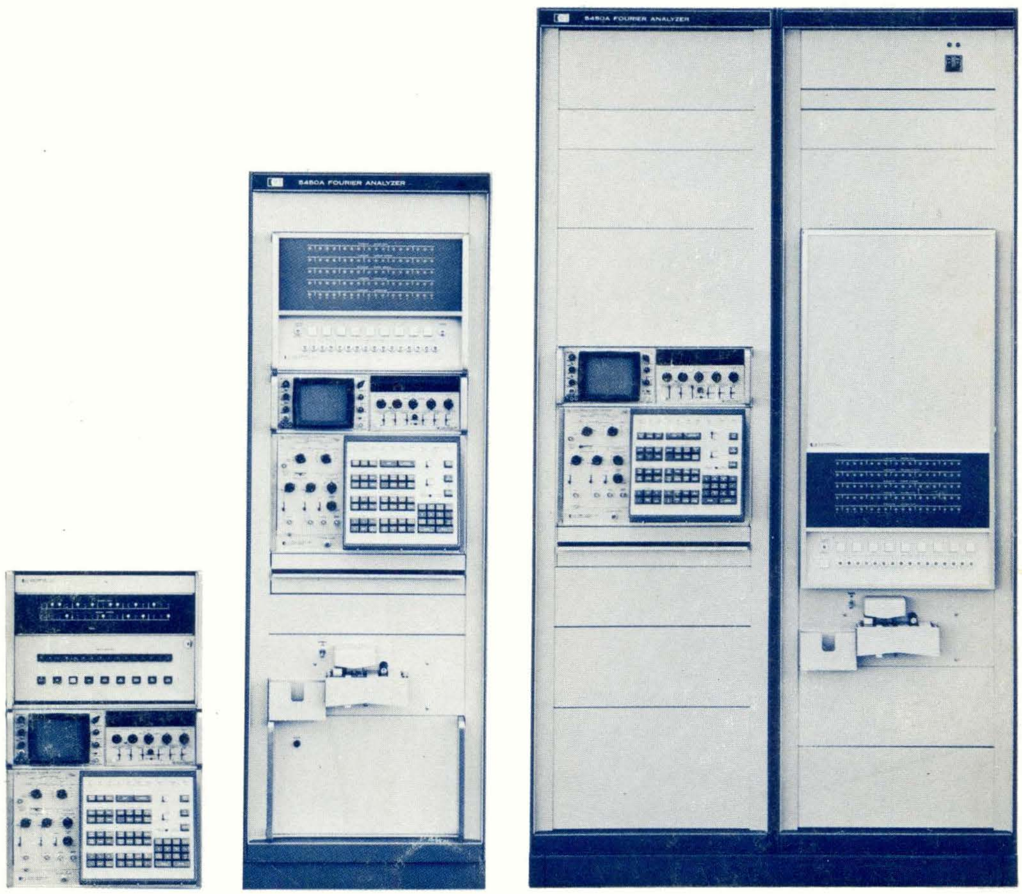
HP 5450A/5452A

SYSTEM OPERATING MANUAL

FOURIER ANALYZER SYSTEM

5450A/5452A



HEWLETT hp PACKARD

HP 5450A/5452A

FOURIER ANALYZER SYSTEM
5450A/5452A

Printed: SEP 1970

HEWLETT PACKARD

TABLE OF CONTENTS

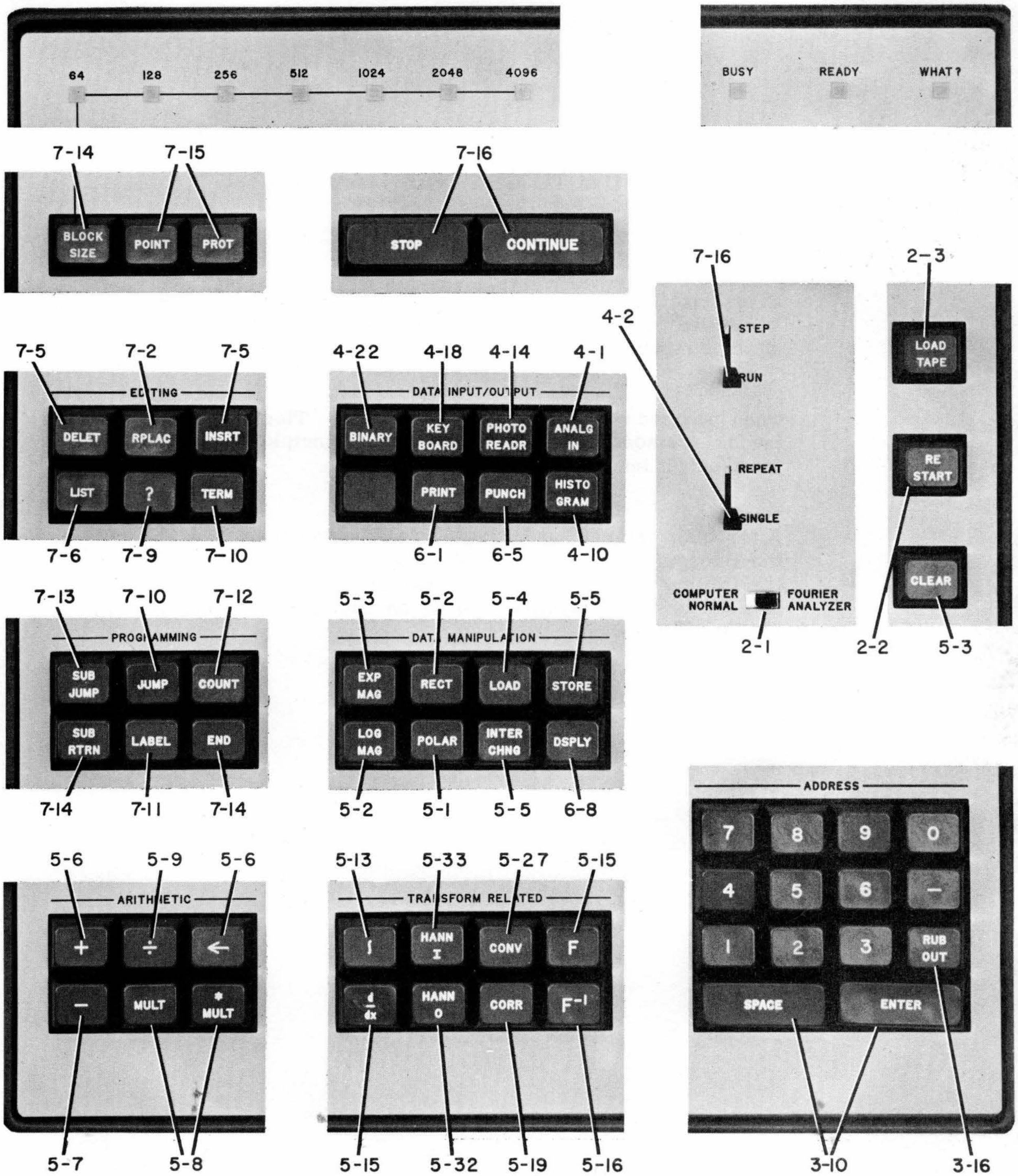
TABLE OF CONTENTS (Cont'd)

TABLE OF CONTENTS (Cont'd)

TABLE OF CONTENTS (Cont'd)

KEYBOARD REFERENCE

This page is intended as a quick reference for keyboard commands. Full details on the use of each key will be found on the pages indicated.



HOW TO USE THIS MANUAL

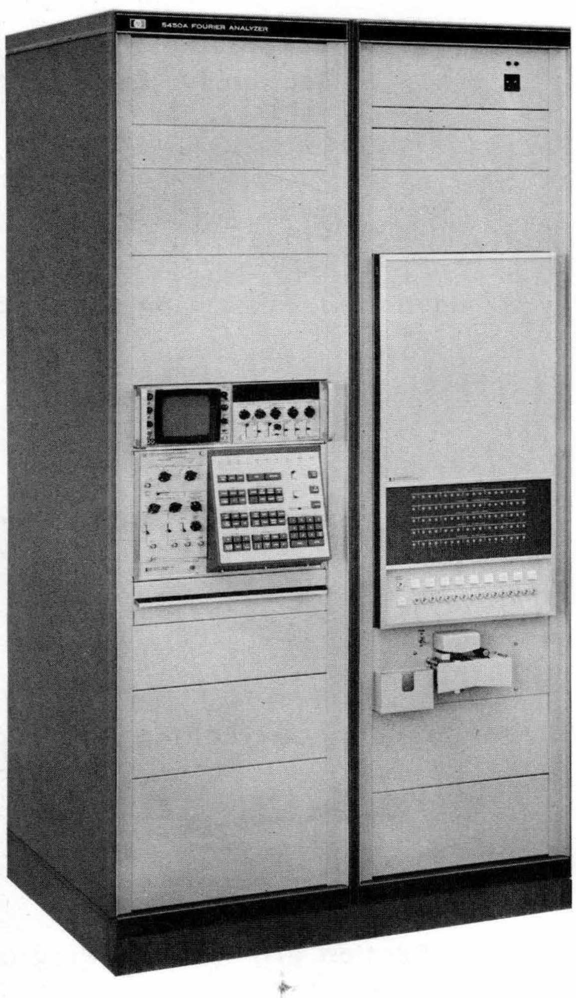
If you have just received your Fourier Analyzer, turn first to Section II, which gives complete instructions on turning on the instrument and loading the Fourier tape.

Then proceed straight through Section III. This is a self-teaching demonstration of some of the principle keys and input/output functions of the instrument.

After that, you can set up your own experiments based on the information in Sections IV through IX. In addition to the Table of Contents, note the Index in the back of the book and the marginal identifier words on each page as aids in finding information.

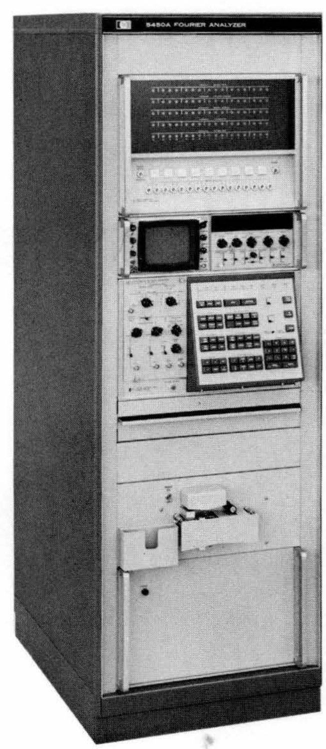
Figure 1-1. Fourier Analyzer Systems

5452A WITH 2114B COMPUTER



5450A WITH 2116B COMPUTER

5450A WITH 2115A COMPUTER



SECTION I
GENERAL INFORMATION

SYSTEM DESCRIPTION

The HP 5450A and 5452A Fourier Analyzers perform analyses of time and frequency data containing frequencies from 0 to 25 kHz. The system analyzes time-series data such as mechanical vibrations, sonar echoes, 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 frequencies, transfer functions and their corresponding coherence functions in complex systems. Both continuous and transient data may be processed.

Keyboard programming allows the user 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 (digital filtering)
Auto and cross correlation
Hanning and other weighting functions
Histogram
Scaling
Cepstrum
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 punched paper tape for easy re-entry into the Analyzer.

Data input and output is likewise controlled from the Keyboard. Data can be entered in analog form through the 10-bit two-channel Analog-to-Digital Converter, or in digital form on punched paper tape, or manually from the Keyboard, or directly from a remote computer.

Results of all operations are displayed on the oscilloscope. In addition, results can be printed out in decimal numbers on the Teleprinter, punched on paper tape, sent to a remote computer, displayed on a separate large-screen oscilloscope or plotted on an external X-Y plotter.

Serial no.'s

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. The 5450A uses the HP 2115A Computer with 8192-word memory; optionally, the 2116B Computer can be used, providing 8192 or 16384-word memory. Processing times with the 2116B are 20% faster. The 5452A uses the 2114A Computer with 8192-word memory. Each can be used as a stand-alone computer by setting a switch on the Keyboard. Computer programming knowledge is not required for operation of the Fourier Analyzer; all operations are controlled through the Keyboard.

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 Configuration Notice, supplied with the System. This Notice also tells what Computer interface channels are used by the various units (ADC, Teleprinter, 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:

Specif. H51-180AR Oscilloscope
Model 2114B, 2115A or 2116B Computer
Model 2748A Punched Tape Reader
Model 2752A Teleprinter
Model 5460A Display Plug-in Unit
Model 5465A ADC Plug-in Unit
Model 5475A Control Unit

Serial Numbers

Fourier Analyzer System

Each Fourier Analyzer System is identified by a two-section system serial number (5450A-000 or 5452A-000). The number is on a stick-on plate mounted inside rear of system cabinet. The 3-digit number is a serial number unique to each Fourier Analyzer system.

Each unit in the Fourier Analyzer System 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).

Computer Subsystem

Components of the Computer Subsystem have stick-on serial number plates with the legend "SYSTEM SERIAL 00000." The "system" serial number on these plates is for the Computer Subsystem of the Fourier Analyzer only, and is not the Fourier Analyzer system serial number.

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

CRT WARRANTY

The Fourier Analyzer and each of its individual units are certified and warranted as stated on inside front cover of this manual. The CRT (Cathode Ray Tube), however, is covered by a warranty separate from the rest of the system. The CRT warranty and warranty claim form are located in the H51-180A/AR Oscilloscope manual. Should CRT fail within time specified on warranty, return CRT with warranty form completed.

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 Fourier Analyzer System Service manual.

SPECIFICATIONS

GENERAL

The 5450A Fourier Analyzer is a digital instrument that performs analysis of time domain and frequency domain data. Analog or digital readout is provided along with a scale factor which indicates vertical amplitudes on the display unit. Control is executed through a keyboard; measurements can be implemented on an operation-by-operation basis, or measurement routines can be automatically executed. Six editing keys on the keyboard operate an on-line resident editor, so that automatically controlled measurement procedures may be changed without need to do off-line editing, compiling, or testing. The 5450A Fourier Analyzer is automatically and absolutely calibrated.

ANALOG INPUT

The Analog-to-Digital Converter (ADC) accepts one or two inputs. In two-channel operation both inputs are sampled simultaneously. Resolution of the ADC is 10 bits.

AMPLITUDE RANGE: 0.1 V to 10 V maximum in steps of 1, 2, 4, 10.

INPUT COUPLING: Ac or dc.

INPUT IMPEDANCE: 1 MΩ ±1% shunted by 45 pF max.

SENSITIVITY: 30 μV rms (sine wave).

CONVERSION GAIN (CHANNEL A):

Accuracy (as a Function of Frequency): ±0.2% ±1 x 10^{-4}%/Hz.

Temperature Stability: 0.005%/°C.

LINEARITY: Integral, ±0.05%; Differential, ±3%.

GAIN AND PHASE CHANNEL A TO B:

Conversion Gain A/B: ±0.2% ±4 x 10^{-4}%/Hz.

Temperature Stability: 0.01%/°C.

Phase and Delay A to B: ±0.2°, ±5 μs.

TRIGGER MODES: Slope and level controls are provided. The trigger input can be ac or dc coupled.

Internal: ADC triggers on signal to Channel A.

External: ADC triggers on signal applied to external input.

Line: ADC triggers on power line frequency.

Free Run: ADC triggers on data request from Digital Processor.

Trigger Output: Pulse output available on trigger.

SAMPLE RATE CONTROL:

Maximum Frequency/Time Between Samples Mode: Maximum frequency is selectable from 0.1 Hz to 25 kHz (0.1 Hz to 10 kHz in two-channel operation) in steps of 1, 2.5, 5, and 10. Equivalently, the sample rate is selected from 0.2 Hz to 50 kHz (0.2 Hz to 20 kHz in two-channel operation) in steps of 1, 2.5, 5, and 10.

Frequency Resolution/Total Time Mode: Frequency resolution is selectable from 0.2 mHz to 100 Hz in steps of 1, 2, 5, and 10. Equivalently, time duration of each ensemble is selected from 5,000 s to 10 ms in steps of 1, 2, 5, and 10.

External Clock: An external time base may be used to allow external control of the sample rate (the maximum sample rate of 50 kHz in single-channel operation to 20 kHz in two-channel operation cannot be exceeded.)

DISPLAY UNIT

Data may be displayed on the 8 x 10 cm oscilloscope or output to a plotter or remote oscilloscope in the following forms:

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

ANALOG DISPLAY ACCURACY: ±1%.

TYPES OF DISPLAY: Points, bars, or continuous (interpolation).

AMPLITUDE SCALE: Data in memory is automatically scaled to give a maximum on-screen calibrated display. The scale factor is given in volts/division, volts²/division, or in dB offset.

Linear Display Range: ±4 divisions with scale factor ranging from 1 x 10^{-150} to 5 x 10^{+150} in steps of 1, 2, 5, and 10.

Log Display Range: 4 decades with a scale factor ranging from 0 to ±998 dB.

Digital Up, Down Scale: Allows 8 up-scale, and 2 down-scale steps (calibrated continuous scale factor).

TIME AND FREQUENCY SCALE:

Linear Sweep Length: 10, 10.24, or 12.8 divisions.

Log Horizontal: 0.5 decade/division.

Markers: Intensity markers every 8th or every 32nd point.

ANALOG PLOTTER OUTPUT:

Amplitude: 0.5 V per oscilloscope display division.

Output Range: 5-300 pts/s (500 pts/s external timing).

Linearity: 0.1% of full scale.

DISPLAY CALIBRATION: Three intensity markers as well as vertical and horizontal position controls are provided for display calibration.

Origin: Left edge of display, zero amplitude.

+FS: Positive full scale, center of display.

—FS: Negative full scale, center of display.

BLOCK SIZES FOR TYPICAL MEASUREMENTS

The following table indicates some of the measurements made by the 5450A as well as the **maximum** block size available for these measurements.

MEASUREMENT	MAX. BLOCK SIZE N (Points/Ensemble) 8 K MEMORY	16 K MEMORY
Power Spectral Density—Ensemble Average	1024	4096
Voltage Spectrum—Ensemble Average	1024	4096
Cross Power Spectral Density—Ensemble Average	1024	2048
Transfer Function	512	2048
Coherence Function	512	1024
Auto Correlation of N/2 Lags	1024	4096
Cross Correlation of N/2 Lags	1024	4096
Cross Correlation of N/2 Lags—Ensemble Average	1024	2048
Auto Correlation of N/2 Lags—Ensemble Average	1024	2048
Power Spectral Density of One Shot Transient	1024	4096
Voltage Spectrum of One Shot Transient	1024	4096

COMPUTATIONAL SPEED

The speeds shown are based on using the 2116B Digital Processor. (These times are increased by 25% with the 2115A Digital Processor.)

FOURIER TRANSFORM:
Block Size 64: 52 ms.
Block Size 1024: 1.4 s.

POWER SPECTRUM ENSEMBLE AVERAGE:
Block Size 64: 110 ms/Spectral Estimate.
Block Size 1024: 2.1 s/Spectral Estimate.

CROSS POWER SPECTRUM ENSEMBLE AVERAGE:
Block Size 64: 210 ms/Spectral Estimate.
Block Size 1024: 4.4 s/Spectral Estimate.

KEYBOARD UNIT

There are 64 keys divided into eight major groups on the Keyboard Unit.

The Transform Related and Arithmetic groups include:

Fourier Transform	Division (Block or Integer)
Inverse Fourier Transform	Multiplication (Block or Integer)
Correlation	Conjugate Multiplication
Convolution	Addition
Linear Hanning	Subtraction
Quadratic Hanning	Integration
	Differentiation

DIGITAL ACCURACY AND RESOLUTION

All calculations use floating point arithmetic on a block basis. Data overflow does not occur. Amplitude resolution is 1 part in 16,000 worst case.

DATA MEMORY SIZE: 3,072 words (8,192 for a 16,384-word memory).

DATA BLOCK SIZE: Any power of 2 from 128 to 1,024 for 8,192-word memory; 64 to 1,024 for 16,384-word memory.

DATA WORD SIZE: 16-bit real and 16-bit imaginary or 16-bit magnitude and 16-bit phase.

COMPUTATIONAL RANGE: ±150 decades.

TRANSFORM ACCURACY: 0.1% worst case error during the forward or inverse calculation.

SPECTRAL RESOLUTION

The element of spectral resolution is the frequency channel width, the maximum frequency divided by ½ the data block size.

MAXIMUM FREQUENCY: 25 kHz single channel; 10 kHz dual channel. Adjustable in steps of 1, 2.5, 5, and 10 down to 0.1 Hz.

FREQUENCY CHANNEL WIDTH: <3.2% down to <0.2% of the maximum frequency in steps of 2 (down to <0.05% for 16,384-word Processor).

SPECTRAL RESOLUTION OF TWO EQUAL AMPLITUDE SINE WAVES: If separated by 3 frequency channel widths, there will be a null of at least 3 dB between them; if separated by 7 frequency channel widths, the relative magnitudes will be correct to within 0.1%. The power spectrum for two equal amplitude sine waves separated by 5 frequency channels will have the correct relative magnitude to within 0.1%.

DYNAMIC RANGE: 4 decades over ±150 decades.

FREQUENCY ACCURACY: ±0.01%.

TIME DOMAIN RESOLUTION

The element of time resolution is the time channel width, the time sample record length divided by the block size.

MAXIMUM SAMPLE RECORD LENGTH: Product of data block size and time channel width. (In Ensemble Averaging up to 32,767 Sample Record Lengths may be used for a statistical estimate.)

TIME CHANNEL WIDTH: 20 μs, single channel; 50 μs, dual channel, up to 5 s in steps of 1, 2, 5. Accuracy 0.01%.

POWER REQUIREMENTS, SIZE, WEIGHT

POWER SOURCE: 115/230 V ±10%, 50/60 Hz.

ENVIRONMENTAL CONDITIONS: 0°C to 55°C using 2116B Digital Processor (10°C to 40°C using 2115A Digital Processor).

SIZE: Dimensions are for a typical system with cabinet (excluding Teleprinter). Height, 64.25 inches (1632 mm). Width, 21 inches (533 mm). Depth, 30 inches (762 mm).

WEIGHT: Shipping weight for a typical system with cabinet (excluding Teleprinter), 662 lb (301 kg).

PRICING/ORDERING INFORMATION

Complete ordering and pricing information for the 5450A Fourier Analyzer Systems is contained on a separate sheet available from Hewlett-Packard Field Sales Offices.

5452A Specs

GENERAL

The 5452A Fourier Analyzer is a digital instrument that performs analysis of time domain and frequency domain data. Analog or digital readout is provided along with a scale factor which indicates vertical amplitudes on the display unit. Control is executed through a keyboard; measurements can be implemented on an operation-by-operation basis, or measurement routines can be automatically executed. Six editing keys on the keyboard operate an on-line resident editor, so that automatically controlled measurement procedures may be changed without need to do off-line editing, compiling, or testing. The 5452A Fourier Analyzer is automatically and absolutely calibrated.

ANALOG INPUT

The Analog-to-Digital Converter (ADC) accepts one or two inputs. In two-channel operation both inputs are sampled simultaneously. Resolution of the ADC is 10 bits.

AMPLITUDE RANGE: 0.1 V to 10 V maximum in steps of 1, 2, 4, 10.

INPUT COUPLING: Ac or dc.

INPUT IMPEDANCE: 1 MΩ ±1% shunted by 45 pF max.

SENSITIVITY: 30 μV rms (sine wave).

CONVERSION GAIN (CHANNEL A):

Accuracy (as a Function of Frequency): ±0.2% ±2 x 10^{-4}%/Hz.

Temperature Stability: 0.005%/°C.

LINEARITY: Integral, ±0.05%; Differential, ±3%.

GAIN AND PHASE CHANNEL A TO B:

Conversion Gain A/B: ±0.4% ±4 x 10^{-4}%/Hz.

Temperature Stability: 0.01%/°C.

Phase and Delay A to B: ±0.2°, ±5 μs.

TRIGGER MODES: Slope and level controls are provided. The trigger input can be ac or dc coupled.

Internal: ADC triggers on signal to Channel A.

External: ADC triggers on signal applied to external input.

Line: ADC triggers on power line frequency.

Free Run: ADC triggers on data request from Digital Processor.

Trigger Output: Pulse output available on trigger.

SAMPLE RATE CONTROL:

Maximum Frequency/Time Between Samples Mode: Maximum frequency is selectable from 0.1 Hz to 25 kHz (0.1 Hz to 10 kHz in two-channel operation) in steps of 1, 2.5, 5, and 10. Equivalently, the sample rate is selected from 0.2 Hz to 50 kHz (0.2 Hz to 20 kHz in two-channel operation) in steps of 1, 2.5, 5, and 10.

Frequency Resolution/Total Time Mode: Frequency resolution is selectable from 0.2 mHz to 100 Hz in steps of 1, 2, 5, and 10. Equivalently, time duration of each ensemble is selected from 5,000 s to 10 ms in steps of 1, 2, 5, and 10.

External Clock: An external time base may be used to allow external control of the sample rate (the maximum sample rate of 50 kHz in single-channel operation to 20 kHz in two-channel operation cannot be exceeded.)

DISPLAY UNIT

Data may be displayed on the 8 x 10 cm oscilloscope or output to a plotter or remote oscilloscope in the following forms:

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

ANALOG DISPLAY ACCURACY: ±1%.

TYPES OF DISPLAY: Points, bars, or continuous (interpolation).

AMPLITUDE SCALE: Data in memory is automatically scaled to give a maximum on-screen calibrated display. The scale factor is given in volts/division, volts²/division, or in dB offset.

Linear Display Range: ±4 divisions with scale factor ranging from 1 x 10^{-150} to 5 x 10^{+150} in steps of 1, 2, 5, and 10.

Log Display Range: 4 decades with a scale factor ranging from 0 to ± 998 dB.

Digital Up, Down Scale: Allows 8 up-scale, and 2 down-scale steps (calibrated continuous scale factor).

TIME AND FREQUENCY SCALE:

Linear Sweep Length: 10, 10.24, or 12.8 divisions.

Log Horizontal: 0.5 decade/division.

Markers: Intensity markers every 8th or every 32nd point.

ANALOG PLOTTER OUTPUT:

Amplitude: 0.5 V per oscilloscope display division.

Output Range: 5-300 pts/s (500 pts/s external timing).

Linearity: 0.1% of full scale.

DISPLAY CALIBRATION: Three intensity markers as well as vertical and horizontal position controls are provided for display calibration.

Origin: Left edge of display, zero amplitude.

+FS: Positive full scale, center of display.

—FS: Negative full scale, center of display.

BLOCK SIZES FOR TYPICAL MEASUREMENTS

The following table indicates some of the measurements made by the 5450A as well as the **maximum** block size available for these measurements.

MEASUREMENT	MAX. BLOCK SIZE N (Points/Ensemble)
Power Spectral Density—Ensemble Average	1024
Voltage Spectrum—Ensemble Average	1024
Cross Power Spectral Density—Ensemble Average	1024
Transfer Function	512
Coherence Function	512
Auto Correlation of N/2 Lags	1024
Cross Correlation of N/2 Lags	1024
Cross Correlation of N/2 Lags—Ensemble Average	1024
Auto Correlation of N/2 Lags—Ensemble Average	1024
Power Spectral Density of One Shot Transient	1024
Voltage Spectrum of One Shot Transient	1024

COMPUTATIONAL SPEED

FOURIER TRANSFORM:
Block Size 128: 150 ms.
Block Size 1024: 1.9 s.

POWER SPECTRUM ENSEMBLE AVERAGE:
Block Size 128: 350 ms/Spectral Estimate.
Block Size 1024: 3.4 s/Spectral Estimate.

CROSS POWER SPECTRUM ENSEMBLE AVERAGE:
Block Size 128: 580 ms/Spectral Estimate.
Block Size 1024: 5.6 s/Spectral Estimate.

KEYBOARD UNIT

There are 64 keys divided into eight major groups on the Keyboard Unit.

The Transform Related and Arithmetic groups include:

Fourier Transform	Division (Block or Integer)
Inverse Fourier Transform	Multiplication (Block or Integer)
Correlation	Conjugate Multiplication
Convolution	Addition
Linear Hanning	Subtraction
Quadratic Hanning	Integration
	Differentiation

DIGITAL ACCURACY AND RESOLUTION

All calculations use floating point arithmetic on a block basis. Data overflow does not occur. Amplitude resolution is 1 part in 16,000 worst case.

DATA MEMORY SIZE: 3,072 words.

DATA BLOCK SIZE: Any power of 2 from 128 to 1,024

DATA WORD SIZE: 16-bit real and 16-bit imaginary or 16-bit magnitude and 16-bit phase.

COMPUTATIONAL RANGE: ±150 decades.

TRANSFORM ACCURACY: 0.5% worst case error during the forward or inverse calculation.

SPECTRAL RESOLUTION

The element of spectral resolution is the frequency channel width, the maximum frequency divided by ½ the data block size.

MAXIMUM FREQUENCY: 25 kHz single channel; 10 kHz dual channel. Adjustable in steps of 1, 2.5, 5, and 10 down to 0.1 Hz.

FREQUENCY CHANNEL WIDTH: <3.2% down to <0.2% of the maximum frequency in steps of 2.

SPECTRAL RESOLUTION OF TWO EQUAL AMPLITUDE SINE WAVES: If separated by 3 frequency channel widths, there will be a null of at least 3 dB between them; if separated by 5 frequency channel widths, the relative magnitudes will be correct to within 1.0%. The power spectrum for two equal amplitude sine waves separated by 4 frequency channels will have the correct relative magnitude to within 1%.

DYNAMIC RANGE: 65 dB voltage magnitude spectrum, 40 dB auto power spectrum ±150 decades.

FREQUENCY ACCURACY: ±0.01%.

TIME DOMAIN RESOLUTION

The element of time resolution is the time channel width, the time sample record length divided by the block size.

MAXIMUM SAMPLE RECORD LENGTH: Product of data block size and time channel width. (In Ensemble Averaging up to 32,767 Sample Record Lengths may be used for a statistical estimate.)

TIME CHANNEL WIDTH: 20 μs, single channel; 50 μs, dual channel, up to 5 s in steps of 1, 2, 5. Accuracy 0.01%.

POWER REQUIREMENTS, SIZE, WEIGHT

POWER SOURCE: 115/230 V ±10%, 50/60 Hz.

ENVIRONMENTAL CONDITIONS: 10°C to 40°C

SIZE: Dimensions are for a typical system (excluding Teleprinter). Height, 33.88 inches (847 mm). Width, 33.5 inches (838 mm). Depth 24.25 inches (606 mm).

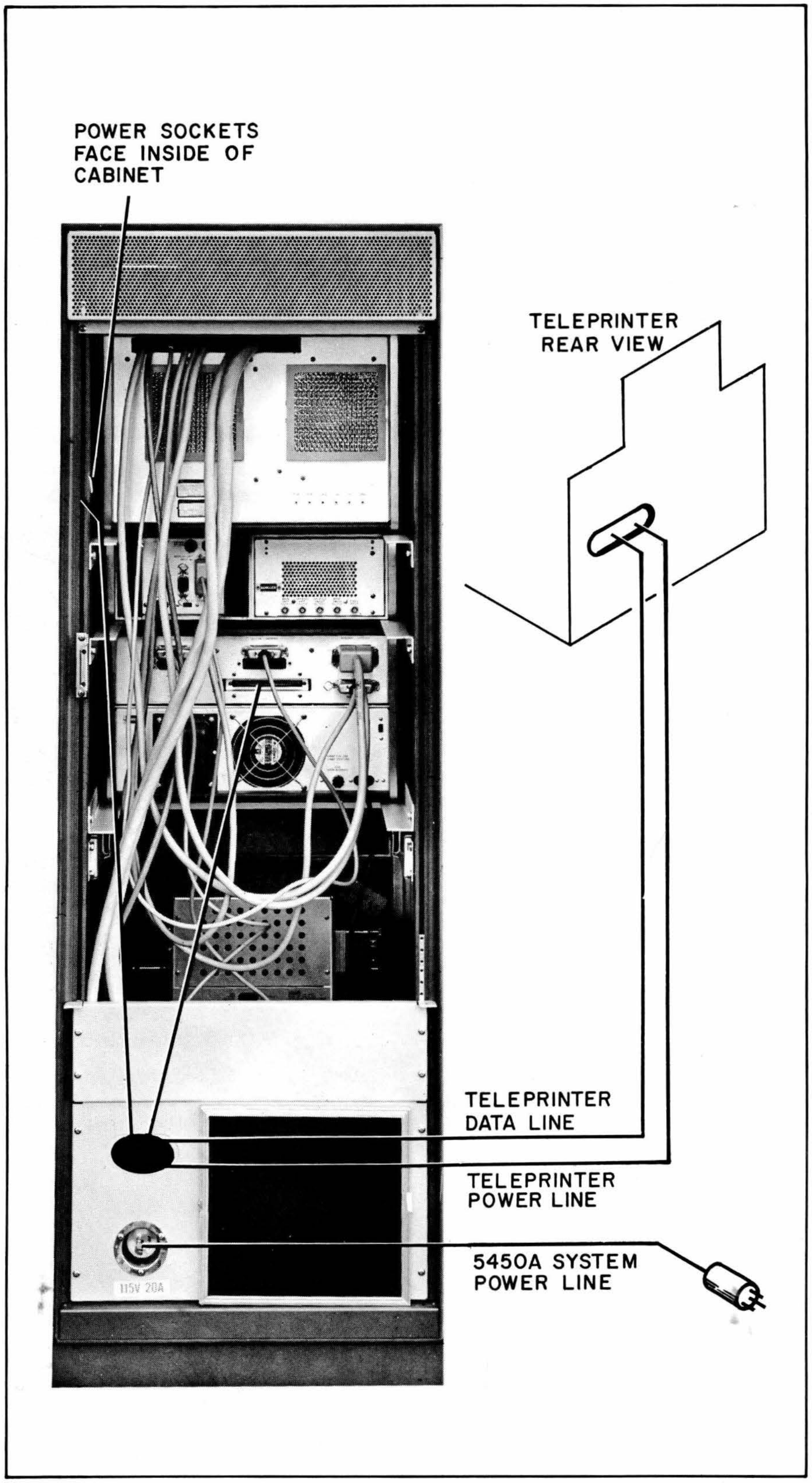
WEIGHT: Shipping weight for a typical system (excluding Teleprinter), 256 lb (116 kg).

PRICING/ORDERING INFORMATION

Complete ordering and pricing information for the 5452A Fourier Analyzer Systems is contained on a separate sheet available from Hewlett-Packard Field Sales Offices.

Figure 2-1. 5450A System ready for turn-on

Turn-on



SECTION II

TURN-ON/OFF PROCEDURE

Power turn-on

ABOUT THIS SECTION

This section provides instructions for initial and routine turn-on/off of the Fourier Analyzer System. It includes instructions on loading the Fourier Master Program Tape, and reloading the Basic Binary Loader.

PREPARATION

The following assumes that the Fourier Analyzer System has been installed and checked out. In the case of the 5450A and of rack-mounted 5452A's, the Teleprinter power cord should be plugged into a socket in the strip on the left side (viewed from rear) of the system cabinet, as shown in Figure 2-1. (Plugging the Teleprinter into this strip ensures that it will be turned off when the system power is turned off; if the Teleprinter is plugged directly into a 115V ac wall socket, it will continue to run when the main system power is turned off.) The Teleprinter data line should be plugged in as shown in Figure 2-1, and the system's main power line plugged into a grounded 115V ac source.

In the case of the 5452A without rack-mounting, plug the Teleprinter power cord into an ac socket. It is preferable though not required to use a multi-socket box with a switch to plug all the power cords into, so that power can be turned on and off for the entire system by a single switch. The Teleprinter data line should be plugged into the back of the computer as shown in Figure 2-1.

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, if your system has one, to turn the Fourier Analyzer on and off.

1. Set Keyboard switch to FOURIER ANALYZER.

2. Turn on cabinet power: the method varies slightly, depending on the cabinet in which your system is mounted.

   In the case of 5450A's and 5452A's with rack-mounting:

   Press button or turn on switch in upper right corner of cabinet.

   In the case of 5450A's and 5452A's without rack-mounting:

   There is no cabinet power switch. Turning on each of the subsystems as described below, turns on the system.

3. Turn on Computer power; the method varies slightly, depending on which computer you have.

Power turn-on

   a. 2114B: open front panel door and flip POWER toggle switch to ON.

   b. 2115A: flip toggle switch lever to its up position.

   c. 2116B: press POWER pushbutton, if it is not lighted.

4. Turn on oscilloscope power by pressing green pushbutton, if it is not lighted.

5. Turn on Keyboard power by pressing green pushbutton, if it is not lighted. When the Keyboard is turned on, either the BUSY or READY lamp will be lighted.

   a. If the READY lamp is lighted, the system is operating properly in the Fourier mode.

   b. If the BUSY lamp is lighted, and the system does not seem to respond to Keyboard commands, press RESTART.

   If the READY lamp does not light, or the BUSY lamp goes out, all of part of the Master Fourier Program has probably been erased. Load the Fourier Program as described below under "Loading Fourier Program Tape".

6. Auxiliary equipment (Tape Reader; Teleprinter; Tape Punch, if your system has one; etc.) should be turned off when not being used to prevent unnecessary wear on the mechanical parts in these devices.

When the system is turned on, the Computer (Digital Processor) returns to the state (address, program, etc.) it was at when the system was turned off. If the Analyzer is in the COMPUTER NORMAL mode, and the Keyboard switch is switched to FOURIER ANALYZER, the RESTART button must be pressed before any Fourier Analysis can begin.

LOADING FOURIER PROGRAM TAPE

The Fourier Analyzer is shipped from the factory with the Fourier Program Tape loaded into the Computer's memory; it should be necessary only to follow the turn-on procedure given above to turn the system on. However, if the Computer has been used as a general-purpose digital computer (i.e., operating in the COMPUTER NORMAL mode), or if some accident has destroyed the Fourier Program in memory, it will be necessary to reload the Fourier Program Tape, using the procedure given below. If you are not sure whether the complete and correct program is in memory, load the tape. Reloading will not modify data that was previously stored in memory by the Fourier program.

Be sure the tape you load has the correct part no., per the following table:

Load Fourier tape

System	Part No. of Fourier Tape
5450A: 8K memory, 2115A Computer	05450-90003
5450A: 8K memory, 2116B Computer	05450-90003
5450A: 16K memory, 2116B Computer	05450-90004
5450A with accessories such as magnetic tape or disc unit: see accessory manual for part no. of Fourier Tape. (Part no. will be 05450-90005 or higher.)	
5452A: 8K memory, 2114B Computer	05452-90003
Note: Some systems may come with two or more Fourier tapes: one of the above plus others for the system configured with any additional options that might have been purchased. Be sure to read the instructional literature for the options to determine the correct part number for the Fourier tape to be loaded.	

Procedure for 5450A with 2748A High Speed Tape Reader
(hereafter called Photoreader)

1. Set Keyboard switch to FOURIER ANALYZER, enabling Keyboard control.

2. a. Place Fourier tape roll in tape holder in Photoreader, feed holes toward rear.

   b. Press POWER pushbutton.

   c. Press LOAD pushbutton on Photoreader.

   d. Run the tape leader underneath the wire-guide and through the pair of feed rollers.

   e. Press READ pushbutton.

3. a. Press LOAD TAPE on Keyboard. Tape should run through Photoreader and stop. The machine is now calibrated such that the vertical log scale is in power dB, that is, 5 dB/division on the scope. If you wish a voltage dB calibration, press LOAD TAPE again; a final section of the tape will run through, changing the vertical log scale calibration to voltage dB, or 10 dB/division.

b. After the tape has run through, press LOAD pushbutton on Photoreader and remove tape. Rewind it and return to its box.

Load Fourier tape

c. Press RESTART.

If tape will not run through:

Check the following:

Is Keyboard switch set to FOURIER ANALYZER?

Are tape feed holes toward instrument front panel?

Are you starting in a blank section of tape (except for feed holes) so that no data is lost?

Is tape running under wire guide in Photoreader?

If you still have problems loading the Fourier Program Tape refer to "Loading Problems," below.

Procedure for 5452A Fourier Analyzer

1. Set Keyboard switch to FOURIER ANALYZER.
2. If system has no Photoreader:

   Place Fourier tape roll in Tape Reader on Teleprinter (lower left front). Be sure switch is OFF. Close the plastic cover.

   If system has Photoreader:

   a. Place Fourier tape roll in tape holder in Photoreader, feed holes toward the rear.

   b. Press POWER pushbutton.

   c. Press LOAD pushbutton on Photoreader.

   d. Run the tape leader underneath the wire-guide and through the pair of feed rollers.

   e. Press READ pushbutton.

3. Open the front panel door of the 2114B Computer; on inside of door, upper left corner, set CONSOLE switch to NORMAL.
4. Close front panel door, press HALT.
5. Press CLEAR REGISTER.
6. If system has no Photoreader:

   a. Simultaneously press PRESET and LOAD and simultaneously release.

   b. Set switch on Teleprinter tape reader to START. Tape should now run through.

If system has Photoreader:

Simultaneously press PRESET and LOAD and simultaneously release. Tape should now run through.

7. When the tape stops, the machine is calibrated such that the vertical log scale is in power dB, that is, 5 dB/division on the scope. If you wish a voltage dB calibration, simultaneously press PRESET and LOAD again, and simultaneously release. A final section of tape will run through, changing the vertical log scale calibration to voltage dB, or 10 dB/division. In either case, when the tape has run through, the M register should show 102077_8.
8. Press CLEAR REGISTER.
9. Set switch 1 on the switch register to the on position. Press LOAD ADDRESS.
10. Press PRESET, then RUN. The Fourier Analyzer is now on and ready to operate. The READY light on the Keyboard should be lighted. If it is not, press RESTART.
11. Open front panel door of Computer, and set CONSOLE switch to LOCK. This will prevent any accidental pressing of a switch register switch from ruining the Fourier program.

LOADING PROBLEMS

Sometimes operator error or dirty equipment can prevent a tape from being loaded properly. If you are having trouble loading tape in the Photoreader consider the possibility of having one or more of the following problems. If the tape still doesn't run, then the Basic Binary Loader—the program which loads tape into the machine—has probably been erased. To reload, see "Reloading the Basic Binary Loader".

1. Tape not correctly placed in Photoreader.
   a. The row of small feed holes (except for bootstrap loader leader) is not centered on the tape; when the tape is placed in the Reader, these holes should be toward the Analyzer front panel, instead of toward you.
   b. Tape placed too far to right through Reader. Unless otherwise instructed, all data holes should be to left of reading element before tape begins its run through Reader.
   c. Tape loaded backward in Reader. All tapes supplied with your system are identified by a label at the beginning of the tape. When rewinding the tape, be sure this leader is at the outside of the roll.
2. Tape torn or dirty. Tape that is faulty, or not clean, may be read incorrectly.
3. Dirty Reader. To clean Reader;
   a. Try blowing dirt away with your breath.
   b. Remove tape and clean Reader using alcohol and brush supplied with your system. Brush feeder holes clean; also lightly brush bottom of Reader lens, if your system has a Photoreader. Blow dirt away with your breath.

RELOADING BASIC BINARY LOADER

Basic Binary Loader

The Basic Binary Loader (BBL) is the program that loads any tape into the Analyzer memory. It occupies the last 64 words in the 8K or 16K word memories. When the system is in the FOURIER ANALYZER mode, this program is automatically protected from erasure. In the case of the 5450A, in the COMPUTER NORMAL mode, the program can be protected by placing the LOADER switch in the PROTECTED position. Thus the only way BBL can be erased is by operating in the COMPUTER NORMAL mode with the LOADER switch ENABLED. In the 5452A, the Basic Binary Loader is protected by setting the LOADER switch on the inside of the front panel door to the LOCK position.

Once the BBL has been erased, it must be reloaded using the "bootstrap" (short length of tape) supplied. Be sure your tape has the correct part no.:

System	Bootstrap part no.
5450A, 8K memory	05450-90006
5450A, 16K memory	05450-90007
5452A without Photoreader	05452-90006
5452A with Photoreader	05452-90005

Systems having accessories such as the magnetic tape or disc unit come with two bootstraps: one for the system configured for the accessories, and one for a standard system (without accessories) in case the user wants to reconfigure his system back to a standard system. In any case, use the appropriate bootstrap for your system. Bootstraps for non-standard systems have part numbers different from those listed above.

To reload the BBL:

1. Set Keyboard switch to COMPUTER NORMAL.
2. If your machine is a 5452A, open the front panel door and, on the inside of the door, set the CONSOLE switch to NORMAL.
3. Enter the octal number 20 in the computer switch register. The switch register number system is explained in Figure 2-2.

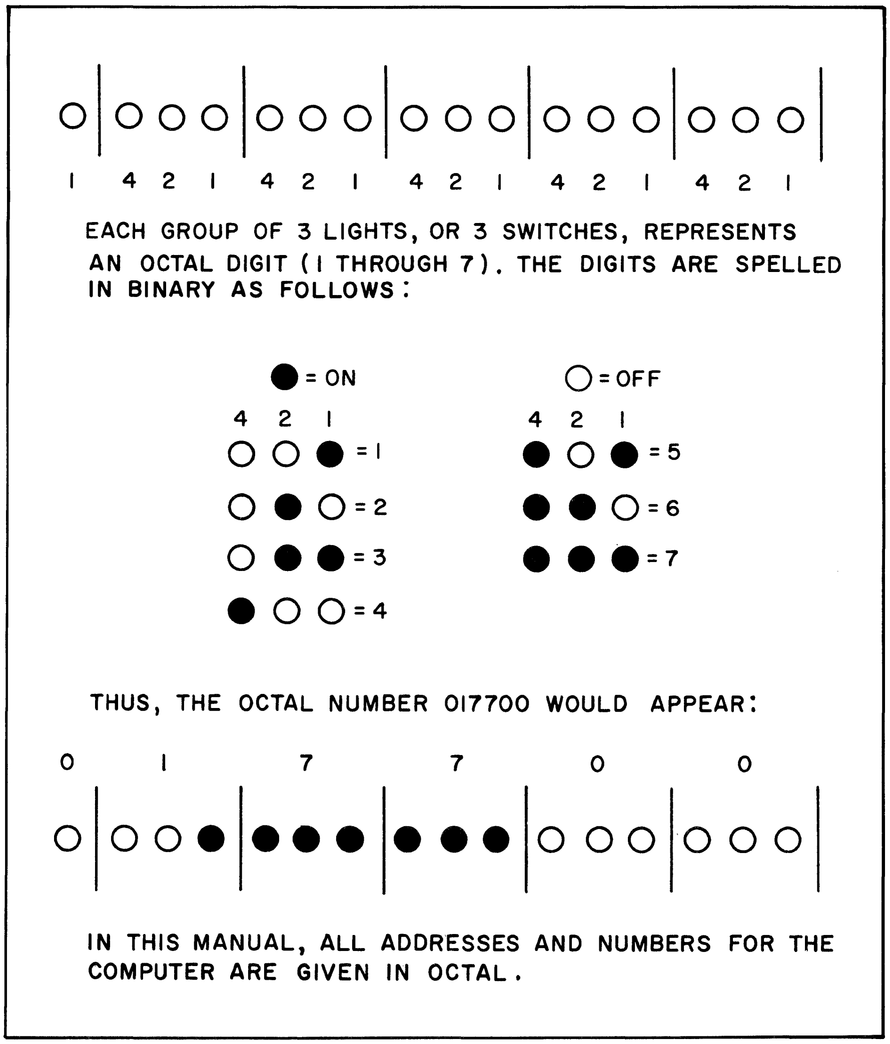
   Press LOAD ADDRESS.

   Enter the following octal numbers into the switch register, pressing LOAD MEMORY after each number. Note that in some cases the numbers will be different if your system doesn't have a Photoreader.

1037CC*
1023CC*
024021
1025CC*
001727
1037CC*
1023CC*
024026
1024CC*
170001
006004
024020

*These two letters represent the tape input device channel. If your machine has a Photoreader, then CC = 10; if it has no Photoreader, then CC = 11, for Teleprinter input. For non-standard systems, such as those with magnetic tape or disc units, check the configuration label for the correct tape input device channel no. This label is located inside the top cover of the 2114B and 2115A Computers, and inside the front panel door of the 2116B.

Figure 2-2. How to read octal numerical system on Computer panel



Basic Binary Loader

4. Set 000001 into the switch register. Push LOAD ADDRESS.
5. Set 077700 into the switch register. Push LOAD MEMORY.
6. Set 000020 into the switch register. Push LOAD ADDRESS.
7. Place the bootstrap in the Photoreader (if your system has one) or otherwise in the Teleprinter Reader the same way as you did the Fourier tape.
8. If you have a 5450A system, set the LOADER switch to ENABLE. If you have a 5452A system, open the front panel door and set the LOADER switch on the inside of the door to ENABLE.
9. Be sure POWER switch on Photoreader is on.
10. Push PRESET.
11. Push RUN. The tape should now run through.
12. On 5450A's, set the LOADER switch to PROTECTED. On the 5452A, set the LOADER switch on the inside of the front panel door to NORMAL.
13. On the 5452A, set the CONSOLE switch on the inside of the front panel door to LOCK. This will prevent any accidental pressing of the switch register switches from ruining the Fourier program. All other switches on the inside of the panel door should be on NORMAL.

SECTION III
LEARNING TO OPERATE THE FOURIER ANALYZER

ABOUT THIS SECTION

This section is intended to be a self-teaching guide for the new owner of the Fourier Analyzer. It is a demonstration of some of the basic Keyboard functions with photos of how the corresponding scope displays should look. The demonstration also explains the use of the rest of this manual, which is set up in reference form so that the user can quickly find the information he needs for his specific purposes.

Data word, block

In order not to interrupt the procedure once it has begun, background information on three important subjects is being placed first: These are (1) data memory format, (2) ADC sampling parameters, and (3) the Keyboard command structure. These should be read before beginning the demonstration.

HOW DATA IS STORED IN THE ANALYZER MEMORY

The Data Word

The smallest element of data that may be entered or taken out of the 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 an input 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 -32768 to +32767, a range of greater than 92 dB. Groups of these data words are collected together to form the next element of data storage, the data block, as shown in Figure 3-1.

The Data Block

The data block may contain a quantity of data words equal to any power of 2 from $2^6 = 64$ to $2^{10} = 1024$ for an Analyzer with an 8192 word memory or from $2^6 = 64$ to $2^{12} = 4096$ data words with the 16,384 word memory. 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.

Associated with each data block are an amplitude scale factor and a frequency code. The amplitude scale factor is a power of 2 which multiplies all N data words in the block. The data word and the block scale factor may thus represent any number from $\pm 1 \times 10^{-150}$ to $\pm 1 \times 10^{+150}$. 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 scale factor is incremented by 1. In this way, full accuracy of the data word is used and no loss of calibration occurs during calculations. The frequency code defines the sampling parameters used by the Analog-to-Digital Converter (ADC), if data was taken in via the ADC mode. The table of frequency codes is provided at appropriate places throughout this manual, for example, page 4-24.

Data block

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. This is a result of the fact that a time series of N independent points results in a frequency spectrum of N/2 independent frequencies. 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 hence are not stored. 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 a 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 3-1).

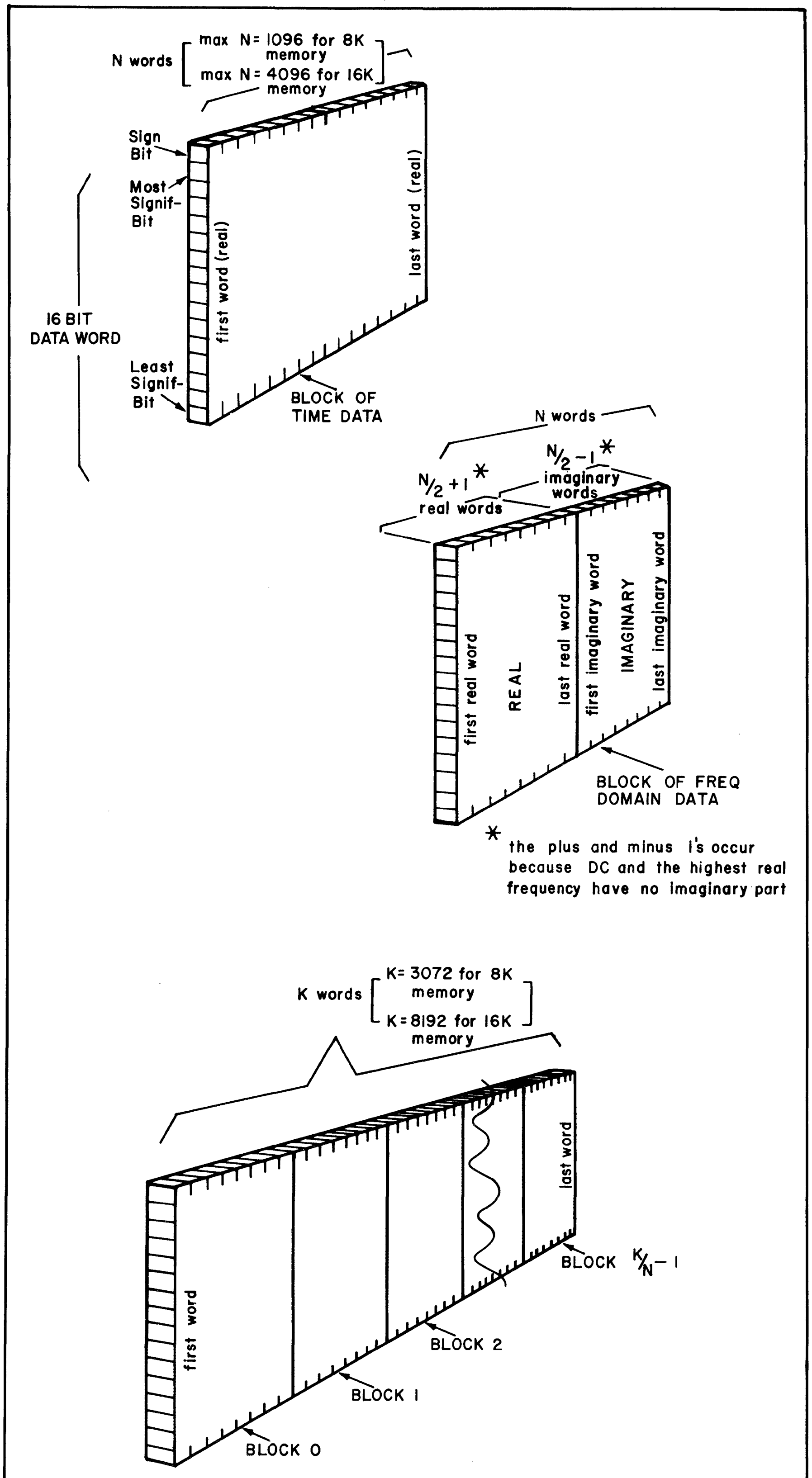
When the data block contains a time record, the first sample point of the time record is contained in the first word (numbered 0) of the data block and the last point in the time record is found in the last word (numbered N - 1) of the data block. But when the data block contains values of a frequency function, the lowest frequency (dc) value of the real part or the magnitude is found in the first word, 0, of the data block and the highest real or magnitude frequency point is found in the N/2 word. The lowest imaginary or phase frequency value is found in the N/2 + 1 word, and the highest imaginary or phase value is found in the N - 1 word.

The Input and the Processing Operations sections of this manual describe in detail how to direct the data word into and out of the real/magnitude or imaginary/phase part of memory. 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.

Since the part of the Analyzer memory reserved for data contains 3,072 words, and the maximum data block size allowed is 1,024 words (8192 and 4096 words respectively for the 16K Analyzer), more than one block of data may be held in memory at one time. This allows flexibility in using the Analyzer for operations such as ensemble averaging, where data from several stages of an operation must be simultaneously held in memory.

Data block

Figure 3-1. Data/memory organization



At any particular time all data blocks must be at 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 the 8K Analyzer (3072 data words total), 6 data blocks numbered from 0 to 5 are available. As shown in Figure 3-1 these data blocks lie in consecutive order in the memory with the first word (0 word) 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:

Data block

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 which will also be the 260th data word of the memory. 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 the 260th 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 5th word of data block 1. This word would be numbered channel 4 of data block 1 since the first channel in the data block is number 0.

Warning: 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^{-3} and the block 1 scale factor is 2×10^{-2}. Now the block size is doubled. Then block 0 contains data having two different scale 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.

Now suppose the original block size is halved. The original block 0 then becomes two blocks: The scale factor for the new block 0 being the same as the scale factor for the original, but the scale factor for the new block 1 (which by rights should have the same factor as the new block 0) is now taken from the original block 1, which probably was different. So again the possibility exists for erroneous scale factor displays.

If the scale factors in the original blocks are all the same, then of course, there is no problem.

ADC SAMPLING PARAMETERS

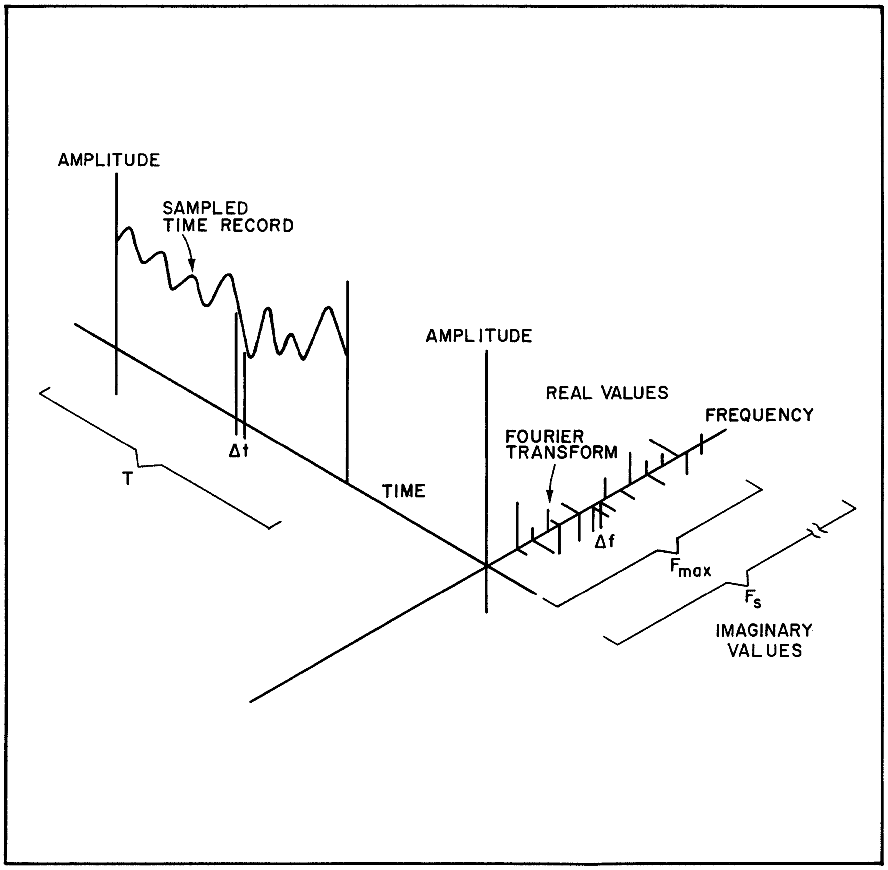
Figure 3-2 shows a sampled time record and its corresponding spectrum. Terms which define the scale of the time and frequency records are Δt, T, F_{max}, F_s, Δf. These terms are related to each other as follows:

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

$$F_s = 1/\Delta t$$

ADC parameters

Figure 3-2. 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 a sample of Δt, 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 Δt, F_{max}, Δf, and T completely determine the time and frequency scales of the Fourier Analyzer.

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

Time parameters

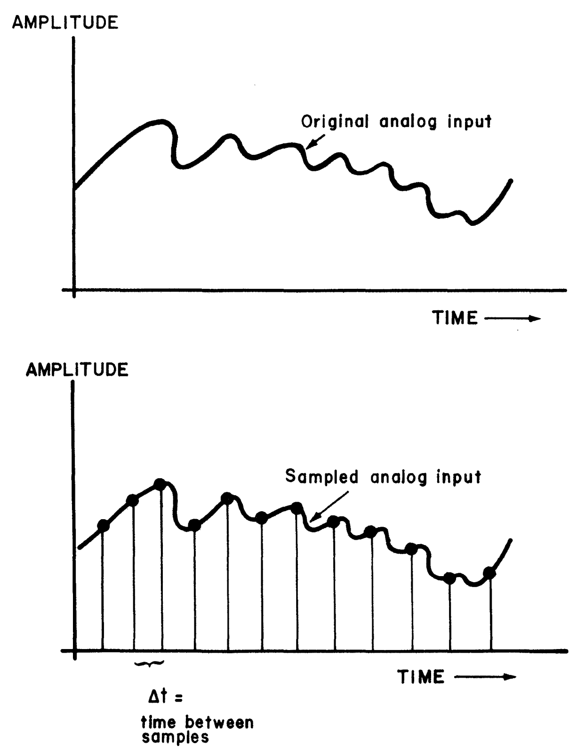
Time Domain Parameters

- 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 3-3 it can be seen that:

total record length = No. of samples x sample interval

$$T = N \times \Delta t$$

Figure 3-3. Time domain parameters



N Samples Δt apart = T

$$T = N\Delta t$$

Frequency Domain Parameters

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 (fundamental frequency); next point is 2 Δf (first harmonic); next 3 Δf (second harmonic), etc. No finer resolution than Δf may be assigned to any frequency value.

N/2 The number of frequency points: this is half the block size, or N/2, because the frequency information is broken down into two displays: real or magnitude (depending on MODE switch setting), and imaginary or phase (depending on MODE switch setting).

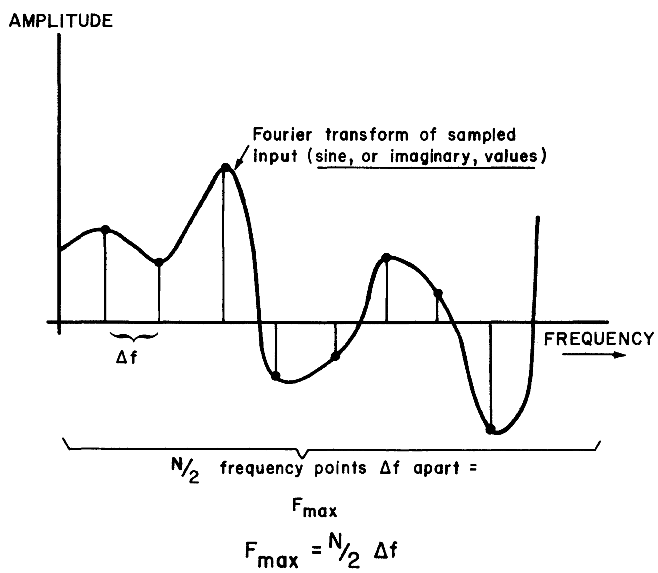
F_{max} The maximum frequency of the display, or in other words, the bandwidth. (MAX FREQ on the ADC panel). From Figure 3-4 it can be seen that:

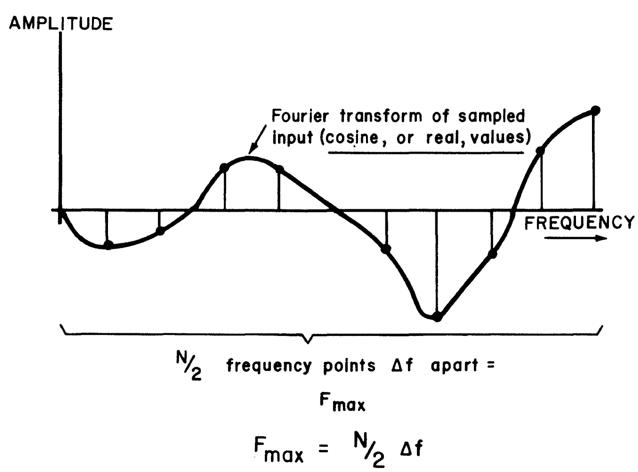
Freq parameters

Maximum frequency = No. of frequency points x frequency resolution

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

Figure 3-4. Frequency domain parameters





Choosing Sample Parameters

Table 3-1 summarizes the equations above, and is also given with the analog-input command on page 4-5. 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 3-1. 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.

Choosing parameters

Suppose you must have a 1 Hz frequency resolution and at the same time a 5 kHz maximum frequency. Go into Table 3-1 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 in the 8K machine is 1,024, and in the 16K machine is 4,096. So an N of 10,000 is impossible. Something has to give. Suppose you agree to settle for a lower maximum frequency. Assuming you have an 8K machine, substitute for N a block size of 1,024. This will give the largest possible F_{max}.

$$F_{max} = 1,024/2 \times 1$$

$$F_{max} = 512 \text{ kHz}$$

If you want a 1 Hz resolution in an 8K machine, you must settle for an F_{max} of 512 Hz.

This is the kind of manipulation of ADC parameters which the user 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.

Aliasing

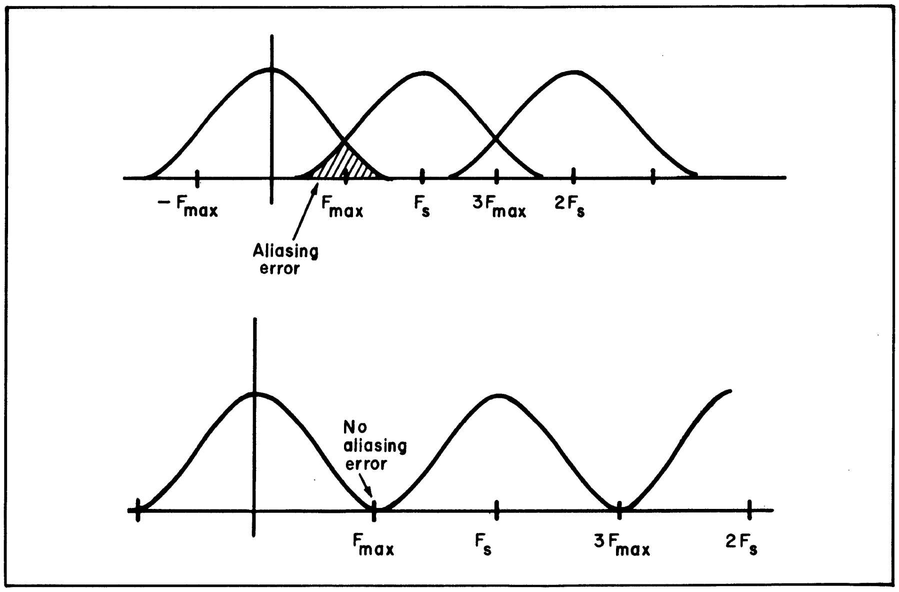
Aliasing is a phenomenon that develops with analog inputs, and which must be kept in mind to avoid possible erroneous results. It comes about from the fact that when an analog input is sampled, the spectrum replicates around multiples of the sample frequency F_S, as shown on page 3-9. Now since F_{max} 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}.

Aliasing

Table 3-1. Selecting values for data sampling parameters

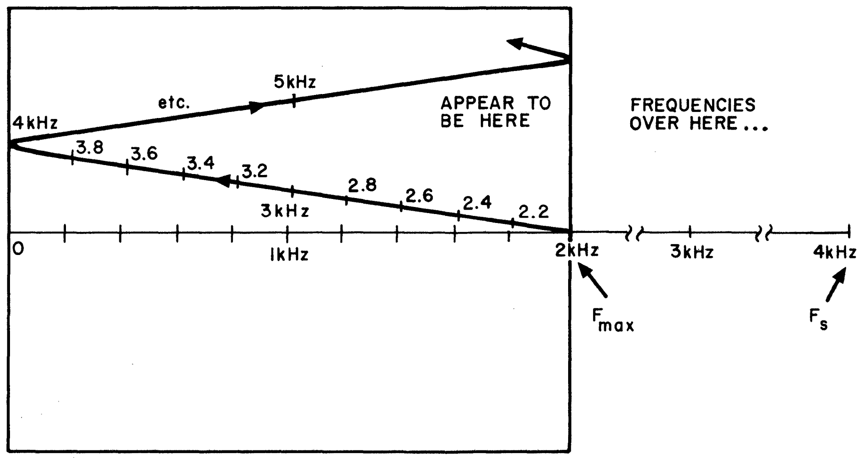
Choose convenient round number for parameter shown.	Chosen parameter automatically fixes the value of parameter below, because of relationship in parentheses.	Then make either of the remaining two parameters (can't be both) as close as possible to the desired value by choosing N* in the relationships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



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, the user has to make sure that the F_{max} he sets is higher than the highest frequency in the data.

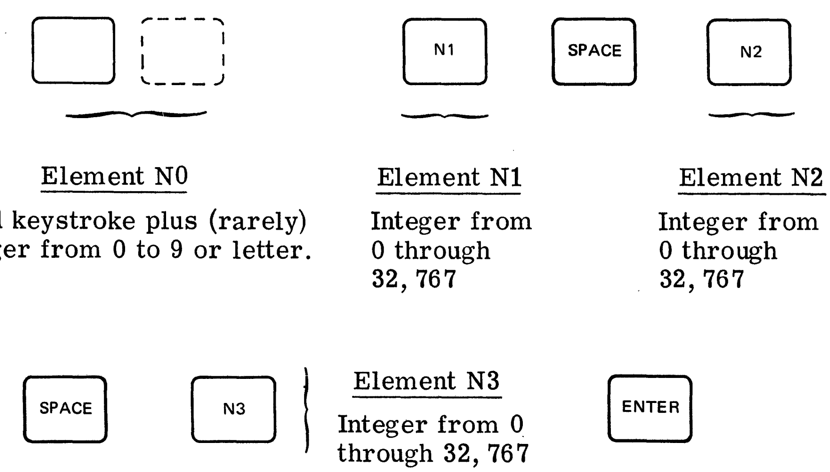


Keyboard command

THE KEYBOARD COMMAND

General Form

All operations on the Fourier Analyzer are initiated from the Keyboard. The general form of the keyboard command is:



Suppose you want to print out, on the Teleprinter, the contents of channels 22 to 31 of data block 2. First turn to the Index and look up the PRINT command. It is on page 6-1, as follows:

[PRINT] [Block N1] [SPACE] [Channel N2]

[SPACE] [Channel N3] [ENTER]

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 this case, the command to be given the machine is:

[PRINT] [2] (N1) [SPACE] [2] [2] (N2) [SPACE]

[3] [1] (N3) [ENTER]

Meaning of Default

Underneath the PRINT command is a table of default values:

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.		

This table tells what the machine assumes if certain elements in the command are omitted. For example, if element N3 is omitted, so that the command is:

[PRINT] [N1] (N1) [SPACE] [2] [2] (N2) [ENTER]

Then only one channel will be printed out, namely, channel N2, because the default value of N3 is N2 - in other words, the stopping and starting channel is the same.

Suppose both N2 and N3 are defaulted, so that the command punched into the machine is:

[PRINT] [2] [ENTER]

Keyboard command

Now, according to the table, the whole of data block 2 will be printed out.

Finally, suppose N1, N2 and N3 are defaulted, so that the command given to the machine is:

[PRINT] [ENTER]

Now the whole of data block 0 will be printed out, because the default value of N1 is block 0.

Thus, a considerable saving in key strokes is possible using the default system. In the above example, to print out all of block 0 requires only two keystrokes, whereas without the default system, it would require seven.

In all cases, the default of elements N1, N2 and N3 causes the operation to be performed on all of data block 0.

No command is executed or entered until the ENTER key is pressed. If, during the entry of a command, a mistake is made, it can be erased by pressing the RUB OUT key. But once the ENTER key has been pressed, the RUB OUT key has no effect: the command has to be re-entered from the beginning.

Operations Between Two Data Blocks

If a command defines an operation between two data blocks (for example, the STORE and LOAD commands, which shift data between blocks), one of these blocks must be block 0, and therefore it is not named in the command. If the operation is a computation, then the result will reside in block 0, and the data in the named block will be left unchanged. For example, the command to multiply block 0 by block 1 is:

[MULT] [1] [ENTER]

The result will be in block 0, block 1 will be left unchanged.

A DEMONSTRATION OF SOME BASIC KEYBOARD FUNCTIONS

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 the coordinate systems available on the Fourier Analyzer. Finally, we will go through an example of a simple Keyboard program.

Instrument Preparation Required

Entering a pulse

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 Section II.

Entering an Analog Pulse

1. Set ADC controls

In order to eliminate the need for an external signal generator, we will use the internally-generated CHECK pulse as an analog input. This pulse has an amplitude of about 51 mV, a length of approximately 1100 μsec, and a repetition rate of the line frequency.

We will begin by taking 11 samples of each pulse. That means the sample interval (Δt) must be 1100 μsec/11 = 100 μsec. Therefore, on the ADC set:

SAMPLE MODE to left side, kHz - μs

MULTIPLIER to black 100

To enable the CHECK signal to be entered into the ADC, set:

INPUT A attenuator to CHECK

Next set:

DISPLAY/INPUT to AA

This tells the Analyzer that this will be a single channel input. The other two positions of the switch are for dual channel input.

Next set:

TRIGGER SOURCE to FREE RUN

This will cause the Computer to take in blocks of data as fast as it can, that is, without waiting for any external trigger signal. Set:

REPEAT switch (on Keyboard) to REPEAT

This makes the Computer display successive blocks of data as they are sampled by the ADC, and permits us to observe the effects of various switch positions on the pulse. In the SINGLE position, the Computer takes in one block only and displays it. The SINGLE position must be used when it is desired to proceed with further Keyboard commands: in the REPEAT position, the ADC continually records and displays analog data, and no other keys except ANALOG IN, RESTART, and STOP have any effect.

2. Set Display Unit controls

Now set the following controls on the Display Unit:

Block size

POSITION to situate x axis on center horizontal line of scope

GAIN to CAL

MODE to REAL/MAGNITUDE

SCALE to straight-up

SWEEP LENGTH to 10

ORIGIN to LEFT

MARKER to OFF

FUNCTION to DISPLAY

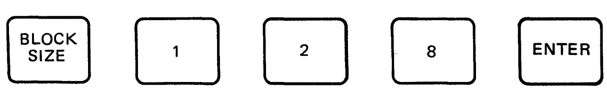
CALIBRATE to ORIGIN

TYPE to BAR

3. Set block size

Next we choose a block size, let us say, arbitrarily, 128 points. Setting block size is a Keyboard command, and to find out what the correct command is, we look up "block size" in the Index. The block size command is given on page 7-14. This procedure of looking up commands will become less and less important as you learn the instrument, but here, in this first demonstration, we will go through the exercise several times. As a further help in locating information, the remaining sections of this book are organized in logical order: Input Modes, Processing Operations, Output Modes, Programming and Editing keys, Sample Programs, and Writing Assembler Language Programs.

As explained on page 7-14, to obtain a block size of 128, the keys to press are:



Note that the "128" light on the Keyboard is now lit, and there are now 128 points across the scope.

Let us enter the pulse into data block 0. To be sure there is no residual data in the block, we will clear it first (though this is not necessary, since analog input operations automatically erase data

Clear block

previously in a block). Looking up CLEAR in the Index, we find it is explained on page 5-3. This command is more complex than BLOCK SIZE. The general form is seen to be

[CLEAR] [Block N1] [SPACE] [Channel N2]

[SPACE] [Channel N3] [ENTER]

where:

N1 is the data block to be cleared.

N2 is the first channel to be cleared.

N3 is the last channel to be cleared.

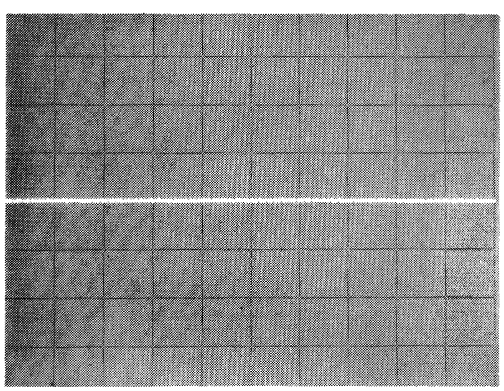
The default table shows that, to clear the whole of block 0, the command should be:

[CLEAR] [ENTER]

Press these keys now. To prove that block 0 is in fact cleared, we use the DISPLAY command. Looking it up in the Index, and checking the default table, we see that block 0 can be displayed by pressing:

[DSPLY] [ENTER]

Display should now look as follows:

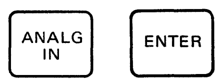


Note: if you make a mistake in entering a command, press the RUB OUT key and re-enter the command. The RUB OUT key, however, can only be used before ENTER is pressed. Otherwise, the whole command must be re-entered. If after entering a command, a WHAT? signal appears on the Keyboard (meaning that the command was illegal), press RESTART, then re-enter the command. (A list of WHAT? signals as typed on the Teleprinter, with their meanings and probable causes is given in the Appendix.)

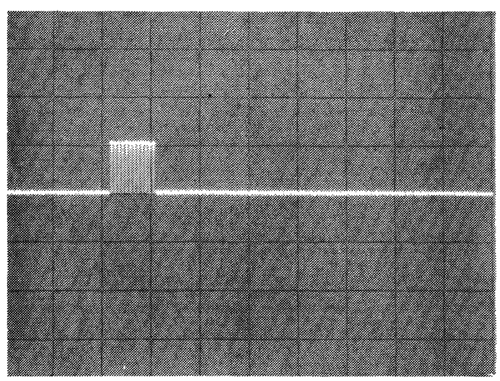
4. <u>Give analog-in command</u>

Analog input

We are now ready to give the analog-in command, which, according to the Index, is on page 4-1. Since the CHECK pulse is going into block 0, and that is also the block we want to view (as opposed to some other block where action might be going on), we make use of the default values and press:



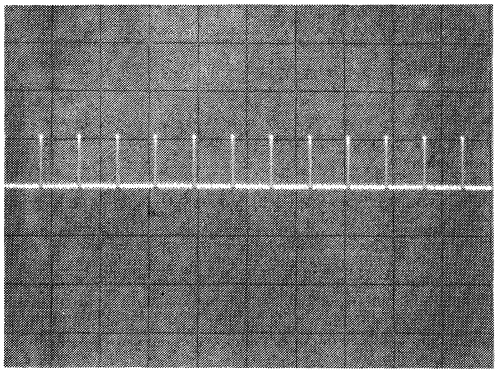
CHECK pulses should now move across the screen somewhat as shown below:



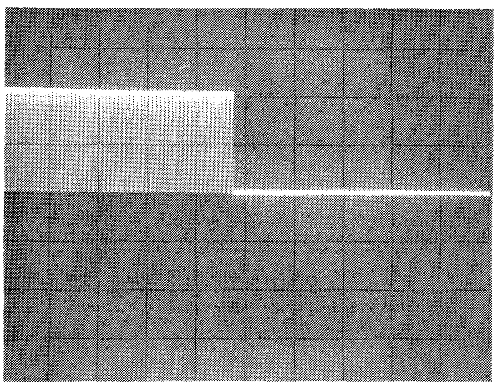
5. <u>Vary sample interval (Δt)</u>

Let us vary the sample interval (Δt) and see how this affects the display. Slowly switch the MULTIPLIER switch on the ADC toward the right, increasing the sample interval (Δt), and note how fewer and fewer samples of each pulse are taken. Turn switch to the left, thus decreasing the interval, and note how more and more samples of each pulse are taken. The two extremes should look approximately as shown on the following page.

Vary Δt



Large sample interval, e.g., 1 – 2K μsec



Small sample interval, e.g., 10 – 20 μsec

Reset:

MULTIPLIER to 100

Relationship of data sampling parameters

Note that the SAMPLE MODE switch setting says kHz-μs. This is because there is a reciprocal relationship between sample interval (Δt) and the maximum frequency (F_{max}) which the ADC can detect without aliasing in the input signal. For example, at the current setting of 100 μs, the maximum frequency detectable is 5 kHz (as can be seen from the blue numbers). There is also a reciprocal relationship between the total record length (T) and the frequency resolution (Δf). These relationships were explained earlier in this section, beginning page 3-4. Some of the information is repeated with the analog-in command in Section IV.

6. Vary trigger controls

The two left-hand positions of the SAMPLE MODE switch (as well as the two right-hand positions) divide or multiply the MULTIPLIER setting by 1000.

Set:

SAMPLE MODE to Hz-ms

Triggering

and note that, when the MULTIPLIER switch is set to its left-most position, the effect of very large sample intervals (i. e., 10 msec for a 1 msec pulse length) can be seen on the display. Also, if you look carefully at the TRIGGERING light, you will see it flicker quite slowly: this is because it takes a long time (128 times 10 msec, or 1.28 seconds) for each data block to be collected in memory. Now reset:

SAMPLE MODE to kHz-μs

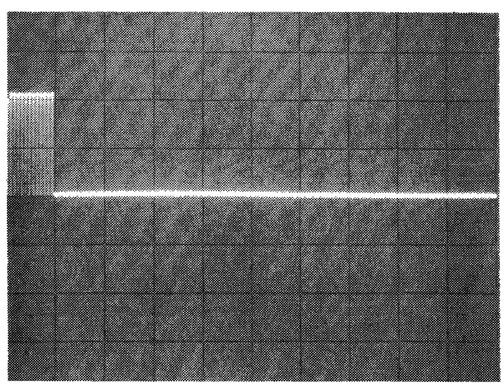
MULTIPLIER to black 100

Now set:

TRIGGER SOURCE to INTERNAL (A) (the "A" means that this mode of triggering can only be done on input channel A)

SLOPE to POS

and adjust the TRIGGER LEVEL until the pulse appears on the left of the scope display. Display should appear as follows:



Here the Analyzer is triggering on the positive slope of the pulse. Set:

SLOPE to NEG

and note that pulse disappears, since now the Analyzer is triggering on the negative slope, beyond which there is no pulse. Set:

SLOPE to POS

Again turn the MULTIPLIER switch right and left several positions to observe how changing the sample interval changes the pulse display. Finally, set:

TRIGGER SOURCE to LINE

Now the Analyzer is triggering at the frequency of the line.

7. Hold block of analog data

Scale factor display

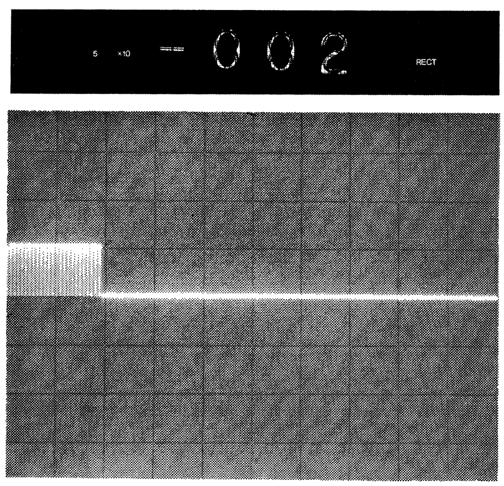
Next, let us hold one block of data in memory. To do this, we must first set the REPEAT switch to SINGLE. Note that the scale factor display is blank when the switch is in the REPEAT position. Now set:

REPEAT switch to SINGLE

The scale factor display now comes on and the block of data in memory is displayed.

Understanding the 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 x 10 -002, meaning 5 x 10^{-2}, so that each vertical division is 0.05, or 50 millivolts.



Thus the CHECK pulse would be on the first division. This scale factor is displayed automatically at all times except for special cases such as REPEAT 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.

Review

So far we have taken an analog pulse into the ADC, seen how some of the sampling controls work, looked at triggering and 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

Keyboard input

Now let us look at another means of entering data, namely, the manual Keyboard method. First, then, we look up the command in the Index (it is found under either "Manual Data Input..." or under "Keyboard Data Input"). The command is given on page 4-18 and is a long one. We will not duplicate any of the information, but instead will simply show the keys that need to be pressed.

First, let us clear the data block. Press:

CLEAR ENTER

Since the pulse will be rectangular, we will use the block-fill mode of entry (same value in a succession of channels). Let the pulse be 9 channels long. Therefore, press:

KEY BOARD 0 SPACE 0

SPACE 8 ENTER

Note that the BUSY light comes on, meaning that the machine is waiting for further Keyboard commands.

8. <u>Enter scale factors</u>

Let the block multiplier be 10^{-4}; since this is a rectangular pulse in the time domain, the coordinate code, per the information on the scale factor command, page 4-20, is 0. The frequency code doesn't matter, so we will arbitrarily use 0. Therefore, the keys to press are as follows (use the minus key in the numerical group, not the Arithmetic key):

KEY BOARD – 4 SPACE

0 SPACE 0 ENTER

Keyboard input

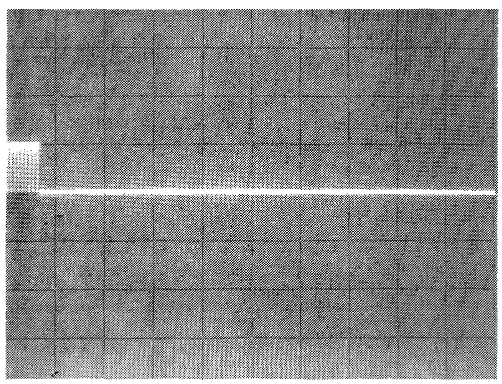
9. <u>Enter data</u>

Since we are in the time domain, we are entering real values. The block multiplier put in was 10^{-4}. Let us aim for a 1 volt amplitude on the pulse. Therefore, we press:

[1] [0] [0]

[0] [0] [ENTER]

The pulse should now appear on the screen as shown below. It is not necessary to press TERM ENTER as this was a block fill. Note that the READY light is now on.



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]

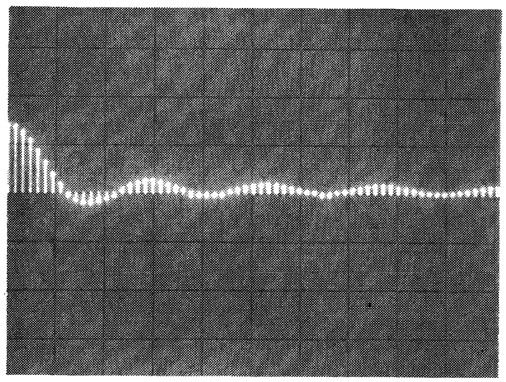
10. Do Fourier transform

We are now ready to do the Fourier transform. This is found in the Index, but by now we should be able to guess intelligently as to what the command should be for a transform in block 0. Namely:

[F] [ENTER]

Fourier trans.

The Fourier transform consists of two spectral series: a real or cosine series and an imaginary or sine series. The MODE switch on the Display Unit should already be set to REAL/MAGNITUDE and the display should appear as follows:



This is the familiar curve, sin x/x.

Determining horizontal scale

The vertical scale we know how to read, from the discussion on page 3-19. The pulse has no frequency, since it was manually entered, but had it been taken in via the ADC, it could be read 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). A more accurate scale is determined by simple calculation. If the SAMPLE MODE switch is in one of the two left-hand positions (as it is in our case), 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 the block size. Δf is the frequency resolution, and the horizontal scale then reads dc on the vertical axis, next point to the right is $1\Delta f$, next $2\Delta f$, etc. In our case, therefore, the calculation is:

$$\frac{5000}{128/2} = 78.12 \text{ Hz}$$

There is 78.12 Hz between each point on the scope, 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.

Freq scale

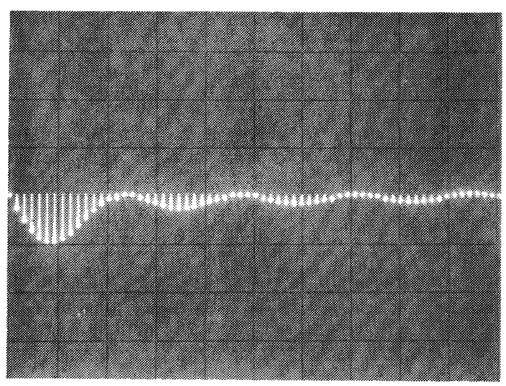
The full explanation of how these various data sampling parameters are related, was given earlier in this section, beginning page 3-4. It is also repeated with the analog-input command information in Section IV. In many cases, of course, the user will be interested in setting frequency parameters, and then will have to use other relationships to determine the values in the time domain.

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 should appear as follows:



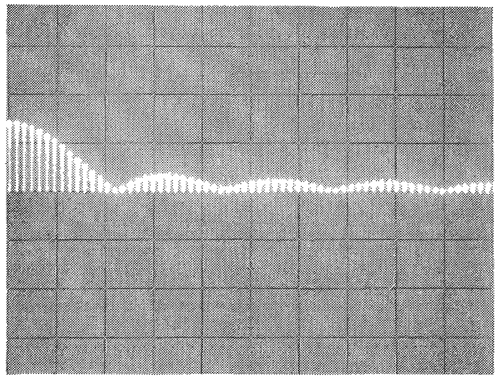
Polar magnitude and phase display

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 should now appear as follows:

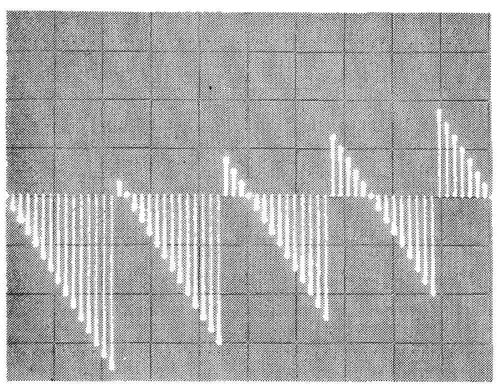


Polar coord's

This is the magnitude display of the spectrum. Set:

MODE switch to IMAGINARY/PHASE

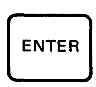
Now the scope shows the phase display, with the scale reading 0 to +180° on the top half of the screen, and to 0 to 180° on the bottom half.



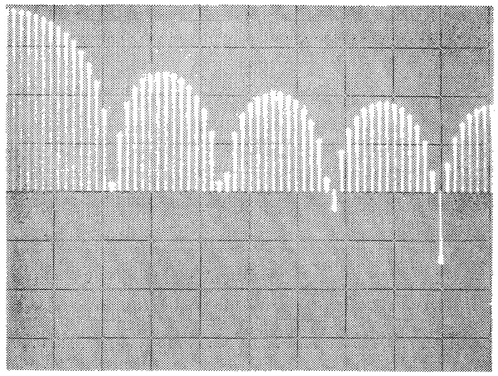
Frequency scale, of course, is as before. Note that the scale factor display is blank. For phase displays, the scale factor is given (and set) on the POLAR ANG/DIV switch. Vary the phase scale by turning the switch left and right. Set:

MODE switch to REAL/MAGNITUDE

and press:



This converts the vertical scale to logarithmic, giving a display as follows:



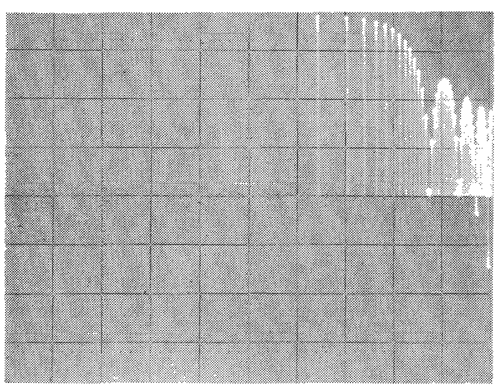
Log scale

At this point, set the TYPE switch on the Display Unit to the POINT and then to the CONT positions and observe the effect on the display. Reset switch to BAR.

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

ORIGIN switch (on Display Unit) to LOG

with the resulting display:



Set:

ORIGIN back to LEFT

and press the following keys to return the display to linear coordinates:

[EXP MAG] [ENTER]

Now again we are looking at the polar magnitude display. Next, to return to rectangular coordinates, press:

RECT ENTER

Markers are available for every 8th or 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 scope intensity may be necessary to make the markers stand out.

Programming

Review

We have now seen how the typical Keyboard command works, exercised some of the Display Unit controls, and in the process observed the different kinds of coordinate systems available in the Fourier Analyzer. Next, and finally for this learning section, we will go through a simple program.

KEYBOARD PROGRAMMING

What is a program in the Fourier Analyzer?

A program is a sequence of commands which the Analyzer will perform automatically. There are locations for 100 command elements in the program memory (200 with 16K memory units). A command element is a word key, or a number, as was described under the discussion of the general form of the Keyboard command, page 3-10. When you begin a program, these memory locations may be empty, but most likely 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--that is, a command involving one of the EDITING keys on the Keyboard.

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, unless this material has been protected by a PROTect command. 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 TERM command is used to exit the programming mode.

To check that the steps have been entered correctly, the LIST command is used, producing a listing on the Teleprinter. In the listing, line numbers are automatically assigned to each command, these line numbers being 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 one of the EDITING keys, in this case, REPLACE, DELETE, or INSERT, referencing the line numbers to be changed or corrected. After the changes have been made, the TERM 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 label, the JUMP command is used, in which the number of the label is named. 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.

Averaging program

A Program Example

As an introduction to programming the Fourier Analyzer, we will do a simple 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 a number of inputs to arrive at an average value of the signal. This, of course, will be a positive value since the CHECK signal is always positive. In brief, the 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.

Add block 0 to block 1. 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) steps from analog input the number of times desired to achieve the final sum.

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

End program, terminate program mode.

Procedure

The CHECK signal has an amplitude of about 51 mV, and a length of 1100 μsec. Since its repetition rate is the power line frequency, the period is about 17 msec. We will be sure to get at least 1 pulse in every input if we choose a sample interval (Δt) of 100 μsec, and a block size of 256. This then gives a total record length of 0. 1 x

256 or 25.6 msec, which is more than the 17 msec period. Therefore set:

SAMPLE MODE switch to left side, kHz - μs

MULTIPLIER switch to black 100

TRIGGER SOURCE switch to FREE RUN

Next, set the block size by pressing:

Initial program steps

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

Set:

Teleprinter switch to LINE

to provide a printout of each Keyboard command. A complete list of the Teleprinter symbols, in alphabetical order, is given in the Appendix.

Now, to begin programming, use the REPLACE command. This, like all commands, can be looked up in the Index. Note that in the REPLACE command, no default is allowed for the N1 element. The command is:

[RPLAC] [0] [ENTER]

Next, we label the starting point of the program:

[LABEL] [1] [ENTER]

Then we clear data block 1, for the following reasons: several steps later, data in block 0 will be added to block 1, an operation which could cause erroneous results the first time around if residual data were in block 1.

For this reason, it is wise to clear block 1 at this point. Press:

[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. Press:

LABEL | 2 | ENTER

Now the analog input is entered. We will take the data into block 0, and display that block. Utilizing default values, the command is:

ANALG IN | ENTER

Summation

Since this program is intended to produce an average, we have to sum up the individual analog inputs. The data comes in block 0: if we add it to block 1, then the sum will reside in block 0 (per the discussion of the general command form, page 3-12, and the definition of the "+" command itself, page 5-6.) Then we will shift the summation into block 1. In short, the sums will accumulate in block 1. Therefore, the next two commands are:

+ | 1 | ENTER

and

STORE | 1 | ENTER

To obtain the average, we will sum 250 inputs 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

"2" being the label. Then press, for the division:

÷ | 1 | SPACE | 2 | 5 | 0 | ENTER

This is the end of the program. Press:

END ENTER

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

Listing

TERM ENTER

The BUSY light is now off and the READY light is on, indicating we are out of the programming mode.

CORRECT LISTING ON TELEPRINTER	MEANING
Ø L1	Label 1
1 CL 1	Clear block 1
3 L2	Label 2
4 RA	Analog input to block 0, display block 0
5 A+ 1	Add block 0 to block 1
7 X> 1	Store contents of block 0 in block 1
9 #2 25Ø Ø	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.
12 : 1 25Ø	Divide block 1 by 250
15 .	End of Program

Corrections and Changes

Program changes

To illustrate how corrections and changes are done, let us modify the analog input command. As it now stands, it displays block 0, which is the block into which each new input is read. It will be more interesting to view block 1, which is the block where the sums of the inputs accumulate. To change this line, we can use either the two commands DELETE and INSERT, or the single command REPLACE. Let us use the latter. Note that all of these commands are in the EDITING group, thus fulfilling the requirement that all programming must begin with an EDITING command.

In the REPLACE command, the line to be replaced is named. The analog input command is line 4, therefore press:

[RPLAC] [4] [ENTER]

The new command is to enter data into block 0 and display block 1. Therefore, press:

[ANALG IN] [0] [SPACE] [1] [ENTER]

Now, to exit the programming mode, press:

[TERM] [ENTER]

Again, to check the program, press:

[LIST] [ENTER]

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

```
 Ø L1
 1 CL      1
 3 L2
 4 RA      Ø       1
 7 A+      1
 9 X>      1
11 #2    25Ø       Ø
14 :       1     25Ø
17 .
```

Note that the line numbers change automatically below the line that was changed, to accommodate the new elements.

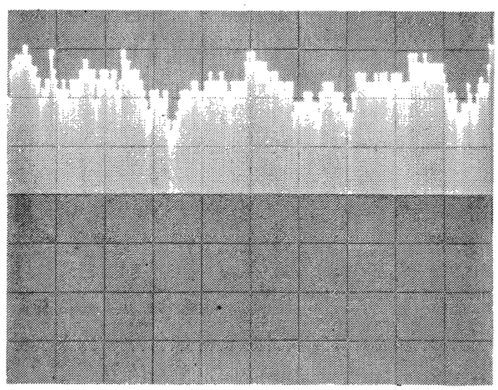
Running the program

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. We will do both. First, the JUMP command. Here, a label, rather than a line number, is used. Press:

Running program

JUMP | 1 | ENTER

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



Note that the average is not 51 millivolts, the height of the pulse, but something less. This is because the pulse duty cycle is less than 100%.

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

POINT | 3 | ENTER

Then press:

CONTINUE

The program should again run through.

If the user wishes the program to be protected, so that another user does not accidentally erase or change it in the course of entering his own program, then he uses the PROTect command. In this case, since the program starts at line 0, press:

PROT ENTER

Protecting program

Now, none of the contents of the program file may be changed or deleted. To unprotect the memory, press:

PROT { 99 / 199 } ENTER

Now only those lines beyond 99 (i.e., no lines, in the case of the 8K memory) will be protected. For the 16K memory unit, 199 should be used instead of 99.

In Section VIII, "Sample Programs", some of the more common Fourier Analyzer programs are given for the user's convenience.

SECTION IV

INPUT MODES

ABOUT THIS SECTION

This section provides operating instructions for the following input modes: analog input, punched tape (Photoreader), manual (Keyboard), and binary. All four are part of the standard 5450A Fourier Analyzer. Analog and manual are part of the standard 5452A; Photoreader is available as an option. Binary input/output is not available for the 5452A.

ANALOG INPUT

Single channel input

The analog input command takes in data from one or both channels of the Analog-to-Digital Converter (ADC). Therefore, there are two types of analog input command: single channel and dual channel. Both commands are unique in that they permit the user to take in data to one data block while simultaneously displaying another. This has its primary use in programs, where it allows the user to obtain an automatic display as the program proceeds. Before using the analog input command, be sure you understand the use of the REPEAT-SINGLE switch, the DISPLAY-INPUT switch, and the TRIGGERING controls, as explained below.

Single Channel Input

Before giving the single channel command, the DISPLAY-INPUT switch on the ADC has to be in the channel A position. In this case, channel A will be inputted. The command is as follows:

[ANALG IN] [Block N1] [SPACE] [Block N2] [ENTER]

where:

N1 is the data block into which input A is to be read.

N2 is the data block to be displayed.

Default table is given below:

Element	Meaning of Element	Default Value of Element
N1	Data block into which data is to be read	data block 0
N2	Data block to be displayed (in SINGLE mode)	N1

Dual Channel Input

The dual channel input command takes in data simultaneously from channels A and B. Before giving the command, the DISPLAY-INPUT switch must be on one of the DUAL positions. The meaning of these two positions is explained below, under "DISPLAY-INPUT switch". The command is:

[ANALG IN] [Block N1] [SPACE] [Block N2]

[SPACE] [Block N3] [ENTER]

REPEAT-SINGLE sw.

where:

N1 is the data block into which input A is to be read.
N2 is the data block into which input B is to be read.
N3 is the block to be displayed.

No defaults are allowed in this command.

REPEAT-SINGLE Switch

The purpose of this switch is to enable the user to continuously observe the analog input. This is done by setting the switch to the REPEAT position. Whatever channel has been set on the DISPLAY-INPUT 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.

WARNING: If either the SAMPLE MODE or MULTIPLIER switches are changed during a read cycle, as is possible in the REPEAT mode, a lock-up may occur. This will be indicated by the display going blank, and the failure of the Analyzer to execute further commands. To remedy this, press the RESTART key and re-enter the analog input command.

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 switch is set to SINGLE, the program will continue. If the switch is in the SINGLE position when the program reaches that step, it will simply display the single input and then proceed.

DISPLAY-INPUT Switch

This switch determines which of the two analog input channels, A or B, will be displayed following the analog input command. Below are the explanations of the three positions:

AA: sets ADC for single channel operation through channel A. In the REPEAT mode, channel A is automatically displayed, and the display part of the analog input command is overridden.

A DUAL: sets ADC for dual channel operation with channel A displayed automatically in REPEAT mode. Again, in the REPEAT mode, the display part of the analog input command is overridden.

B DUAL: sets ADC for dual channel operation with channel B displayed automatically in REPEAT mode. In REPEAT mode, display part of analog input command is overridden.

TRIGGERING Controls

When the analog input command is executed, the trigger circuit is armed. Triggering then occurs in accordance with the TRIGGER SOURCE position, as described below:

Free-run: The arming function is the trigger source; a sample record is immediately read into the Analyzer memory at the rate set by the SAMPLE MODE and MULTIPLIER switches. As soon as the block is filled, the trigger starts a new sample record in.

Internal: The ADC triggers on the signal applied to channel A, as long as the signal has one division peak-to-peak amplitude on the display. The voltage level at which triggering occurs is set by the TRIGGER LEVEL control. + = positive amplitude; - = negative amplitude. The PRESET position sets the level at about 0. The SLOPE control sets the slope at which triggering will occur (positive = increasing side, negative = decreasing side).

External: Same as Internal, except that the user brings in the trigger through the EXT TRIGGER input. 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 mode.

Line: ADC triggers synchronously with the 50 or 60 Hz line. No slope or polarity setting is available for this position.

Input Attenuators

These are the two switches on the ADC marked A and B, one for each input channel. The OVERLOAD LIGHT is between them. The numbers represent the maximum plus or minus voltage which the ADC can accept without overload. When overload occurs, the OVERLOAD light comes on, and the attenuator should be set to the next highest position.

The CHECK position inputs an internally-generated rectangular pulse to the ADC; the pulse has a nominal 51 mV amplitude, a 1.1 msec duration, and the frequency of the line (50 or 60 Hz). It is used to obtain an analog input when no external sources are available.

A OVERLOAD B

The damage level for any particular range is approximately 46 dB (200 times peak level).

The Analyzer takes into account the OVERLOAD VOLTAGE switch setting whenever the Analyzer receives data from the ADC. Thus, all further data operations are on a calibrated basis. 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.

How to Set Sampling Rates

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 the frequency resolution (Δf), and the maximum frequency, (F_{max}). The user should have an understanding of how these parameters are related by reading the discussion in Section III beginning page 3-4. 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:

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

$$T = N\Delta t$$

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:

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

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

The following table, a duplicate of the one on page 3-9, sums up the above relationships, and makes the choosing of desired parameters a little easier.

ADC sampling rates

Table 4-1. Selecting values for data sampling parameters

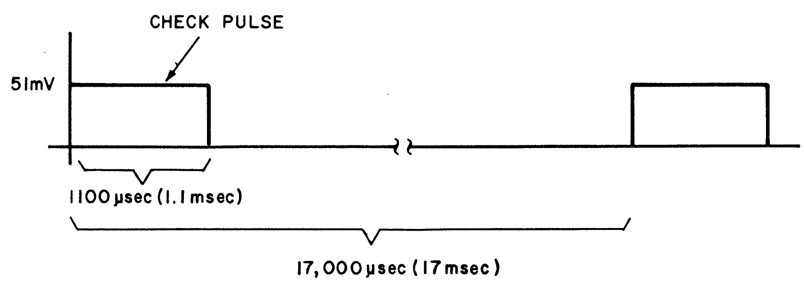
Choose convenient round number for parameter shown.	Chosen parameter automatically fixes the value of parameter below, because of relationship in parentheses.	Then make either of the remaining two parameters (can't be both) as close as possible to the desired value by choosing N* in the relationships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

*N, the data block size, is always a power of 2.

Example of Inputting Analog Data

Single input. We will use the CHECK signal on INPUT A as the analog input signal. This signal is a rectangular pulse of about 51 mV amplitude and 1100 μsec (1.1 msec) 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 μsec (17 msec) for a 60 Hz line. Thus, the pulse is present in only 1/17th of the time T.

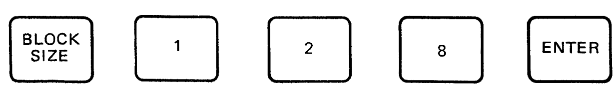
Analog input example



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. 11 samples would be adequate: the pulse is 1100-μsec long. 1100/11 gives us a Δt of 100 sec. Therefore, we set

SAMPLE MODE to left side, kHz-μs
MULTIPLIER to black 100

None of the remaining sample parameters F_{max}, Δf, or T are critical in this case, so we can choose any data block size, N, we wish. Let it be 128. Press



As far as triggering is concerned, we can let the triggering frequency be the same as the frequency of the pulse, so set

TRIGGER SOURCE to LINE

The remainder of the triggering adjustments are not important in this case. Next, to give us an input, we set

A OVERLOAD VOLTAGE to CHECK

And, because this is only a single channel input from INPUT A, set

DISPLAY/INPUT to AA

As a start, we will only look at one sweep of the input, so set:

Keyboard switch to SINGLE

Finally, check that the display controls are set as follows:

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

Since we have a single channel input, the form of the analog input command (from page 4-1) must be:

[ANALG IN] [Data block into which input A is to be read] [SPACE] [Data block to be displayed] [ENTER]

Setting Δf

Let us read the CHECK pulse into block 0, and of course display that same block. Therefore we press:

[ANALG IN] [ENTER]

If we wanted to continuously read in and display the pulse, we would simply set

Keyboard switch to REPEAT

and give the analog-in command. The slight flicker in the display is the signal coming in at the triggering frequency (LINE).

Examples of Setting the Sample Controls

Setting a desired frequency resolution (Δf). A signal with a spectrum that is known to contain no energy above a frequency of about 5 kHz, is to be measured with a resolution of 10 Hz per channel. Looking at Table 4-1, we can see we are on the third line, because we want to go in with Δf as a convenient round number.

Set Δf = 10 Hz by setting:

SAMPLE MODE switch to right side, white Hz

then set:

MULTIPLIER SWITCH to white 10

Table 4-1. 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 $(T = N\Delta t)$ Δf $\left(\Delta f = \frac{1}{N\Delta t}\right)$
2. F_{max}	Δt $\left(\Delta t = \frac{1}{2F_{max}}\right)$	T $(T = N\Delta t)$ Δf $\left(\Delta f = \frac{1}{N\Delta t}\right)$
3. Δf	T $\left(T = \frac{1}{\Delta f}\right)$	Δt $\left(\Delta t = \frac{T}{N}\right)$ F_{max} $\left(F_{max} = \frac{N}{2} \cdot \Delta f\right)$
4. T	Δf $\left(\Delta f = \frac{1}{T}\right)$	Δt $\left(\Delta t = \frac{T}{N}\right)$ F_{max} $\left(F_{max} = \frac{N}{2} \cdot \Delta f\right)$
*N, the data block size, is always a power of 2.		

Setting Δf

The sample record length T (TOTAL TIME) has now been fixed at 100 msec, as can be read from the above (black) switch settings. As we would like F_{max} to be about 5 kHz, we use the relationship $F_{max} = (N/2) \cdot \Delta f$ (from third line of Table 4-1) and see that if N = 1024, F_{max} will have closed to the desired value.

$$F_{max} = 1024/2 \cdot 10 = 5.12 \text{ kHz}$$

So, set the block size to 1024.

From the remaining equation (third line, Table 4-1), Δt can be computed as follows:

$$\Delta t = \frac{T}{N} = \frac{100 \text{ msec}}{1024} = 97.6\ \mu\text{sec}$$

Setting a desired maximum frequency (F_{max}). The signal in the previous example will have a display calibration of 512 Hz/Div (F_{max}/ 10 cm = display calibration). This may not be the most desirable number to work with. From Table 4-1, second line, now, we see that we can trade off on the value of Δf, in return for a convenient round number for F_{max}, or in other words a convenient round number for display calibration. This is done as follows:

Set F_{max} = 5 kHz by setting:

SAMPLE MODE switch to left side, blue kHz.

then set

MULTIPLIER switch to blue 5.

Setting F_{max}

Δt is now fixed at 100 μsec as can be read from the black switch settings. We are now on the second line of Table because we have chosen F_{max} to be a convenient value (i.e., 5 kHz). Therefore we experiment with values of N in the equation $\Delta f = 1/(N \cdot \Delta t) = F_{max}/(N/2)$ until we get as near to the desired value of 10 Hz for Δf as possible. It turns out this occurs when N remains at its value of 1024.

$$\Delta f = \frac{5 \text{ kHz}}{1024/2} = 9.6 \text{ Hz}$$

Setting sample rate (F_s) via external clock: Example 1. It is desired to remotely control the sample rate so that it will be between 1 and 2 kHz. A programmed oscillator will be used as the external clock unit; frequencies available are 100 kHz and 200 kHz. What should be the MULTIPLIER switch setting?

The relationship between the external clock frequency (F_c) and the sample rate (F_s) is given under the explanation of the EXT CLOCK input, Figure 4-1, and is:

$$F_s = \frac{(F_M)(F_c)}{5000}$$

where:

F_s = sampling frequency

F_c = external clock frequency

F_M = MULTIPLIER switch setting (blue number)

$$2 \times 10^3 = \frac{(F_M)(2 \times 10^5)}{5000} \qquad 1 \times 10^3 = \frac{F_M(1 \times 10^5)}{5000}$$

$$F_M = 50 \qquad F_M = 50$$

So the external clock signal should be brought in through the EXT CLOCK input, the SAMPLE MODE switch should be set to EXT CLOCK, and the MULTIPLIER switch set to blue 50.

Example 2. A sample rate of 100 Hz is desired using a clock that is recorded on an analog magnetic tape to remove the effects of tape flutter.

The clock frequency is 200 kHz. What should the MULTIPLIER be set to?

$$100 = \frac{F_M \times (2 \times 10^5)}{5000}$$

$$F_M = 2.5$$

Bring in external clock through EXT CLOCK input, set SAMPLE MODE to EXT CLOCK, set MULTIPLIER to blue 2.5.

Histogram

Histograms

The histogram input can only be made through INPUT A, not INPUT B. The form of the command is:

[HISTO GRAM] [Block N1] [ENTER]

where:

N1 is the data block into which the histogram is to be entered. Default value = block 0.

Since the histogram command does not automatically clear the data block before accumulating a new set of values, the user must be certain to do this himself, unless additional values are to be collected on an old histogram. To clear the block, press:

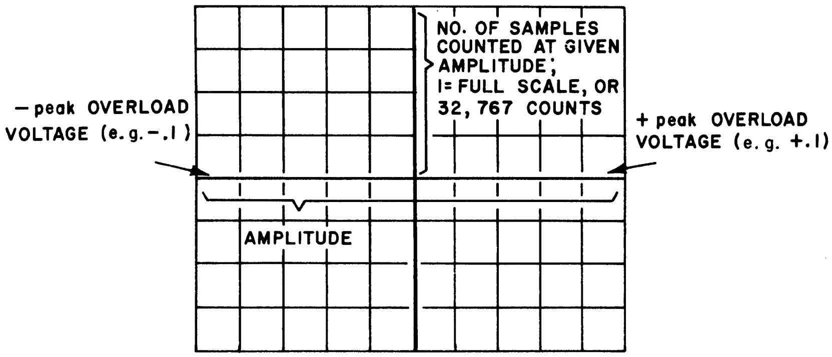
[CLEAR] [Block N1] [ENTER]

where:

N1 is the data block. Default value = block 0.

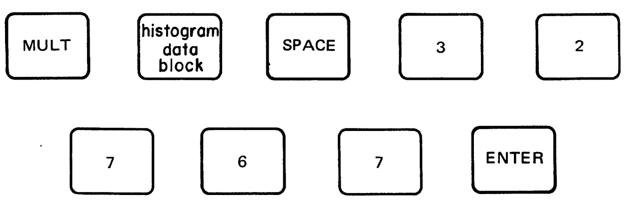
The ADC samples the input at the rate set by the SAMPLE MODE and MULTIPLIER switches, but the sample interval Δt for histograms cannot be less than 200 μsec. If the switches are set for a shorter sample than this, a WHAT? signal will result.

The histogram display is automatically set up as follows:



Histogram

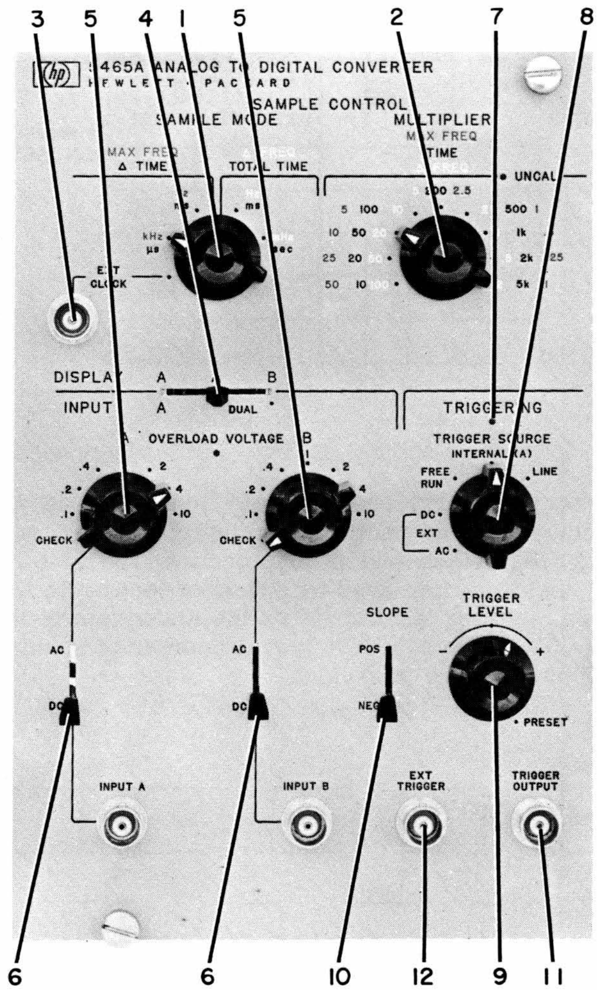
In the Analyzer memory, each voltage amplitude is an address. Counts are accumulated at each address until the number at any one address is 32, 767. When and if that occurs, the histogram stops automatically, and is redisplayed with the correct scale factor. Further counts are ignored. Full scale on the histogram is indicated by the value 1. 0. To convert this to actual number of counts, multiply the data block by 32767 by pressing:



Note

A Time-Interval Histogram accessory is available for the Fourier Analyzer, consisting of an HP 10618A BCD-Binary Converter and an HP 5325B or a 5326A/B Counter, plus interfaces.

Figure 4-1. 5465A Analog-to-Digital Converter (ADC)



ADC controls

1. SAMPLE MODE: in left half of range, selects maximum frequency (F_{max}), called MAX FREQ, and sample interval (Δt), called Δ TIME.

   In right half of 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 4-1, page 4-8, for instructions on choosing parameters.

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

PHOTOREADER (PUNCHED TAPE) INPUT

How Done

Punched tape containing data (as opposed to programs) is entered into the Fourier Analyzer via the High Speed Paper Tape Reader (called "the Photoreader" throughout this manual). If a Fourier Analyzer has no Photoreader, it can accept data input via the Paper Tape Reader in the Teleprinter, provided the tape is prepared as described on page

Input Command

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

Photoreader 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.

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.

Data Format Required

All tapes generated by the PUNCH command on the Fourier Analyzer can be entered in the Photoreader. A tape generated by the PRINT command cannot be entered into the Photoreader because it is punched in a radix 10 rather than a radix 2 number system.

Tapes from other computers can be entered if they have the correct data format, which is spelled out in Figure 4-2.

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 when these points are part of the data.

If in doubt whether a tape is OK, print it out on the Teleprinter and see if it conforms to the correct format.

Photoreader tape loading

Data tape is loaded in the same way as the Fourier Program Tape. That is:

1. a. Place Fourier tape roll in tape holder in Photoreader, feed holes toward the rear.

   b. Press POWER pushbutton.

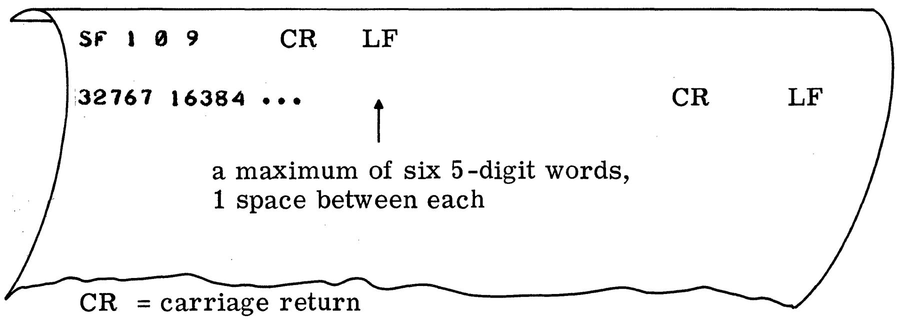
   c. Press LOAD pushbutton on Photoreader.

   d. Run the tape leader underneath the wire guide and through the pair of feed rollers.

   e. Press READ pushbutton.

2. Give Photoreader input command.

Figure 4-2. Required data format for punched tape



CR = carriage return

LF = line feed

Data format

<u>SF</u> stands for "scale factors"

1 is "k" in the expression "2^k". All data words, when they arrive in the Analyzer memory, are multiplied by 2^k, giving the number system called "floating point on a block basis." Thus, in the above example, all data words are multiplied by $2^1 = 2$. (But note on next page what these data words stand for.)

Ø coordinate code from following table. In above example, data is in frequency domain, rectangular coordinates.

Coordinate Code	Time	Frequency	Rectangular	Polar	Log	Linear
0	x		x			x
2	x				x	
4		x	x			x
5		x		x		x
7		x		x	x	

9 any arbitrary number from 0 through 99, OR, a frequency code from the following table. The table expresses data sampling parameters in terms of SAMPLE MODE and MULTIPLIER switch settings on the ADC of the 5450A. 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 99 is inserted in its place.

In the example above, pertinent data sampling parameters were F_{max} of 0.5 Hz, Δt of 1000 msec.

Data format

Figure 4-2. Required data format for punched tape (cont'd)

Freq. Code	F_{max} (MAX FREQ)	Δt (Δ TIME)	Freq. Code	Δf (Δ FREQ)	T (TOTAL TIME)
47	50 kHz	10 μsec	63	100 Hz	10 msec
46	25 kHz	20 μsec	62	50 Hz	20 msec
45	10 kHz	50 μsec	61	20 Hz	50 msec
44	5 kHz	100 μsec	60	10 Hz	100 msec
43	2.5 kHz	200 μsec	59	5 Hz	200 msec
42	1 kHz	500 μsec	58	2 Hz	500 msec
41	.5 kHz	1000 μsec	57	1 Hz	1000 msec
40	.25 kHz	2000 μsec	56	.5 Hz	2000 msec
39	0.1 kHz	5000 μsec	55	.2 Hz	5000 msec
15	50 Hz	10 msec	31	100 mHz	10 sec
14	25 Hz	20 msec	30	50 mHz	20 sec
13	10 Hz	50 msec	29	20 mHz	50 sec
12	5 Hz	100 msec	28	10 mHz	100 sec
11	2.5 Hz	200 msec	27	5 mHz	200 sec
10	1 Hz	500 msec	26	2 mHz	500 sec
→ 9	0.5 Hz	1000 msec	25	1 mHz	1000 sec
8	0.25 Hz	2000 msec	24	.5 mHz	2000 sec
7	0.1 Hz	5000 msec	23	.2 mHz	5000 sec

32767 16384. . Data words. Data word system is as follows: 0 = 0; 32767 = 1. So, in above example, first data word is $32767 \times 2^1 = 32767 \times 2 = 65534 = 2$. Second data word is $16384 \times 2^1 = 16384 \times 2 = 32768 = 1$. No data word can be higher than 32767 or lower than 0.

To convert any set of data into this number system:

1. Choose a k that will make 2^k slightly higher than the highest number in the data.

   For example, if data is 0.5, 1, 2, 3, choose k = 2, because $2^2 = 4$, which is one higher than 3.

   Second example: if data is 0.0005, 0.001, 0.002, 0.003, choose k = -8, because $2^{-8} = 0.0039$.

2. Then convert data into Fourier Analyzer data words per following formula:

   $$\frac{\text{data value}}{2^k} \times 32767 = \text{Fourier Analyzer data word}$$

   For example, if data value = 3, as in previous example, then data word is:

   $$3/4 \times 32767 = 24575$$

   And, in second example, taking data value = 0.002, data word is:

   $$\frac{0.002}{0.0039} \times 32767 = 16803$$

MANUALLY ENTERING DATA FROM THE KEYBOARD

How Done

Data can be manually entered into the Fourier Analyzer directly from the Control Unit Keyboard. This mode of input, and the Teleprinter output, are a direct data interface between man and machine because they use a radix 10 (i.e., decimal) number system. The punched tape mode (Photoreader input and Punch output), on the other hand, is a data interface between the Analyzer and itself or other computers, and uses a radix 2 number system.

Command to Manually Enter Data

Keyboard data entry

The manual data input command is in four parts, as shown in Figure 4-3, which is intended as a quick reference. The following expands on each of the steps.

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 in the block-fill command. (The block-fill command with N3 defaulted of course becomes the point-by-point fill.)

   After either of these commands is given, the BUSY light comes on, meaning that no other commands except the scale factors (step 2) and data (step 3) will be executed, unless the TERM ENTER keys are pressed. In that case, the Analyzer reverts immediately to the READY state.

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 number 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 less than 32767, however. Thus, 1 could not be entered as 100,000 (x 10^{-5}). 4 would have to be entered as 4000 (x 10^{-3}), not as 40,000 (x 10^{-4}).

   N2--the coordinate code, from the table below, which tells the machine the type of data entry you will be making.

Keyboard data entry

Figure 4-3. Manual data entry – a 4-part command

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.

Or

Point-by-Point Fill

[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.
N2 is the coordinate code.
N3 is the frequency code (default value = previous value).

3. **Enter Data**

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

OR

[magnitude word] [SPACE] [phase word (in .01 degree)] [ENTER]

4. **Press**

[TERM] [ENTER] (if point-by-point fill)

Coordinate Codes

N2	Time	Frequency	Rectangular	Polar	Log	Linear
0	x		x			x
4		x	x			x
5		x		x		x

N3--the frequency code. This can be any number you wish from 0 through 32767. The term "frequency code" comes from the fact that, when analog data is punched or 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. (The complete frequency code table is shown on pages 4-24 and 6-7.) If N3 is defaulted, the machine assumes the previous value.

Keyboard data entry

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 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]

Note: a 0 must always be entered for the imaginary or phase dc value and the imaginary or phase value of the highest frequency whenever these two points are part of the data. The reason for this is explained under the discussion of the Fourier Transform used in the Fourier Analyzer, page

4. Press TERM ENTER

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

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.

Examples of Keyboard Data Entry

An example of entering data from the Keyboard is given in Section II, page 3-20.

Entering Data on Paper Tape through Teleprinter

Teleprinter data entry

On Fourier Analyzers with no Photoreader, data on paper tape can be entered via the Tape Reader on the Teleprinter as long as the tape is prepared in the same manner as steps 2, 3 and 4 of the manual Keyboard entry. Step 1 is entered on the Keyboard itself. In step 2, instead of the KEYBOARD key, use the letters SF on the Teleprinter. In all cases, instead of ENTER use RETURN, then LINE FEED. In step 3, there can be a maximum of six 5-digit data words per line, with 1 space between each word. In step 4, instead of TERM use the slash (/).

It should be realized, however, that this method is much slower than the Photoreader method, and also, because the number system is decimal rather than binary, the accuracy will be somewhat less than that obtainable with the binary tape/Photoreader mode.

BINARY INPUT/OUTPUT

The Binary Input/Output mode provides a link between the Fourier Analyzer and another computer or analyzer. It is available as an option and requires using the special Fourier Program Tape provided. Binary Input/Output prevents the use of the continuous plot mode on an external plotter.

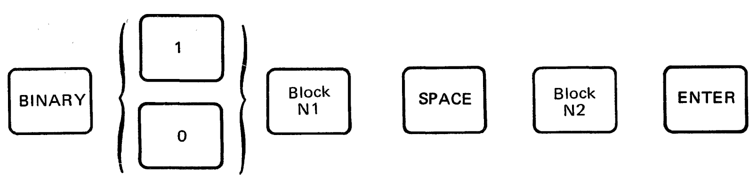
How Done

Binary input/output is done through Microcircuit interface cards (Part No. 12566A) or general purpose registers (Part No. 12554A) in channels 16 and 17 of the Analyzer. (Card or register can be used in either or both channels.) Full instructions on connecting the card or register to the other device and installing it in the Fourier Analyzer are contained in the manual that comes with card or register. Note: with 8K programs, only single channel Binary Input/Output is permitted. Dual channel is allowed with 16K programs.

Binary input/output

Binary Input/Output Command

The command structure is:



where:

BINARY 1 means input to the Fourier Analyzer.

BINARY 0 means output from the Fourier Analyzer.

N1 is the data block whose data is transferred via channel 16 in the Analyzer.

N2 is the data block whose data is transferred via channel 17 in the Analyzer. If N2 is defaulted, there will be no input or output through channel 17.

It is possible to input both blocks, or output both blocks, at the same time, but the simultaneous input of one and output of the other is not permitted.

Each channel may proceed at its own independent word rate, but when using both channels, the Analyzer will remain on BUSY until both channels have completed their transfer. Then the Analyzer goes to READY or proceeds with a program.

Required Format of Data

The sending or receiving of a data block of information must be preceeded by three header words, each 16 bits long. When outputting data from the Fourier Analyzer, a word consisting of all zeroes must preceed the three header words. See Figure 4-4 for flow charts relating to this discussion.

1st header word. The block size as a binary power of 2 in bits 6 through 13, where the bit number is the power. That is, bit 6 on = block size of 2^6 or 64; bit 10 on = block size of 2^{10} or 1024. Bit 15 is a "handshake" bit which establishes whether input or output will be executed. If the Analyzer has been commanded to input data, then the handshake bit is 1; if commanded to output data, then the handshake bit is 0.

Header word

2nd header word. A 16 bit word in which bits 6 through 15 are a power of 2 scale factor (i. e., radix 2), expressing a number which is to multiply each data word. Scale factor = 2^k, where k is the value of the binary number expressed by bits 6 through 15. Thus a scale factor of 2^{10} would be expressed by bits 9 and 7 on, and all others between 6 and 15 off. Bits 0 through 2 spell out the coordinate code below in binary. This number specifies the coordinates of the data.

Coordinate Code	Time	Frequency	Rectangular	Polar	Log	Linear
0	x		x			x
2	x				x	
4		x	x			x
5		x		x		x
7		x		x	x	

3rd header word. A frequency code which can be any arbitrary number between 0 and 32,767. This code was originally intended to describe analog input parameters, as shown in the table, next page. However, the number need not correspond to such parameters.

Data words. The next 2^N words sent or received go into the data block(s) specified and must be in 2's complement format. If the Analyzer is sending data, then, after the last data word, an "illegal number" is sent consisting of a 1 in bit 15 followed by all 0's. This signals the end of the transmission. Likewise, if the Analyzer is receiving data, it expects such an "illegal number" at the end of the remote device's transmission.

Note

The binary input/output interface cards employ negative logic.

Frequency code

Freq. Code	F_{max} (MAX FREQ)	Δt (Δ TIME)	Freq. Code	Δf (Δ FREQ)	T (TOTAL TIME)
47	50 kHz	10 μsec	63	100 Hz	10 msec
46	25 kHz	20 μsec	62	50 Hz	20 msec
45	10 kHz	50 μsec	61	20 Hz	50 msec
44	5 kHz	100 μsec	60	10 Hz	100 msec
43	2.5 kHz	200 μsec	59	5 Hz	200 msec
42	1 kHz	500 μsec	58	2 Hz	500 msec
41	.5 kHz	1000 μsec	57	1 Hz	1000 msec
40	.25 kHz	2000 μsec	56	.5 Hz	2000 msec
39	0.1 kHz	5000 μsec	55	.2 Hz	5000 msec
15	50 Hz	10 msec	31	100 mHz	10 sec
14	25 Hz	20 msec	30	50 mHz	20 sec
13	10 Hz	50 msec	29	20 mHz	50 sec
12	5 Hz	100 msec	28	10 mHz	100 sec
11	2.5 Hz	200 msec	27	5 mHz	200 sec
10	1 Hz	500 msec	26	2 mHz	500 sec
9	0.5 Hz	1000 msec	25	1 mHz	1000 sec
8	0.25 Hz	2000 msec	24	.5 mHz	2000 sec
7	0.1 Hz	5000 msec	23	.2 mHz	5000 sec

Binary in/out flow chart

Figure 4-4. Binary Input/Output flow charts

Single-Channel Input From Other Device

Events in channel 16 on Fourier Analyzer shown below.

Single-Channel Output to Other Device

Events in channel 16 on Fourier Analyzer shown below.

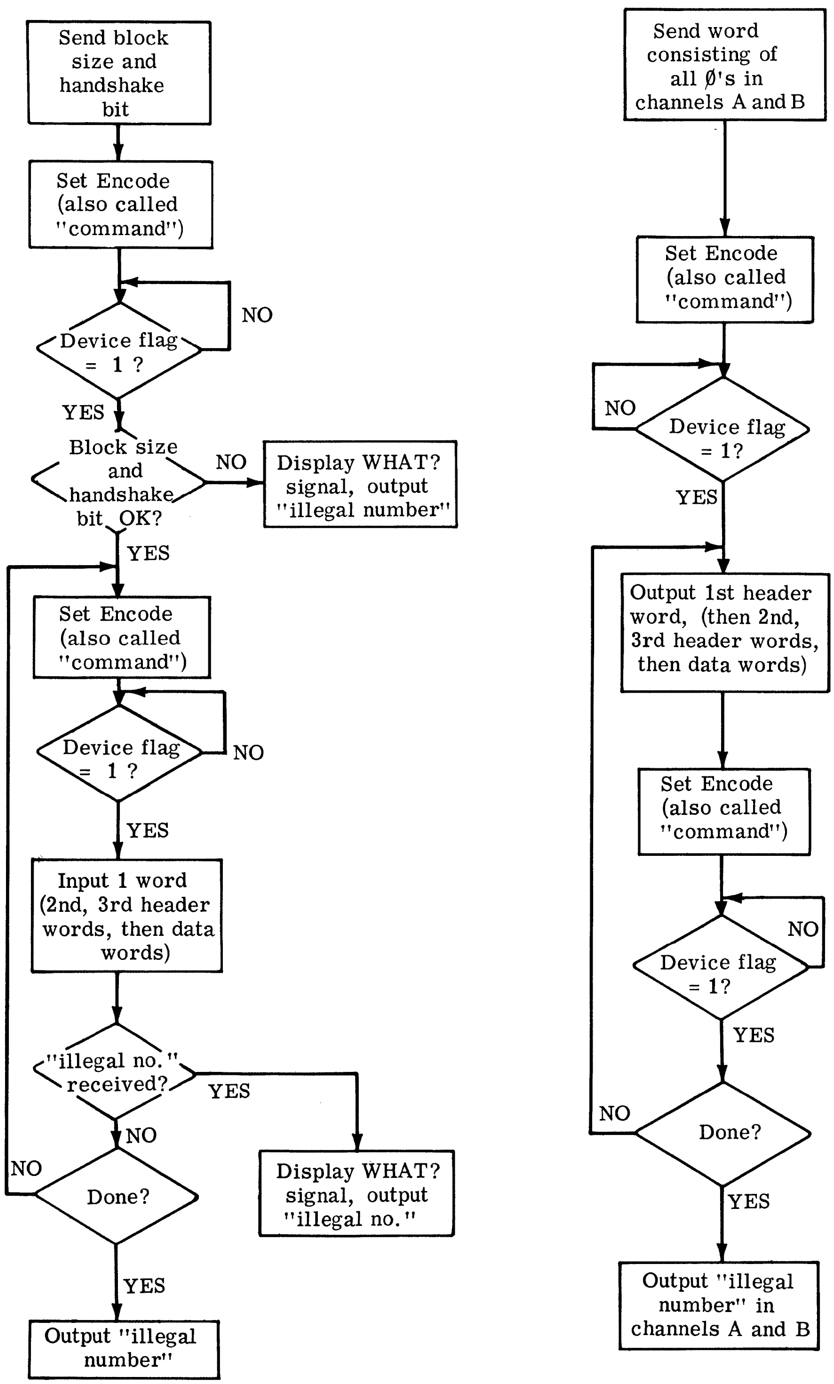


Figure 4-4. Binary Input/Output flow charts (cont'd)
Two-Channel Input from Other Device
Events in channels 16 and 17 on Fourier Analyzer shown below.
Channel A = 16; channel B = 17

Binary in/out flow chart

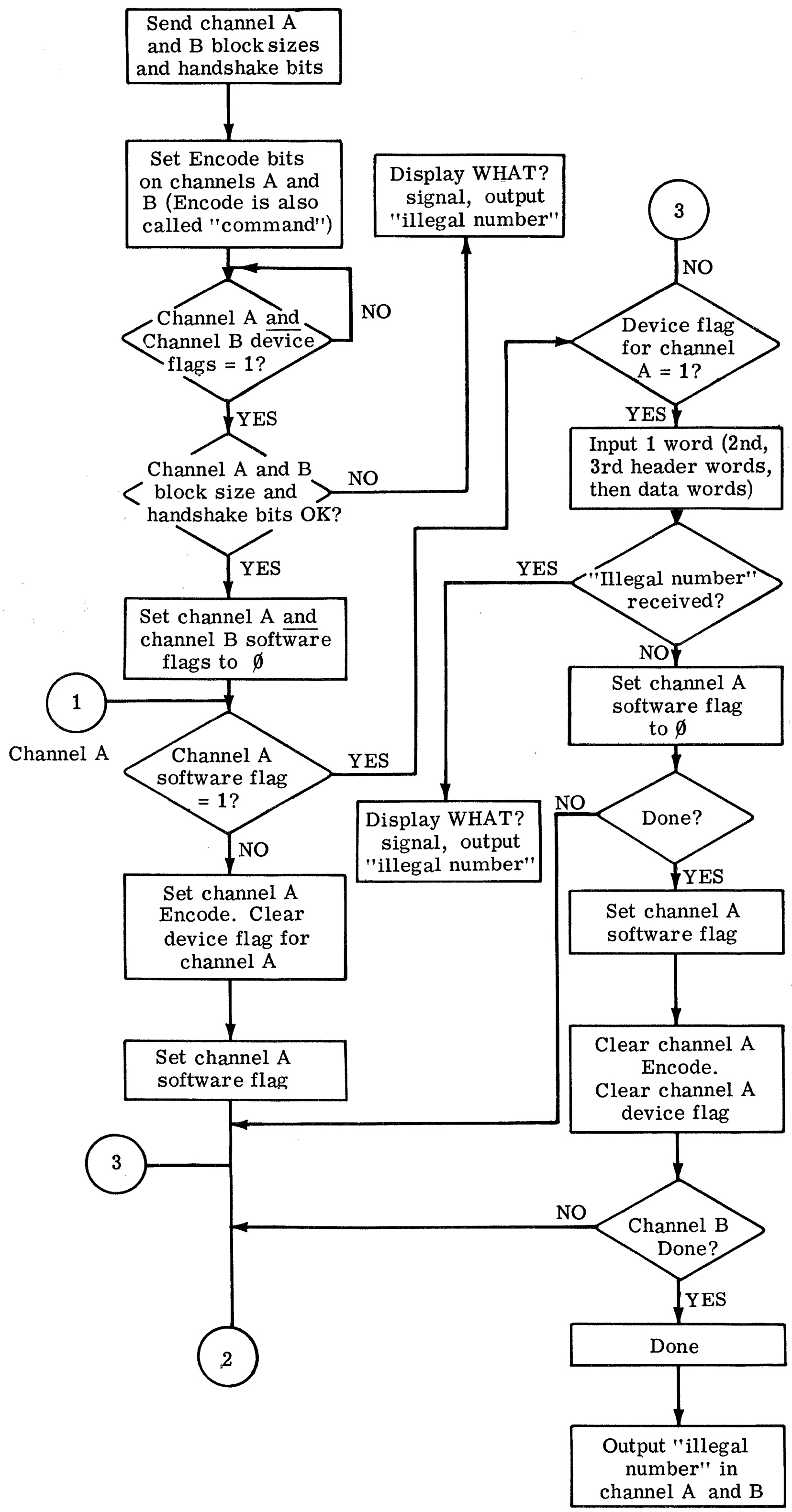
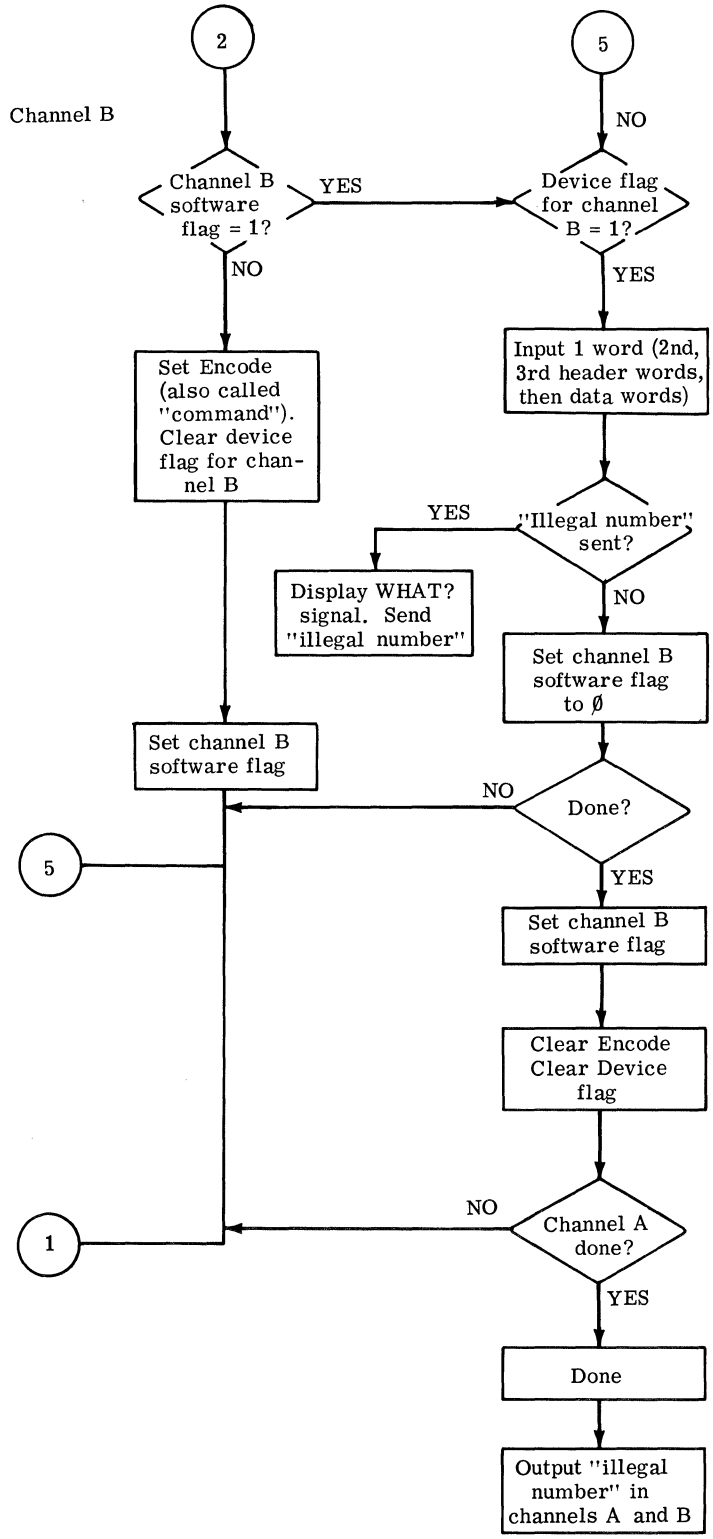


Figure 4-4. Binary Input/Output flow charts (cont'd)
Two-Channel Input from Other Device (cont'd)



Binary in/out flow chart

Binary in/out flow chart

Figure 4-4. Binary Input/Output flow charts (cont'd)
Two-Channel Output to Other Device
Events in channels 16 and 17 on Fourier Analyzer shown below.
Channel A = 16; channel B = 17

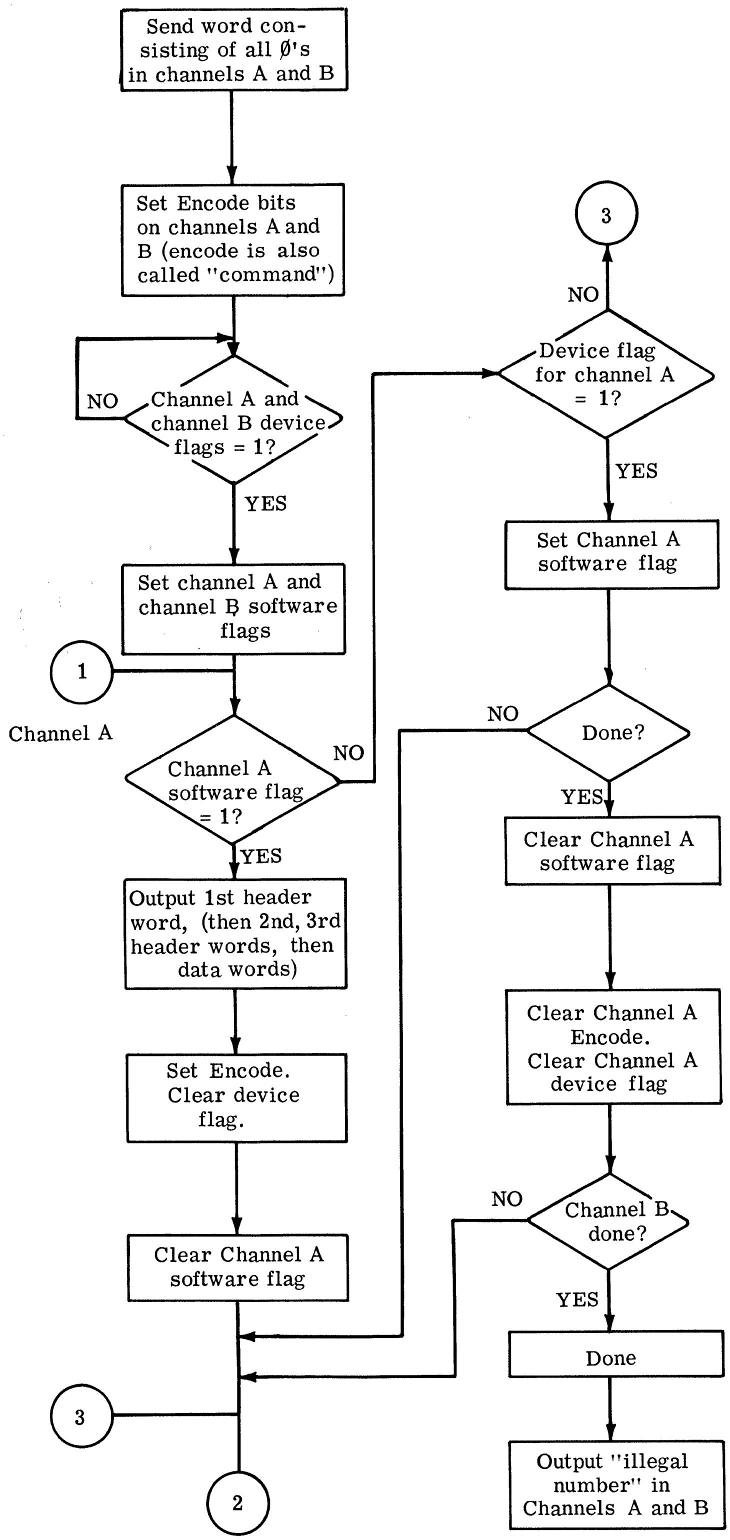
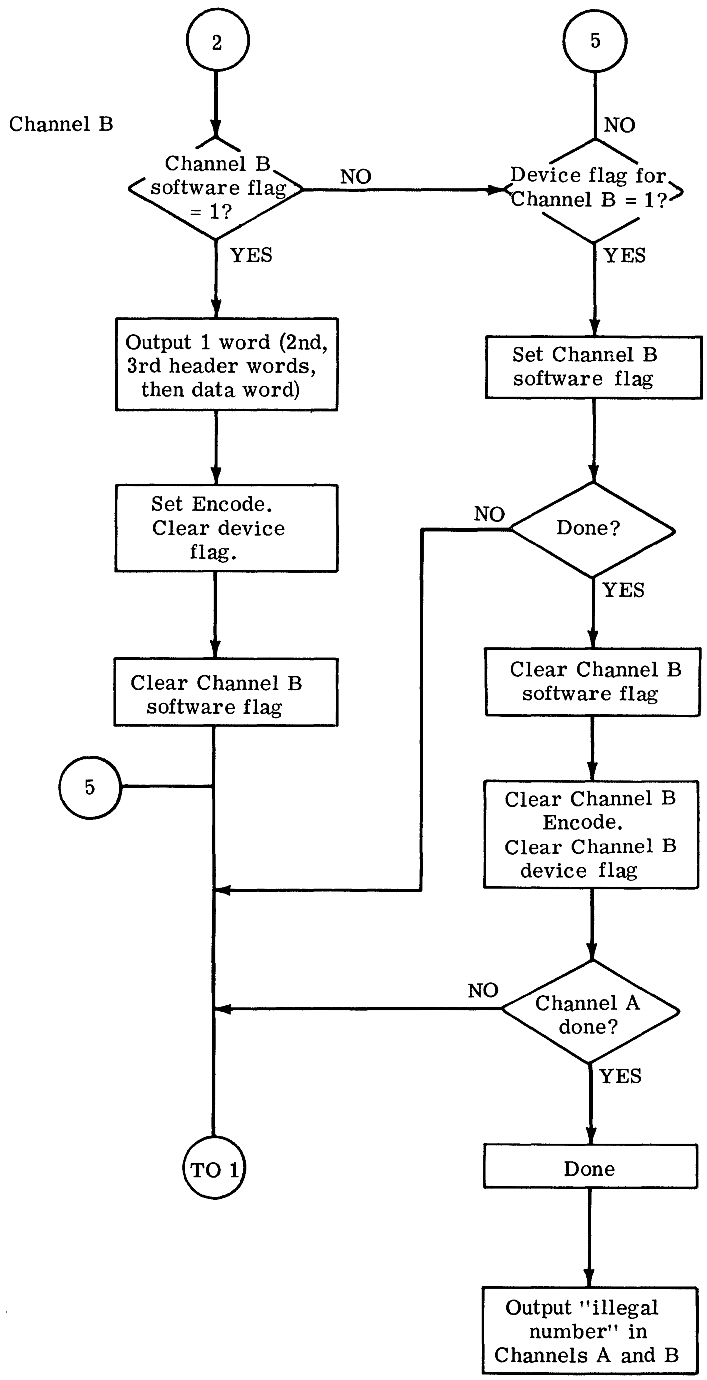


Figure 4-4. Binary Input/Output flow charts (cont'd)
Two-Channel Output to Other Device (cont'd)



Binary in/out flow chart

SECTION V
PROCESSING OPERATIONS

ABOUT THIS SECTION

This section describes Keyboard functions concerned with the processing of data. The functions are broken down in the same groups as on the Keyboard: Data Manipulation 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. Transform Related keys involve complex operations, most of which change the basic units of the data. Fourier Transform (F), for example, changes one of the data coordinates from time to frequency, or vice versa in the case of the inverse transform (F^{-1}). Arithmetic keys perform basic arithmetic operations on data blocks.

[POLAR] Coordinates

Polar coord's

To convert from rectangular coordinates [real (cosine) series and imaginary (sine) series] to polar coordinates (magnitude and phase), the following command is used:

[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 frequency. In the phase display, the vertical axis is degrees, and the horizontal axis again frequency. Switching between the two displays is accomplished by the MODE switch on the Display Unit panel. 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 panel. 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

To convert from polar coordinates (magnitude and phase) to rectangular [real (cosine) series and imaginary (sine) series], the command 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 panel, one can display either the real (cosine) components or the imaginary (sine) components of a spectrum.

LOG MAG

Logarithmic vertical scale

Rectangular coord's

To convert the vertical axis of the display to logarithmic scale, the following command is used:

LOG MAG | Block N1 | ENTER

where:

N1 is the data block. Default value = block 0.

If the display was in polar coordinates, then the logarithm is taken on the magnitude values. If the display was in rectangular coordinates, then the logarithm is taken on the real part only. Logarithm is taken on all the time domain data because it is composed solely of real numbers.

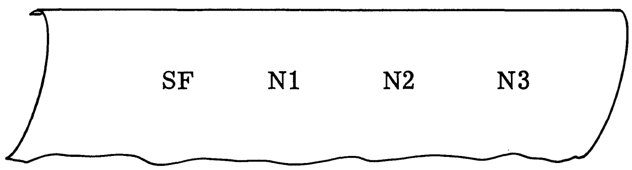
Scale factor display

The scale factor display specifies the offset, i.e., the value of the top line of the scope display in dB. When the Fourier tape was first loaded into the machine, the log scale was automatically calibrated in power dB, that is: 5 dB/division on the scope display. If you wish to convert this to a voltage dB calibration (10 dB/division), the tape must be reloaded and then, after it stops, the LOAD TAPE key must be pressed again. A final section of tape will then be loaded automatically which calibrates the log scale in voltage dB.

The SCALE switch on the Display Unit permits the user to adjust the offset to a decade value such as 0, 20, 40, etc. +2 power dB or +4 voltage dB are added for each position to the right the switch is turned. Thus, for example, if the offset was -12 dB, then one position to the right would bring it to the decade value of -10 dB.

Log printouts

In a logarithmic printout on the Teleprinter (accomplished by the PRINT command), the dB values are printed in hundredths. The scale factor N1 at the start of the printout, i.e.,



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 machine automatically makes sure that the offset is converted to a multiple-of-ten offset.

Exponential magnitude,

Clear block

[EXP MAG] Exponential Magnitude Coordinates

This command is the inverse of LOG MAG and is used to return to a linear scale from a logarithmic vertical scale. The form is:

[EXP MAG] [Block N1] [ENTER]

where:

N1 is the data block. Default value = block 0.

[CLEAR]

There are two types of CLEAR functions: block and partial block. The command to clear an entire data block is:

[CLEAR] [Block N1] [ENTER]

where: N1 is the default value. Default value = block 0.

The command to partially clear a block, i.e., a given range of channels, is:

[CLEAR] [Block N1] [SPACE] [Channel N2]

[SPACE] [Channel 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 only channel N2 will be cleared.		

Of course, the block clear is simply the defaulted version of the partial block clear.

Load block

[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 left untouched. The command is:

[LOAD] [Block N1] [ENTER]

where:

N1 is any data block except block 0. No default allowed.

Note that, since this is an operation between two data blocks, one of them must be block 0, as per the discussion of the general Keyboard command structure, page 3-12. Hence, block 0 is not named in the command.

To remember the difference between the LOAD and STORE commands think: "LOAD 0, STORE elsewhere."

STORE

This function transfers data from block 0 to block N1. Block 0 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 any data block except block 0. No default allowed.

Note that, since this is an operation between two data blocks, one of them must be block 0, as per the discussion of the general Keyboard command structure, page 3-12. Hence, block 0 is not named in the command.

To remember the difference between the LOAD and STORE commands, think: "LOAD 0, STORE elsewhere."

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 any data block except block 0. No default allowed.

Note that, since this is an operation between two data blocks, one of them must be block 0, as per the discussion of the general Keyboard command structure, page 3-12. Hence, block 0 is not named in the command.

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 INTERCHG 1 ENTER yields...	B	A	B
The command INTERCHG 2 ENTER yields...	C	A	B
The command INTERCHG 1 ENTER yields...	A	C	B

[←] **Block Shift or Rotate**

This shifts or rotates the entire block a given number of channels to the left. In this case, the block is assumed to be circular, i.e., the last channel is connected to the first. The command form is:

Shift,

Addition

[←] [Block N1] [SPACE] [N2 Channels] [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.

In the frequency domain, as far as the shift command is concerned, the end of the real portion is assumed to be connected to the beginning of the imaginary portion, and the end of the imaginary portion then circles around to the beginning of the real.

[+] **Block Addition**

This function adds the data in any block, point-by-point, to the data in block 0, and stores and displays the result in block 0. The data in the other block is left unchanged. The command structure is:

[+] [Block N1] [ENTER]

where:

N1 is any data block. Default value = block 0 (i.e., in that case, data block 0 is added to itself.)

All block additions must be done in the same domain (both blocks in time domain, or both blocks in frequency). Otherwise the answer is nonsensical. Data should be in rectangular linear coordinates. However, the magnitudes in polar coordinates may be added (for example, to obtain an ensemble average), but the resulting phase sum will be nonsensical.

[–] **Block Subtraction**

This function 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. The command structure is:

[–] [Block N1] [ENTER]

where:

N1 is any data block. Default value = data block 0 (in that case, result would be zero as data is being subtracted from itself.)

All block subtractions must be done in the same domain (both blocks in time domain, or both blocks frequency). Otherwise, the answer is nonsensical. The magnitudes in polar coordinates may be subtracted (for example, to obtain an ensemble average), but the resulting phase difference will be nonsensical.

[MULT] **Loop Counter Multiply**

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 the user attempts 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 Multiplication**

This is the point-by-point multiplication of block 0 by any data block, N1, with the results being stored and displayed in 0, and the data in the other block left unchanged. 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 data in both block 0 and the other block must be in rectangular coordinates, and both must be in the same domain, otherwise the result is nonsensical.

Conjugate multiplication

[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] [Block N1] [SPACE] [Integer N2] [ENTER]

where:

N1 is the data block. No default allowed.

N2 is the integer, which can be any positive value from 1 through 32,767. 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] **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 result in block 0. The data in the other block is left untouched. The command form is:

[* 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.)

Data must be in rectangular coordinates.

[÷] Integer Divide

This function divides block N1 by the integer N2. The result is stored and displayed in block N2. 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.

[÷] Loop Counter Divide

This command is used solely inside a count loop in a 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 the user attempts 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.

[÷] Block Division

This function divides the data in block 0, point-by-point by the data in block N1. The result is stored and displayed in block 0, the data in N1 is left untouched. The command form is:

[÷] [Block N1] [ENTER]

where:

N1 is the data block by which block 0 is to be divided.
No default allowed.

Both blocks must be in linear, rectangular coordinates, and both must be in the same domain (both time or both frequency). Otherwise, the results may be nonsensical.

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 0 and 1.0 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 divided, 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 one are divided by 2 and the scale factor is multiplied by 2, 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 5 scale-downs are allowed. This means that the quotient and dividend will be divided by 32 in the worst case. When this number of scale-downs is reached, any further overflows result in the quotient being "saturated" or overflowed at the largest number. When this happens, the largest numbers in the block are the overflowed channels and are not valid. The block division is then completed for all channels, and the WHAT? light comes on. If a program is being processed, the program will halt at this point. Should the

WHAT? light come on before the division is started, this is an indication that the command itself was in error, for example, that a nonexistent data block was named. In either case, a press of the CONTINUE key will restart the program from the WHAT? signal.

If, in dividing two data blocks, you repeatedly get overflows in a certain range, the solution is to clear out (using the CLEAR command) before the division operation is executed, the channels in the numerator that typically overflow.

<u>Example</u>: The following example illustrates the use of a number of processing operation keys and specifically illustrates the effects of the dividend overflow. 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] [8] [ENTER]

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

[3] [2] [0] [0] [0] [ENTER]

A constant value as shown in Figure 5-1(a) is now in data block 0. Store this function in data block 1 with the following command:

[STORE] [1] [ENTER]

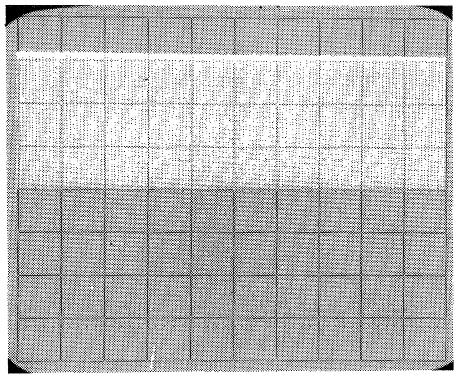
Note that the function stored in block 1 is identical to that which is left in block 0. This can be checked with the following commands:

[DSPLY] [1] [ENTER]

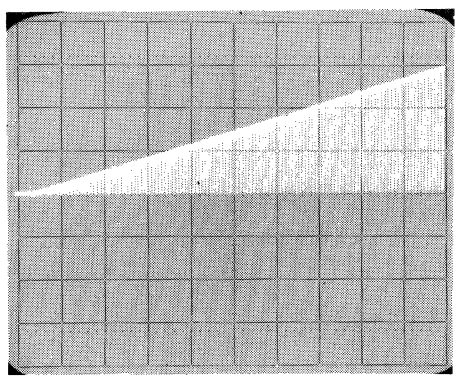
[DSPLY] [ENTER]

Division

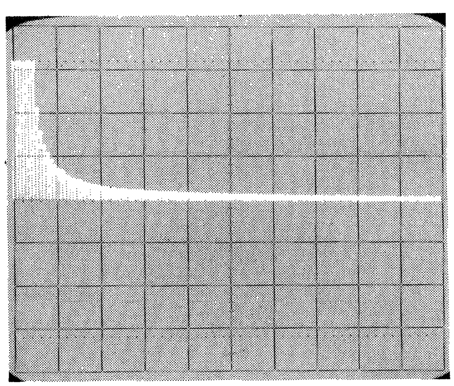
Figure 5-1. Example of block division, showing overflow



(a) Constant of 3. 2 in all channels



(b) After integration of channels 5 through 128



(c) After block division; note overflow in first seven channels

Divide the value in data block 1 by the integer value 64 with the following integer divide command:

[÷] [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 128, 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:

[∫] [1] [SPACE] [5] [SPACE]

[1] [2] [8] [ENTER]

Integration

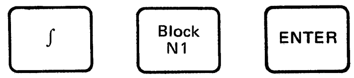
The result of this integration is shown in Figure 5-1(b). If the constant value in data block 0 is divided by data block 1 the low values in the first few channels will cause the divide to overflow. Do this division with the following command:

[÷] [1] [ENTER]

Note that the division has caused a WHAT? signal but that the division operation was completed. The result of the block division is shown in Figure 5-1(c). Note that the first seven channels are overflow values.

[∫] Integration

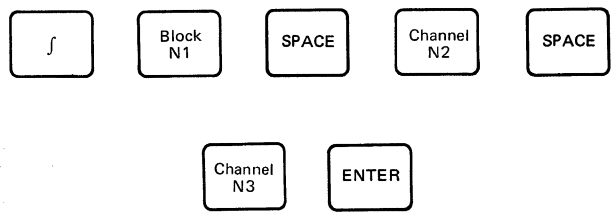
There are two types of integration in the Fourier Analyzer: block integration, in which the entire data block is integrated, and partial block integration, in which only a given number of channels in the block are integrated. The command for block integration is:



where:

N1 is the data block. Default value = block 0.

The command for partial block integration is:



Integration

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.

[$\frac{d}{dx}$] Differentiation

The command structure for differentiation is:

[$\frac{d}{dx}$] [Block N1] [ENTER]

where:

N1 is the data block. Default value = block 0.

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 Analyzer. Repeated integrations cannot bring these lost values back, hence, the slight discrepancy between the reintegrated function and the original.

[F] Fourier Transform

The Fourier transform can be taken on one data block, or on two data blocks in succession, via the command:

[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 is taken only on block N1.

The result of a Fourier transform is always in rectangular coordinates. The value of the imaginary part of the dc and the highest frequency channels is always 0, as explained below under "Fourier Transform used in Fourier Analyzer."

[F⁻¹] Inverse Fourier Transform

Like the Fourier transform, the inverse Fourier transform can be taken on one data block, or on two data blocks in succession, vai the command:

[F⁻¹] [Block N1] [SPACE] [Block N2] [ENTER]

Inverse Fourier

where:

N1 is a data block. Default value = block 0.

N2 is another data block. Default value = transform is taken only on block N1.

<u>Fourier transform used in Fourier Analyzer</u>

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 hence are not stored. 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 two 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 - 2, or N/2 -1 points. Adding the real and imaginary points together and 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 or the Photoreader, zero must be entered for the imaginary or phase value of 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, 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 (MULT*).

In the case of polar coordinates, any 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 spectrums

Since the Fourier algorithm 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 sine wave 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 MULT command on page 5-8.

A complete tabulation of the values given by the Fourier Transform, and the constants needed to correct these, are shown in the table below.

Fourier tr.

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 page 4-4). Since the effective noise power bandwidth of the 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 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.

[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, page 3-12, 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.

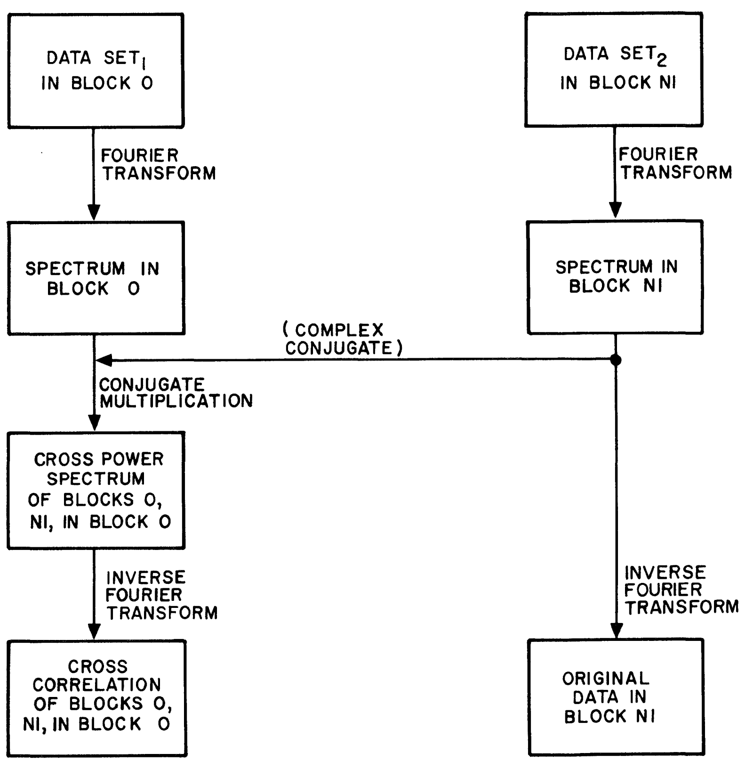
If the correlation is between block 0 and another block, then the result is a cross correlation. If between 0 and itself, the result is an autocorrelation, as shown in Figure 5-2. The command structure is:

[CORR] [Block N1] [ENTER]

where:

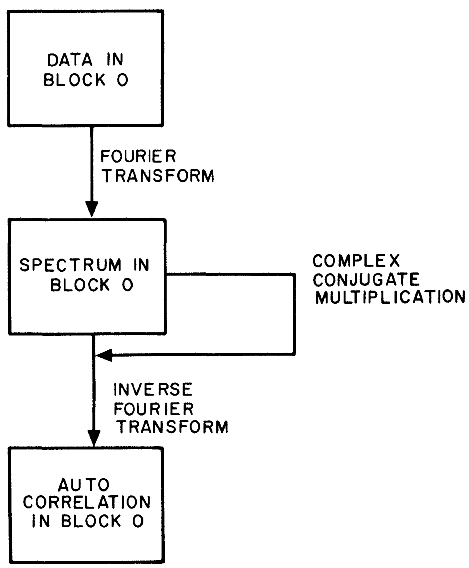
N1 is any data block. Default value = block 0 (in that case, an autocorrelation will result).

Figure 5-2(a). Flow graph of cross correlation computational procedure



The means by which the Analyzer obtains the correlation function is as follows: first the Analyzer takes the Fourier transform of both blocks (cross correlation) or of block 0 (autocorrelation). The Analyzer 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. Flow graphs of these computations are shown in Figure 5-2. Note: when beginning a correlation with data in the frequency domain, it is necessary to multiply the result by the block size to achieve the proper scaling and calibration.

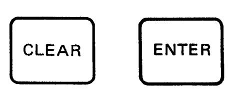
Figure 5-2(b). Flow graph of autocorrelation computational procedure

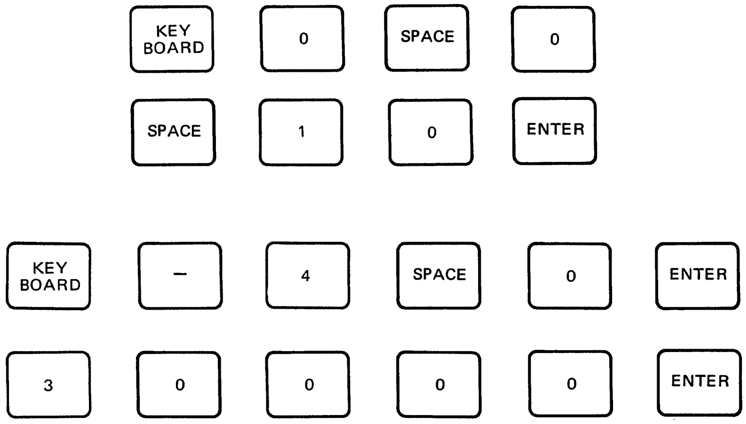


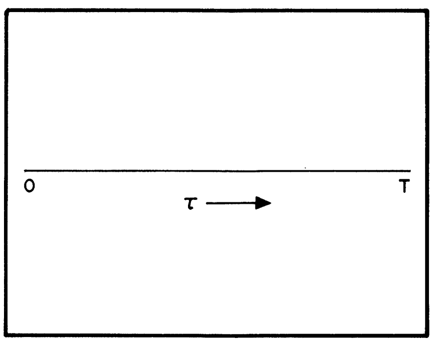
Correlation

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 on page 5-21. 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. Channel 10 is the tenth channel from the left origin, channel N-10 is the N-10th channel from the left origin. When the origin is set to CENTER, these channel numbers retain their same relative positions.

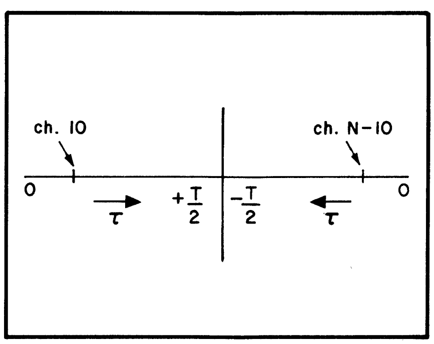
Example 1: Set a block size of 128, then manually create a rectangular pulse in block 0, 10 channels wide, via the following commands:



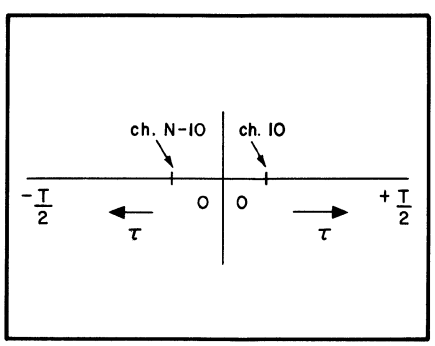




NORMAL TIME DOMAIN DISPLAY



AFTER CORRELATION, ORIGIN SWITCH ON LEFT



AFTER CORRELATION, ORIGIN SWITCH ON CENTER

Correlation

Take the autocorrelation of the pulse by pressing:

CORR ENTER

The result should be as shown in Figure 5-3(a). Switching the ORIGIN switch to CENTER should give the result shown in Figure 5-3(b).

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

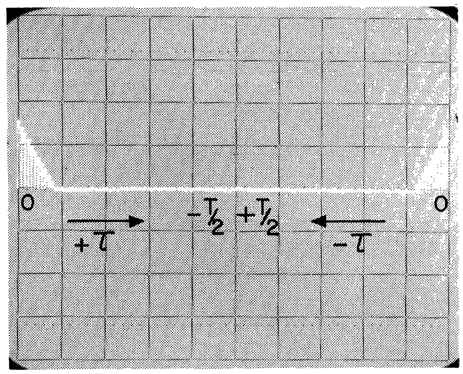
The result should be as shown in Figure 5-3(c), and indicates a lag between the two signals of 28 channels.

Wrap-around error

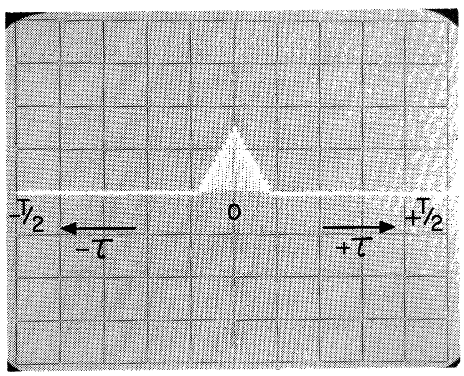
In any correlation or convolution operation using the Fast Fourier Transform, the effect known as wrap-around error must be taken into account. It occurs whenever one of the signals to be correlated or convolved is 0 for less than half of the record length T.

To understand the cause of wrap-around error, we must realize that correlation or convolution in the Fourier Analyzer are done by going

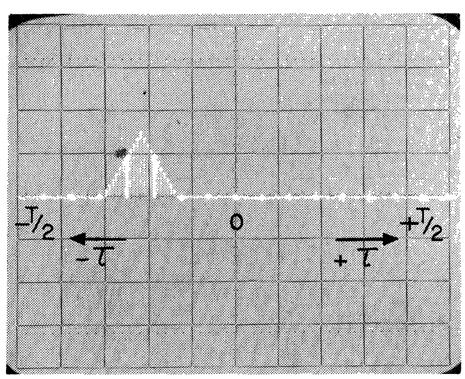
Figure 5-3. Autocorrelation (a and b) and Cross correlation (c)



(a) ORIGIN switch on LEFT

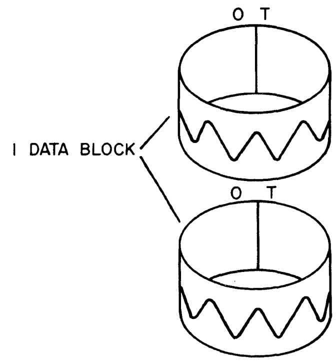


(b) ORIGIN switch on CENTER



(c) After original pulse was shifted 100 channels
(8 point markers)

through the frequency domain. This means that whatever portion of the original signal the Analyzer takes in, it assumes the data block to be periodic, with the period having the record length T, as shown in Figure 5-4, 3. Now when correlation or convolution are 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 5-4, 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 5-4, 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:



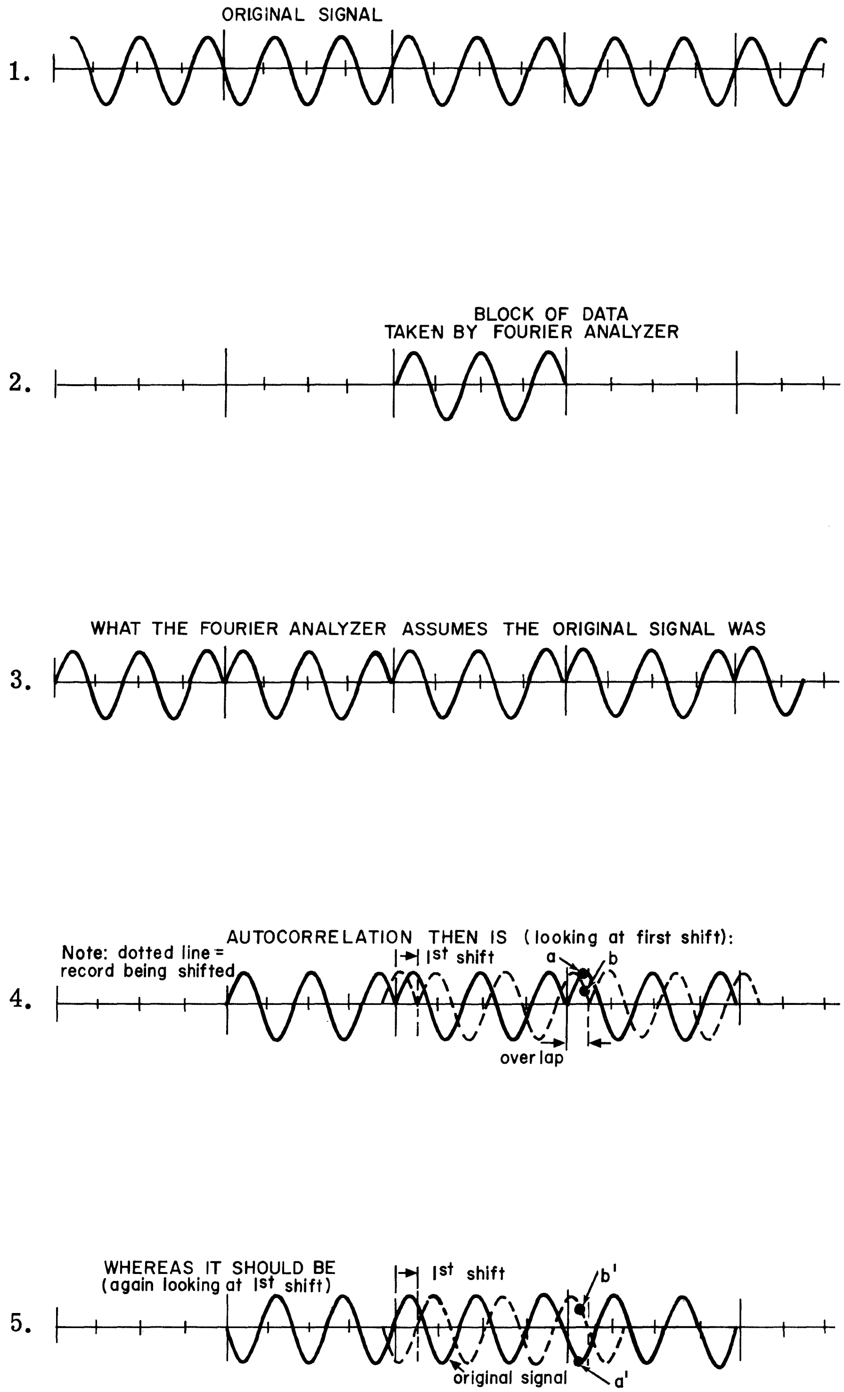
Wrap-around error

The way to eliminate the problem is shown in Figure 5-4, 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 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 5-5, 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 5-5, 4) the very 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 Section VIII.)

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, as shown in Figure 5-4, CLEAR out the first and last quarter of either of the blocks.
3. Correlate the blocks, using the CORR 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.

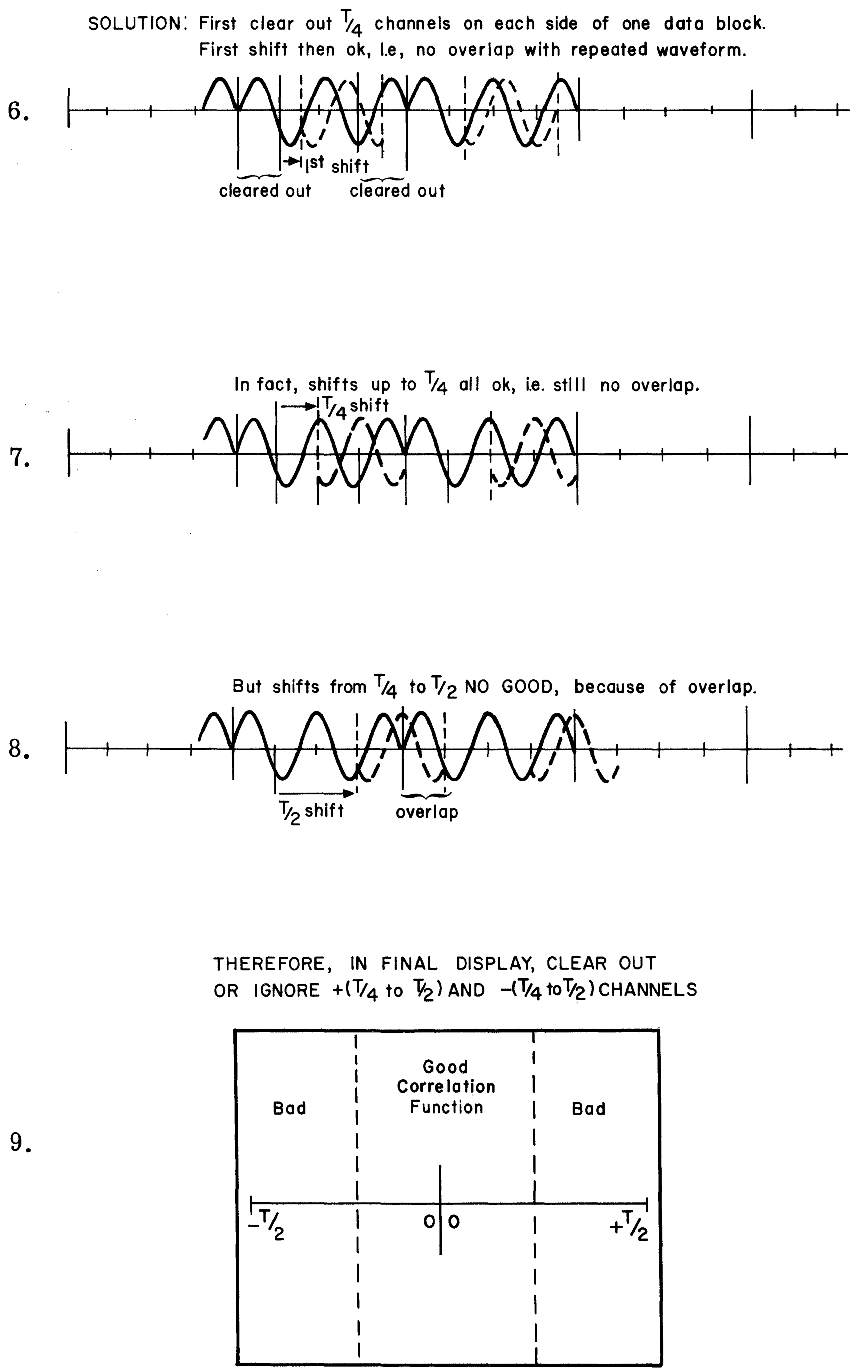
Figure 5-4. Wrap-around error



Wrap-around error

Figure 5-4. Wrap-around error (cont'd)

Wrap-around error



[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, page 3-12, 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.

If the convolution is between block 0 and another block, then the result is a cross convolution. If between 0 and itself, the result is an auto convolution. The command structure is:

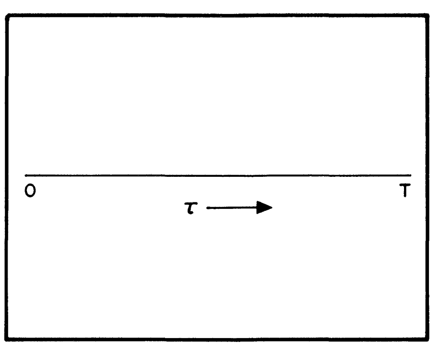
[CONV] [Block N1] [ENTER]

where:

N1 is any data block. Default value = block 0 (in that case, an autoconvolution will result).

The means by which the Analyzer obtains the convolution function is as follows: First the Analyzer takes the Fourier transform of both blocks (cross convolution) or of block 0 (auto convolution). The Analyzer does a block 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 convolution function. Flow graphs of these computations are shown in Figure 5-5. Note: when beginning a convolution with data in the frequency 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:



NORMAL TIME DOMAIN DISPLAY

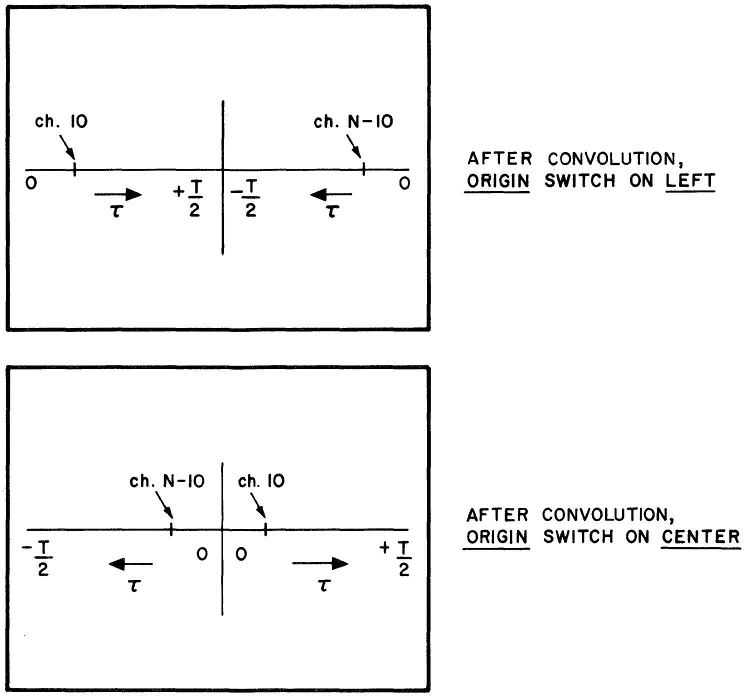
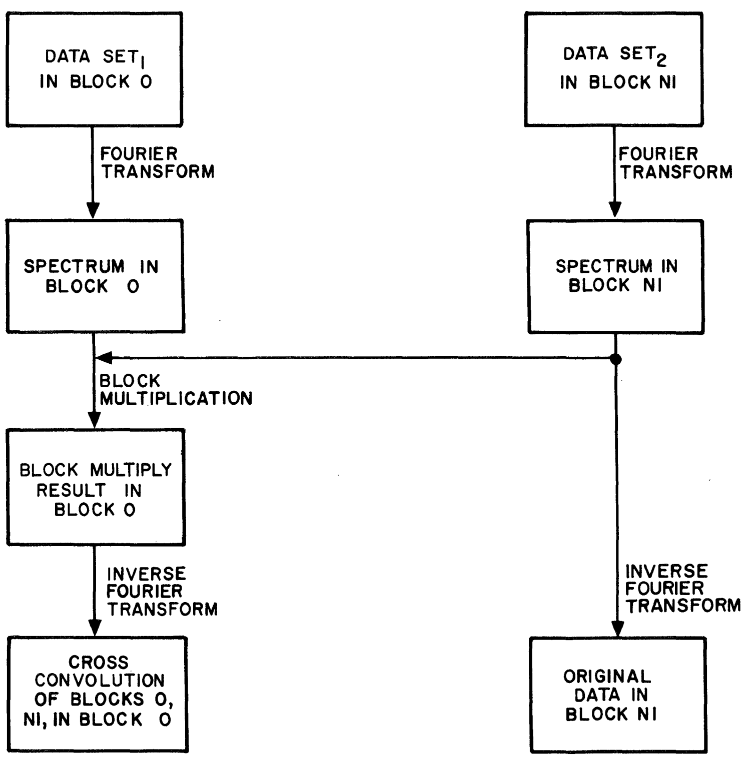


Figure 5-5(a). Flow graph of cross convolution computational procedure

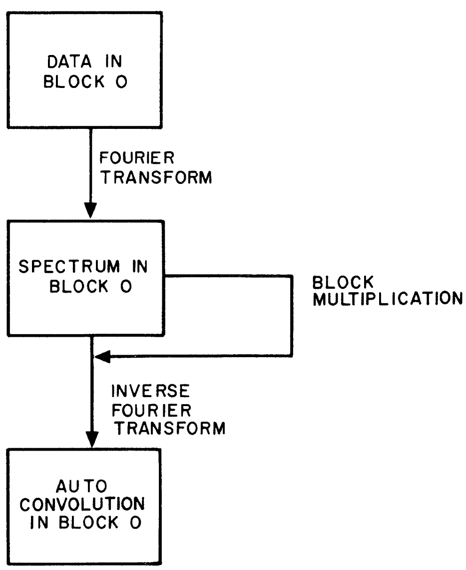
Convolution



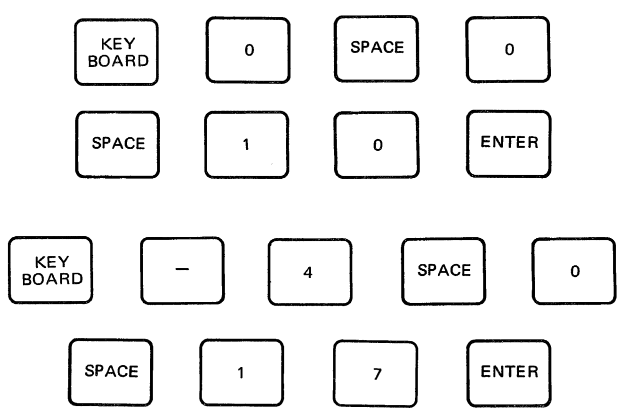
Note that the position of the origin can be switched with the ORIGIN switch on the Display Unit panel. The switch of origins is a display function, not a change in channel locations in the data block. With the ORIGIN switch on LEFT, the channels run from zero on the left to N-1 on the right. Channel 10 is the 10th channel from the left origin, channel N-10 is the N-10th channel from the left origin. When the origin is set to CENTER, these channel numbers retain their same relative positions.

Be sure to see the remarks on wrap-around error beginning page 5-22. This error applies to both the convolution and correlation functions.

Figure 5-5(b). Flow graph of auto convolution computational procedure



Example 1: Set a block size of 128, then manually create a rectangular pulse in block 0, 10 channels wide, via the following commands:



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

Take the auto convolution of the pulse by pressing:

[CONV] [ENTER]

The results should be as shown in Figure 5-6(a). Setting the ORIGIN switch to CENTER should give the results shown in Figure 5-6(b).

Convolution

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]

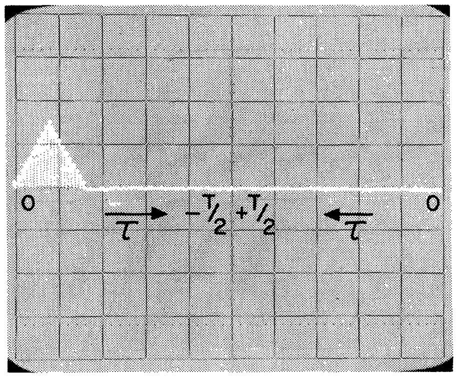
[0] [ENTER]

Take the cross convolution of blocks 0 and 1 by:

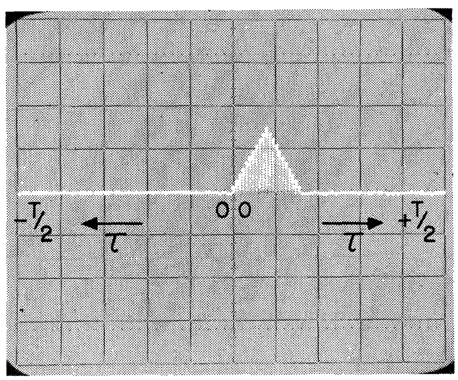
[CONV] [1] [ENTER]

The result should be as shown in Figure 5-6(c), and indicates a lag between the two signals of 28 channels.

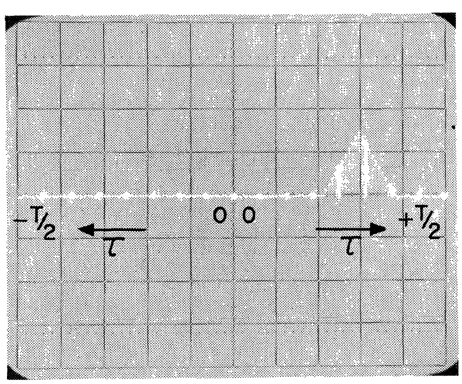
Figure 5-6. Autoconvolution (a and b) and cross convolution (c)



(a) ORIGIN switch on LEFT



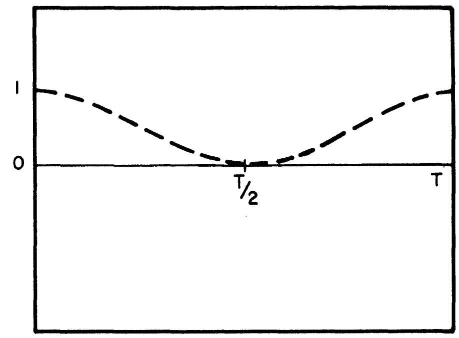
(b) ORIGIN switch on CENTER



(c) After original pulse was shifted 100 channels
(8 point markers)

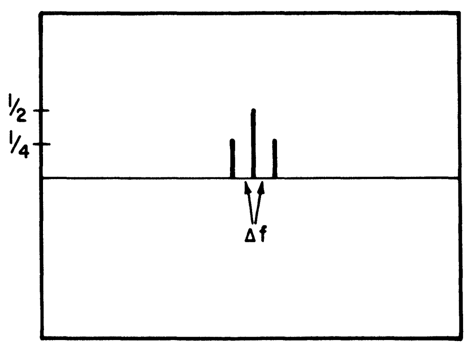
[HANN O] Origin-Centered Hanning

This provides the window function, $1/2 + 1/2 \cos 2\pi\ t/T$:



Hanning

which can be used to window quadratic functions such as correlation functions and power spectra formed from correlation functions. Hann 0 is always done by convolving the transform of the function in question with the following delta function, which is the Fourier transform of the window above:



Thus, to window in the time domain, one first takes the Fourier transform of the function in question, then does a Hann 0 on it. Repeated Hann 0's are permitted.

The command for Hann 0 is:

[HANN O] [Block N1] [ENTER]

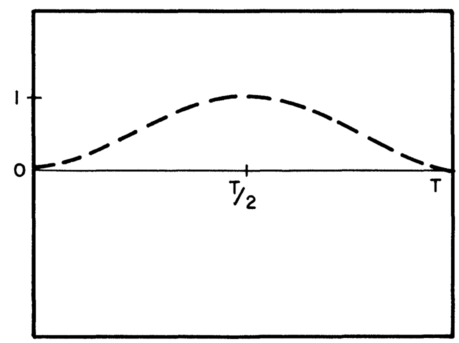
where: N1 is the data block. Default value = block 0.

Hann 0 has one additional application, that of a smoothing function in the domain to which it is directly applied. This is because convolution by the delta function above is a form of running average across the function in question. In brief: windowing in the time domain must be done by a Hann 0 in the frequency domain. Smoothing in either the time or the frequency domain is done by a Hann 0 in that domain.

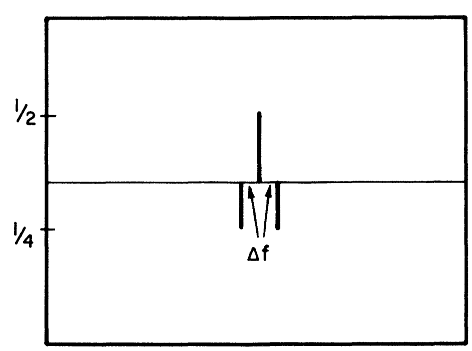
HANN I

Interval-centered Hanning

This provides the window function, $1/2 - 1/2 \cos 2\pi t/T$:



The function can be used, for example, to suppress leakage lobes in the frequency domain. Hann I always is done by convolving the transform of the function in question with the following delta function, which is the transform of the window function above:



Thus, windowing in the time domain is achieved by first taking the Fourier transform of the function in question, then doing a Hann I on it.

The command for Hann I is:

[HANN I] [Block N1] [ENTER]

where:

N1 is the data block. Default value = block 0.

Repeated Hann I's are permitted: there is a leakage reduction of 12 dB/octave each time the window is applied.

Hanning

SECTION VI

OUTPUT MODES

ABOUT THIS SECTION

This section provides operating instructions for the following output modes: print (Teleprinter), punched tape, scope display and plotter. Binary output is covered with binary input in Section IV, page 4-22. The print (Teleprinter) and scope display outputs are standard on all Fourier Analyzers. Punched tape and plotter are options.

PRINTED DATA OUTPUT VIA TELEPRINTER

How Done

The 2752A Teleprinter is used to obtain a printed copy of the data in any block, or in any part of a data block. The data is printed in the decimal number system with radix 10 scale factor.

Print Command

The form of the command for printing out data is:

Teleprinter output

[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.

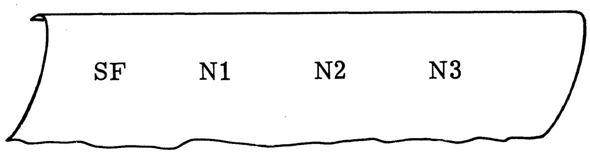
N3 is the final channel to be printed out.

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 print-out	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 Out 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:



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.

Teleprinter output

When the Fourier tape was loaded into the machine, the log scale was automatically calibrated in power dB (5 dB/division on scope); to obtain a voltage dB calibration (10 dB/division), reload the Fourier tape, then, when it stops, press LOAD TAPE again, causing a final section of tape to be loaded which enters the voltage dB calibration.

N2 is the coordinate code from the table below.

Coordinate Code	Time	Frequency	Rectangular	Polar	Log	Linear
0	x		x			x
2	x				x	
4		x	x			x
5		x		x		x
7		x		x	x	

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	50 kHz	10 μsec	63	100 Hz	10 msec
46	25 kHz	20 μsec	62	50 Hz	20 msec
45	10 kHz	50 μsec	61	20 Hz	50 msec
44	5 kHz	100 μsec	60	10 Hz	100 msec
43	2.5 kHz	200 μsec	59	5 Hz	200 msec
42	1 kHz	500 μsec	58	2 Hz	500 msec
41	.5 kHz	1000 μsec	57	1 Hz	1000 msec
40	.25 kHz	2000 μsec	56	.5 Hz	2000 msec
39	0.1 kHz	5000 μsec	55	.2 Hz	5000 msec
15	50 Hz	10 msec	31	100 mHz	10 sec
14	25 Hz	20 msec	30	50 mHz	20 sec
13	10 Hz	50 msec	29	20 mHz	50 sec
12	5 Hz	100 msec	28	10 mHz	100 sec
11	2.5 Hz	200 msec	27	5 mHz	200 sec
10	1 Hz	500 msec	26	2 mHz	500 sec
9	0.5 Hz	1000 msec	25	1 mHz	1000 sec
8	0.25 Hz	2000 msec	24	.5 mHz	2000 sec
7	0.1 Hz	5000 msec	23	.2 mHz	5000 sec

Examples of typical printouts are given in Figure 6-1.

Teleprinter output

At the start of each printout line, the channel number of the first data 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. 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 out in hundredths of a degree.

After the printout, the display sometimes shifts up scale (with corresponding change in scale factor). This occurs because, prior to printout, the data is expanded to full range in memory, for maximum resolution in the printout.

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

Teleprinter output

Figure 6-1. Examples of printouts

1. Time domain data

N1: amplitude scale factor; multiply all data words by 10^{-4}

N2: coordinate code; Ø = time rectangular data

N3: frequency code; if ADC was used to input data, indicates Δf = 100 Hz; T = 10 msec otherwise is arbitrary number

```
SF     -4      Ø     63
(   Ø)    249Ø   2487   25Ø1   2495   25Ø1   2496   2499   2498
(   8)    25Ø3   25ØØ   25Ø7   2499   25Ø3
/
```

Channel no. of first word in line

data

2. Frequency domain data, rectangular coordinates

multiply all data words by 10^{-6}

4 = frequency rectangular data

same meaning as in 1.

real value / imaginary value

real / imaginary

```
SF     -6      4     63
(   Ø)    8293       Ø     7418  -3Ø89     522Ø  -5229     2391  -583Ø
(   4)    -1Ø  -4964
/
```

ch. 4 · ch. 1 · ch. 2 · ch. 3

channel no. of first <u>two</u> words in line

data

3. Frequency domain data, polar coordinates (phase in .01 degree)

5 = frequency polar coordinates

magnitude value / phase value

magnitude / phase

```
SF     -6      5     63
(   Ø)    8293       Ø     8Ø35  -2261     7388  -45Ø4     63ØØ  -6769
(   4)    4963  -9Ø11
/
```

ch. 4 · ch. 1 · ch. 2 · ch. 3

data

4. Frequency domain data, polar log coordinates (magnitude in .01 dB; phase in .01 degree)

Add -10 dB to each data word

7 = frequency polar log coordinates

magnitude value / phase value

magnitude / phase

```
SF    -1Ø      7     63
(   Ø)   -1Ø81       Ø    -1Ø95  -2261    -1132  -45Ø4    -12Ø1  -6769
(   4)   -13Ø4  -9Ø11
/
```

ch. 4 · ch. 1 · ch. 2 · ch. 3

data

PUNCHED TAPE OUTPUT

How Done

Punched tape is outputted either by the optional High Speed Paper Tape Punch or, if a punch is lacking (or turned off), then the Fourier Analyzer will automatically punch out the tape on the Teleprinter. It must be clearly understood that here we are talking about tape which is produced by the PUNCH command (described below)--that is, a tape punched in ASCII code with a radix 2 number system and which therefore can be re-entered, if desired, in the Photoreader. Punched tape can also be produced on the Teleprinter by the PRINT and LIST commands, but this tape is a decimal, not a radix 2 number system, and hence cannot be re-entered into the Photoreader. In the case of the PRINT command, it is used simply for reprinting material on the Teleprinter. In the case of the LIST command, it is used to re-enter programs only. A data tape can be entered on the Teleprinter if the tape is prepared as described on page 4-21.

The Punch-Photoreader mode loses no accuracy on repeated inputs and outputs of the same tape. and therefore should be used in preference to the decimal Keyboard-Teleprinter mode.

Output Command

The command to punch a paper tape is given below. If the instrument does not have a High Speed Punch, set the Teleprinter switch to LINE and press the ON button on the punch.

Punch output

The command 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 channel to be punched out.
N3 is the last channel to be punched out.

Following table has default values.

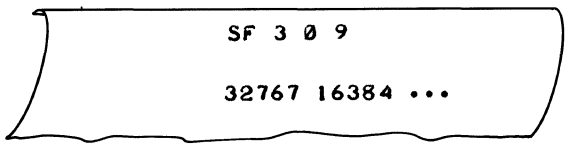
Element	Meaning of Element	Default Value of Element
N1	data block to be punched out	data block 0
N2*	first channel to be punched out	whole data block is punched out
N3*	last channel to be punched out	N2
*If N2 and N3 are defaulted, entire data block will be punched out.		

The PUNCH command causes a slash symbol (/) to be punched at the end of the data.

Data Format

The data format on any tape produced by the PUNCH command (via High Speed Punch or Teleprinter) is shown in Figure 6-2.

Figure 6-2. Data format on punched tape



SF stands for "scale factors"

3 is "k" in the expression "2^k". All data words are multiplied by 2^k, giving the number system called "floating point on a block basis." Thus, in the above example, all data words are multiplied by $2^3 = 8$. (But note on next page what these data words stand for.)

Ø coordinate code from following table. In above example, data is in time domain, rectangular coordinates.

Punch output

Coordinate Code	Time	Frequency	Rectangular	Polar	Log	Linear
0	x		x			x
2	x				x	
4		x	x			x
5		x		x		
7		x		x	x	

9 frequency code from following table, 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 100 msec.

Figure 6-2. Data format on punched tape (continued)

Freq. Code	F_{max} (MAX FREQ)	Δt (Δ TIME)
47	50 kHz	10 μsec
46	25 kHz	20 μsec
45	10 kHz	50 μsec
44	5 kHz	100 μsec
43	2.5 kHz	200 μsec
42	1 kHz	500 μsec
41	.5 kHz	1000 μsec
40	.25 kHz	2000 μsec
39	0.1 kHz	5000 μsec
15	50 Hz	10 msec
14	25 Hz	20 msec
13	10 Hz	50 msec
12	5 Hz	100 msec
11	2.5 Hz	200 msec
10	1 Hz	500 msec
9	0.5 Hz	1000 msec
8	0.25 Hz	2000 msec
7	0.1 Hz	5000 msec

Freq. Code	Δf (Δ FREQ)	T (TOTAL TIME)
63	100 Hz	10 msec
62	50 Hz	20 msec
61	20 Hz	50 msec
60	10 Hz	100 msec
59	5 Hz	200 msec
58	2 Hz	500 msec
57	1 Hz	1000 msec
56	.5 Hz	2000 msec
55	.2 Hz	5000 msec
31	100 mHz	10 sec
30	50 mHz	20 sec
29	20 mHz	50 sec
28	10 mHz	100 sec
27	5 mHz	200 sec
26	2 mHz	500 sec
25	1 mHz	1000 sec
24	.5 mHz	2000 sec
23	.2 mHz	5000 sec

Punch output

32767 16384. . data words. Data word system is as follows: 0 = 0; 32767 = 1.

To convert data words into actual physical values:

Use the following formula:

$$\frac{\text{data word}}{32767} \times 2^k = \text{actual physical value}$$

Thus, in the above example, actual physical value of first data word, 32767 is

$$\frac{32767}{32767} \times 2^3 = 8.00$$

Actual physical value of second data word is:

$$\frac{16384}{32767} \times 2^3 = 4.00$$

SCOPE DISPLAY OUTPUT

The 5460A Display Plug-In and the H51-180AR Oscilloscope form the Display Unit of the Fourier Analyzer. The purpose of the Display Unit is to convert digital data in the Analyzer memory to analog signals that may be displayed on the 8 x 10 cm screen of the scope, or may be sent to an external X-Y scope, or plotted on a standard X-Y plotter. Control of Analyzer operations connected with the display--such as digital expansion and log horizontal scale--are provided on the Display Unit.

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 Display Unit. Thus, display operations do not cause computational errors to build up. Only operations initiated by the Keyboard cause changes in data form in the memory.

Display Command

The command to display any data block has the following form:

[DSPLY] [Block N1] [ENTER]

where:

N1 is the data block. Default value = data block 0.

Scope output

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 or N3 of the command, or on the default value, or, in the REPEAT mode, on the setting of the DISPLAY switch on the ADC.

If a DISPLAY command is included in a program, the program will display and stop, until ordered to continue by pressing the CONTINUE key or by another command. In other words, the DISPLAY is a pause statement. However, in a machine with 16K memory, using the 16K program (Part No. 05450-90004), the following command will make the display take one sweep of the block specified, then continue the program automatically:

[DSPLY] [Block N1] [SPACE] [0] [ENTER]

where:

N1 is the data block, default value = block 0.

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 Plug-In 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 scope. 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 scope, on the center vertical axis.

Example of Calibrating the Scope for Positive Readings Only

In many cases, such as, power spectra, the function displayed on the scope will be positive only. Thus, a full plus and minus scale will be wasteful of the resolution powers of the scope. To calibrate the scope for full-screen positive display only, set:

FUNCTION to CAL
CALIBRATE to ORIGIN

Scope output

Now use the POSITION control to situate the calibration dot in the lower left corner of the screen. Use the HORIZONTAL POSITION control (on scope panel) to make any horizontal adjustments necessary to put the dot exactly on the corner. Now set:

CALIBRATE to +FS

and use the GAIN control to situate the dot exactly on the top line of the screen. It should be on the center vertical axis. Any positive display will now be spread out over the full screen.

Example 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 vertical 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]

Scope output

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

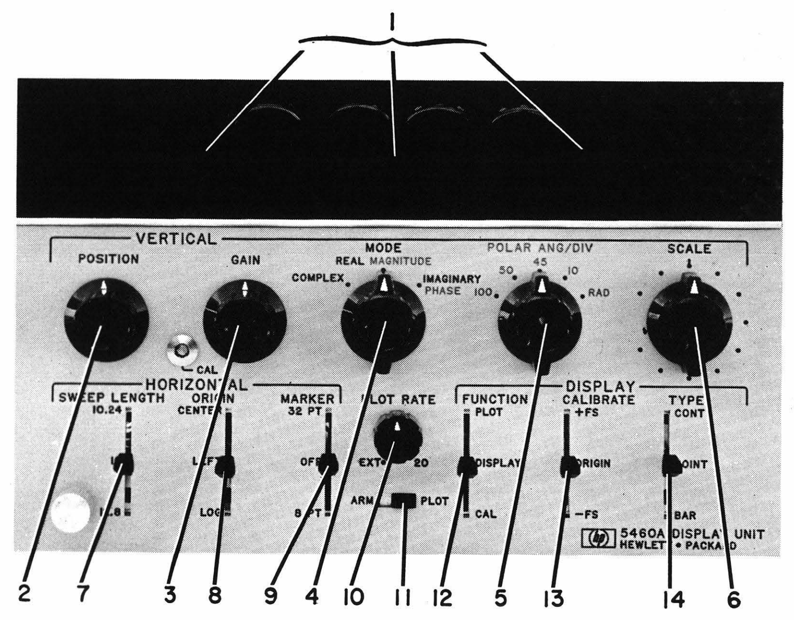
To determine what channel the point is in, set the MARKER switch to 8 PT. Note that the data point is more intensified (as is every 8th point thereafter, across the screen). Therefore, we know that the data point is in the 8th channel.

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 6-3. 5460A Display Unit



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, 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 scope.

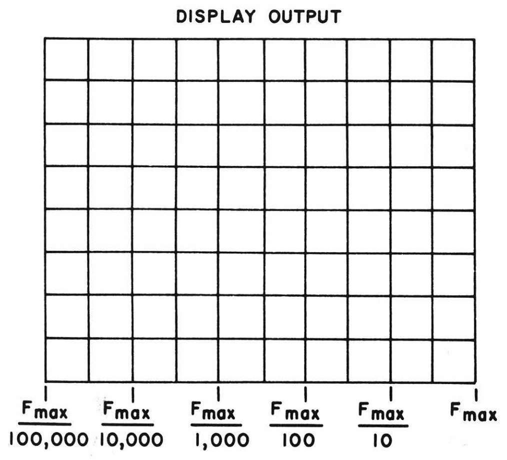
8. ORIGIN:

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. This LOG mode operates in the frequency domain only.

The horizontal log scale is illustrated below.



9. MARKER:

32 PT: gives an intensity marker every 32 channels on the horizontal axis, counting origin as channel 0.

OFF: no markers.

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

10. PLOT RATE:

EXT: allows "seek command" from Analyzer to tell external X-Y plotter to plot a point. Analyzer will not give next point until it receives plotter's "completed plot" signal for previous point.

"internal" (i.e., off EXT position): external X-Y plotter plots points at rate set by Analyzer. This control varies that rate from 5 points/sec (counterclockwise position) to 300 points/sec (clockwise position). 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 Analyzer available to external X-Y plotter, via connectors at rear of Display Unit. Shuts off scope display.

DISPLAY: makes display output from 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.

Scope output

Scope output

13. CALIBRATE:

ORIGIN: allows calibration dot to be centered on horizontal axis, left edge of screen, using VERTICAL POSITION control on Display Plug-In 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 then +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 obviate the need for precise horizontal gain (to identify a frequency point, one simply has to count the dots and multiply by Δf).

Calibration examples are on page 6-9.

Figure 6-3. 5460A Display Unit (Cont'd)

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.

PLOTTER OUTPUT

How Done

Plotter output is obtained with the HP 7004 Plotter, or with any other plotter that can handle the Display Unit output described below under "Using Other Plotters." As long as the vertical GAIN control on the Display Unit is in the CAL position, whatever is displayed on the scope will be plotted. All display controls except two affect the plotter: these are the GAIN and POSITION controls.

The following discussion applies primarily to the HP 7004 plotter.

Comparison Between Point-Plotting and Continuous Plotting

Two types of plotter output are available from the Fourier Analyzer: point-plotting and continuous plotting. Point-plotting is done with the HP 7004 Plotter and the 17012B Point-Plotting Head. The Display Unit sends a point to the Plotter, the Plotter servos null at that point, the Plotter pen lowers and places the point on the paper, the pen rises and the Plotter then sends back a flag to the Display Unit indicating it is ready for the next point. This mode is fairly rapid, having a maximum rate of 50 points/second. Its main disadvantage is that when the points are far apart (as is often the case with spectrums), the plot may be hard to read.

Plotter output

One might argue that the above problem would be solved by simply using a continuous pen, that is, a pen which is on the paper at all times, thus drawing a connecting line between each point. However, when the points are far apart, there is some overshoot in the pen, and furthermore any slight difference between the rate of movement of each servo will cause not a straight line between points, but a curved one.

A solution is provided in all Fourier Analyzers where Binary Input/Output is not used, namely the second of the two types of plotter outputs, called continuous. (The 8K 5450A is limited to the point-plot mode only.) For continuous plotting, a 7004 Plotter is used, but this time with the standard pen, which is continuously on the paper. Also, a different display command is required, namely:

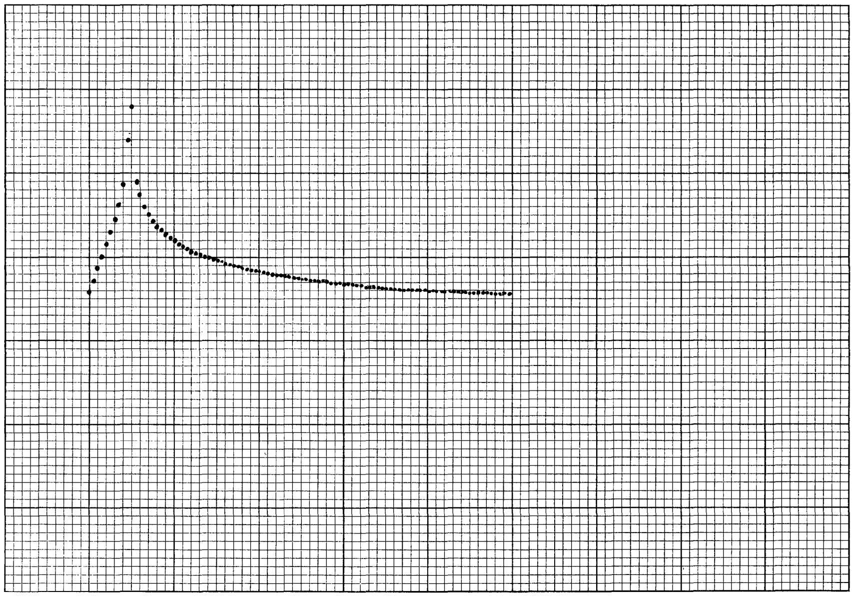
[DSPLY] [Block N1] [SPACE] [1] [ENTER]

where:

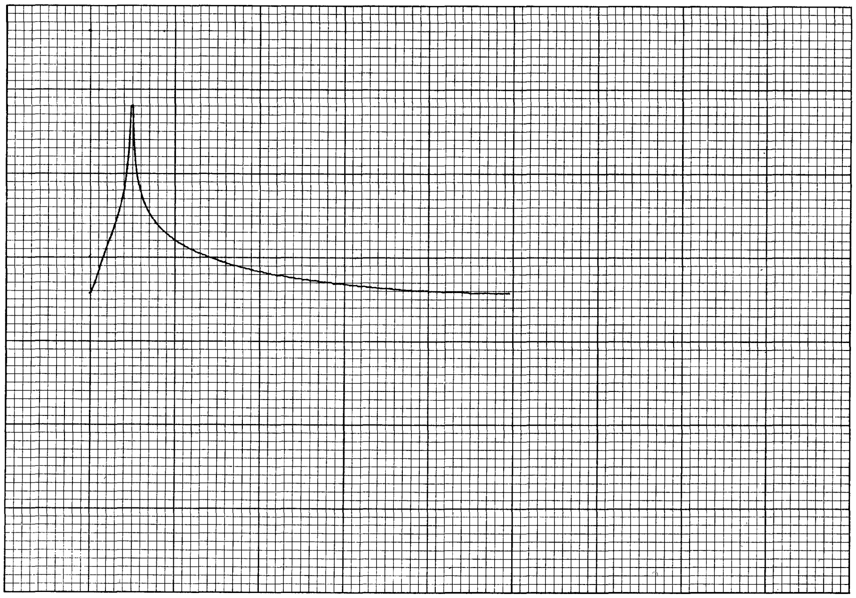
N1 is the block to be displayed.

This command calls up an interpolating display routine. The Display Unit outputs points at a rate set by the PLOT RATE control (rate is variable between 5 and 300 points per second). The Display Unit does not wait for a flag after each point. The distance between each point is so small (0.5% of scope screen width), that the overshoot for each point is negligible. In this mode the pen moves at a rate of 2.5 inches/second.

Figure 6-4. Point and continuous plots



Point plot



Continuous plot

Plotter output

Obtaining a Point-Plot

To obtain a point-plot with the HP 7004, the Plotter must be installed as described in the Plotter manual. Then, the procedure is as follows:

1. Insert graph paper in Plotter.
2. Be sure the ON-OFF switch on the 17173A Null Detector is ON. This is required on the 7004 to make the servo enable/disable function.
3. Place a 17012B Plotting Head in the arm.
4. Set the PLOT RATE control on the Display Unit to EXT.
5. Set RECORDER on plotter to MUTE. This will insure that the servos will be shut off between points.
6. Be sure the vertical GAIN control on the Display Unit is in the CAL position.
7. Set the FUNCTION switch on the Display Unit to PLOT. This causes the scope to finish its current sweep and stop. The servo-disable line from the Display Unit to the Plotter is open, turning on the plotter servos.
8. Set ARM-PLOT switch to ARM. The pen should now move to the first point on the plot and stop.
9. Set ARM-PLOT switch to PLOT. The pen should now plot the display. At the conclusion of the plot, the pen should lift from the paper.

Plotter output

To return to the display mode, set the FUNCTION switch on the Fourier Analyzer to DISPLAY. This will ground the disable line, turning off the plotter servos.

Obtaining a Continuous Plot

To obtain a continuous plot with the HP 7004, the Plotter must be installed as described in the Plotter manual. Then, the procedure is as follows:

1. Insert graph paper in Plotter.
2. Be sure the ON-OFF switch on the 17173A Null Detector is ON. This is required on the 7004 to make the servo enable/disable function.
3. Be sure a continuous plotting pen (Part No. 5080-7979, red, or 5080-7980, blue) is installed in the arm.
4. Set the PLOT RATE control about 3/4 full rotation.
5. Set RECORDER switch on plotter to NO MUTE.
6. Be sure the vertical GAIN control on the Display Unit is in the CAL position. Enter special display command (see page 6-14).

7. Set the FUNCTION switch on the Display Unit to PLOT. This causes the scope to finish its current sweep and stop. The servo-disable line from the Display Unit to the plotter is open, turning on the Plotter servos.

8. Set ARM-PLOT switch to ARM. The pen should now move to the first point on the plot and stop.

9. Set ARM-PLOT switch to PLOT. The pen should now plot the display. At the conclusion of the plot the pen should lift from the paper.

To return to the display mode, set the FUNCTION switch on the Fourier Analyzer to DISPLAY. This will ground the disable line, turning off the plotter servos.

Calibration for Point and Continuous Plotting

Before beginning a plot or a series of plots, the plotter should be calibrated. The following procedure applies to both the point and continuous modes.

1. Be sure the ON-OFF switch on the 17173A Null Detector is ON.

2. Set the RECORDER switch to the NO MUTE position.

3. Set the FUNCTION switch on the Fourier Analyzer Display Unit to CAL.

4. Set CALIBRATE switch to ORIGIN.

5. Set the X ZERO and Y ZERO controls on the Plotter to position pen in the same location as the dot on the scope.

6. Now set CALIBRATE switch to +FS and set the plotter GAIN to match the dot position.

7. Set CALIBRATE to -FS, and check pen and dot position.

8. Return RECORDER switch to MUTE.

Note: The Display Unit puts out $.500 \pm .1\%$ volts/display division.

Using Other Plotters

Plotters other than the HP models can be operated from the Display Unit by wiring a plug for connector J112 on the back of the Display Unit, according to the table below.

Output of the Display Unit is .5V per scope division.

J112 PIN NO.	FUNCTION
1	+X OUTPUT
2	-X OUTPUT
3	Not used
4	Pen Control No. 1
5	Servo disable
6	Completed plot
7	Ground
8	+Y OUTPUT
9	-Y OUTPUT
10	Not used
11	Pen Control No. 2
12	Not used
13	Seek
14	Ground

If the plotter does not have a highly damped response, and if the X and Y channel servos are not balanced as to time constant, there is likely to be hysteresis in the pen movement in the point plot mode, causing an unsatisfactory plot. The point plot mode is obtained by setting the PLOT RATE control to EXT, causing the Display Unit to output a point to the plotter then wait for a flag indicating the point has been plotted before outputting the next point. The hysteresis can be overcome by using the following display command.

[DSPLY] [Block N1] [SPACE] [1] [ENTER]

where:

Plotter output

N1 is the data block to be displayed.

This calls up an interpolation routine that will output points at the rate set by the PLOT-RATE control (5 to 300 points/sec) but only 0.5% of the scope screen width apart, thus reducing hysteresis to a negligible amount. This is called the continuous plot mode.

In normal operation the disable line is grounded when the FUNCTION switch on the Display Unit is in the DISPLAY position. When the FUNCTION switch in in the CAL or PLOT position the disable line is ungrounded (open circuit). This allows the servos of an external plotter to be muted when the Display Unit is in the display mode--that is, prevents the pen from attempting to follow the display. Pen controls 1 and 2 are shorted when the PLOT-ARM switch moves from the ARM to the PLOT position and a plot output is being delivered. The sense of contact closure can be reversed by moving the OC-CC jumper on Display Unit card 5460-60004 to OC or CC, whichever is appropriate.

Each time a new point is sent to the plotter a seek pulse is sent. If the PLOT-RATE switch is in the EXT position, the new point is not sent until a completed plot pulse (flag) is received from the plotter. This is the point plot mode. If the PLOT-RATE switch is off the EXT position, the signal goes out at the rate set, i. e., 5 to 300 points/sec. Assuming the special display command had been given, this would be the continuous mode of plotting.

SECTION VII

PROGRAMMING AND EDITING OPERATIONS

ABOUT THIS SECTION

This section gives a brief explanation of how a program is set up on the Fourier Analyzer, and details on the programming and editing keys. Keys are grouped as on the Fourier Analyzer Keyboard, i. e., Editing and Programming. Some additional keys which pertain to the programming function are grouped at the end of the section.

WHAT IS A PROGRAM

A program is a sequence of commands which the Analyzer will perform automatically. Power spectra and averaging functions are two prime examples of programming applications.

A typical printout (called a listing) of a program on the Teleprinter might appear as follows.

				Meaning
0	L1			Label 1
1	CL	1		Clear block 1
3	L2			Label 2
4	RA	0	1	Analog into block 0, display block 1
7	A+	1		Add block 0 to block 1
9	X>	1		Store block 0 to block 1
11	#2	100	0	Repeat steps from label 2, 100 times
14	.			End program

Writing a program

The following paragraphs give an outline of the kind of procedure one uses to set up and run a program on the Fourier Analyzer. Details on each of the Key commands are provided in this section. A detailed, step-by-step explanation of how to set up and run a program is given in Section III, beginning page 3-26. Examples of some typical programs will be found in Section VIII.

SETTING UP A PROGRAM

There are 100 element locations in the program memory (200 with 16K machines), which, when you begin a program, may be empty, but most likely will be partially filled with commands from a previous program. Therefore, setting up a program consists of editing the contents of the program memory. Hence, every program entry must begin with an editing command--that is, a command involving one of the EDITING keys on the Keyboard.

In general, you will enter a program with a RePLACe command, for example:

RPLAC 0 ENTER

which starts to enter the program at line 0 (i. e., at the beginning of the program memory). 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 steps, or lines as they are also called in this manual, which you enter will automatically displace downward in memory any previous lines, unless this material has been protected by a PROTect command.

After, or during the setting up of a program, the steps can be listed via a LIST command. Such a listing is shown above. The line numbers are automatically assigned by the Analyzer: the number being the number of elements to the end of the previous line (these are the same command elements that have been mentioned several times throughout this manual).

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, one uses a POINT command, to set the pointer to a given line number, followed by a press of 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, one uses a JUMP-to-a-label command; the program runs through from that label, displays results and stops.

Running a program

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, an interrogate command (symbol ?) is used: the line number and also the range of protected lines, are then printed out on the Teleprinter. The interrogate command is useful if a WHAT? signal should appear during the running of a program. The CONTINUE key will restart the program after an interrogation.

RPLAC

This command replaces a single line or a range of lines with another line or range of lines. <u>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.</u> The first line to be replaced (for example, line 0 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, in accordance with whether more or less elements were replaced; in other words, the program memory re-compacts itself for most efficient use of memory space.

The command form is:

[RPLAC] [Line N1] [SPACE] [Line N2] [ENTER]

where:

N1 is the first line number to be replaced. No default value allowed.

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, for example, a program with an END statement on line 10, 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 (line 10 in the example) and then this line must be re-entered, as is, and the new program entered following it. The reason here 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 line 11 and on). The command in that case would go back until it found a full line (line 10) 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:

				Meaning
Ø	RA	Ø	1	Analog in block 0, display block 1
3	F	Ø		Fourier transform block 0
5	*-			Conj. multiply block 0
6	A+	1		Add block 1 to block 0
8	•			End program

Now it is desired to replace line 5 with commands to clear block 0, channel 0; perform an interval-centered Hanning function in block 0; then do a conjugate multiply on the same block. The command sequence to accomplish all this is as follows:

[RPLAC] [5] [ENTER]

Note that Analyzer now goes to BUSY state.

CLEAR | 0 | SPACE | 0 | ENTER

HANN I | ENTER

* MULT | ENTER

TERM | ENTER

Note that the Analyzer now goes to the READY state.

Now we call for a printout of the entire program via the command:

LIST | ENTER

and the resulting printout would be:

```
 0 RA    0    1
 3 F     0
 5 CL    0    0
 8 H1
 9 *-
10 A+    1
12 .
```

Replace

Example 2. Now, in the above program, it is desired to replace lines 5 through 9 with a single interval-centered Hanning command. The sequence of commands is:

RPLAC | 5 | SPACE | 9 | ENTER

(BUSY light now comes on.)

[HANN I] [ENTER]

[TERM] [ENTER]

Now, a listing (obtained by pressing LIST ENTER) would look as follows:

```
0 RA      0       1
3 F       0
5 H1
6 A+      1
8 •
```

Insert, Delete

[INSRT]

This permits the addition of any number of lines following any given line in a program. 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 Analyzer then goes to READY. Program steps beyond those inserted are shifted farther down in memory.

[DELET]

This is used to delete one step or a range of steps from a program. All lines below the delete move up to re-compact the program. The command form is:

[DELET] [Line N1] [SPACE] [Line N2] [ENTER] where:

N1 is the number of the first line to be deleted. No default allowed.

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 causes any program, or any part of a program, to be printed out on the Teleprinter.

To list an entire program from line 0: 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, i.e., line 99.

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.

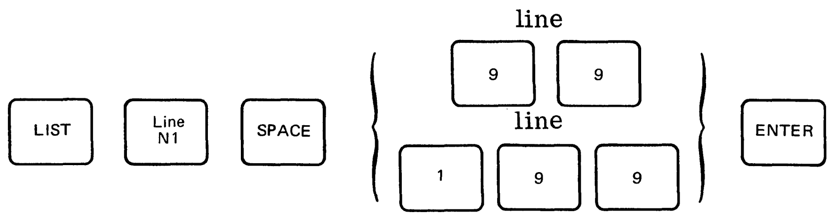
To list a range of lines: Use the command form:

[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:



where:

N1 is the first line to be listed. No default allowed.

99 is the number of the last line in 8K program memory, 199 is the number of the last line in 16K 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 99 or 199.

To punch a program on paper tape: There are two ways to punch a program on paper tape: Method 1 leaves the initial editing command (i.e., REPLACE) off the tape, so that to re-enter the tape, the command must be given separately, as described later under "Entering a program punched according to Method 1." Method 2 puts the editing command on the tape.

Method 1 (No REPLACE command on tape)

1. Set Teleprinter switch to LINE.
2. Turn off Teleprinter punch (switch on box at middle of left side of Teleprinter).
3. Enter the LIST command for the type of listing desired (see above) but do not press ENTER.
4. Turn on the Teleprinter punch.
5. Press the HERE IS key on the Teleprinter about three times to generate adequate tape leader.
6. Now press the ENTER key. The LIST command will be executed and simultaneously punch the program or portion of program on the paper tape.
7. Press the HERE IS key to generate tape trailer.

Method 2 (REPLACE command is on tape)

1. Set the Teleprinter switch to LOCAL.
2. Turn on the Teleprinter punch (switch on box at middle of left side of Teleprinter).
3. Press the HERE IS key on the Teleprinter about three times to generate adequate tape leader.
4. Type the following command on the Teleprinter.

[/] [TAPE R] [Line N1] [RE-TURN] [LINE FEED]

where:

N1 is the line number at which you want to re-enter the program; this will normally be line 0.

5. Turn off the Teleprinter punch.
6. Set Teleprinter to LINE.
7. Give the equivalent LIST command on the Fourier Analyzer Keyboard for that given in step 4 above, but do not press ENTER.
8. Turn on Teleprinter punch.
9. Press ENTER.
10. When the program has been punched out, turn off the Teleprinter punch.
11. Set the Teleprinter switch to LOCAL.
12. Turn the Teleprinter punch back on. (This repeated turning on and off of the punch is to prevent the intermediate actions, such as setting the Teleprinter switch to LOCAL from punching the tape.)
13. Type the following on the Teleprinter:

[/] [RE-TURN] [LINE FEED]

14. Generate adequate leader by pressing the HERE IS key three times.

List

To re-enter a program punched via Method 1, use the following procedure:

1. Set Teleprinter tape reader to OFF (switch is at left front of Teleprinter.)
2. Place paper tape in reader, lowering plastic cover in place.
3. Give the appropriate REPLACE command on the Fourier Analyzer 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 line N1 must be the last line of the other program, and you must immediately re-enter that line after giving the REPLACE command.

4. Turn on Teleprinter reader. The tape should now read in.
5. When the tape stops, press the TERM key on the Fourier Analyzer Keyboard.

To re-enter a program punched via Method 2, use the following procedure:

1. Set Teleprinter tape reader to OFF (switch at left front of Teleprinter).
2. Place paper tape in reader, lowering plastic cover in place.
3. Turn on Teleprinter reader. The tape should now read in, and stop when completed. The Keyboard should automatically go to READY state.

LIST Example 1: It is desired to list all steps in a program starting at line 10 (8K memory). The command is:

[LIST] [1] [0] [SPACE]

[9] [9] [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]

[?] Interrogate

This command is used to determine where the pointer is at any given time. The pointer follows down the list of steps in a program as they are executed. The interrogate command can be used, for example, after a WHAT? signal, in order to determine which line produced that signal and caused the program to stop. The command form is simply:

? ENTER

The Analyzer then prints out on the Teleprinter the pointer location and also the protected range, if any, in the following format:

PT N1

PR N2

where:

N1 is the pointer location, i. e., line number.

N2 is the first protected line, that line through line 99 (or line 199 in the case of 16K machines) being the protected range.

TERM

This command is used to indicate to the Analyzer the end of a replace or insert editing function, and also to indicate the end of a Keyboard data entry. It returns the Keyboard to the READY mode. The command is:

TERM ENTER

Terminate,

Jump

JUMP

When a program reaches this command, it immediately goes to a label specified by N, and proceeds from that point. This command can also be used to start a program running, simply by specifying the label (N) at the beginning of the program. The command form is:

JUMP Label N ENTER

where:

N is a label integer, 0 through 9. No default allowed.

Letters can be substituted for integers by entering the entire command from the Teleprinter keyboard. To do this, set the Teleprinter switch to LINE, then enter the command in the following form:

(J) (Label N) (RE-TURN) (LINE FEED)

where:

N is a letter. See LABEL command for instructions on entering a letter as a label.

The JUMP command must be executed as part of a program or to start a program running. It has no meaning as an isolated command.

Example. If, at a certain point in a program, we wished to go back to Label 2, and have the program proceed from there, the command would be:

[JUMP] [2] [ENTER]

Label

[LABEL]

This command labels a line in the program irrespective of the line number. The line can then be gotten to via the JUMP, SUB-JUMP, and COUNT commands. The form of the command is:

[LABEL] [N] [ENTER]

where:

N is an integer, 0 through 9. No default allowed.

Letters can be substituted for integers by entering the entire LABEL command from the Teleprinter keyboard. To do this, set the Teleprinter switch to LINE, then enter the command in the following form:

(FORM L) (N) (RE-TURN) (LINE FEED)

where:

N is a letter. See JUMP command for instructions on jumping to a letter label.

The LABEL command must be executed as part of a program. It has no meaning as an isolated command.

[COUNT]

This causes a portion of the program whose starting point is Label N, to be repeated N1 times. In other words, this command is used to form a loop. The command form is:

[COUNT] [SPACE] [from Label N] [SPACE] [N1 times] [ENTER]

where:

N is the label of the starting point (must be 0 through 9, or a letter). No default allowed.

N1 is the number of times the portion of the program is to be repeated. N1 must be an integer from 0 through 32, 767. No default allowed.

LABEL N

This portion of the program is done N1 times.

COUNT N SPACE N1 ENTER

When a program which includes a COUNT command is listed on the Teleprinter, the COUNT line will have the following form:

#N N1 N2

Count

where:

N is the label (an integer o through 9, or a letter).

N1 is the number of times the portion of the program is to be repeated.

N2 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 N2 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, re-enter the 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.

Example. If the user wanted to repeat the steps of a program 100 times, the following would be placed at the bottom of the list of steps, immediately prior to the END command:

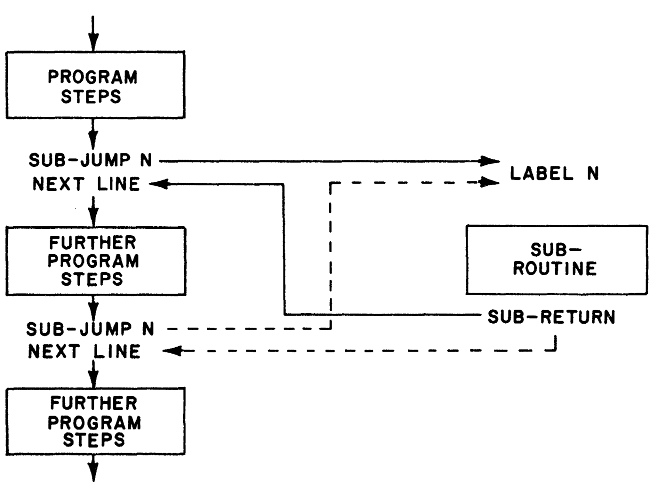
COUNT 0 SPACE 1

0 0 ENTER

Sub-jump

SUB JUMP

This command allows a short sequence of steps (i. e., a sub-routine) to be executed from any point in a program. The sub-routine must have a label N to identify its beginning, and a SUB-RETURN command to identify its conclusion. When a program reaches a SUB-JUMP command, it goes from that line to the first line in the sub-routine, executes all the sub-routine steps, then returns to the line following the one from which it entered the sub-routine. The following diagram illustrates this:



The command form is:

[SUB JUMP] [Label N] [ENTER]

where:

N is the label number of the beginning of the sub-routine. The sub-routine will only be run through once.

The SUB-JUMP command must be executed as part of a program. It has no meaning as an isolated command.

Sub-return,

End,

Block size

[SUB RTRN]

This command must be placed at the conclusion of any sub-routine in order to return the program control to where it entered the sub-routine, i. e., the step immediately following SUB-JUMP. A diagram of the sub-routine format can be found under "SUB-JUMP." The form of the SUB-RETURN command is simply:

[SUB RTRN] [ENTER]

[END]

This is used to indicate the conclusion of a program. This command should always be used in case there are residual steps from earlier programs still in memory following the new program. The command form is:

[END] [ENTER]

The END command must be executed as part of a program. It has no meaning as an isolated command.

[BLOCK SIZE]

This establishes the number of channels in each data block. Permissible block sizes are shown above the indicator lights on the top of the Control Unit Keyboard. An Analyzer with an 8K program can have block sizes from 128 to 1024; with a 16K program, from 64 to 4096. If a block size that is not allowed is commanded, the Analyzer will give a WHAT? signal. The command form is:

[BLOCK SIZE] [Block N1] [ENTER]

where:

N1 is the number of channels, as 64, 128, 256, etc.

[POINT]

The command sets the pointer to any line number (not label number) in the program. Then, when the CONTINUE key is pressed, the program will restart at that line number. The purpose of this is to restart the program at any point not specified by a label. Line numbers are obtained on a printout of the program via the LIST command. The command form is:

[POINT] [Line N1] [ENTER]

where:

N1 is the line number at which the program is to restart. Default value = line 0, in which case the program restarts at the beginning of the program memory.

Note that the CONTINUE key must be pressed before this command will be executed.

[PROT] **Protect**

This command is used to prevent a step or a range of steps from being edited or deleted, for example, by a user who was not aware that a previous user wished to preserve a portion of a program. The command form is:

[PROT] [from Line N1] [ENTER]

where:

N1 is the number of the first line to be protected. Default value = line 0, or in other words, the beginning of the program memory.

To unprotect the memory, use the command:

[PROT] [Line N1] [ENTER]

where:

N1 is 99 for 8K machines, 199 for 16K. That is, one "protects" beyond the last line only, unprotecting the rest of the program memory.

[STOP] [CON TINUE]

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 thus leaves meaningless data in memory. Neither STOP nor RESTART, however, have any effect on the Fourier Master Program, that is, they do not turn off the 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 which line the pointer is set at, use the interrogate command:

[?] [ENTER]

Stop, Continue, Step-Run

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.

SECTION VIII

SAMPLE PROGRAMS

ABOUT THIS SECTION

This section provides a number of typical sample programs, plus a discussion of the considerations that went into the development of each. A Teleprinter listing of each program is also included. Because of the length of some of the programs, the illustrated key format will no longer be used: key names will simply be spelled out in capitals.

For details on how programs are entered into the Fourier Analyzer, see Section III, beginning page 3-26.

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 spectrums (or linear spectrums) averaged in different ways. We shall begin with some remarks on the power spectrum.

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 energy 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. To obtain the required positive quantities, therefore, the Fourier transform must be conjugate multiplied, yielding a positive quantity at each frequency which is the energy at that frequency. If we sum a number of such spectrums, and then divide each frequency channel by that number, we achieve the desired average.

Power spectrum

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:

$$\sigma = \frac{1}{K} \times 100$$

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 the user 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 the user 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, the user 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, the user 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

Linear Spectrum

To accumulate a linear spectrum rather than a power spectrum, one uses a polar coordinate command in place of the conjugate multiply command. The result will be the magnitude values in the first half of the data block, and the phase in the last half. The magnitudes then sum as repeated spectral estimates are taken and stored. The phase values will randomly sum and so must be disregarded.

There are some advantages and disadvantages to a linear spectrum. 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, as a matter of fact, about 15% longer. The advantage is in increased dynamic range. The computer uses 16-bit data words (15-bits plus sign) thus providing a range of 0 to 65, 534 numbers (-32, 767 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. Thus the linear spectrum has less speed but greater dynamic range than the power spectrum.

Summation Average

The summation average is the simplest and fastest average, because it consists solely of summations, followed by a division at the end.

$$A_n = \frac{\sum_{o}^{n} I_i}{n}$$

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:

$$A_{n-1} = \frac{\sum_{o}^{n-1} I_i}{n-1}$$

Then, multiplying:

$$(A_{n-1})(n-1) = \sum_{o}^{n-1} I_i$$

If we now add the nth estimate, I_n, to both sides, we get:

$$(A_{n-1})(n-1) + I_n = \sum_{o}^{n-1} I_i + I_n$$

Dividing both sides by n:

$$\frac{(A_{n-1})(n-1) + I_n}{n} = \frac{\sum_{o}^{n-1} I_i + I_n}{n}$$

But the right hand side is A_n, the average after n estimates. And, rewriting the left hand side, we get:

$$\frac{(A_{n-1})\not{n}}{\not{n}} - \frac{A_{n-1}}{n} + \frac{I_n}{n} = A_n$$

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 computer time required to do the subtracting, division and addition after each estimate.

Exp. decaying average

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:

$$A_n = A_{n-1} + \frac{I_n - A_{n-1}}{n}$$

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

$$E\left[A_n\right] = \frac{1}{k} \sum_{i=1}^{n} E\left[I_i \left(\frac{k-1}{k}\right)^{n-i}\right]$$

Exp. decaying average

$$E\left[A_n\right] = \frac{1}{k} \sum_{i=1}^{n} \left(\frac{k-1}{k}\right)^{n-i} E\left[I_i\right]$$

$$E\left[A_n\right] = E\left[I_i\right]\left(\frac{k-1}{k}\right)^n \frac{1}{k} \sum_{i=1}^{n} \left(\frac{k-1}{k}\right)^{-i}$$

But since

$$\sum_{i=0}^{n-1} ar^i = a\left(\frac{r^n - 1}{r - 1}\right)$$

and letting $a = 1$, $r = \left(\frac{k-1}{k}\right)^{-1} = \frac{k}{k-1}$,

$$\sum_{i=1}^{n} \left(\frac{k-1}{k}\right)^{-i} = \frac{\left(\frac{k}{k-1}\right)^{n+1} - \frac{k}{k-1}}{\frac{k}{k-1} - 1}$$

$$E\left[A_n\right] = E\left[I_i\right] \frac{1}{k} \left(\frac{k-1}{k}\right)^n \left[\frac{\left(\frac{k}{k-1}\right)^{n+1} - \frac{k}{k-1}}{\frac{k}{k-1} - 1}\right]$$

$$E\left[A_n\right] = E\left[I_i\right] \frac{1}{k} \left[\frac{1 - \left(\frac{k-1}{k}\right)^n}{1 - \frac{k-1}{k}}\right]$$

$$E\left[A_n\right] = E\left[I_i\right]\left[1 - \frac{k-1}{k}^{\,n}\right]$$

Exp. decaying average

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_n, $E[I_i]$ 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 sum. avg.

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 spectrums $G_{xx}(f)$. 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
MULT* ENTER[1]	$G_{xx}(f)$ of current record	"	Forms $G_{xx}(f)$ of current record
+ 1 ENTER	Sum of current plus past $G_{xx}(f)$'s	"	Forms sum of current plus past $G_{xx}(f)$'s in block 0
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 N1 ENTER	"	"	Loop back to target label 1 N1 times, for N1 sums

[1]Replace this command with POLAR ENTER if you want to accumulate voltage rather than power spectrums.

AUTO SPECTRUM SUMMATION AVERAGE PROGRAM (Cont'd)

PROGRAM COMMANDS	CONTENTS OF BLOCK 0	CONTENTS OF BLOCK 1	PURPOSE OF COMMAND
÷ 0 SPACE N1 ENTER	Average of N1 $G_{xx}(f)$ sums	Sum of current plus past $G_{xx}(f)$'s	Forms average of N1 $G_{xx}(f)$ sums
END ENTER	Average of N1 $G_{xx}(f)$ sums	"	Ends program

LISTING OF AUTO SPECTRUM
SUMMATION AVERAGE PROGRAM

(N = 100)

```
 0 LØ
 1 CL      1
 3 L1
 4 RA      Ø        1
 7 F
 8 *-
 9 A+      1
11 X>      1
13 #1    1ØØ        Ø
16 :       Ø      1ØØ
19 .
```

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 SPACE 2 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
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 $G_{yx}(f)$'s to sum of past $G_{yx}(f)$'s
STORE 2 ENTER	"	"	"	Stores result of current pass for next pass
COUNT 2 SPACE N1 ENTER	"	"	"	Loop back to target label 1 N1 times for N1 sums

Cross spectrum sum avg.

CROSS SPECTRUM SUMMATION AVERAGE PROGRAM (Cont'd)

PROGRAM COMMANDS	CONTENTS OF BLOCK 0	CONTENTS OF BLOCK 1	CONTENTS OF BLOCK 2	PURPOSE OF COMMAND
÷ 0 SPACE N1 ENTER	Average of N1 $G_{yx}(f)$'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

LISTING FOR CROSS SPECTRUM SUMMATION AVERAGE PROGRAM

(N1 = 50)

```
 0 L0
 1 CL     2
 3 L1
 4 RA     0     1     2
 8 F      0     1
11 *-     1
13 A+     2
15 X>     2
17 #2     50    0
20 :      0     50
23 .
```

Cross spectrum

sum avg.

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 spectrums $G_{xx}(f)$'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
MULT* ENTER[1]	$G_{xx}(f)$ of current record	"	Forms $G_{xx}(f)$ of current record
- 1 ENTER	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	Result of dividing above difference by no. 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., no. of $G_{xx}(f)$'s averaged so far

[1]Replace this command with POLAR ENTER if you want to accumulate the average of voltage rather than power spectrums.

Auto spectrum stab. avg.

AUTO SPECTRUM STABLE AVERAGE PROGRAM (Cont'd)

PROGRAM COMMANDS	CONTENTS OF BLOCK 0	CONTENTS OF BLOCK 1	PURPOSE OF COMMAND
+1 ENTER	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	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 N1 ENTER	"	"	Loop back to label 1 N1 times, for N1 averages
END ENTER	"	"	Ends program

LISTING OF AUTO SPECTRUM STABLE AVERAGE PROGRAM

(N = 100)

```
 0 LØ
 1 CL     1
 3 L1
 4 RA     Ø     1
 7 F
 8 *-
 9 A-     1
11 :      Ø     Ø
14 X>     1
16 #1   100     Ø
19 .
```

Auto spectrum

stab. avg.

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]	$G_{xx}(f)$ of current record	"	Forms $G_{xx}(f)$ of current record
-1 ENTER	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	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]Replace this command with POLAR ENTER if you wish to average voltage rather than power spectrums.

Auto spectrum exp. avg.

AUTO SPECTRUM EXPONENTIALLY
DECAYING AVERAGE PROGRAM (Cont'd)

PROGRAM COMMANDS	CONTENTS OF BLOCK 0	CONTENTS OF BLOCK 1	PURPOSE OF COMMAND
+1 ENTER	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	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
JUMP 1 ENTER	"	"	Loop back to Label 1. Note: program keeps running until STOP key is pressed
END ENTER	"	"	Ends program

LISTING OF SPECTRUM
EXPONENTIALLY DECAYING AVERAGE PROGRAM

(K = 20)

```
 0 L0
 1 CL     1
 3 L1
 4 RA     0     1
 7 F
 8 *-
 9 A-     1
11 :      0    20
14 A+     1
16 X>     1
18 J1
19 .
```

Auto spectrum

exp. avg.

TIME ENSEMBLE AVERAGE

If a signal is buried in noise, and a time synchronization is available so the user knows the periodicity of the signal, the time averaging program is an efficient, very fast 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 (re-moves 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, dis-plays block 1, where sums are accum-ulating
+1 ENTER	Sum of cur-rent record plus past sums	"	Forms sum of current record plus past sum in block 0
STORE 1 ENTER	"	Sum of cur-rent record plus past sum	Stores new sum in block 1 for next pass

Time avg.

TIME ENSEMBLE AVERAGE PROGRAM (Cont'd)

PROGRAM COMMANDS	CONTENTS OF BLOCK 0	CONTENTS OF BLOCK 1	PURPOSE OF COMMAND
COUNT 1 SPACE N1 ENTER	New sum	New sum	Loop back to target label 1 N1 times for N1 sums
÷ 0 SPACE N1 ENTER	Average of N1 sums	"	Forms average of N1 sums
END ENTER	"	"	Ends program

LISTING OF TIME ENSEMBLE AVERAGE PROGRAM

(N1 = 50)

```
 0 LØ
 1 CL     1
 3 L1
 4 RA     Ø     1
 7 A+     1
 9 X>     1
11 #1     5Ø    Ø
14 :      Ø     5Ø
17 .
```

NORMALIZING A HISTOGRAM TO UNIT AREA

In the program to normalize a histogram to unity, we see how the 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.

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

Normalized histogram

NORMALIZING A HISTOGRAM TO UNIT AREA (Cont'd)

PROGRAM COMMANDS	CONTENTS OF BLOCK 0	CONTENTS OF BLOCK 1	PURPOSE OF COMMAND
$\int$ 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

LISTING OF NORMALIZING HISTOGRAM PROGRAM

(Block size = 512)

```
 0 X>    1
 2 $     1
 4 ←     1   511
 7 CL    1     1    511
11 $     1
13 :     1
15 .
```

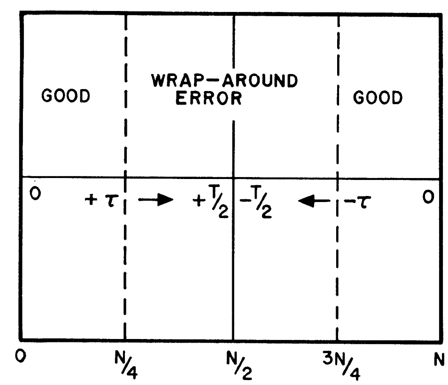
Normalized histogram

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, beginning page 5-22.

The subroutine is designed as follows: For a cross correlation, the data is read into block 0 and block 1. For an auto correlation, the single set of data is read into block 0, then stored in 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 block 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:



Correlation

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. The user can set the ORIGIN switch to CENTER in order 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 affectively makes the length of the integral in the function greater than T.

AUTO[1] AND CROSS-CORRELATION SUBROUTINE

(Eliminates wrap-around error; for discussion of this error, see page 5-22.)

PROGRAM COMMANDS	CONTENTS OF BLOCK 0	CONTENTS OF BLOCK 1	PURPOSE OF COMMAND
LABEL L ENTER			Establishes initial label point
ANALOG IN 0 SPACE 1 SPACE 0 ENTER[1]	Current time record from channel A	Current time record from channel B	Input data
CLEAR 1 SPACE 0 SPACE N/4 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 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

[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

AUTO AND CROSS-CORRELATION SUBROUTINE (Cont'd)

PROGRAM COMMANDS	CONTENTS OF BLOCK 0	CONTENTS OF BLOCK 1	PURPOSE OF COMMAND
CLEAR 0 SPACE N/4 SPACE 3/4 (N) 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.			

LISTING OF CROSS CORRELATION SUBROUTINE

(N = 512)

```
 Ø L9
 1 RA     Ø     1     Ø
 5 CL     1     Ø   128
 9 CL     1   384   512
13 CR     1
15 CL     Ø   128   384
19 •
```

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 or Y[1]. The ensemble estimates are summed as in the summation average programs, although if the user desires, he 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 spectrums 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 spectrums 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 ($\overline{G_{YY}}$).

Two sources of computational error should be kept in mind when computing transfer and coherence functions. Both tend to occur at those frequencies where the value of the power spectrum becomes very small, for example, at the high end of the filter roll-off curve. The first source of errors occurs as follows: As the power spectrum values get very small, they are in effect truncated from the right because the Computer uses a data word of finite length, i.e., 16 bits. Therefore the numerator or denominator of the transfer or coherence function may be reduced to numbers only 4 or 5 bits long.

Transf. & coher. functions

---

[1] Roth, Peter, "Digital Fourier Analysis", Hewlett-Packard Journal, June 1970, pp. 7-8.

As a result, the accuracy of these numbers is also diminished. Thus it can happen that the denominator becomes smaller than the numerator and produces a divide overflow (as indicated by the WHAT? light), or a coherence function greater than 1 (which is not valid), or an erroneous and noisy transfer function. Note that these errors only occur for low power spectrum values. The second source of error also is caused by low power spectrum values, and occurs when the randomness in the numerator and the denominator are not the same, so that the denominator may go to 0, producing a divide overflow. To remedy these errors, simply clear out, before dividing, by using the CLEAR command, the numerator of those channels that are causing the error. It is important that only the numerator channels be cleared. This in effect puts zeroes in those channels, and hence a 0 in the final function at these frequencies.

LISTING OF COMBINED TRANSFER AND COHERENCE FUNCTION PROGRAM

(N1 = 10)

```
 Ø LØ
 1 CL    1
 3 CL    2
 5 CL    3
 7 L1
 8 RA    4    5    1
12 F     4    5
15 CL    4    Ø
18 CL    5    Ø
21 X<    5
23 *-    4
25 A+    1
27 X>    1
29 X<    4
31 *-
32 A+    2
34 X>    2
36 X<    5
38 *-
39 A+    3
41 X>    3
43 #1   1Ø    Ø
46 X<    1
48 :     2
5Ø X>    4
52 TP    4
54 X<    1
56 *-
57 :     Ø    2
6Ø :     Ø    3
63 .
```

Transf. & coher. functions

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 label point
CLEAR 1 ENTER		Cleared					Initializes block 1 (removes old data)
CLEAR 2 ENTER			Cleared				Initializes block 2 (removes old data)
CLEAR 3 ENTER				Cleared			Initializes block 3 (removes old data)
LABEL 1 ENTER		Sum of past $G_{YX}(f)$'s. 0 first time.	Sum of past $G_{XX}(f)$'s. 0 first time.	Sum of past $G_{YY}(f)$'s. 0 first time.			Establishes target point for power spectrum summation
ANALOG IN 4 SPACE 5 SPACE 1 ENTER		"	"	"	Current time record from channel A (input, X)	Current time record from channel B (output, Y)	Data input
F 4 SPACE 5 ENTER		"	"	"	Fourier transform of channel A record	Fourier transform of channel B record	Obtain Fourier transform of data
CLEAR 4 SPACE 0 ENTER		"	"	"	Fourier transform of channel A record minus dc value	"	Clears dc value from channel A record
CLEAR 5 SPACE 0 ENTER		"	"	"	"	Fourier transform of channel B record minus dc value	Clears dc value from channel B record
LOAD 5 ENTER	Fourier transform of channel B record minus dc value	"	"	"	"	"	Prepare for conjugate multiply in block 0
MULT* 4 ENTER	Cross power spectrum ($G_{YX}(f)$) of channel A and B records	"	"	"	"	"	Obtain $G_{YX}(f)$

COMBINED TRANSFER AND COHERENCE FUNCTION PROGRAM (Cont'd)

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
+1 ENTER	Sum of current plus past $G_{YX}(f)$'s	Sum of past $G_{YX}(f)$'s. 0 first time.	Sum of past $G_{XX}(f)$'s. 0 first time.	Sum of past $G_{YY}(f)$'s. 0 first time.	Fourier transform of channel A record minus dc value	Fourier transform of channel B record minus dc value	Add current $G_{YX}(f)$ to sum of past $G_{YX}(f)$'s
STORE 1 ENTER	"	Sum of current plus past $G_{YX}(f)$'s	"	"	"	"	Store results of current pass for next pass
LOAD 4 ENTER	Fourier transform of channel A record minus dc value	"	"	"	"	"	Prepare for conjugate multiply in block 0
MULT* ENTER	Auto power spectrum of input ($G_{XX}(f)$)	"	"	"	"	"	Form auto power spectrum of input, $G_{XX}(f)$
+2 ENTER	Sum of current plus past $G_{XX}(f)$'s	"	"	"	"	"	Add current $G_{XX}(f)$ to sum of past $G_{XX}(f)$'s
STORE 2 ENTER	"	"	"	"	"	"	Store results of current pass for next pass
LOAD 5 ENTER	Fourier transform of channel B record (output, Y)	"	"	"	"	"	Prepare for conjugate multiply in block 0
MULT* ENTER	Auto power spectrum of output ($G_{YY}(f)$)	"	"	"	"	"	Form auto power spectrum of output, $G_{YY}(f)$
+3 ENTER	Sum of current plus past $G_{YY}(f)$'s	"	"	"	"	"	Add current $G_{YY}(f)$ to sum of past $G_{YY}(f)$'s

COMBINED TRANSFER AND COHERENCE FUNCTION PROGRAM (Cont'd)

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
STORE 3 ENTER	Sum of current plus past $G_{YY}(f)$'s	Sum of current plus past $G_{YX}(f)$'s	Sum of past $G_{XX}(f)$'s. 0 first time.	Sum of current plus past $G_{YY}(f)$'s	Fourier transform of channel A record minus dc value	Fourier transform of channel B record minus dc value	Store results of current pass for next pass
COUNT 1 SPACE N1 ENTER	"	"	"	"	"	"	Loop back to target label 1 N1 times
LOAD 1 ENTER	Sum of current plus past $G_{YX}(f)$'s	"	"	"	"	"	Prepare for division of G_{YX} by G_{XX}
÷ 2 ENTER	Transfer function $H(f) = \frac{\overline{G_{YX}(f)}}{\overline{G_{XX}(f)}}$	"	"	Sum of past $G_{YY}(f)$'s. 0 first time.	"	"	Obtain transfer function
STORE 4 ENTER	Transfer function $H(f) = \frac{\overline{G_{YX}(f)}}{\overline{G_{XX}(f)}}$	"	"	"	Transfer function	"	Store transfer function in block 4
POLAR 4 ENTER	Transfer function $H(f) = \frac{\overline{G_{YX}(f)}}{\overline{G_{XX}(f)}}$	"	"	"	Transfer function in polar coordinates	"	Convert transfer function to polar coordinates (magnitude and phase) - optional
LOAD 1 ENTER	Sum of current plus past $G_{YX}(f)$'s	"	"	"	"	"	Prepare for conjugate multiply in block 0
MULT* ENTER	$\lvert G_{YX}(f)\rvert^2$	"	"	"	"	"	Obtain $\lvert G_{YX}(f)\rvert^2$ in block 0

COMBINED TRANSFER AND COHERENCE FUNCTION PROGRAM (Cont'd)

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
÷ 0 SPACE 2 ENTER	$\frac{\|G_{YX}(f)\|^2}{G_{XX}(f)}$	Sum of current plus past $G_{YX}(f)$'s	Sum of past $G_{XX}(f)$'s. 0 first time.	Sum of past $G_{YY}(f)$'s. 0 first time.	Transfer function in polar coordinates	Fourier transform of channel B record minus dc value	Obtain $\frac{\|G_{YX}(f)\|^2}{G_{XX}(f)}$ in block 0
÷ 0 SPACE 3 ENTER	Coherence function $\gamma^2 = \frac{\|G_{YX}(f)\|^2}{G_{XX}(f) \cdot G_{YY}(f)}$	"	"	"	"	"	Obtain coherence function in block 0
END ENTER	"	"	"	"	"	"	End program

REAL TIME (BUFFERED) ADC INPUT

In the 5450A with the 16K program only, it is possible to input a single channel ADC signal into one block (a buffer block) and simultaneously do an operation on another block. If the length of the operation in the other block is less than that of the input, then no data will be lost in successive input records, and a real time analysis can be performed.

There are several limitations to the process: (1) We have to give up a data block for the buffer; (2) It only can be used for single channel ADC input; (3) F_{max} can never be set higher than 5 kHz because above this frequence the ADC goes into a different processing mode. Because of the length of the processing time, the real time capability for power spectrums is limited typically to inputs with maximum frequencies between 150 and 300 Hz.

The buffer block is always the highest numbered block as follows:

Block Size	Input Buffer Data Block
4096	1
2048	3
1024	7
512	15
256	31
128	63
64	127

A special form of the analog input command is required for buffered input. It is:

ANALOG IN Block N1 SPACE 9 9 8 ENTER

where:

N1 is the data block into which the data is to be entered. Obviously N1 cannot be the buffer block.

The user must know the maximum frequency (F_{max}) in his input. From this he can compute the minimum data record length (T) as follows:

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

$$T = N\Delta t$$

where:

N is the block size
Δt is the sample interval

Real time input

Knowing T, the user can determine the processing time for all but the analog input command by writing a program consisting of these steps. For example in a power spectrum program, the steps would include everything from the label referenced by the COUNT command through the COUNT command itself, excluding the analog-in command. (The timing is facilitated by choosing a sufficiently large number of COUNT loops to time. An ordinary wrist watch with a second hand or stop watch is used, then the resulting figure is divided by the number of counts. Note: the time required to transfer data from the buffer block to the operation block is an additional, though small, factor, amounting to 100 μsec/point.) If the processing time computed above, is less than T, then the user can obtain real time processing for his data. If not, he can't, unless he is able to shorten the program.

Since in this mode both blocks are being processed, there can be no simultaneous display. If the user desires a display, he can insert one into the program using the special display command

DISPLAY	Block N1	SPACE	0	ENTER

where:

N1 is the data block to be displayed.

This will give one sweep of the display each time and then cause the program to continue. But the price is the additional processing time which further limits F_{max}. Note that the above display command is different from the normal display command which causes a pause in the program.

A typical buffered input program for a power spectrum might be as follows:

LABEL 1 ENTER

CLEAR 1 ENTER

LABEL 2 ENTER

ANALOG IN 0 998 ENTER

F ENTER

MULT* ENTER

+1 ENTER

STORE 1 ENTER

COUNT 2 SPACE 50 ENTER

END ENTER

Real time input

SECTION IX

WRITING ASSEMBLER LANGUAGE PROGRAMS FOR THE FOURIER ANALYZER

ABOUT THIS SECTION

This section provides information so that the user can write his own Assembler language programs for special processing of his data, in addition to that available on the Fourier Analyzer Keyboard. Pertinent addresses and data word formats are given so that the user can write his own ADC and Display unit drivers. A special command, called the Y command is described which permits the user to interface his own program with the Fourier program. Full details on HP Assembler language can be found in the Assembler/BCS Training Manual or the Computer Micromanual.

TWO TYPES OF PROGRAMS

The user may write two types of Assembler language programs for the Fourier Analyzer. The first type is used to process data which has been gathered by the Fourier program. In this case the user's program is entered after the data has been gathered, and erases part or all of the Fourier program. This type of program can be used with all three Fourier Analyzers: the 8K and 16K 5450A, and the 8K 5452A. The second type of program does not displace the Fourier program and will be primarily written for the 16K 5450A, because this system has space for 1472 ($2{,}700_8$) user-entered steps. This type of program employs a special command, called the Y command (described on page 9-10) which permits the user to automatically pass control from his Keyboard entered program to his own Assembler-language program(s) and then back. In the 8K programs there is no room for user Assembler program steps. However, the user can put program steps in data memory space, and thus expand this capability. Of course, data storage space is then correspondingly limited. This technique can also be used in the 16K 5450A if necessary.

DATA STORAGE

A complete chart of the 8K and 16K memories is given in Figure 9-1. This will provide a handy reference for the following discussion.

Data in the 8K and 16K memory is stored as 16-bit words in 2's complement format. In all systems the first data word is at address 2000_8. In the 8K program, the data memory goes to address $7{,}777_8$; in the 16K program data goes to 21777_8. Data blocks are sequentially located beginning with data block 0 and running through data block (K/N)-1, where K is the size of the data memory in decimal and N is the block size in decimal.

Figure 9-1. Memory locations pertinent to user-written Assembler language programs (all addresses in octal)

5450A/5452A 8K MEMORY

17, 777 17, 700	Basic Binary Loader, 64_{10} words (100_8)
	Fourier Program
7777	Data block $\frac{3,072_{10}}{N} - 1$ (N = block size)
	Data block 1, etc.
2000	Data block 0
	Fourier Program
622	Y command entry: must contain 1st address in user prog.
457	Exit from user program to Fourier program
447	Contains address of 1st frequency code word
441	SIZE, = block size
440	POW, SIZE = 2^{POW}
437	MSZ, = negative of SIZE
425	Contains address of 1st scale factor word
424	N3 in Y command
423	N2 in Y command
422	N1 in Y command
421	Y in Y command X in Y command

Assembler

prog. memory

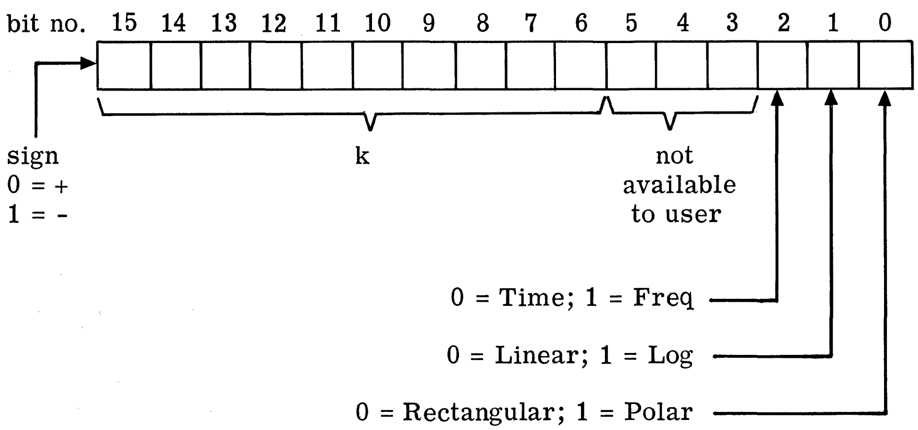
Figure 9-1. Memory locations pertinent to user-written Assembler language programs (all addresses in octal)

5450A/5452A 16K MEMORY

37,777 37,700	Basic Binary Loader, 64_{10} words (100_8)
37,677 35,000	Space for user-written programs
	Fourier Program
21,777	Data block $\frac{8192_{10}}{N} - 1$ (N = block size) etc.
	Data block 2
	Data block 1
2000	Data block 0
1630 1560	Linkages for user-written programs
	Fourier Program
632	Y command entry: must contain 1st address in user prog.
457	Exit from user program to Fourier program
447	Contains address of 1st frequency code word
441	SIZE, = block size
440	POW, SIZE = 2^{POW}
437	MSZ, = negative of SIZE
425	Contains address of 1st scale factor word
424	N3 in Y command
423	N2 in Y command
422	N1 in Y command
421	Y in Y command X in Y command

SCALE FACTOR WORD

With each data block, two Computer words are associated: one contains the scale factor and coordinate code, the other contains the frequency code. The address of the first scale factor word is contained in address 425_8 (all three systems). Scale factor words are then given in sequence from that address on, corresponding to the data block sequence. Thus the scale factor for data block 0 is at the address given in address 425_8; the scale factor word for data block 1 is given in the next address, etc. The format of the scale factor words is as follows:



The scale factor is the "k" which immediately follows the SF at the beginning of any punched tape input or output. It is the floating point power, in radix 2, of the corresponding data block. Thus, for example, if the block has a scale factor of 2^{10}, then bits 7 and 9 would be 1's, and the remainder of bits 6 through 15 would be 0's. The scale factor displayed by the Display Unit of course is in radix 10; likewise the scale factor entered with a manual (Keyboard) data entry. Decimal-to-binary, binary-to-decimal and radix conversion for these scale factors is done by the Fourier program. The coordinate code is given in bits 0, 1, and 2 of the scale factor word. Note that these bits spell out in binary the coordinate code which we have seen throughout this manual:

Coordinate Code	Time	Freq	Rect	Polar	Log	Linear
0	x		x			x
2	x		x		x	
4		x	x			x
5		x		x		x
7		x		x	x	

Scale factor word

Thus, time rectangular linear coordinates are expressed by a coordinate code of 0, or 0's in bits 0, 1 and 2; frequency rectangular linear coordinates are expressed by a coordinate of 4, or a 1 in bit

2 and 0's in bits 0 and 1. Regardless of what shifting of the scale factor the user might do, he should be certain he does not in any way truncate the 10 bits or improperly modify the coordinate code.

FREQUENCY CODE WORD

The frequency code word is a 16-bit word in 2's complement form which can have values ranging from 0 through +32, 767.

bit no.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

sign (bit 15)
0 = +
1 = -

These frequency codes have been seen at various places throughout this manual, for example page . The code has its principal application when data has been taken in by the ADC. In this case, it gives the SAMPLE MODE and MULTIPLIER switch settings. The user, however, can enter any number from 0 through +32, 767 for his own identification purposes when he is inputting data via paper tape or via manual Keyboard entry. The frequency code stays with the data block throughout all operations on it, and is never modified. The address of the first frequency code word is located at address 447_8. Frequency code words are then given in sequence from that address, the word for data block 0 being in that address, the word for data block 1 being in the next address, etc.

WRITING ADC DRIVERS

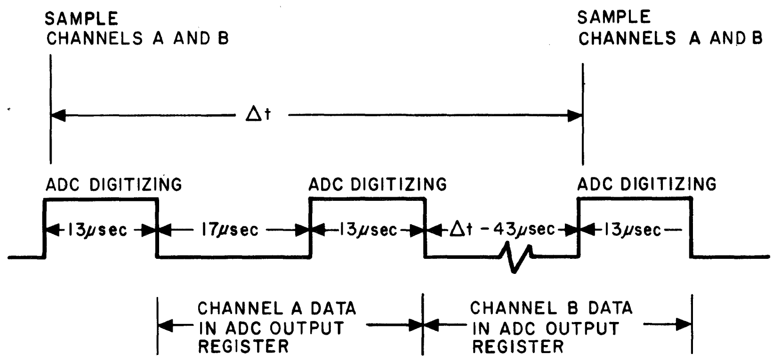
The user can write his own driver to utilize the ADC. The following information will assist him.

The ADC interface Microcircuit card is located in channel 14. The ADC is armed by issuing Set Control, Clear Flag commands to channel 14. After the Set Control command is given and after the trigger circuit is triggered, the ADC will begin to send data points back to the Computer. In the LINE mode the ADC is triggered by the next line frequency zero crossing. In the INTERNAL mode it is triggered by the A channel input as determined by the SLOPE and LEVEL control settings on the ADC panel. In the FREE RUN mode the Set Control signal also triggers the ADC so that in this mode a data response comes back within 12 μsec of Set Control. In the EXTernal mode, the EXTernal trigger starts the process.

The Set Control command sets the ADC Encode. The Microcircuit card is bussed so that the device flag from the ADC does not clear the Encode signal through the ADC. This means that once the ADC starts sending data, the only way it can be turned off is by a Clear Control command. When a Clear Control is received, the ADC does a self-calibration and cannot be turned on again in less than 50 μsec.

Freq. code word

In the single channel mode, data points enter the Microcircuit card register on the ADC at the rate set by the ADC sampling control. In the dual channel mode, data points enter the channel in pairs, first the A channel point, then the B channel point, then A, etc. The A word remains in the register 30 μsec ± 3, as shown in Figure 9-2. The B word enters the register and remains until the next A word. Thus, the distance between A words is the sample interval, Δt, as set by the ADC control, but the distance between the A word and the B word is a fixed quantity independent of the sampling rate.



ALL TIMES ± 10 %

ADC Data Word

The ADC data word is a 10-bit word in 2's complement form, the first bit being the sign.

bit no. 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

sign
0 = +
1 = -

Channel B attenuator setting code in binary (bits 5–3):
0 = CHECK
:
7_{10} = 10

Channel A attenuator setting code in binary (bits 2–0):
0 = CHECK
:
7_{10} = 10

ADC data word

The attenuator setting code in bits 0 through 5 are transmitted with every ADC word, single or dual channel. Consequently before these words are stored in the Computer memory, bits 0 through 5 must be removed so that an erroneous value will not result.

The ADC is adjusted so that 0 volts is on the edge between 0 and -1; that is, 0 volts has equal likelihood of producing all bits equal 0 or all bits equal 1 in the ADC word. A plus full-scale-overload is indicated by a 0 sign bit and all 1's in the data; a negative full scale overload is indicated by a 1 sign bit and all 0's.

WRITING DISPLAY UNIT DRIVERS

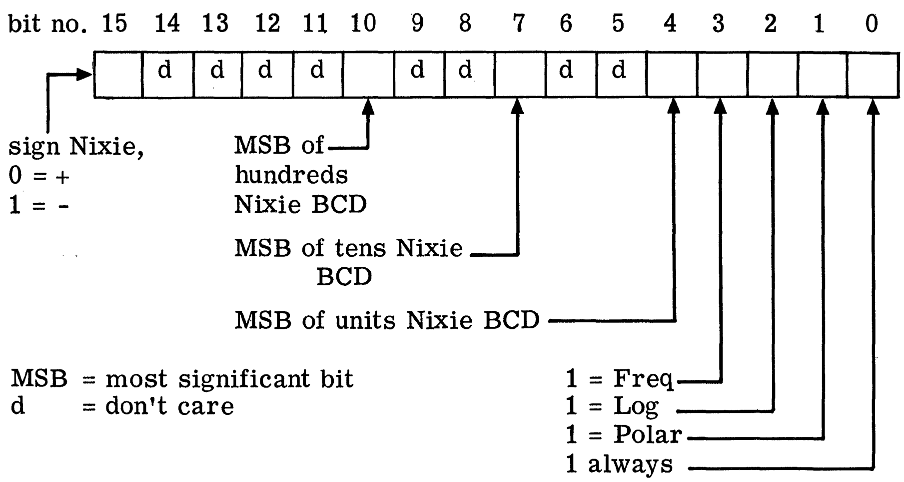
The user can also write his own Display Unit Drivers. The following information will be of assistance.

The Display Unit is interfaced through the Microcircuit card in channel 15. To output a word to the Microcircuit card from the Computer, the Set Control, Clear Flag commands are given. Then 25 μsec are required to process the word and return the device flag, after which the Display Unit is ready for the next word.

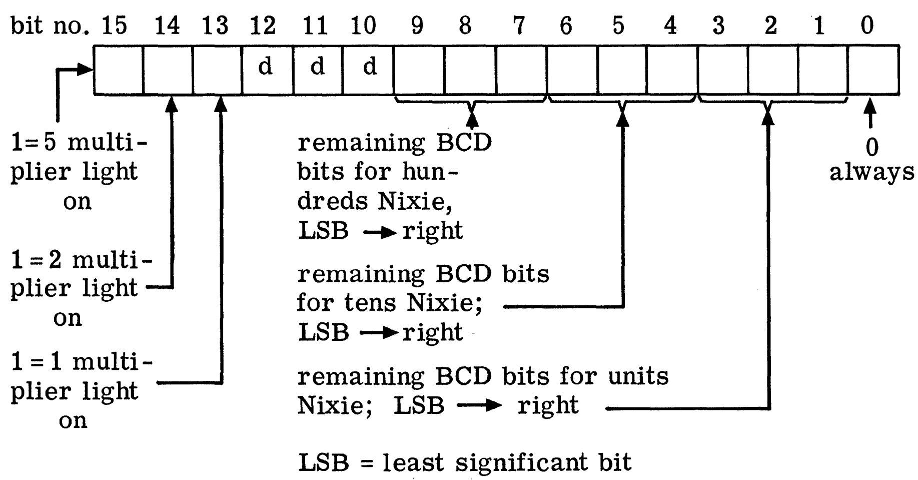
Display Unit Word - From Computer

Display Unit words coming from the Computer have the format shown on next page. The first data word is identified by a 1 in bit 0. The first and second words contain scale factor information. The scale factor digits displayed on the Nixie tubes are defined in BCD: the most significant bits are in word 1, the remaining in word 2. The third word is then the first X point and the fourth is the first Y point; X's and Y's are then sent alternatively until the end of the data block. Each data word to the Computer is an 11-bit word in 2's complement form, left adjusted. The first bit is the sign; 00 is located in the center of the screen. When the ORIGIN switch is used to shift the origin to the left of the screen, this is done via software.

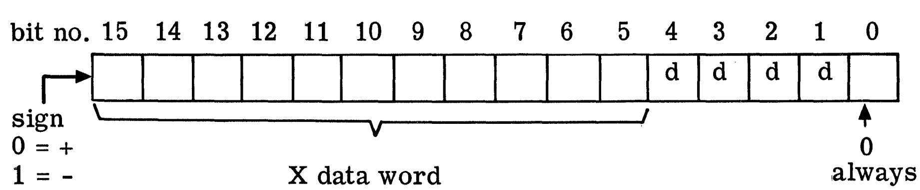
WORD 1



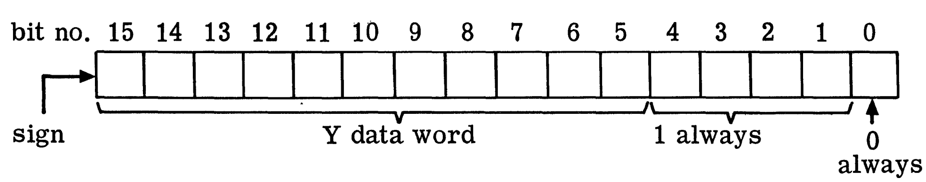
WORD 2



WORD 3



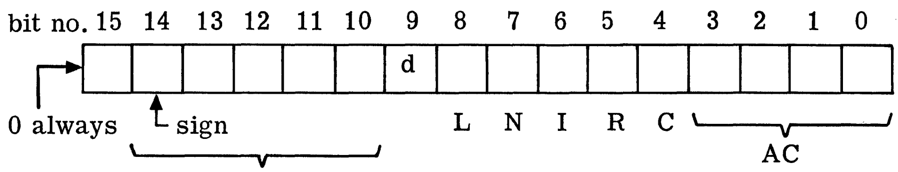
WORD 4



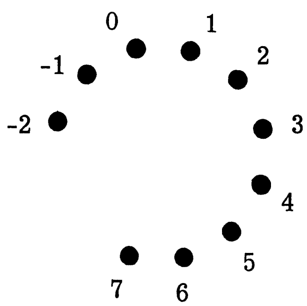
Display Unit word

Display Unit Word-To Computer

When the Display Unit sends back the device flag it also sends a data word back indicating the Display Unit control settings. The format of this word is as shown below:



VC = Vertical SCALE Switch change. Binary number equal to number of upscale or downscale step changes on SCALE switch.



Position	no. of changes / bit no.	sign 15	14	3 13	2 12	1 11	0 10
-2	. . . .	0	1	0	0	1	0
-1	. . . .	0	1	0	0	0	1
0	. . . .	0	0	0	0	0	0
1	. . . .	0	0	0	0	0	1
2	. . . .	0	0	0	0	1	0
3	. . . .	0	0	0	0	1	1
4	. . . .	0	0	0	1	0	0
5	. . . .	0	0	0	1	0	1
6	. . . .	0	0	0	1	1	0
7	. . . .	0	0	0	1	1	1

If L = 1, horizontal log display
If N = 1, complex display
If I = 1, imaginary part display
If R = 1, real part display
If C = 1, origin at left of screen

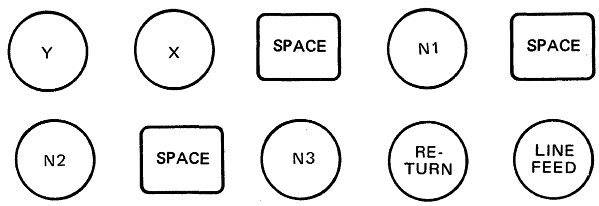
AC = Phase angle scale code, as follows:

bit no.	3	2	1	0
100°/div	0	1	0	0
50°/div	1	0	0	0
45°/div	0	0	0	0
10°/div	0	0	1	0
RAD/div	1	1	0	0

Y COMMAND

The Y command transfers control from the Fourier program to the user-written Assembler-language program or programs. If one of the user's programs occurs as part of a program which the user has entered and stored via the Keyboard, then the Y command must appear at the point where the user wants to initiate his own program. After the conclusion of the user program(s), control is passed to the next step in the Keyboard-entered program.

The Y command, which can only be entered on the Teleprinter, has the following form:



where:

X is a space or any letter or integer. If X is a space, then the following SPACE may be omitted.

N1, N2 and N3 are optional program parameters having values from 0 to +32, 767.

Elements may be defaulted, beginning with element N3 and working back through X. Default value in each case is 0.

When the Keyboard entered program comes to the Y command, it jumps indirect to base page address 622_8 in the case of the 8K programs or base page address 632_8 in the case of the 16K program. In these addresses the user may place the address of the first step of his program. The program then proceeds through the user's program. At the conclusion of the user's program, there must be a Jump to base page address 457_8 (in both 8K and 16K programs). From there, control will return to the Fourier program. The Analyzer program will then continue with the stored program or go to the READY state.

The 5450A 16K memory has the largest space for user-written programs, amounting to 1472 ($2,700_8$) words, that is, from address $35,000_8$ through $37,677_8$, plus 40 (50_8) words for linkages available on the base page from addresses 1560_8 to 1630_8. See Figure 9-1. In the 8K program there is no room for program words. However, the user can put program words in data memory space and thus expand the amount of program memory available. This applies to the 16K 5450A system also. Of course, data storage space is then correspondingly limited.

Y command

The terms of the Y command are contained in addresses 421_8 through 424_8 as shown in Figure 9-1. The Y command indirect address is located in address 622_8 in the 8K programs, and in address 632_8 in the 16K program.

A NOTE ON THE INTERRUPT SYSTEM

In all Fourier programs, the interrupt system is always on when the Fourier program is on. If, in the course of a user Assembler language program, the interrupt system is turned off, then before control is returned to the Fourier program, all I/O channels must be turned off (with the CLC, 00 command) and the interrupt turned on (with the STF, 00 command).

OPTION 900

Option 900 is a special software configuration that enables the user with a 16K 5450A to expand the space available for his own Assembler language programs to 7,959 words ($17,427_8$). The software is created at the factory by assembling the 8K Fourier program with a 16K Basic Control System and inserting the Y command capability. The trade-off is that the available data memory in the 16K system is now reduced to that of an 8K system, namely, 3,072 words.

With Option 900, the following addresses are available for user-written Assembler programs: 1100 to 1630_8 on the base page; and 21,000 through $37,677_8$ in upper core. All other pertinent addresses (except for the Basic Binary Loader) are the same as in the 8K memory chart in Figure 9-1.

APPENDIX 1: TELEPRINTER SYMBOLS

b = space

Non-alphabetical symbols are listed at end.

No default values for Keyboard commands are shown. These can be found under the detailed explanations of each command in Sections VI and VII.

Teleprinter Symbol	Keyboard Command
A+	Block Add: Add block N1 to block 0. Leave the result in block 0. + N1 ENTER
A-	Block Subtract: Subtract block N1 from block 0. Leave the result in 0. - N1 ENTER
BS	Block Size: Set block size to N1. BLOCK SIZE N1 ENTER
CL	Clear Total Block: Clear block N1. CLEAR N1 ENTER Clear Partial Block: Clear block N1 from channel N2 to N3. CLEAR N1 SPACE N2 SPACE N3 ENTER
CR	Correlation: Correlate block 0 with block N1 and leave the result in block 0. CORR N1 ENTER
CV	Convolution: Convolve block 0 with block N1 and leave the result in block 0. CONV N1 ENTER
/D	Deleting Program Steps: Delete steps N1 through N2. DELET N1 SPACE N2 ENTER
Db	Display: Display block N1. DISPLAY N1 ENTER

TELEPRINTER SYMBOLS (Cont'd)

Teleprinter Symbol	Keyboard Command
Fb	Fourier Transform: Fourier transform blocks N1 and N2. F N1 SPACE N2 ENTER
F^{-1}	Inverse Fourier Transform: Inverse Fourier transform blocks N1 and N2. F^{-1} N1 SPACE N2 ENTER
H1	Interval-centered Hanning Window: Hann block N1 with the interval-centered Hanning window. HANN I N1 ENTER
H0	Origin-centered Hanning Window: Hann block N1 with origin-centered Hanning window. HANN 0 N1 ENTER
/I	Inserting Program Steps: Insert after step N1 any number of steps. INSRT N1 ENTER
I0	Binary Output: Binary output from block N1 through interface channel 16 and from N2 through interface channel 17. BINARY 0 N1 SPACE N2 ENTER
I1	Binary Input: Binary input into block N1 through interface channel 16 and to interface channel 17. BINARY 1 N1 SPACE N2 ENTER
JN	Jump: Jump unconditionally to label N. JUMP N ENTER
Kb	Keyboard Input Command, Block Fill: Keyboard one channel value into block N1 from channels N2 to N3. KEYBOARD N1 SPACE N2 SPACE N3 ENTER

TELEPRINTER SYMBOLS (Cont'd)

Teleprinter Symbol	Keyboard Command
Kb (cont'd)	Keyboard Input Command, Point-by-Point Fill: Keyboard successive values into block N1 starting with channel N2. KEYBOARD N1 SPACE N2 ENTER Keyboard Scale Factor Input: Input amplitude scale factor N1, coordinate code N2, frequency code N3. KEYBOARD N1 SPACE N2 SPACE N3 ENTER Keyboard Data Input, Time: Input real part, N1 N1 ENTER Keyboard Data Input, Frequency: Input real part/magnitude, N1; imaginary part/phase N2. N1 SPACE N2 ENTER
/L	List an entire program from line 0. LIST ENTER List a Single Line, N1: LIST N1 ENTER List a Range of Lines from Line N1 to Line N2: LIST N1 SPACE N2 ENTER List from a Certain Line N1 to the end of the Program: LIST N1 SPACE (99 for 8K machines, 199 for 16K machines) ENTER
LN	Label: Label N, where N is an integer from 0 to 9 or a letter. LABEL N ENTER
> N	Sub-Routine Jump: Jump to sub-routine labeled N. SUB JUMP N ENTER

TELEPRINTER SYMBOLS (Cont'd)

Teleprinter Symbol	Keyboard Command
#N	Loop: Loop back to label N, N1 times. COUNT N SPACE N1 SPACE N2 ENTER
Pb	Punch Output: Punch out block N1 from channel N2 to N3. PUNCH N1 SPACE N2 SPACE N3 ENTER
PR	Protect: Set program protect range from line N1 to line 99 (in 8K machines) or line 199 (in 16K machines). PROT N1 ENTER
RA	Single-channel Analog Input: Analog input into block N1 from input A. Display block N2. ANALOG N N1 SPACE N2 ENTER
	Dual-Channel Analog Input: Analog input into block N1 from input A, to block N2 from input B. Display block N3. ANALOG IN N1 SPACE N2 SPACE N3 ENTER
/R	Replacing Program Steps: Replace steps N1 through N2 with any number of steps. RPLAC N1 SPACE N2 ENTER
Rb	Photoreader Input: Input data via Photoreader into block N1 from channels N2 to N3. PHOTOREADER N1 SPACE N2 SPACE N3 ENTER
RH	Histogram: Histogram from input A into block N1. HISTOGRAM N1 ENTER
TE	Exponential Magnitude: Convert the magnitude (or real part) of block N1 from log to linear. EXP MAG N1 ENTER

TELEPRINTER SYMBOLS (Cont'd)

Teleprinter Symbol	Keyboard Command
TL	Log Magnitude: Convert the magnitude (or real part) of block N1 to log. LOG MAG N1 ENTER
TP	Rectangular to Polar Coordinate Change: Convert block N1 from rectangular to polar coordinates. POLAR N1 ENTER
TR	Polar to Rectangular Coordinate Change: Convert block N1 from polar to rectangular coordinates. RECT N1 ENTER
Wb	Print Output: Print out block N1 from channels N2 to N3. PRINT N1 SPACE N2 SPACE N3 ENTER
Xb	Interchange: Interchange block 0 with block N1. INTERCHNG N1 ENTER
X>	Store: Store data block 0 in block N1. STORE N1 ENTER
X<	Load: Load data block N1 into block 0. LOAD N1 ENTER
:b	Block Divide: Divide block 0 by block N1. Leave the result in 0. ÷ N1 ENTER Integer Divide: Divide block N1 by the integer N2 (N2 positive only). ÷ N1 SPACE N2 ENTER
*b	Block Multiply: Multiply block 0 by block 1. Leave the result in 0. MULT N1 ENTER Integer Multiply: Multiply block N1 by the integer N2 (N2 positive only). MULT N1 SPACE N2 ENTER

TELEPRINTER SYMBOLS (Cont'd)

Teleprinter Symbol	Keyboard Command
*_	Block Conjugate Multiply: Multiply block 0 by the complex conjugate of block N1. Leave the result in 0. MULT* N1 ENTER
←	Rotate or Shift: Rotate or shift block N1, N2 channels to the left. ← N1 SPACE N2 ENTER
$b	Integrate: Integrate block N1 from channel N2 to N3. Result in block N1. ∫ N1 SPACE N2 SPACE N3 ENTER
%b	Differentiate: Differentiate block N1. d/dx N1 ENTER
b	Sub-Routine Return: Return to next program step after sub-routine jump step. SUB RTRN ENTER
.	Program End: End of program. END ENTER
/?	Interrogate Pointer and Protected Locations: List pointer location and start of protected range. ? ENTER
/b	Terminate an Edit or Keyboard Entry: Terminate and close edit file. TERM ENTER
/.	Pointer: Set program pointer to line N1. POINT N1 ENTER

APPENDIX 2: WHAT? MESSAGES ON TELEPRINTER

When the WHAT? light on the Keyboard comes on, this is an indication that an illegal command has been given. In addition, if the Teleprinter switch is set to LINE, an error message will be typed. The most common causes of errors are as follows:

1. N1, N2 or N3 in the command is an illegal block or channel for the block size set.
2. N1, N2 or N3 is an illegal use of a negative integer.
3. The symbol used for element 0 is incorrect. This would apply primarily to the keying-in of commands from the Teleprinter rather than from the Fourier Analyzer Keyboard.

Some additional causes of WHAT? messages are given below in alphabetical order by the identifying symbol.

Printout on Teleprinter

BS WHAT? (BLOCK SIZE)

1. N1 is an illegal block size. Note: smallest allowable block size for 8K programs is 128.

CL WHAT? (CLEAR)

1. N1 is a non-existent block for the block size set.
2. N2 or N3 is an illegal channel number for the block size set.

CR WHAT? (CORR)

1. Block 0 and N1 are in different domains (frequency and time).

CV WHAT? (CONV)

1. Block 0 and N1 are in different domains (frequency and time).

I WHAT? (BINARY)

1. The block size or handshake bit received from the remote device was not correct.
2. An illegal number was received.

J WHAT? (JUMP)

1. Label N does not exist.

WHAT? MESSAGES ON TELEPRINTER (Cont'd)

RA WHAT? (ANALOG IN)

1. ADC is set in DUAL. Single channel command has been given.
2. ADC is set in A. A. mode. Dual channel command has been given.

RH WHAT? (HISTOGRAM)

1. Sample rate greater than 5 kHz ($\Delta t < 200\ \mu s$).
2. ADC in DUAL mode.

W WHAT? (PRINT)

1. N2 or N3 exceeds block size for time domain or one-half block size for frequency domain.

WHAT? (COUNT)

1. No label N exists.

:WHAT? (÷)

1. In block divide, 0 and N1 are in different domains.
2. Overflow has occurred in a pair of channels being divided during a block division. This WHAT? signal is different from all others in that it occurs after the operation (division of both blocks) has been completed. All other WHAT? signals occur before the operation is completed, and halt the operation.

< WHAT? (SUB JUMP)

1. Label N does not exist.

> WHAT?

1. Label N does not exist.
2. Sub-routines nested more than 3 deep.

< WHAT? (SUB RTRN)

1. No sub-routine entry.
2. Sub-routines nested more than 3 deep.

←WHAT? (←)

1. N2 larger than the block size.

$ WHAT? ($\int$)

1. N2 or N3 exceeds the block size.
2. N2 or N3 exceeds one-half the block size for the frequency domain.

INDEX

-A-

-B-

-C-

-C- (Cont'd)

-F-

-G-

-H-

-I-

-I- (Cont'd)

-O-

-P-

-R-

-R- (Cont'd)

-T- (Cont'd)

HEWLETT·PACKARD

ELECTRONIC INSTRUMENTATION SALES AND SERVICE

UNITED STATES, CENTRAL AND SOUTH AMERICA, CANADA

UNITED STATES

ALABAMA
P.O. Box 4207
2003 Byrd Spring Road S.W.
Huntsville 35802
Tel: (205) 881-4591
TWX: 810-726-2204

ARIZONA
3009 North Scottsdale Road
Scottsdale 85251
Tel: (602) 945-7601
TWX: 910-950-1282

5737 East Broadway
Tucson 85716
Tel: (602) 298-2313
TWX: 910-952-1162

CALIFORNIA
1430 East Orangethorpe Ave.
Fullerton 92631
Tel: (714) 870-1000

3939 Lankershim Boulevard
North Hollywood 91604
Tel: (213) 877-1282
TWX: 910-499-2170

1101 Embarcadero Road
Palo Alto 94303
Tel: (415) 327-6500
TWX: 910-373-1280

2220 Watt Ave.
Sacramento 95825
Tel: (916) 482-1463
TWX: 910-367-2092

1055 Shafter Street
San Diego 92106
Tel: (714) 223-8103
TWX: 910-335-2000

COLORADO
7965 East Prentice
Englewood 80110
Tel: (303) 771-3455
TWX: 910-935-0705

CONNECTICUT
508 Tolland Street
East Hartford 06108
Tel: (203) 289-9394
TWX: 710-425-3416

111 East Avenue
Norwalk 06851
Tel: (203) 853-1251
TWX: 710-468-3750

DELAWARE
3941 Kennett Pike
Wilmington 19807
Tel: (302) 655-6161
TWX: 510-666-2214

FLORIDA
P.O. Box 24210
2806 W. Oakland Park Blvd.
Ft. Lauderdale 33307
Tel: (305) 731-2020
TWX: 510-955-4099

P.O. Box 20007
Herndon Station 32814
621 Commonwealth Avenue
Orlando
Tel: (305) 841-3970
TWX: 810-850-0113

GEORGIA
P.O. Box 28234
450 Interstate North
Atlanta 30328
Tel: (404) 436-6181
TWX: 810-766-4890

ILLINOIS
5500 Howard Street
Skokie 60076
Tel: (312) 677-0400
TWX: 910-223-3613

INDIANA
3839 Meadows Drive
Indianapolis 46205
Tel: (317) 546-4891
TWX: 810-341-3263

LOUISIANA
P.O. Box 856
1942 Williams Boulevard
Kenner 70062
Tel: (504) 721-6201
TWX: 810-955-5524

MARYLAND
6707 Whitestone Road
Baltimore 21207
Tel: (301) 944-5400
TWX: 710-862-0850

P.O. Box 1648
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
24315 Northwestern Highway
Southfield 48075
Tel: (313) 353-9100
TWX: 810-224-4882

MINNESOTA
2459 University Avenue
St. Paul 55114
Tel: (612) 645-9461
TWX: 910-563-3734

MISSOURI
11131 Colorado Ave.
Kansas City 64137
Tel: (816) 763-8000
TWX: 910-771-2087

2812 South Brentwood Blvd.
St. Louis 63144
Tel: (314) 962-5000
TWX: 910-760-1670

NEW JERSEY
W. 120 Century Road
Paramus 07652
Tel: (201) 265-5000
TWX: 710-990-4951

1060 N. Kings Highway
Cherry Hill 08034
Tel: (609) 667-4000
TWX: 710-892-4945

NEW MEXICO
P.O. Box 8366
Station C
6501 Lomas Boulevard N.E.
Albuquerque 87108
Tel: (505) 265-3713
TWX: 910-989-1665

156 Wyatt Drive
Las Cruces 88001
Tel: (505) 526-2485
TWX: 910-983-0550

NEW YORK
1702 Central Avenue
Albany 12205
Tel: (518) 869-8462
TWX: 710-441-8270

1219 Campville Road
Endicott 13760
Tel: (607) 754-0050
TWX: 510-252-0890

82 Washington Street
Poughkeepsie 12601
Tel: (914) 454-7330
TWX: 510-248-0012

39 Saginaw Drive
Rochester 14623
Tel: (716) 473-9500
TWX: 510-253-5981

1025 Northern Boulevard
Roslyn, Long Island 11576
Tel: (516) 869-8400
TWX: 510-223-0811

5858 East Molloy Road
Syracuse 13211
Tel: (315) 454-2486
TWX: 710-541-0482

NORTH CAROLINA
P.O. Box 5188
1923 North Main Street
High Point 27262
Tel: (919) 885-8101
TWX: 510-926-1516

OHIO
25575 Center Ridge Road
Cleveland 44145
Tel: (216) 835-0300
TWX: 810-427-9129

3460 South Dixie Drive
Dayton 45439
Tel: (513) 298-0351
TWX: 810-459-1925

1120 Morse Road
Columbus 43229
Tel: (614) 846-1300

OKLAHOMA
2919 United Founders Boulevard
Oklahoma City 73112
Tel: (405) 848-2801
TWX: 910-830-6862

OREGON
Westhills Mall, Suite 158
4475 S.W. Scholls Ferry Road
Portland 97225
Tel: (503) 292-9171
TWX: 910-464-6103

PENNSYLVANIA
2500 Moss Side Boulevard
Monroeville 15146
Tel: (412) 271-0724
TWX: 710-797-3650

1021 8th Avenue
King of Prussia Industrial Park
King of Prussia 19406
Tel: (215) 265-7000
TWX: 510-660-2670

RHODE ISLAND
873 Waterman Ave.
East Providence 02914
Tel: (401) 434-5535
TWX: 710-381-7573

TEXAS
P.O. Box 1270
201 E. Arapaho Rd.
Richardson 75080
Tel: (214) 231-6101
TWX: 910-867-4723

P.O. Box 22813
6300 Westpark Drive
Suite 100
Houston 77027
Tel: (713) 781-6000
TWX: 910-881-2645

231 Billy Mitchell Road
San Antonio 78226
Tel: (512) 434-4171
TWX: 910-871-1170

UTAH
2890 South Main Street
Salt Lake City 84115
Tel: (801) 487-0715
TWX: 910-925-5681

VERMONT
P.O. Box 2287
Kennedy Drive
South Burlington 05401
Tel: (802) 658-4455
TWX: 710-224-1841

VIRGINIA
P.O. Box 6514
2111 Spencer Road
Richmond 23230
Tel: (703) 285-3431
TWX: 710-956-0157

WASHINGTON
433-108th N.E.
Bellevue 98004
Tel: (206) 454-3971
TWX: 910-443-2303

***WEST VIRGINIA**
Charleston
Tel: (304) 768-1232

FOR U.S. AREAS NOT LISTED:
Contact the regional office nearest you: Atlanta, Georgia . . . North Hollywood, California . . . Paramus, New Jersey . . . Skokie, Illinois. Their complete addresses are listed above.

*Service Only

CENTRAL AND SOUTH AMERICA

ARGENTINA
Hewlett-Packard Argentina
S.A.C.e.I
Lavalle 1171 - 3
Buenos Aires
Tel: 35-0436, 35-0627, 35-0431
Telex: 012-1009
Cable: HEWPACKARG

BRAZIL
Hewlett-Packard Do Brasil
I.e.C Ltda.
Rua Coronel: Oscar Porto, 691
Sao Paulo - 8, SP
Tel: 288-7111
Cable: HEWPACK Sao Paulo

Hewlett-Packard Do Brasil
I.e.C. Ltda.
Rua da Matriz 29
Botafogo ZC-02
Rio de Janeiro, GB
Tel: 246-4417
Cable: HEWPACK Rio de Janeiro

CHILE
Héctor Calcagni y Cia, Ltda.
Bustos, 1932-3er Piso
Casilla 13942
Santiago
Tel: 4-2396
Cable: Calcagni Santiago

COLOMBIA
Instrumentacion
Henrik A. Langebaek & Kier Ltda.
Carrera 7 No. 48-59
Apartado Aereo 6287
Bogota, 1 D.E.
Tel: 45-78-06, 45-55-46
Cable: AARIS Bogota
Telex: 044-400

COSTA RICA
Lic. Alfredo Gallegos Gurdián
Apartado 3243
San José
Tel: 21-86-13
Cable: GALGUR San José

ECUADOR
Laboratorios de Radio-Ingenieria
Calle Guayaquil 1246
Post Office Box 3199
Quito
Tel: 12496
Cable: HORVATH Quito

EL SALVADOR
Electrónica
Apartado Postal 1589
27 Avenida Norte 1133
San Salvador
Tel: 25-74-50
Cable: ELECTRONICA San Salvador

GUATEMALA
Olander Associates Latin America
Apartado Postal 1226
Ruta 4, 6-53, Zona 4
Guatemala City
Tel: 63958
Cable: OLALA Guatemala City

MEXICO
Hewlett-Packard Mexicana, S.A. de C.V.
Moras 439
Col. del Valle
Mexico 12, D.F.
Tel: 5-75-46-49

NICARAGUA
Roberto Terán G.
Apartado Postal 689
Edificio Terán
Managua
Tel: 3451, 3452
Cable: ROTERAN Managua

PANAMA
Electrónico Balboa, S.A.
P.O. Box 4929
Ave. Manuel Espinosa No. 13-50
Bldg. Alina
Panama City
Tel: 30833
Cable: ELECTRON Panama City

PERU
Fernando Ezeta B.
Avenida Petit Thouars 4719
Miraflores
Casilla 3061
Lima
Tel: 45-2335
Cable: FEPERU Lima

PUERTO RICO
San Juan Electronics, Inc.
P.O. Box 5167
Ponce de Leon 154
Pda. 3-Pta. de Tierra
San Juan 00906
Tel: (809) 725-3342
Cable: SATRONICS San Juan
Telex: SATRON 3450 332

URUGUAY
Pablo Ferrando S.A.
Comercial e Industrial
Avenida Italia 2877
Casilla de Correo 370
Montevideo
Tel: 40-3102
Cable: RADIUM Montevideo

VENEZUELA
Hewlett-Packard De Venezuela C.A.
Apartado 50933
Caracas
Tel: 71.88.05, 71.88.69, 71.99.30
Cable: HEWPACK Caracas

FOR AREAS NOT LISTED, CONTACT:
Hewlett-Packard
INTERCONTINENTAL
3200 Hillview Ave.
Palo Alto, California 94304
Tel: (415) 326-7000
TWX: 910-373-1267
Cable: HEWPACK Palo Alto
Telex: 034-8461

CANADA

ALBERTA
Hewlett-Packard (Canada) Ltd.
11745 Jasper Ave.
Edmonton
Tel: (403) 482-5561
TWX: 610-831-2431

BRITISH COLUMBIA
Hewlett-Packard (Canada) Ltd.
4519 Canada Way
North Burnaby 2
Tel: (604) 433-8213
TWX: 610-922-5059

MANITOBA
Hewlett-Packard (Canada) Ltd.
511 Bradford Ct.
St. James
Tel: (204) 786-7581
TWX: 610-671-3531

NOVA SCOTIA
Hewlett-Packard (Canada) Ltd.
2745 Dutch Village Rd.
Suite 203
Halifax
Tel: (902) 455-0511
TWX: 610-271-4482

ONTARIO
Hewlett-Packard (Canada) Ltd.
880 Lady Ellen Place
Ottawa 3
Tel: (613) 722-4223
TWX: 610-562-1952

Hewlett-Packard (Canada) Ltd.
50 Galaxy Blvd.
Rexdale
Tel: (416) 677-9611
TWX: 610-492-4246

QUEBEC
Hewlett-Packard (Canada) Ltd.
275 Hymus Boulevard
Pointe Claire
Tel: (514) 697-4232
TWX: 610-422-3022
Telex: 01-20607

FOR CANADIAN AREAS NOT LISTED:
Contact Hewlett-Packard (Canada) Ltd. in Pointe Claire, at the complete address listed above.

E 7-70

HEWLETT · PACKARD

ELECTRONIC INSTRUMENTATION SALES AND SERVICE

EUROPE, AFRICA, ASIA, AUSTRALIA

EUROPE

AUSTRIA
Unilabor GmbH
Wissenschaftliche Instrumente
Rummelhardtgasse 6/3
P.O. Box 33
Vienna A-1095
Tel: (222) 42 61 81, 43 13 94
Cable: LABORINSTRUMENT Vienna
Telex: 75 762

BELGIUM
Hewlett-Packard Benelux S.A.
348 Boulevard du Souverain
Brussels 1160
Tel: 72 22 40
Cable: PALOBEN Brussels
Telex: 23 494

DENMARK
Hewlett-Packard A/S
Datavej 38
DK-3460 **Birkeroed**
Tel: (01) 81 66 40
Cable: HEWPACK AS
Telex: 66 40

EASTERN EUROPE
Hewlett-Packard S.A. Genf.
Korrespondenz Büro Für Ost-europa
(Czechoslovakia, Hungary, Poland, DDR, Rumania, Bulgaria)
Innstrasse 23/2
Postfach
A-1204 Vienna, Austria
Tel: (222) 33 66 06/09
Cable: HEWPACK Vienna

FINLAND
Hewlett-Packard Oy
Bulevardi 26
P.O. Box 12185
Helsinki 12
Tel: 13-730
Cable: HEWPACKOY-Helsinki
Telex: 12-1563

FRANCE
Hewlett-Packard France
Quartier de Courtaboeuf
Boite Postale No. 6
91 Orsay
Tel: 1-920 88 01
Cable: HEWPACK Orsay
Telex: 60048

Hewlett-Packard France
4 Quai des Etroits
69 **Lyon** 5ème
Tel: 78-42 63 45
Cable: HEWPACK Lyon
Telex: 31617

GERMANY
Hewlett-Packard Vertriebs-GmbH
Lietzenburgerstrasse 30
1 **Berlin** 30
Tel: (0311) 211 60 16
Telex: 18 34 05

Hewlett-Packard Vertriebs-GmbH
Herrenbergerstrasse 110
703 **Böblingen,** Württemberg
Tel: 07031-6671
Cable: HEPAG Böblingen
Telex: 72 65 739

Hewlett-Packard Vertriebs-GmbH
Achenbachstrasse 15
4 **Düsseldorf** 1
Tel: (0211) 68 52 58/59
Telex: 85 86 533

Hewlett-Packard Vertriebs-GmbH
Berliner Strasse 117
6 **Nieder-Eschbach/Frankfurt 56**
Tel: (0611) 50 10 64
Cable: HEWPACKSA Frankfurt
Telex: 41 32 49 FRA

Hewlett-Packard Vertriebs-GmbH
Beim Strohhause 26
2 **Hamburg** 1
Tel: (0411) 24 05 51/52
Cable: HEWPACKSA Hamburg
Telex: 21 53 32

Hewlett-Packard Vertriebs-GmbH
Reginfriedstrasse 13
8 **München** 9
Tel: (0811) 69 59 71/75
Cable: HEWPACKSA München
Telex: 52 49 85

GREECE
Kostas Karayannis
18, Ermou Street
Athens 126
Tel: 230301,3,5
Cable: RAKAR Athens
Telex: 21 59 62 RKAR GR

IRELAND
Hewlett-Packard Ltd.
224 Bath Road
Slough, Bucks, England
Tel: Slough 753-33341
Cable: HEWPIE Slough
Telex: 84413

ITALY
Hewlett-Packard Italiana S.p.A.
Via Amerigo Vespucci 2
20124 **Milano**
Tel: (2) 6251 (10 lines)
Cable: HEWPACKIT Milan
Telex: 32046

Hewlett-Packard Italiana S.p.A.
Palazzo Italia
Piazza Marconi 25
00144 **Rome** - Eur
Tel: 6-591 2544
Cable: HEWPACKIT Rome
Telex: 61514

NETHERLANDS
Hewlett-Packard Benelux, N.V.
Weerdestein 117
P.O. Box 7825
Amsterdam, Z 11
Tel: 020-42 7777
Cable: PALOBEN Amsterdam
Telex: 13 216

NORWAY
Hewlett-Packard Norge A/S
Box 149
Nesveien 13
N-1344 **Haslum**
Tel: 2-53 83 60
Cable: HEWPACK Oslo
Telex: 16621

PORTUGAL
Telectra
Empresa Tecnica de Equipamentos Electricos, S.a.r.l.
Rua Rodrigo da Fonseca 103
P.O. Box 2531
Lisbon 1
Tel: 68 60 72
Cable: TELECTRA Lisbon
Telex: 1598

SPAIN
Ataio Ingenieros SA
Ganduxer 76
Barcelona 6
Tel: 211-44-66
Cable: TELEATAIO BARCELONA

Ataio Ingenieros SA
Enrique Larreta 12
Madrid, 16
Tel: 215 35 43
Cable: TELEATAIO Madrid
Telex: 27249E

SWEDEN
Hewlett-Packard (Sverige) AB
Hagakersgatan 9C
S 431 04 **Mölndal** 4
Tel: 031 - 27 68 00

Hewlett-Packard (Sverige) AB
Svetsarvägen 7
S171 20 **Solna** 1
Tel: (08) 98 12 50
Cable: MEASUREMENTS Stockholm
Telex: 10721

SWITZERLAND
Hewlett Packard (Schweiz) AG
Zurcherstrasse 20
8952 Schlieren
Zurich
Tel: (051) 98 18 21/24
Cable: HPAG CH
Telex: 53933

Hewlett Packard (Schweiz) A.G.
Rue du Bois-du-Lan 7
1217 Meyrin 2 **Geneva**
Tel: (022) 41 54 00
Cable: HEWPACKSA Geneva
Telex: 2 24 86

TURKEY
Telekom Engineering Bureau
P.O. Box 376 - Galata
Karakoy
Istanbul
Tel: 49 40 40
Cable: TELEMATION Istanbul

UNITED KINGDOM
Hewlett-Packard Ltd.
224 Bath Road
Slough, Bucks
Tel: Slough 33341
Cable: HEWPIE Slough
Telex: 84413

Hewlett-Packard Ltd.
The Graftons
Stamford New Road
Altrincham, Cheshire
Tel: 061 928-8626
Telex: 668068

USSR
Please Contact
Hewlett-Packard S.A.
Rue du Bois-du-Lan 7
1217 Meyrin 2 **Geneva**
Tel: (022) 41 54 00
Cable: HEWPACKSA Geneva Switzerland
Telex: 2.24.86

YUGOSLAVIA
Belram S.A.
83 avenue des Mimosas
Brussels 1150, Belgium
Tel: 34 33 32, 34 26 19
Cable: BELRAMEL Brussels
Telex: 21790

FOR AREAS NOT LISTED, CONTACT:
Hewlett-Packard S.A.
Rue du Bois-du-Lan 7
1217 Meyrin 2 **Geneva** Switzerland
Tel: (022) 41 54 00
Cable: HEWPACKSA Geneva
Telex: 2.24.86

AFRICA, ASIA, AUSTRALIA

ANGOLA
Telectra Empresa Técnia de Equipamentos Eléctricos SAR
Rua de Barbosa Rodrigues 42-1°
Box 6487
Luanda
Cable: TELECTRA Luanda

AUSTRALIA
Hewlett-Packard Australia Pty. Ltd.
22-26 Weir Street
Glen Iris, 3146
Victoria
Tel: 20.1371 (6 lines)
Cable: HEWPARD Melbourne
Telex: 31024

Hewlett-Packard Australia Pty. Ltd.
61 Alexander Street
Crows Nest 2065
New South Wales
Tel: 43.7866
Cable: HEWPARD Sydney
Telex: 21561

Hewlett-Packard Australia Pty. Ltd.
97 Churchill Road
Prospect 5082
South Australia
Tel: 65.2366
Cable: HEWPARD Adelaide

Hewlett Packard Australia Pty. Ltd.
2nd Floor, Suite 13
Casablanca Buildings
196 Adelaide Terrace
Perth, W.A. 6000
Tel: 21-3330
Cable: HEWPARD Perth

Hewlett-Packard Australia Pty. Ltd.
10 Woolley Street
P.O. Box 191
Dickson A.C.T. 2602
Tel: 49-8194
Cable: HEWPARD Canberra ACT

CEYLON
United Electricals Ltd.
P.O. Box 681
Yahala Building
Staples Street
Colombo 2
Tel: 5496
Cable: HOTPOINT Colombo

CYPRUS
Kypronics
19 Gregorios & Xenopoulos Road
P.O. Box 1152
Nicosia
Tel: 6282-75628
Cable: HE-I-NAMI

ETHIOPIA
African Salespower & Agency Private Ltd., Co.
P. O. Box 718
58/59 Cunningham St.
Addis Ababa
Tel: 12285
Cable: ASACO Addisababa

HONG KONG
Schmidt & Co. (Hong Kong) Ltd.
P.O. Box 297
1511, Prince's Building 15th Floor
10, Chater Road
Hong Kong
Tel: 240168, 232735
Cable: SCHMIDTCO Hong Kong

INDIA
Blue Star Ltd.
Kasturi Buildings
Jamshedji Tata Rd.
Bombay 20BR, India
Tel: 29 50 21
Telex: 2396
Cable: BLUEFROST

Blue Star Ltd.
Band Box House
Prabhadevi
Bombay 25DD, India
Tel: 45 73 01
Telex: 2396
Cable: BLUESTAR

Blue Star Ltd.
14/40 Civil Lines
Kanpur, India
Tel: 6 88 82
Cable: BLUESTAR

Blue Star, Ltd.
7 Hare Street
P.O. Box 506
Calcutta 1, India
Tel: 23-0131
Telex: 655
Cable: BLUESTAR

Blue Star Ltd.
Blue Star House,
34 Ring Road
Lajpat Nagar
New Delhi 24, India
Tel: 62 32 76
Telex: 463
Cable: BLUESTAR

Blue Star, Ltd.
96 Park Lane
Secunderabad 3, India
Tel: 7 63 91
Cable: BLUEFROST

Blue Star, Ltd.
23/24 Second Line Beach
Madras 1, India
Tel: 2 39 55
Telex: 379
Cable: BLUESTAR

Blue Star, Ltd.
1B Kaiser Bungalow
Dindli Road
Jamshedpur, India
Tel: 38 04
Cable: BLUESTAR

INDONESIA
Bah Bolon Trading Coy. N.V.
Djalah Merdeka 29
Bandung
Tel: 4915 51560
Cable: ILMU
Telex: 809

IRAN
Telecom, Ltd.
P. O. Box 1812
240 Kh. Saba Shomali
Teheran
Tel: 43850, 48111
Cable: BASCOM Teheran

ISRAEL
Electronics & Engineering Div. of Motorola Israel Ltd.
17 Aminadav Street
Tel-Aviv
Tel: 36941 (3 lines)
Cable: BASTEL Tel-Aviv
Telex: Bastel Tv 033-569

JAPAN
Yokogawa-Hewlett-Packard Ltd.
Nisei Ibaragi Bldg.
2-2-8 Kasuga
Ibaragi-Shi
Osaka
Tel: 23-1641

Yokogawa-Hewlett-Packard Ltd.
Ito Building
No. 59, Kotori-cho
Nakamura-ku, **Nagoya** City
Tel: 551-0215

Yokogawa-Hewlett-Packard Ltd.
Nitto Bldg.
2300 Shinohara-cho,
Kohoku-ku
Yokohama 222
Tel: (405) 432-1504/5

Yokogawa-Hewlett-Packard Ltd.
Ohashi Building
59 Yoyogi 1-chrome
Shibuya-ku, **Tokyo**
Tel: 03-370-2281/7
Telex: 232-2024YHP
Cable: YHPMARKET TOK 23-724

KENYA
R. J. Tilbury Ltd.
P. O. Box 2754
Suite 517/518
Hotel Ambassadeur
Nairobi
Tel: 25670, 68206, 58196
Cable: ARJAYTEE Nairobi

KOREA
American Trading Co., Korea, Ltd.
P.O. Box 1103
7th & 8th Floors
Dae Kyung Bldg.
107 Sejong Ro
Chongro Ku
Seoul
Tel: 75-5841 (4 lines)
Cable: AMTRACO Seoul

LEBANON
Constantin E. Macridis
Clemenceau Street
P.O. Box 7213
Beirut
Tel: 220846
Cable: ELECTRONUCLEAR Beirut

MALAYSIA
MECOMB Malaysia Ltd.
2 Lorong 13/6A
Section 13
Petaling Jaya, **Selangor**
Cable: MECOMB Kuala Lumpur

MOZAMBIQUE
A. N. Goncalves, LDA.
4.1 Apt. 14 Av. D. Luis
P.O. Box 107
Lourenco Marques
Cable: NEGON

NEW ZEALAND
Hewlett-Packard (N.Z.) Ltd.
32-34 Kent Terrace
P.O. Box 9443
Wellington, N.Z.
Tel: 56-559
Cable: HEWPACK Wellington

Hewlett Packard (N.Z.) Ltd.
Box 51092
Pukuranga
Tel: 573-733

PAKISTAN (EAST)
Mushko & Company, Ltd.
Zirat Chambers
31, Jinnah Avenue
Dacca
Tel: 280058
Cable: NEWDEAL Dacca

PAKISTAN (WEST)
Mushko & Company, Ltd.
Oosman Chambers
Victoria Road
Karachi 3
Tel: 511027, 512927
Cable: COOPERATOR Karachi

PHILIPPINES
Electromex Inc.
Makati Commercial Center
2129 Pasong Tamo
Makati, Rizal D 708
P.O. Box 1028
Manila
Tel: 89-85-01
Cable: ELEMEX Manila

SINGAPORE
Mechanical and Combustion Engineering Company Ltd.
9, Jalan Kilang
Red Hill Industrial Estate
Singapore, 3
Tel: 642361-3
Cable: MECOMB Singapore

SOUTH AFRICA
Hewlett Packard South Africa (Pty.), Ltd.
Breecastle House
Bree Street
Cape Town
Tel: 3-6019, 3-6545
Cable: HEWPACK Cape Town
Telex: 5-0006

Hewlett Packard South Africa (Pty.), Ltd.
P.O. Box 31716
Braamfontein Transvaal
Milnerton
30 De Beer Street
Johannesburg
Tel: 724-4172 724-4195
Telex: 0226 JH
Cable: HEWPACK Johannesburg

Hewlett Packard South Africa (Pty.), Ltd.
30B Glenwood Centre
Corner Hunt & Moore Roads
Durban
P.O. Box 99
Overport, Natal
Tel: 347536

TAIWAN REP. OF CHINA
Hwa Sheng Electronic Co., Ltd.
P. O. Box 1558
Room 404
Chia Hsin Building
No. 96 Chung Shan North Road, Sec. 2
Taipei
Tel: 555211 Ext. 532-539
Cable: VICTRONIX Taipei

TANZANIA
R. J. Tilbury Ltd.
P.O. Box 2754
Suite 517/518
Hotel Ambassadeur
Nairobi
Tel: 25670, 26803, 68206, 58196
Cable: ARJAYTEE Nairobi

THAILAND
The International Engineering Co., Ltd.
P. O. Box 39
614 Sukhumvit Road
Bangkok
Tel: 910722 (7 lines)
Cable: GYSOM
TLX INTENCO BK-226 Bangkok

UGANDA
R. J. Tilbury Ltd.
P.O. Box 2754
Suite 517/518
Hotel Ambassadeur
Nairobi
Tel: 25670, 26803, 68206, 58196
Cable: ARJAYTEE Nairobi

VIETNAM
Peninsular Trading Inc.
P.O. Box H-3
216 Hien-Vuong
Saigon
Tel: 20.805
Cable: PENINSULA Saigon

ZAMBIA
R. J. Tilbury (Zambia) Ltd.
P.O. Box 2792
Lusaka
Zambia, Central Africa

FOR AREAS NOT LISTED, CONTACT:
Hewlett-Packard INTERCONTINENTAL
3200 Hillview Ave.
Palo Alto, California 94304
Tel: (415) 326-7000
TWX: 910-373-1267
Cable: HEWPACK Palo Alto
Telex: 034-8461

HEWLETT hp PACKARD

05450-90029

PRINTED IN U.S.A.