About this Manual

We've added this manual to the Agilent website in an effort to help you support your product. This manual is the best copy we could find; it may be incomplete or contain dated information. If we find a more recent copy in the future, we will add it to the Agilent website.

Support for Your Product

Agilent no longer sells or supports this product. Our service centers may be able to perform calibration if no repair parts are needed, but no other support from Agilent is available. You will find any other available product information on the Agilent Test & Measurement website, www.tm.agilent.com.

HP References in this Manual

This manual may contain references to HP or Hewlett-Packard. Please note that Hewlett-Packard's former test and measurement, semiconductor products and chemical analysis businesses are now part of Agilent Technologies. We have made no changes to this manual copy. In other documentation, to reduce potential confusion, the only change to product numbers and names has been in the company name prefix: where a product number/name was HP XXXX the current name/number is now Agilent XXXX. For example, model number HP8648A is now model number Agilent 8648A.

HEWLETT PACKARD

PROGRAMMING MANUAL

HP 70700A DIGITIZER

SERIAL NUMBERS

This manual applies directly to HP 70700A Digitizers with serial numbers prefixed 2709A and below.

FIRMWARE VERSIONS

This manual applies directly to HP 70700A Digitizers with firmware versions of 870501 and earlier.

1212 VALLEY HOUSE DRIVE, ROHNERT PARK, CALIFORNIA 94928-4999 U.S.A.

Manual Part Number 70700-90021
Microfiche Part Number 70700-90022

Printed: MAY 1987

CERTIFICATION

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

WARRANTY

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

For warranty service or repair, this product must be returned to a service facility designated by HP. Buyer shall prepay shipping charges to HP and HP shall pay shipping charges to return the product to Buyer. However, Buyer shall pay all shipping charges, duties, and taxes for products returned to HP from another country.

HP warrants that its software and firmware designated by HP for use with an instrument will execute its programming instructions when properly installed on that instrument. HP does not warrant that the operation of the instrument, or software, or firmware will be uninterrupted or error free.

LIMITATION OF WARRANTY

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

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

EXCLUSIVE REMEDIES

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

ASSISTANCE

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

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

SAFETY SYMBOLS

The following safety symbols are used throughout this manual and in the instrument. Familiarize yourself with each of the symbols and its meaning before operating this instrument.

Instruction manual symbol. The instrument will be marked with this symbol when it is necessary for the user to refer to the instruction manual in order to protect the instrument against damage. Location of pertinent information within the manual is indicated by use of this symbol in the table of contents.

Indicates dangerous voltages are present. Be extremely careful.

CAUTION

The CAUTION sign denotes a hazard. It calls attention to a procedure which, if not correctly performed or adhered to, could result in damage to or destruction of the instrument. Do not proceed beyond a CAUTION sign until the indicated conditions are fully understood and met.

WARNING

The WARNING sign denotes a hazard. It calls attention to a procedure which, if not correctly performed or adhered to, could result in injury or loss of life. Do not proceed beyond a WARNING sign until the indicated conditions are fully understood and met.

GENERAL SAFETY CONSIDERATIONS

WARNING

BEFORE THIS INSTRUMENT IS SWITCHED ON, make sure it has been properly grounded through the protective conductor of the ac power cable to a socket outlet provided with protective earth contact. Any interruption of the protective (grounding) conductor, inside or outside the instrument, or disconnection of the protective earth terminal can result in personal injury.

WARNING

There are voltages at many points in the instrument which can, if contacted, cause personal injury. Be extremely careful. Any adjustments or service procedures that require operation of the instrument with protective covers removed should be performed only by trained service personnel.

CAUTION

BEFORE THIS INSTRUMENT IS SWITCHED ON, make sure its primary power circuitry has been adapted to the voltage of the ac power source. Failure to set the ac power input to the correct voltage could cause damage to the instrument when the ac power cable is plugged in.

CONTENTS

CONTENTS, Continued

CONTENTS, Continued

CONTENTS, Continued

Appendices

LIST OF ILLUSTRATIONS

HP 70000 MODULAR MEASUREMENT SYSTEM DOCUMENTATION OUTLINE

Instruments and modules of the HP 70000 Modular Measurement System are documented to varying levels of detail. Modules that serve as masters of an instrument require operation information in addition to installation and verification instructions. Modules that function as slaves in a system require only a subset of installation and verification information. Service documentation is available for every module of the HP 70000 Modular Measurement System family.

USER MANUALS, SUPPLIED WITH MODULE

Installation and Verification Manual

Topics covered by this manual include installation, specifications, verification of module operation, and some troubleshooting techniques. Manuals for modules that serve as instrument masters will supply information in all these areas; manuals for slave modules will contain only information needed for slave module installation and verification. Master module documentation may also include some system-level information.

Operation Manual

Information in this manual usually pertains to multiple- and single-module instrument systems. The Operation Manual describes manual operation of the module, with explanations of softkeys and their use.

Programming Manual

Information in this manual usually pertains to multiple- and single-module instrument systems. The Programming Manual defines commands that enable remote operation of the module, and describes remote command syntax.

SERVICE MANUAL, AVAILABLE SEPARATELY

Technical Reference

This manual provides service information for a module, including performance verification, adjustments, troubleshooting, replaceable parts lists, replacement procedures, schematics, and component location diagrams. For ordering information, contact an HP Sales and Service Office.

Chapter 1
GENERAL INFORMATION

HOW TO USE THIS MANUAL

The HP 70700A Digitizer Programming Manual describes how to operate a digitizer system by remote, computer control. It is not necessary to read this manual from cover to cover; however, it may be useful to note what is in each chapter.

Chapter 1, General Information, provides descriptions of the digitizer system, the HP 70700A Digitizer module, and front- and rear-panel features of an HP 70700A Digitizer module.

NOTE

Before any remote operations are performed with your digitizer system, it must first be installed and configured properly. Refer to the HP 70700A Digitizer Installation and Verification Manual for detailed instructions on correct installation and configuration information.

Chapter 2, Programming Fundamentals, provides information on the following topics: installation for remote operation, communication with the system, the Status Reporting Structure, synchronization of events and commands, and data transfer. Information on multiple-digitizer remote slaving is also included in this chapter.

Chapter 3, Language Reference, provides a complete, detailed description of each command within each functional subsystem, including syntax diagrams and functional parameters.

Appendices consist of four cross-reference listings and/or summaries for the digitizer remote commands, and a general overview of the Hewlett-Packard Interface Bus (HP-IB):

Appendix A	Command Listing by Subsystem
Appendix B	Alphabetical Mnemonic Listing
Appendix C	Alphabetical Command Description Listing
Appendix D	Alphabetical Command Summary
Appendix E	HP-IB Review

DIGITIZER SYSTEM OVERVIEW

A single-channel digitizer system consists of an HP 70700A Digitizer master module, an HP 70205A or HP 70206A Display, and an HP 70001A Mainframe. Additional digitizer modules may be slaved to the master module to provide a multiple-channel digitizer system. See Figure l-l.

The digitizer system measures repetitive or transient waveforms with improved accuracy, resolution, and dynamic range over other measurement techniques. With the use of the internal analog-to-digital converter

(ADC) combined with digital memory, a computer can be used to analyze the data, providing better results and allowing automation. Simultaneous digitization and memory access are available (i.e., as data is being measured, the results may be accessed immediately with the use of a controller). Multi-channel operation may be achieved by slaving multiple digitizer modules together, with the slave digitizer module synchronized by the master digitizer module "clock out" and "sync out" signals.

Data is always digitized at a 20 MHz rate and then a digital detection algorithm is applied to "reduce" the data before it is stored. The four detection modes that are available for selection are sample, peak, pit, and alternate.

Interpolation may be used to provide a better visual display of high-frequency waveforms.

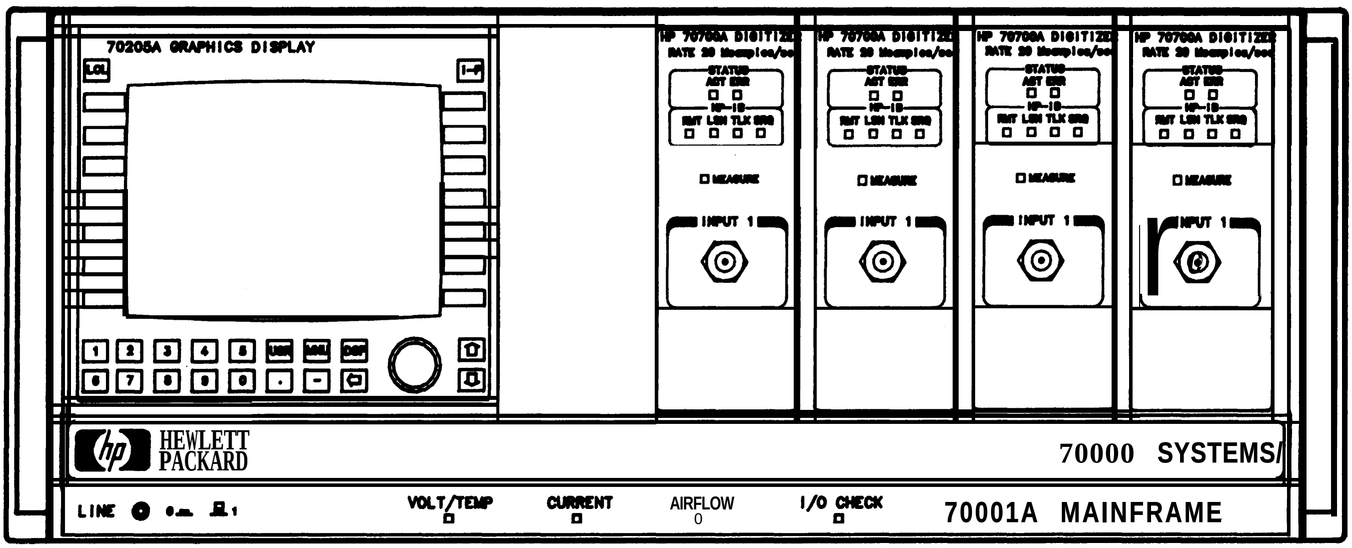


Figure l-l. Multiple-Channel Digitizer System

The digitizer system has triggering capabilities which are especially necessary when measuring single-shot transient waveforms. These triggering capabilities are the functions of EITHER EDGE, ABOVE LEVEL, BELOW LEVEL, or OUTSIDE RANGE. Additionally, hysteresis may be used to adapt the triggering to match the input signal.

The Random Event Capture mode retains up to 1000 trigger events of a measurement with a guaranteed amount of pre- and post-trigger data for each event. There is no dead time between events, the time of the event is always stored, and the event time may be queried from a controller without outputting the event data. The memory is dual ported so that captured events may be examined while the measurement is still proceeding.

The Equivalent Time Sampling mode provides a technique for looking at stable, periodic, repetitive signals by using multiple trigger events to form a composite waveform. It will allow rise-time measurements of less than 10 ns.

HP 70700A DIGITIZER MODULE OVERVIEW

The HP 70700A Digitizer is a l/8-module with a 20 MHz sampling rate designed to work in an HP 70000 Series mainframe. It has both HP-IB (Hewlett-Packard Interface Bus) and HP-MSIB (Hewlett-Packard Modular System Interface Bus) communication capabilities. The HP 70700A Digitizer can function in the following configurations:

- Single-channel digitizer system (consisting of one module)
- Multiple-channel digitizer system (with one HP 70700A Digitizer functioning as a master to other HP 70700A Digitizer modules)
- Slave to an HP 70900A Local Oscillator when configured in an HP 70000 Series Modular Spectrum Analyzer System
- Single-channel digitizer instrument configured with an HP 70000 Series Modular Spectrum Analyzer System (When the HP 70700A Digitizer is used in this configuration, it is not a slave of the spectrum analyzer but is used to view the spectrum analyzer video output.)

The last configuration described above is essentially the same as a single-channel digitizer system. The disadvantage of this configuration is that the spectrum analyzer and digitizer operate independently, so sweep-time coupling and amplitude calibration are lost. However, the advantage of this configuration is that it allows all the digitizer features, such as the Measure ALL function and the Random Event Capture mode, to operate on the video output of the spectrum analyzer. Refer to the HP 70700A Digitizer Installation and Verification Manual for diagrams of correct digitizer system configurations.

HP 70700A DIGITIZER FRONT-PANEL FEATURES

LED Indicators

1. STATUS ACT indicates that the HP 70700A Digitizer is active. The ACTIVE LED lights when:

   a. the keyboard of the display is allocated to the digitizer.

   b. any Display function indicates the digitizer (e.g., when the cursor of the Display Address Map is at the HP-MSIB address of the digitizer).

   c. a digitizer is a slave to another digitizer that is designated as a master module, and it is being used by that master digitizer.

2. STATUS ERR indicates errors. (Refer to Chapter 5, Troubleshooting, in the HP 70700A Digitizer Installation and Verification Manual.)

3. HP-IB RMT indicates that the module is being remotely controlled and local control is disabled.

4. HP-IB LSN indicates a state in which the module is ready to accept information from the controller.

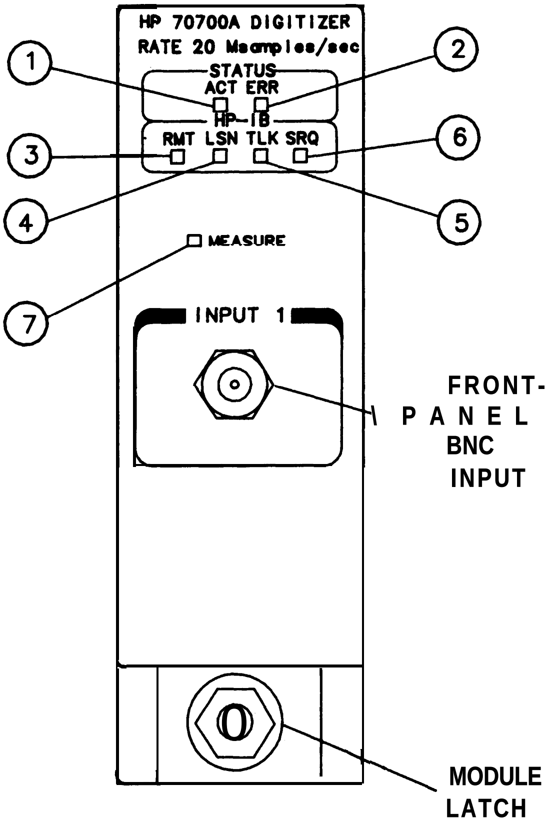


Figure 1-2. HP 70700A Digitizer Front Panel

5. HP-IB TLK indicates a state in which the module is ready to send information to the controller.

6. HP-IB SRQ indicates a condition requested or set by the user (e.g., errors, operation complete status, power-on condition). Refer to Chapter 2, Programming Fundamentals, for more information on the Service Request LED and the Standard Status Data Structure.

7. MEASURE indicates that the module is making a measurement.

INPUTS

INPUT 1 (type BNC connector) can be utilized only when the HP 70700A Digitizer is used in a digitizer system. Refer to Table 1-l for more information about input selection and input impedance.

MODULE LATCH

The module hex-nut latch is used for installing the module in an HP 70000 Series mainframe. An 8 mm hex-ball driver is required to turn the hex-nut latch.

Table 1-1. HP 70700A Digitizer Input Selection and Input Impedance

HP 70700A Digitizer used in a digitizer system, or as a digitizer instrument in a non-digitizer system:	
Preset Input	INPUT 1
Softkey-Selected Input	INPUT 2
Preset Input Impedance	DC Coupled 1 MΩ
Softkey-Selected Input Impedance	AC Coupled 1Ω, or DC Coupled 50Ω
HP 70700A Digitizer used as a slave to a non-digitizer master:	
Preset Input	INPUT 2
Softkey-Selected Input	none
Preset Input Impedance	DC Coupled 1 MΩ
Softkey-Selected Input Impedance	none

HP 70700A DIGITIZER REAR-PANEL FEATURES

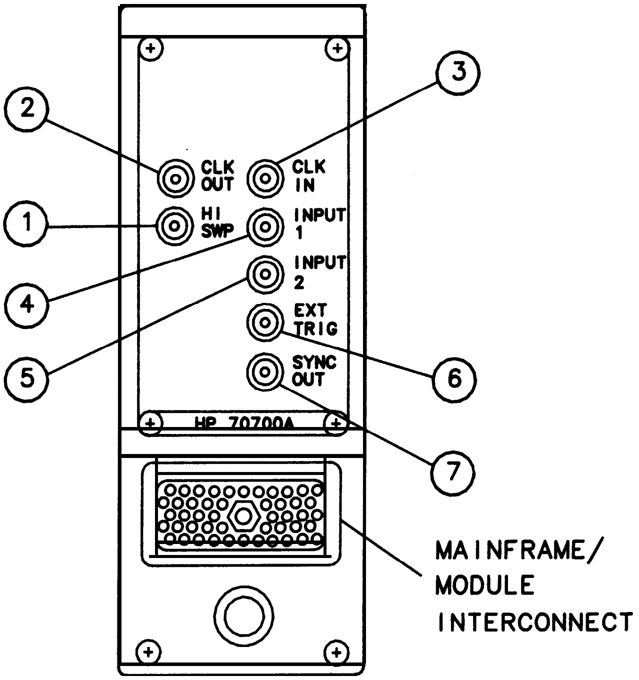


Figure 1-3. HP 70700A Digitizer Rear Panel

Rear-Panel SMB Connectors

1. HI SWP (High Sweep) is an input/output that is connected to HSWP of the HP 70900A Local Oscillator when an HP 70700A Digitizer is used as a slave to an HP 70900A Local Oscillator in a spectrum analyzer system. This connection is necessary to synchronize the digitizer and local oscillator and their starting and stopping of measurements.

2. CLK OUT (Clock Out) provides a TTL-level 20 MHz clock output. In a single-channel digitizer system, the CLK OUT is connected to the CLK IN on the same module. In a multiple-channel digitizer system, the CLK OUT of the master HP 70700A Digitizer module must be connected to both its own CLK IN and the CLK IN of all of its slaves. (Refer to the HP 70700A Digitizer Installation and Verification Manual, Chapter 2, for detailed instructions on correct installation and configuration information.)

3. CLK IN (Clock In) requires a 50% duty cycle, TTL-level clock input with a 10 MHz to 20 MHz frequency. (See CLK OUT, above.)

4. INPUT I can be utilized only when the HP 70700A Digitizer is used in a digitizer system. The front-panel INPUT 1 and rear-panel INPUT 1 are connected together and are electrically the same. Refer to Table 1-I for more information about input selection and input impedance.

5. INPUT 2 can be utilized when the HP 70700A Digitizer is used in any type of configuration. When the HP 70700A Digitizer is used in a digitizer system, INPUT 2 is preset “open” (i.e., no connection). Refer to Table l-l for more information about input selection and input impedance.

6. EXT TRIG (External Trigger) allows an external input signal to be used to trigger the digitizer module externally. The input signal must be TTL with a pulse width of’ at least one clock period, typically 50 ns. (See SYNC OUT, below.)

7. SYNC OUT (Synchronizing Output) provides a TTL-level signal used to svnchronize the ‘HP 70700A Digitizer slave modules of a multiple-channel digitizer system. In a multiple-channel digitizer system, the SYNC OUT of the master HP 70700A Digitizer must be connected to the EXT TRIG inputs of the HP 70700A Digitizer slave modules. (Refer to the HP 70700A Digitizer Installation and Verification Manual, Chapter 2, for detailed instructions on correct installation and configuration information.)

MAINFRAME/MODULE INTERCONNECT

The mainframe provides the power supply, HP-MSIB connections, and HP-IB connections for the HP 70700A Digitizer module through this mainframe/module interconnect.

Chapter 2
PROGRAMMING FUNDAMENTALS

This chapter provides the information necessary to operate a digitizer system via a computer. The topics covered in this chapter are listed below:

- Setup Procedures for Remote Operation
- Address Switches
- Communication with the System
- Status Reporting Structure
- Synchronization of Events and Commands
- Data Transfer
- Multiple-Digitizer Remote Slaving Procedure

SETUP PROCEDURES

The following procedure describes how to connect your equipment for remote operation of a digitizer system.

NOTE

Refer to the HP 70700A Digitizer Installation and Verification Manual for more detailed and specific information on installation, configuration, and addressing of digitizer systems.

1. Connect computer, digitizer system, and other peripherals with HP-IB cables.

2. After the HP-IB cables are installed, reset all instruments connected to the bus. (If you are not sure how to reset a device, switch its line power off, then on, to reset it.)

3. Check the HP-IB address of the master digitizer module on the address map. To view the address map, **press the [DISPLAY] key on the display front panel, then press the** *ADDRESS MAP* **softkey. Turn the** RPG knob on the front panel of the display until the first digitizer module appears in the address map.

NOTE

The master digitizer module must be located in row 0 for HP-IB access and error-reporting capabilities.

4. Check the system configuration on the address map. For single-channel digitizer system operation, the digitizer and display modules must be positioned in the bottom row (row. 0) of the address map. For multi-channel digitizer systems, the other digitizer modules must be positioned either in the same column as the master digitizer above row 0, or in any other higher-address column above row 0 of the master module. For more information on multi-channel system configurations, refer to Digitizer Channel Number Assignment, below.

DIGITIZER CHANNEL NUMBER ASSIGNMENT

The channel number assignment of a digitizer system is determined by addressing. The master digitizer, located at row 0, will always be assigned as CHANNEL 1. The next HP 70700A Digitizer encountered by the master in its search will be assigned as CHANNEL 2. This process continues until all of the channels have been assigned, or until all of the HP 70700A Digitizers have been assigned channels. The sequence in which the Address Map is searched is from bottom to top and left to right. A maximum of four channels are available when a digitizer system is controlled from a display front panel. A maximum of eight channels are available when the system is controlled by a computer.

Figure 2-1 displays the address map of a four-channel digitizer system. Note that the master digitizer module is CHANNEL 1 and that it is located in row 0. The other three digitizers (or Channels) are defined by their address positions.

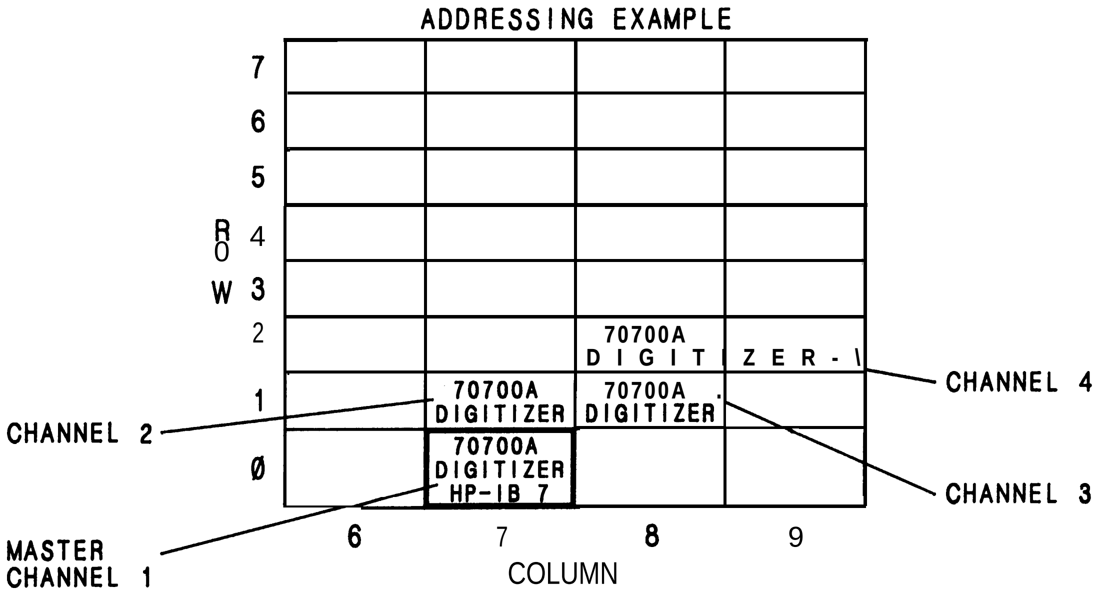


Figure 2-l. Multiple-Channel Digitizer System Addressing

ADDRESS SWITCHES

Address switches set the HP-MSIB address of an element (module); the column address switches also set the HP-IB address for masters and independent elements. Some master elements can also have their HP-IB address set through the use of **softkeys** (i.e., soft-set address).

The hard address switches for the digitizer are located on the top of the module.

NOTE

HP-IB address 31 is an Illegal address and should not be used.

Descriptions of the HP 70700A Digitizer address switches are given below.

HP-IB ON-OFF When this is set to OFF, the HP 70700A Digitizer is switched off the HP-IB.

Column ADDRESS Switches l-5 These set the HP-MSIB column address, which is also the HP-IB address.

Row ADDRESS Switches l-3 These set the HP-MSIB row address.

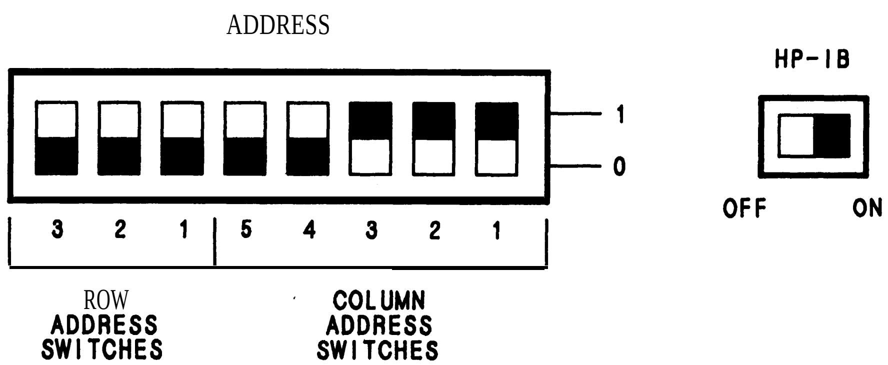


Figure 2-2. HP 70700A Digitizer Address Switches

SOFT-SET HP-IB ADDRESSES

The HP-IB address of the digitizer master module may be changed from the front panel of the display. The soft-set address remains in effect until the hard address switches are changed and power is cycled, or until another soft-set address is entered. At power-up, the soft-set address will override the hard address switch settings.

NOTE

Changing the HP-IB address via the display front **panel** does not affect the position of the modules on the address map.

Use the following procedure to enter a soft-set HP-IB address.

1. Press the [DISPLAY] key on the display front panel.
2. When the display Main Menu appears, press the ADDRESS MAP softkey.
3. When the next menu appears, press SET HP-IB.
4. Enter the new HP-IB address using the numeric keys on the display front panel.
5. Press ENTER.

COMMUNICATION WITH THE SYSTEM

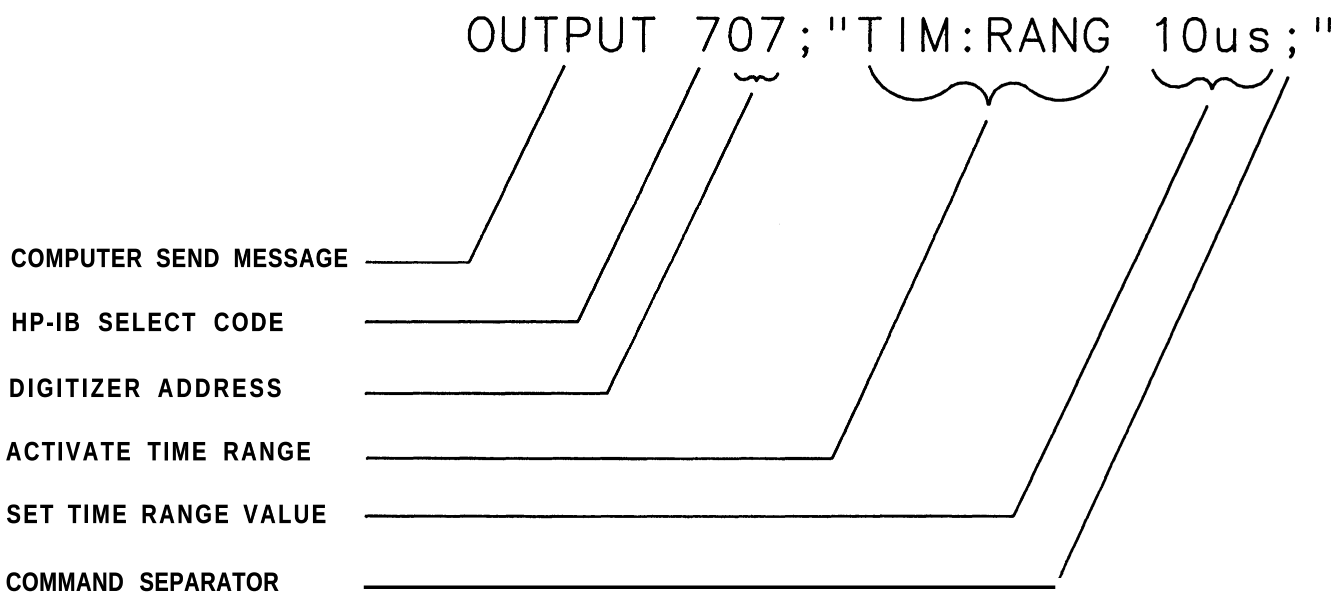
This section develops some fundamental techniques for controlling the digitizer and obtaining sound measurement results. Remote operation of the digitizer is controlled with commands that correspond in general to front-panel softkey functions.

It is important to understand how messages are communicated to the digitizer; therefore, enter and output statements and command syntax discussed in this section should be understood before proceeding. It should be noted that HP BASIC is used for all examples in this manual.

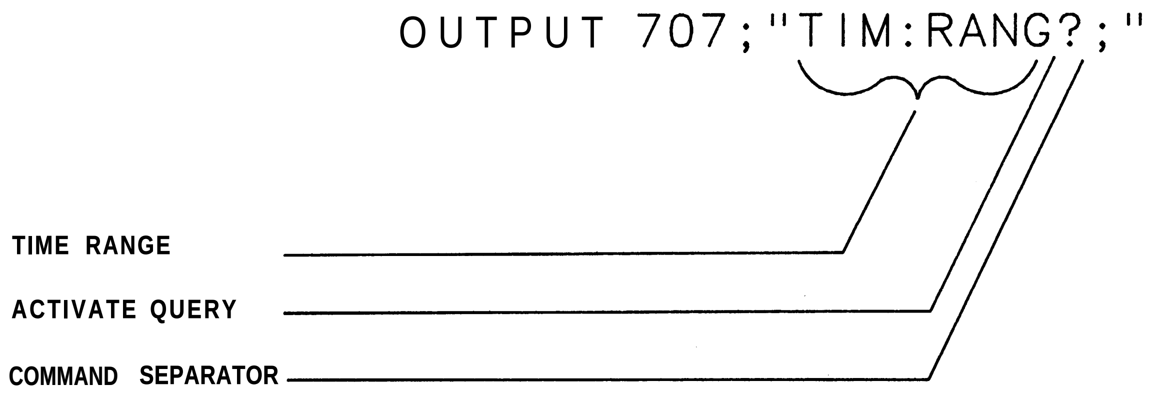
EXECUTING REMOTE COMMANDS

Digitizer programs control the passage of digitizer commands and data between the digitizer and the computer on the Hewlett-Packard Interface Bus (HP-IB), using HP BASIC OUTPUT and ENTER statements.

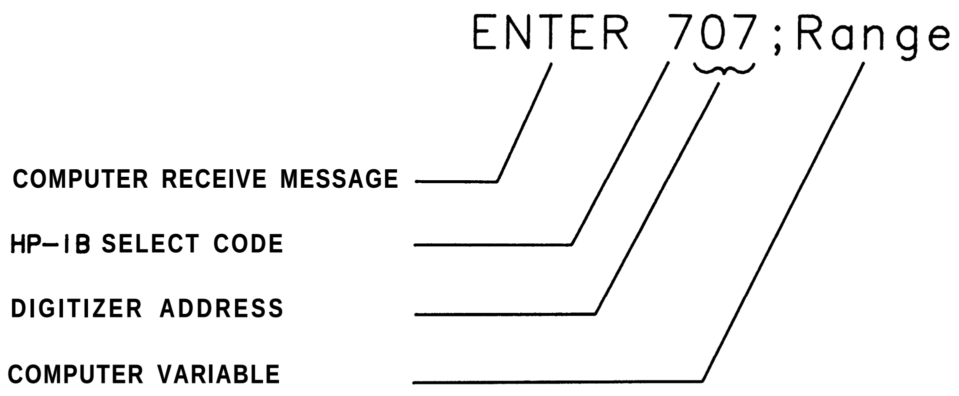
An OUTPUT statement tells the computer to send a message to the digitizer. For example, executing the output statement below sets the time range to 10 µs:



An ENTER statement used in conjunction with a digitizer query returns information to the computer. To return the time range value to the computer, first form a query by adding a question mark (?) to the command:



Next, the enter statement can be used to assign the returned value to a variable in the computer:



The value of the time range above is equated to the computer variable “Range”. The variable may be printed, stored, or used for any other computer function.

Syntax Requirements

All of the program examples in this manual show recommended command syntax. All digitizer commands must be constructed according to specific syntactical rules which are outlined in Chapter 3, Language Reference. The Language Reference lists all of the remote digitizer commands in alphabetical order according to each functional group subsystem, and contains a syntax diagram for each subsystem.

Local and Remote Control

Whenever the digitizer is remotely addressed, the display front-panel softkeys are disabled. Pressing the [LOCAL] key or executing the HP BASIC command “LOCAL” reenables operation of the softkeys.

INITIAL PROGRAM CONSIDERATIONS

Programs should begin with a series of HP BASIC and digitizer commands that form a good starting point for digitizer measurements. The following example shows how to initialize the digitizer to form a good starting point.

```
10 ASSIGN @DIG to 707
20 CLEAR @DIG
30 OUTPUT @DIG;"*RST;TIM:MODE ASIN;RUN;"
```

The ASSIGN command is an HP BASIC command that creates an I/O path name and assigns that name to an I/O resource. In the example above, the I/O path name is "@DIG" and is assigned to the device at HP-IB address 7. (Note: all program examples in this manual assume that the digitizer is addressed at HP-IB address 707.)

The ASSIGN command offers several advantages when included in a digitizer program. For example, the digitizer address is easily changed in the computer program and the program can transfer data to a mass storage unit.

The RESET command, *RST, presets all of the analog parameters of the digitizer and provides a good starting point for all measurement processes. Executing *RST is actually the same as executing a number of digitizer commands that set the digitizer to a known state.

The CLEAR command is an HP BASIC command that clears the input buffer, the output buffer, and the command parser of the specified instrument; that is, a device on HP-IB is "cleared" so that it is ready for operation. This command may be used to clear devices on the bus singly or in unison. It is often desirable to reset only one instrument so that other instruments on the bus are not affected.

To clear the digitizer, the "CLEAR @DIG" statement may be entered into the computer.

To clear all devices at select code 7, the "CLEAR 7" statement may be entered into the computer.

STATUS REPORTING STRUCTURE

This section describes and defines the status reporting structure used in the HP 70700A Digitizer. In general, the status data structure is used to "request service" or indicate a specific condition (e.g., operation complete) via SRQ (Service Request). This structure may be used to alert the user that certain events have occurred without the user's actually initiating a request for this information. Refer to Figure 2-3 for a model of the status data structure.

Each of the integral parts of the status data structure are described below in more detail.

STATUS BYTE REGISTER

The Status Byte Register contains the device's Status Byte (STB) summary messages, Request Service (RQS) messages, and Master Summary Status (MSS) messages. See Figure 2-4.

The Status Byte Register can be read with either a serial poll or the READ STATUS BYTE common query (*STB?). Both of these methods read the status byte message identically. However, the value sent for the bit 6 position depends on the method used.

When the Status Byte Register is read with a serial poll "SPOLL (@DIG)", the device returns the seven-bit status byte message plus the single-bit RQS message to the controller as a single data byte.

When the Status Byte Register is read with the *STB? common query, the device returns the seven-bit status byte message plus the single-bit MSS message as a single <NR1 NUMERIC PROGRAM DATA> element. The response to *STB? is identical to the response to a serial poll except that the MSS summary message appears in the bit 6 position in place of the RQS message. The MSS summary message indicates that the device has at least one reason for requesting service.

Standard Event Status Bit (ESB) Summary Message

The ESB summary message is a defined message which appears in bit 5 of the Status Byte Register. Its state indicates whether or not one or more of the enabled events have occurred since the last reading or clearing of the Standard Event Status Register. Refer to Figure 2-3.

The ESB summary message is TRUE when an enabled event in the Standard Event Status Register is set TRUE. Conversely, the ESB summary message is FALSE when no enabled events are TRUE.

Message Available (MAV) Queue Summary Message

The MAV summary message is a defined message which appears in bit 4 of the Status Byte Register. The state of the message indicates whether or not the Output Queue is empty. Whenever the device is ready to accept a request by the controller to output data bytes, the MAV summary message shall be TRUE. The MAV summary message shall be FALSE when the Output Queue is empty. Refer to Figure 2-3.

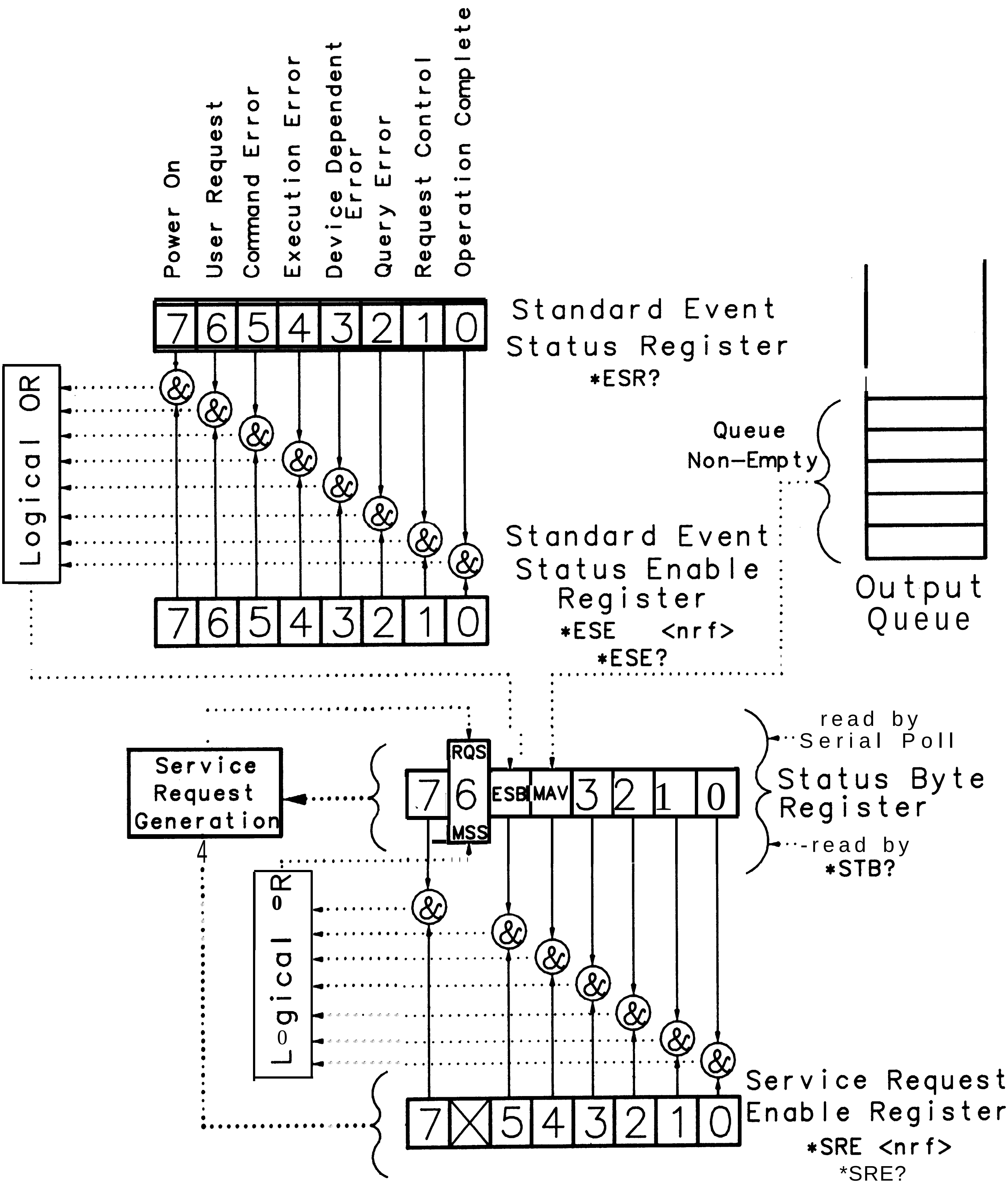


Figure 2-3. Status Data Structure

The MAV summary message is used to synchronize information exchange with the controller. The controller can, for example, send a query command to the device and then wait for MAV to become TRUE. The system bus is available for other use while an application program is waiting for a device to respond. If an application program begins a read operation of the Output Queue without first checking for MAV, all system bus activity is held up until the device responds.

NOTE

Bits 0 through 3 and bit 7 are not used at this time.

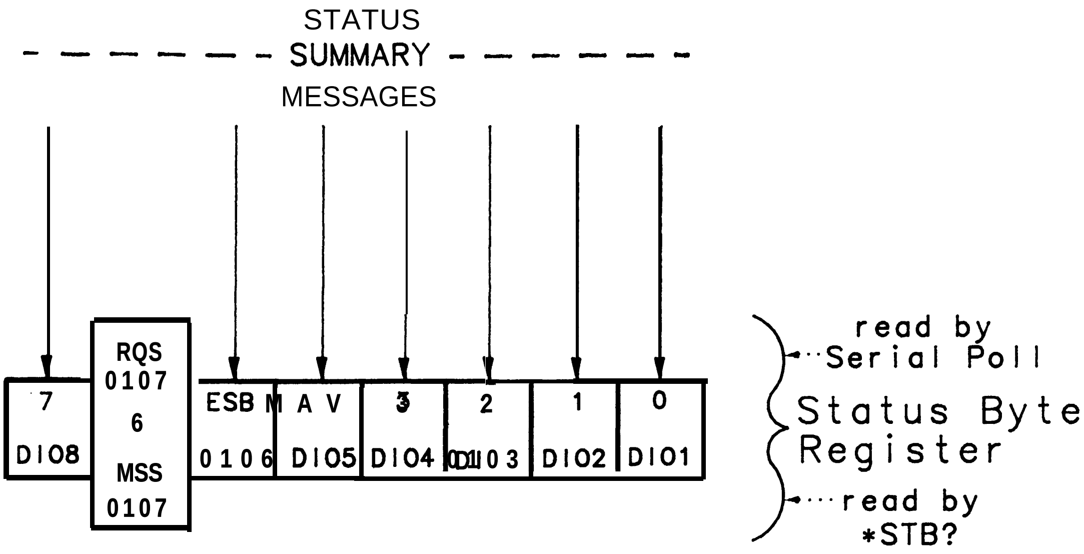


Figure 2-4. Status Byte Register

Clearing the Status Byte Register

The CLEAR STATUS (*CLS) common command causes the Event Registers and Queues of the status data structure to be cleared so that the corresponding summary messages are clear. The Output Queue and its MAV summary message are an exception and are unaffected by *CLS.

SERVICE REQUEST ENABLE REGISTER

The Service Request Enable Register is an eight-bit register that can be used by the programmer to select which summary messages in the Status Byte Register may cause service requests. The programmer may select reasons for the device to issue a service request by altering the contents of the Service Request Enable Register. Refer to Figure 2-5.

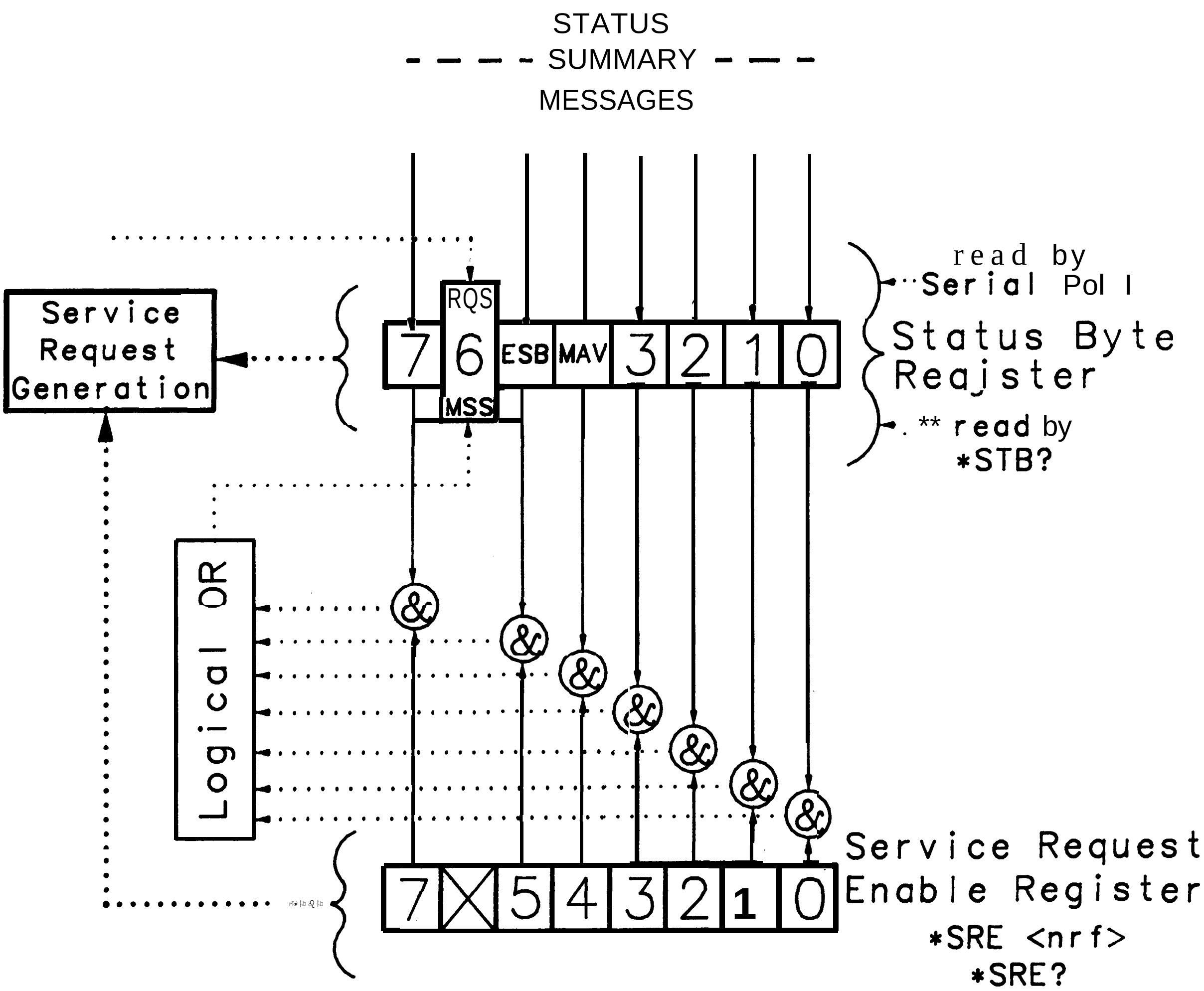


Figure 2-5. Service Request Enable Register

Reading/Writing the Service Request Enable Register

The Service Request Enable Register is read with the SERVICE REQUEST ENABLE (*SRE?) common query. The response message to this query is a **<NR1** NUMERIC PROGRAM DATA> element that represents the sum of the binary-weighted values of the Service Request Enable Register (2 raised to the power of the bit number). The value of unused bit 6 is always zero.

The Service Request Enable Register is written to with the SERVICE REQUEST ENABLE **(*SRE)** common command followed by a <DECIMAL NUMERIC PROGRAM DATA> element. The <DECIMAL NUMERIC PROGRAM DATA>, when rounded to an integer value and expressed in base two (binary), represents the bit values of the Service Request Enable Register. A bit value of one indicates an enabled condition. A bit value of zero indicates a disabled condition.

NOTE

The device always ignores the value of bit 6.

Service Request Generation

The Service Request function provides the device with the ability to request service from the controller via the Service Request interface (SRQ, a line on HP-IB), and report that it has requested service via the Request Service message (RQS, bit 6 of the Status Byte Register).

The generation of service requests ensures that the device shall:

1. Assert an SRQ when a previously "enabled" condition occurs.
2. Keep SRQ asserted until the controller has recognized the service request and polled the device, or has taken specific action to cancel the request (e.g., *CLS command).
3. Release SRQ when polled so that the controller can detect an SRQ from another device.
4. Assert an SRQ again if another condition occurs, whether or not the controller has cleared the first condition. If the previous condition has not been cleared, the next condition must be different than the first for an SRQ to be asserted again.

Whenever the contents of the Status Byte Register or the Service Request Enable Register are changed, the device must determine whether the change affects the service request state of the device. Device status transitions do not affect the state of the SRQ interface directly. Instead, changes to the Status Byte Register and the Service Request Enable Register generate two local messages which either assert (Request Service TRUE) or unassert (Request Service FALSE) the hardware.

The device shall generate a new service request (assert Request Service TRUE) when:

1. A bit in the Status Byte Register changes from FALSE to TRUE while the corresponding bit in the Service Request Enable Register is TRUE.
2. A bit in the Service Request Enable Register changes from FALSE to TRUE while the corresponding bit in the Status Byte Register is TRUE.
3. A bit in the Status Byte Register changes from FALSE to TRUE and the corresponding bit in the Service Request Enable Register changes from FALSE to TRUE simultaneously.

In general, the controller application program must never assume that an SRQ indicates that a new reason for service has occurred, but only that it may have occurred. The application program should check the device Status Byte Register to determine whether it is indeed the case that a new reason for service exists.

Clearing the Service Request Enable Register

The SERVICE REQUEST ENABLE (*SRE) common command followed with a <DECIMAL NUMERIC PROGRAM DATA> element value of zero clears the Service Request Enable Register. A cleared register does not allow status information to generate a hardware Request Service message; thus, no service requests are issued.

STANDARD EVENT STATUS REGISTER

The Standard Event Status Register structure has specific, defined events assigned to specific bits. Refer to Figure 2-6.

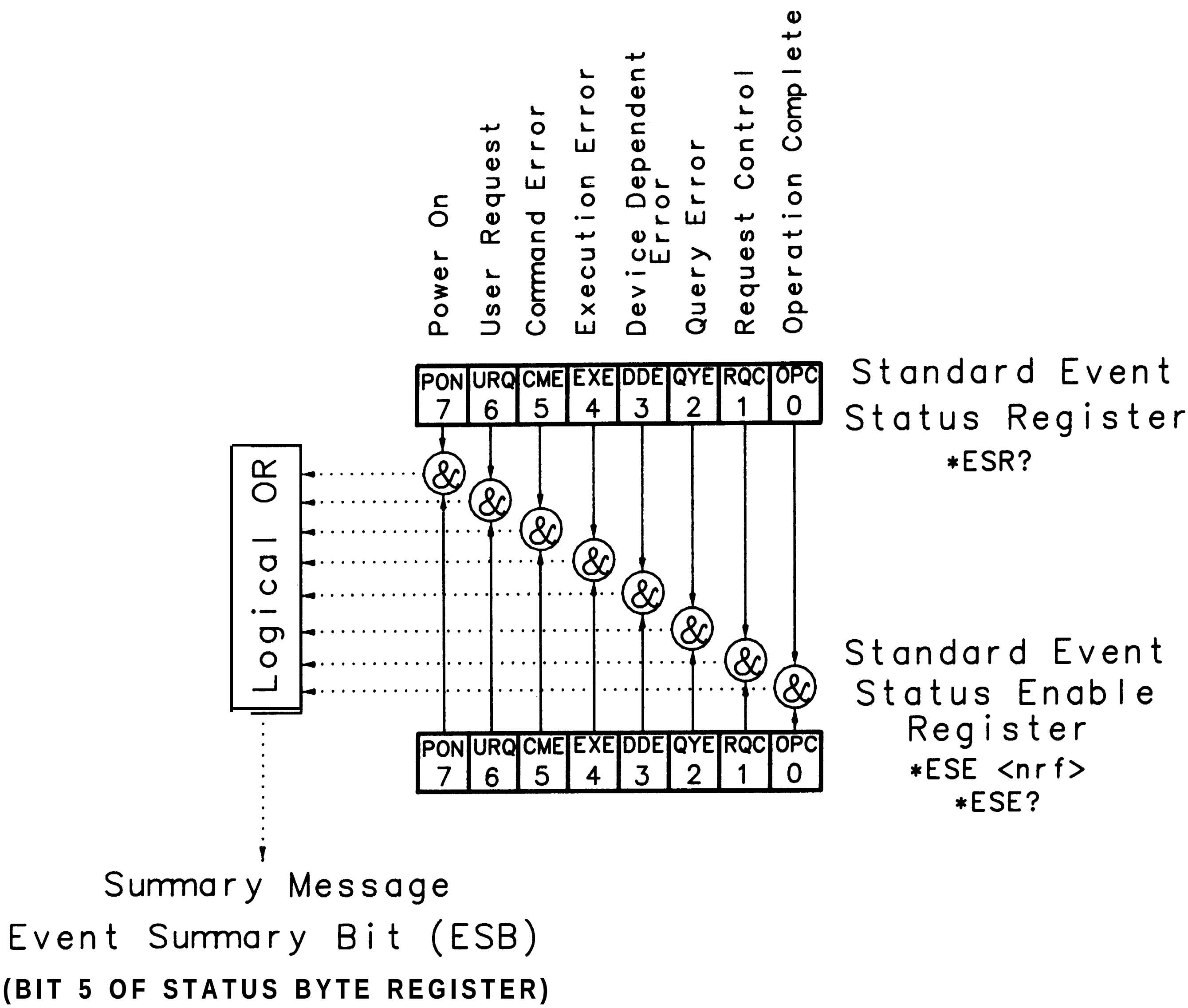


Figure 2-6. Standard Event Status Register

Bits 7 through 0

Bit 7 — Power On (PON) This event bit indicates that an off-to-on transition has occurred in the power supply of the device.

Bit 6 — User Request **(URQ)** This event bit is not used at this time.

Bit 5 — Command Error (CME) This event bit indicates that one of the following events has occurred:

1. A syntax error (controller-to-device message) has been detected. Possible errors include a data element which violates the device listening formats or whose type is unacceptable to the device.

2. A semantic error has occurred, indicating that an unrecognized header was received. Unrecognized headers include incorrect device-dependent headers and incorrect or unimplemented common commands.
3. A Group Execute Trigger (GET) was entered into the Input Buffer inside of a <PROGRAM MESSAGE>. A GET message is a controller-to-device message defined as an addressed command.

The Command Error bit is not set to report any other device-dependent condition. Events that are reported as Command Errors are not reported as Execution Errors, Query Errors, or Device-Dependent Errors. Refer to the appropriate bit definitions for more information.

Bit 4 – Execution Error (EXE) This event bit indicates that:

1. A <PROGRAM DATA> element following a header was evaluated by the device as outside of its legal input range, or is otherwise inconsistent with the capabilities of the device.
2. A valid program message could not be properly executed due to some device condition.

Following an Execution Error, the device continues to process the input stream.

Execution Errors are reported by the device after rounding and expression-evaluation operations have taken place, For example, rounding a numeric data element is not reported as an Execution Error.

Events that generate Execution Errors do not generate Command Errors, Query Errors, or Device-Dependent Errors. Refer to the appropriate bit definitions for more information.

Bit 3 – Device-Dependent Error (DDE) This event bit indicates that an error has occurred which is neither a Command Error, a Query Error, nor an Execution Error.

A Device-Dependent Error is any executed device operation that was not properly completed due to some condition, such as overrange.

Following a Device-Dependent Error, the device continues to process the input stream.

Events that generate Device-Dependent Errors do not generate Command Errors, Query Errors, or Execution Errors. Refer to the appropriate bit definitions for more information.

Bit 2 – Query Error (QYE) This event bit indicates that:

1. An attempt is being made to read data from the Output Queue when no output is either present or pending.
2. Data in the Output Queue has been lost.

The Query Error bit is not set to report any other condition. Events that generate Query Errors do not generate Execution Errors, Command Errors, or Device-Dependent Errors.

Bit 1 – Request Control (RQC) This event bit is not used at this time.

Bit 0 – Operation Complete (OPC) This event bit responds to the OPERATION COMPLETE (*OPC) common command. It indicates that the device has completed any pending operations and that the parser is ready to accept more program messages. The parser is the logical portion of the device which takes Data Byte Messages, END messages, and hardware Group Execute Trigger messages from the Input Buffer and analyzes them by separating out the various syntactic elements.

Reading/Writing the Standard Event Status Register

The Standard Event Status Register is destructively read (that is, read and cleared) with the STANDARD EVENT STATUS REGISTER common query (*ESR?).

The Standard Event Status Register cannot be written to remotely.

Clearing the Standard Event Status Register

The Standard Event Status Register shall only be cleared by:

1. A CLEAR STATUS common command (*CLS).
2. A power-on sequence which initially clears the Standard Event Status Register then records any subsequent events during the power-on sequence of the device, including setting the PON event bit (bit 7).

STANDARD EVENT STATUS ENABLE REGISTER

The Standard Event Status Enable Register allows one or more events in the Standard Event Status Register to be reflected in the ESB summary-message bit (bit 5 of the Status Byte Register). This register is defined for eight bits, each corresponding to the bits in the Standard Event Status Register. Refer to Figure 2-6.

Reading/Writing the Standard Event Status Enable Register

The Standard Event Status Enable Register is read with the STANDARD EVENT STATUS ENABLE common query (*ESE?). Data is returned as a binary-weighted <NR1 NUMERIC RESPONSE DATA>.

The Standard Event Status Enable Register is written to by the STANDARD EVENT STATUS ENABLE common command (*ESE). Data is encoded as <DECIMAL NUMERIC PROGRAM DATA>.

Clearing the Standard Event Status Enable Register

The Standard Event Status Enable Register shall be cleared by the following:

1. Sending the *ESE common command with a data value of zero.
2. A power-on event.

The Standard Event Status Enable Register is specifically not affected by the RESET common command (*RST).

OUTPUT QUEUE

The Output Queue stores response messages until they are read. The availability of the output is summarized by the Message Available (MAV) summary message (bit 4 of the Status Byte Register). The MAV summary message is used to synchronize information exchange with the controller. Refer to Figure 2-7.

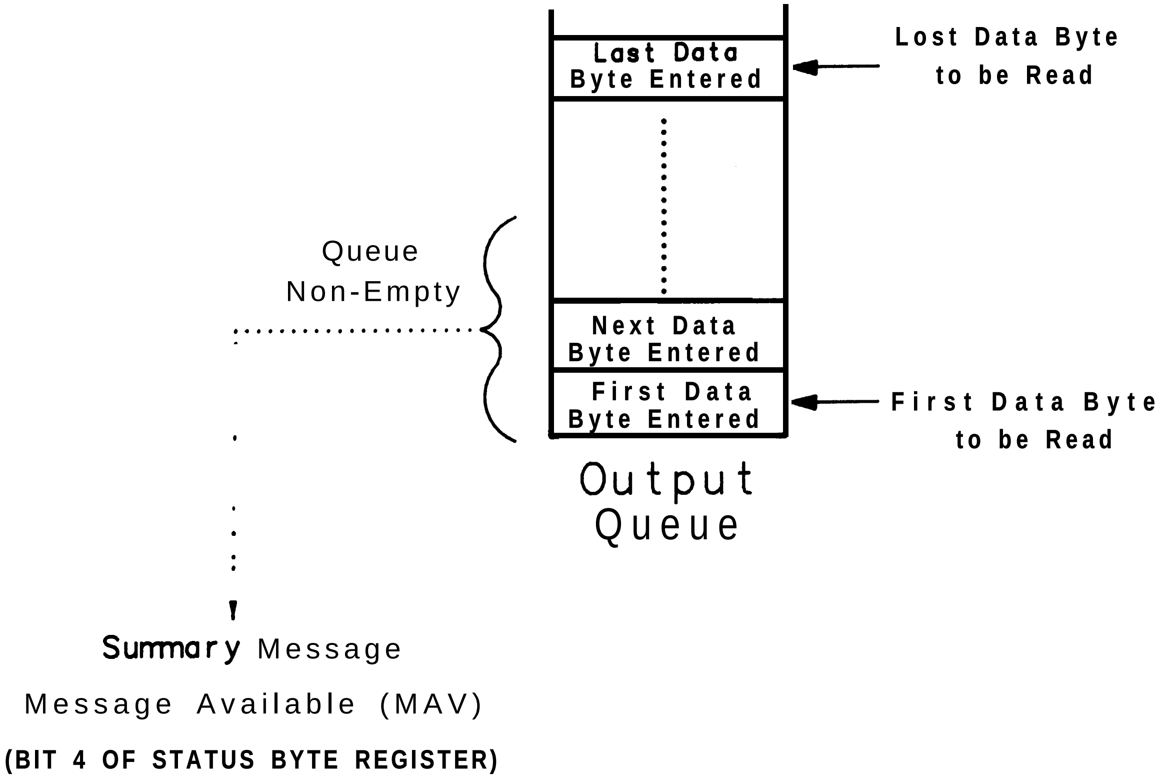


Figure 2-7. Output Queue

The Response Formatter places Data Byte Messages and END messages into the Output Queue in response to query commands. These bytes are removed from the Output Queue as they are read by the controller. As long as the Output Queue contains one or more bytes, MAV is TRUE.

The Output Queue is cleared upon power-on, Device Clear Active State Message (dcas), or the RESET **(*RST)** common command, without causing a Query Error. A Query Error is generated if the contents of the Output Queue are discarded for any other reason.

SYNCHRONIZATION OF EVENTS AND COMMANDS

This section describes techniques which may be used to ensure synchronization between events and commands, which in turn ensures valid measurements.

A potential problem with commands that take appreciable time to finish is that the application program needs to know when the commands have finished; therefore, to make measurements remotely, the controller must know when trace data is available to the digitizer before making a measurement. This potential problem can be avoided by using synchronization commands that instruct the module to wait until a measurement is finished before making any other measurements.

For example, consider a data-logging device which is commanded to take a measurement with the -RUN command and then to make a frequency measurement using MEAS:FREQ?. The RUN command is a command that allows execution of subsequent commands while the device operations initiated by the RUN command are still in progress. The RUN command therefore takes appreciable time to perform. Figure 2-8 shows a timing diagram of this operation without the use of a synchronization command.

It should be noted that, without the use of a synchronization command, the RUN command is still in progress when the FREQUENCY measurement is initiated. If the RUN command has not completed its operation, any data used to make other measurements (e.g., FREQUENCY) while the RUN command is still in progress may be invalid. Only after an operation has been completed can the data be assumed to be valid.

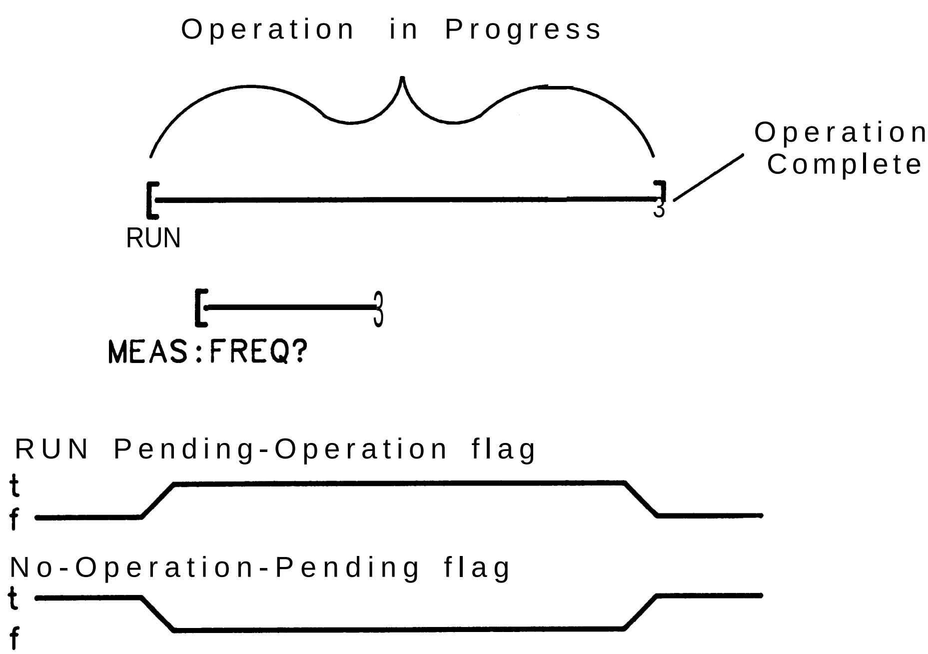


Figure 2-8. Timing Diagram Without A Synchronization Command

Figure 2-9 illustrates the same example as described above except that a synchronization command is used. It should be noted that, with the use of a synchronization command, the RUN command has completed its operation before the FREQUENCY measurement is initiated; therefore, the data used to make the FREQUENCY measurement is valid.

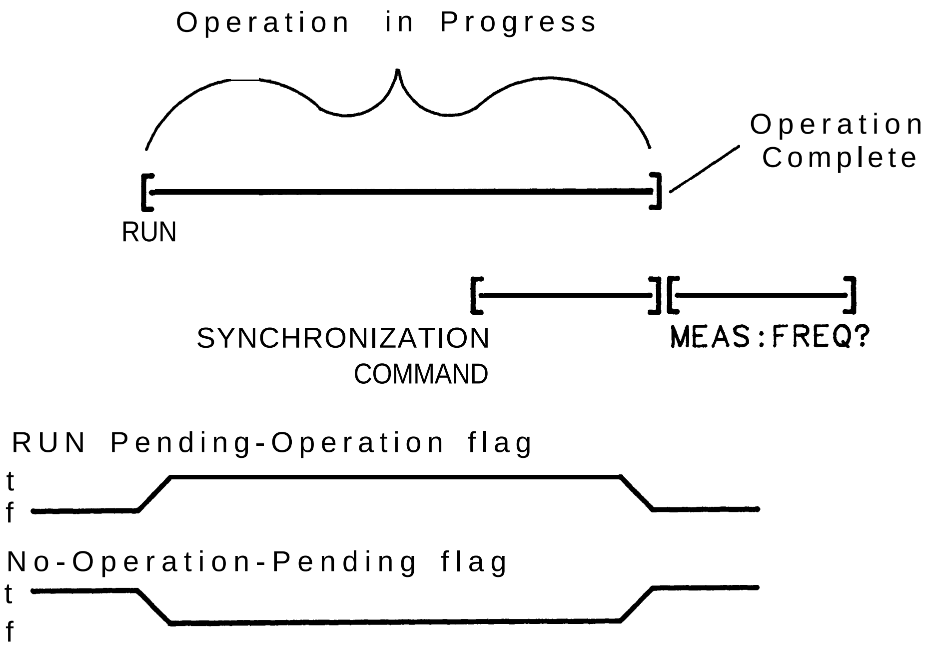


Figure 2-9. Timing Diagram Using A Synchronization Command

GENERAL SEQUENCE

The following example HP BASIC statements illustrate the general sequence that should be followed when making measurements via remote control.

1. OUTPUT @DIG;"*RST;TIM:MODE ASIN;..."

   RESET the module, put it into SINGLE SWEEP mode, then set up all desired parameters (e.g., voltage range, time range, etc.).

2. OUTPUT @DIG;"RUN"

   Invoke a sweep. (In SINGLE SWEEP mode, a single measurement will be taken.)

3. SYNCHRONIZATION COMMAND

   Use a synchronization command to allow the module to complete one operation before starting another. Refer to Synchronization Commands below for more detailed information on the three forms of synchronization commands that are available.

4. OUTPUT @DIG;"MEAS:SOUR CHAN 1;FREQ?" ENTER @DIG;Frequency

   Ask for and enter measurement data.

SYNCHRONIZATION COMMANDS

There are three forms of synchronization commands that can be used in the general sequence for remote measurements listed above. The three forms are listed and described below.

Wait-To-Continue **(*WAI)** This command instructs the module to not use the bus until all operations are complete. If using triggered sweeps, this command may cause the bus to "wait" indefinitely if the trigger does not occur.

Operation Complete (*OPC) This command instructs the module to send a **"1"** to the Output Queue of the status data structure when all operations are complete. Similar to the ***WAI** command, the *OPC command may result in the bus's waiting indefinitely for an event to occur.

Assert **SRQ** (Service Request) This form of synchronization uses a sequence of commands indicating that after the *OPC command is received, the module is to assert SRQ when all operations are complete. This method has the advantage of the HP-IB's never being in a state where it is waiting indefinitely. The sequence of commands for this synchronization method is:

```
OUTPUT @DIG;"*ESE 1;"        ! This command only needs to be sent once
OUTPUT @DIG;"*SRE 32;"       ! This command only needs to be sent once
OUTPUT @DIG;"*CLS;*OPC;"
REPEAT
UNTIL BIT (6,SPOLL (dig))=1
```

NOTE

"All operations complete" is defined as: 1) there is no trace in progress, including reading trace data (which is NOT the same as trace complete, since a trace may not have been started); and 2) all remote commands have been parsed and processed.

NOTE

Because reading trace data is an operation, the synchronization commands should not be used after the request of trace data **(e.g., WAV:DATA?;** or DIG **CHAN1;)** until all data has been read. If the synchronization commands are used, the system may be placed in a state where it is "waiting" indefinitely. In this state, the digitizer system waits for the data to be read, and the computer waits for all operations to be completed.

DATA TRANSFER

The HP 70700A Digitizer represents trace data and non-trace data differently. For non-trace data, the digitizer will send information back in ASCII character code as either text or numeric data. As an example of ASCII text, the **Timebase** REFERENCE query (TIM:REF?) returns **LEFT|CENT|RIGH** as a response. As an example of ASCII numeric data, the **Timebase** RANGE query (TIM:RANG?) returns a value X.XXEXX as a response.

NOTE

Some numeric responses are integers (i.e., they have no decimal point or exponent).

For trace data, the digitizer currently supports only **16-bit** binary transfers (**MSByte** first, **LSByte** second). The trace data is preceded by a header to indicate that binary data is about to be received. The header is **"#0"**, which indicates indeterminate length binary data. When using the digitizer via HP-IB, the last data byte will have the EOI (END OR INTERRUPT) status line asserted with it.

Listed below is an HP BASIC example program that reads trace data from the digitizer.

```
OUTPUT @DIG;"WAV:DATA?"                    ! Request trace data.
ENTER @DIG USING "#,2A",Header$            ! Read the header.
FOR J=1 TO Points
   ENTER @DIG USING "#,W";Value(J)         ! Read each point.
NEXT J
```

MULTIPLE DIGITIZER REMOTE SLAVING PROCEDURE

A digitizer system, consisting of one to eight modules, may be remotely controlled or "slaved" via a computer. In this mode of operation, the computer functions as the "master" or controlling device and each channel is essentially a slave. This allows more than four channels as well as provides features not accessible through a normal four-channel digitizer system (e.g., a different timebase for each channel). Also in this mode of operation, one digitizer module must still be designated as the reference channel which controls the triggering of all the digitizer channels. More than one digitizer *system* is also possible, but only one module for each designated digitizer system can control the triggering in a slaved digitizer system that is remotely controlled. For example, two four-channel digitizer systems may reside in the same mainframe. However, one channel in each system must be designated as the reference channel for triggering the other three channels.

The commands used to effect this mode of operation are:

TRIG:SOUR SLAV
MCABT
MCSET
MCSNC
MCARM
MCINF
MCTRG
MCREC

The "MC" in these command mnemonics stands for "multiple channel". The remainder of this section explains how these commands are used in multiple-channel digitizing.

These "multiple channel" commands differ from the other digitizer commands in that they must be employed strictly in a particular sequence. Each step in this sequence must be completed in the module before continuing, The OPC (Operation Complete) bit in the Standard Event Status Register is used to indicate the completion of a step. Corresponding lines of program code from an example HP BASIC program (listed at the end of this section) are shown in each step of the procedure for example purposes. It is not intended that these lines of program code be used by themselves, but that they may be used as part of the whole program example listed at the end of this section.

CAUTION

The commands described in this section must be used exactly as presented. Misuse of these commands may put the modules into a state that can only be recovered from by cycling power.

The first three steps of the procedure are a sequence for slaving multiple modules to the controller (i.e., one module is designated as the reference channel for the other modules), and must be performed only once when the system is initialized. The remaining steps allow the user to set up measurement parameters, and synchronize and trigger the system to obtain the sampled data.

1. Configure the address switches of the modules and properly install the cabling. The cabling for this mode is the same as for a normal multiple-channel digitizer system. The CLOCK OUT of the reference channel is connected to the CLOCK IN of all the other channels. The SYNC OUT of the reference channel is connected to the EXT TRIG of ail the other channels.

NOTE

Each module needs a unique **HP-IB** address so that it may be accessed individually; therefore, the address of each module must be configured in row 0 and at a different column address.

2. Reset all modules, stop all modules from making measurements (allowing the computer to perform the appropriate setup), and put all modules in SINGLE SWEEP mode (which is required since the computer must perform operations between sweeps):

```
REFERENCE CHANNEL:   *RST;STOP;TIM:MODE ASIN|SING;
   OTHER CHANNELS:   *RST;STOP;TIM:MODE SING;
```

Program line examples:

```
OUTPUT @Ref;"*RST;STOP;TIM:MODE ASIN;*WAI;"
OUTPUT Khans; '*RST;STOP;TIM:MODE SING;*WAI;'
```

3. Use this command sequence to inform the channels that they are being controlled from an external source:

```
OTHER CHANNELS: TRIG:SOUR SLAV;
```

Program line example:

```
OUTPUT @Chans;"TRIG:SOUR SLAV;"
```

4. In this step, each module is set up with the desired measurement parameters (e.g., VOLTS/DIVISION, SECONDS/DIVISION, etc.):

NOTE

If the **timebase** settings are not the same on all channels, then only simple digitizing can be performed. Random Event Capture (REC), interpolation, and other features are not available in this case. System operation is not guaranteed if different **timebase** settings are used.

```
REFERENCE CHANNEL/CHANNELS:  set up parameters (only parameters that
                             have changed from the previous measure-
                             ment need to be sent)
```

Program line examples:

```
N=200
OUTPUT @Ref;"CHAN1:RANG 2.0V;TIM:RANG 10us;ACQ:POIN";N
OUTPUT @Chan1;"CHAN1:RANG 2.0V;TIM:RANG 10us;ACQ:POIN";N
OUTPUT @Chan2;"CHAN1:RANG 3.0V;TIM:RANG 10us;ACQ:POIN";N
```

5. Use this command sequence only if using REC or aborting a trace that is waiting for a trigger:

NOTE

Most of the multi-channel system commands begin with the letters **"MC".**

REFERENCE CHANNEL/CHANNELS: *CLS;MCABT;

The OPC bit in the Standard Event Status Register is set at this time. Do not use the *OPC or *WAI commands; they will not function as expected.

6. In this step, the reference channel sets its hardware to take a measurement. This is the first step in synchronizing all the other channels with the reference channel. (The trigger for the measurement is not armed at this time.):

REFERENCE CHANNEL: *CLS;MCSET;

The OPC bit in the Standard Event, Status Register is set at this time. Do not use the *OPC or *WAI commands; they will not function as expected.

Program line example:

```
OUTPUT @Ref;"*CLS;MCSET;"
```

7. In this step, all channels set their hardware to take a measurement and are ready to be synchronized with the reference channel:

OTHER CHANNELS: *CLS;MCSET;

The OPC bit in the Standard Event Status Register is set at this time. Do not use the *OPC or *WAI commands; they will not function as expected.

Program line example:

```
OUTPUT @Chans;"*CLS;MCSET;"
```

8. In this step, the reference channel completes its synchronization with the other channels:

REFERENCE CHANNEL: *CLS;MCSNC;

The OPC bit in the Standard Event Status Register is set at this time. Do not use the *OPC or *WAI commands; they will not function as expected.

Program line example:

```
OUTPUT @Ref;"*CLS;MCSNC;"
```

9. The digitizer action for all channels, except the reference channel, is armed at this time. It should be noted that since the triggering is performed through the reference channel, the digitizer action is still effectively unarmed:

OTHER CHANNELS: *CLS;MCARM;

The OPC bit in the Standard Event Status Register is set at this time. Do not use the *OPC or *WAI commands; they will not function as expected.

Program line example:

```
OUTPUT @Chans;"*CLS;MCARM;"
```

10. The trace acquisition for the reference channel is armed at this time. At this point, when the trigger condition is satisfied, a measurement will be made:

REFERENCE CHANNEL: *CLS;MCARM;

Step 10 should be carried out soon enough after step 9 so that the triggers armed in step 9 do not “time out”. If necessary, lengthen the time-out times using the TRIG:TOUT command.

The OPC bit in the Standard Event Status Register is set at this time. Do not use the *OPC or *WAI commands; they will not function as expected.

Program line example:

```
OUTPUT @Ref;"*CLS;MCARM;"
```

I I. The sampled data for the reference channel is available at this time. It can be obtained as follows:

REFERENCE CHANNEL: WAV:DATA?

The *OPC or *WAI commands may be used at this time.

Program line examples:

```
OUTPUT @Ref;"*WAI;"
OUTPUT @Ref;"WAV:DATA?"
ENTER @Ref USING "#,2A";A$
For J=O TO N-l
   ENTER @Ref USING"#,W";Value
NEXT J
```

12. When using REC mode, use the following command sequence to transfer information from the reference channel to the other channels:

```
REFERENCE CHANNEL: MCINF?
                   ENTER @Ref;Datl,Dat2,Dat3,Dat4

   OTHER CHANNELS: *CLS;MCINF Datl,Dat2,Dat3,Dat4
```

The command form "MCINF Datl,Dat2,Dat3,Dat4" sets the OPC bit in the Standard Event Status Register at this time. Do not use the ***OPC** or ***WAI** commands; they will not function as expected,

Program line examples:

```
OUTPUT @Ref;"MCINF?"
ENTER @Ref; Datl,Dat2,Dat3,Dat4
OUTPUT Khans;"*CLS;MCINF";Datl,Dat2,Dat3,Dat4
```

13. If interpolation is on, indicated by an "i" appearing on the display screen or querying the system remotely, information about the trigger must be transferred from the reference channel to the other channels. If interpolation is off, this command form is effectively a no-operation instruction (i.e., it may or may not be used).

```
REFERENCE CHANNEL: MCTRG?
                   ENTER. @Ref;Trigger_point

   OTHER CHANNELS: MCTRG Trigger_point
```

NOTE

If the other channels (not the reference channel) have a link with a display Instrument, the channel does not walt for this command before drawing Its trace to the display screen. As a result, there may be up to **50** ns of "jitter" in the data of the channel, resulting from a lack of trigger timing Information. The data acquired by the computer will be correct (i.e., it will not necessarily match the data on the display screen).

Program line examples:

```
OUTPUT @Ref;"MCTRG?"
ENTER @Ref;Temp
OUTPUT @Chans;"MCTRG";Temp
```

14. The sampled data from all the other channels (not the reference channel) is available at this time. It should be noted that each channel must be addressed individually.

```
OTHER CHANNELS: WAV:DATA?
```

The ***OPC** or ***WAI** commands may be used at this time.

Program line examples:

```
OUTPUT @Chan1;"*WAI;"
OUTPUT @Chan1;"WAV:DATA?"
ENTER @Chan1 USING "#,2A";A$
FOR J=O TO N-1
   ENTER @Chan1 USING "#,W";Value
NEXT J

OUTPUT @Chan2;"*WAI;"
OUTPUT @Chan2;"WAV:DATA?"
ENTER Khan2 USING "#,2A";A$
FOR J=O TO N-l
   ENTER @Chan2 USING "#,W";Value
NEXT J

     .
     .      (repeat for each channel)
     .
```

A sample HP BASIC program utilizing the remote slaving procedure is provided at the end of this section.

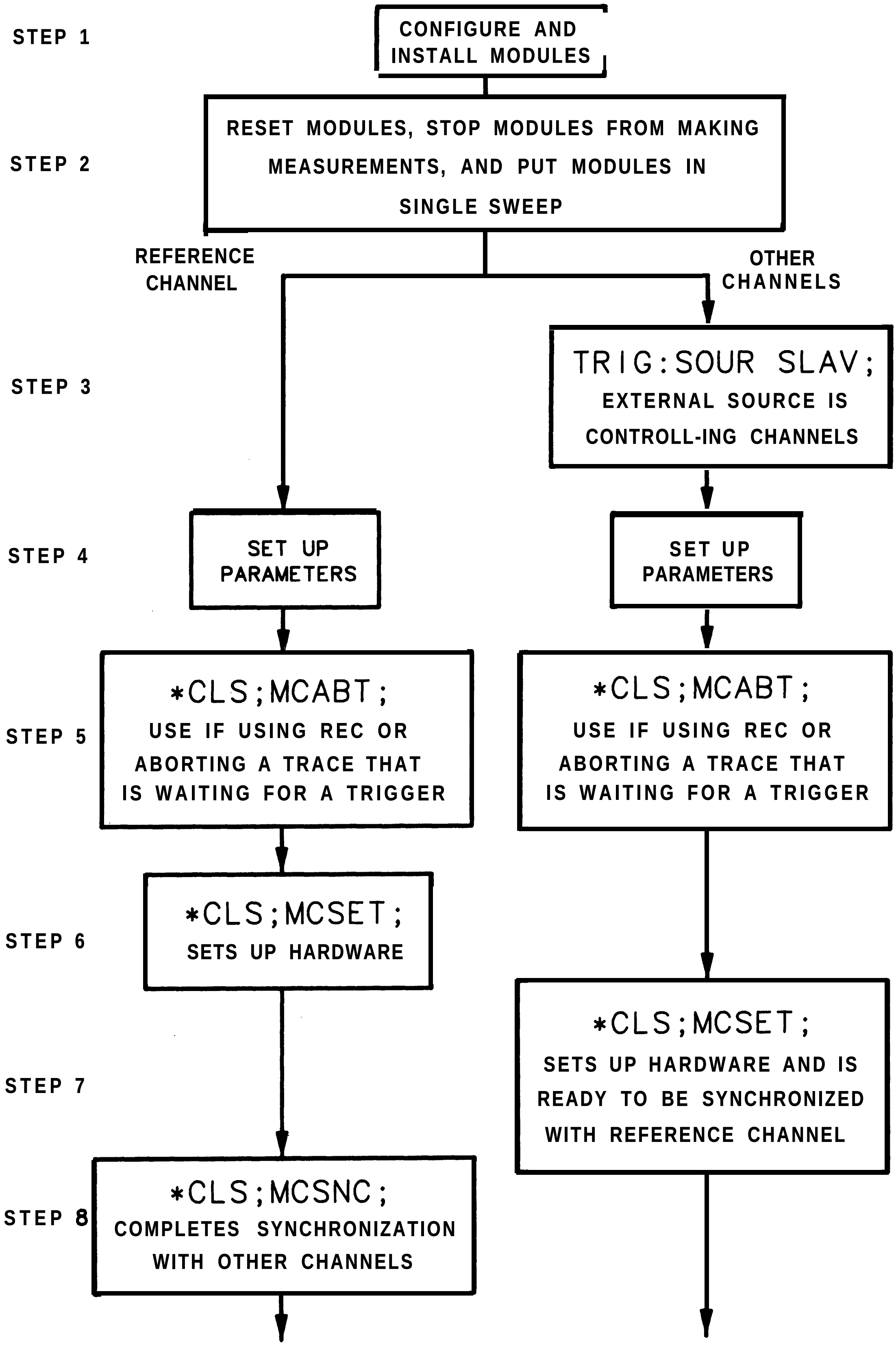


Figure 2-10. Remote Slaving Proceclure Diagram (1 of 2)

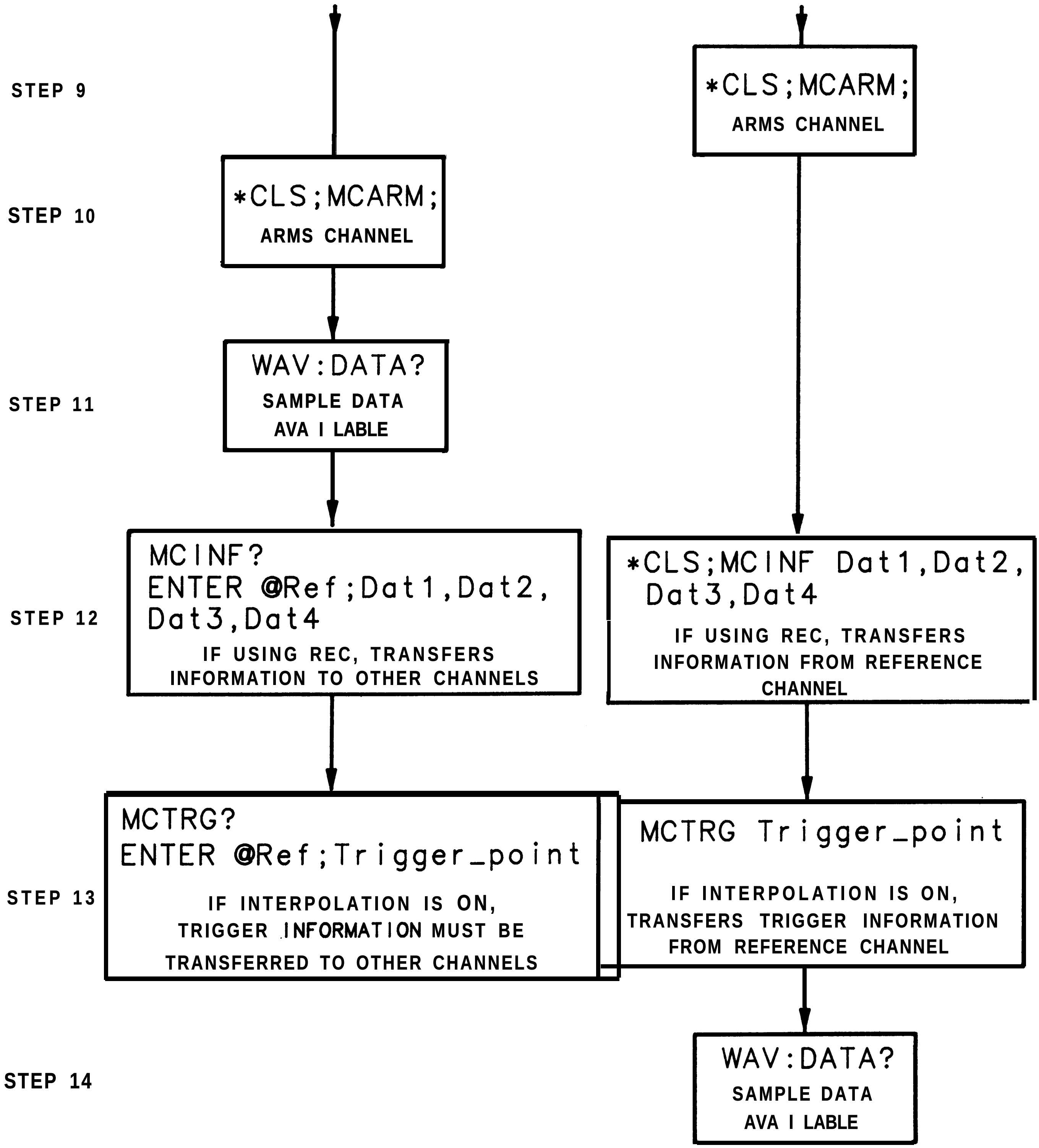


Figure 2-1 0. Remote Slaving Procedure Diagram (2 of 2)

EXAMPLE PROGRAM

```
10    ASSIGN @Ref TO 707
20    ASSIGN @Chans TO 705,706
30    ASSIGN @Chan1 TO 705
40    ASSIGN @Chan2 TO 706
50    !
60    ! STEP 2
70    !
80    OUTPUT @Ref;"*RST;STOP;TIM:MODE ASIN;*WAI;"
90    OUTPUT @Chans;"*RST;STOP;TIM:MODE SING;*WAI;"
100   !
110   ! STEP 3
120   !
130   OUTPUT @Chans;"TRIG:SOUR SLAV;"
140   !
150   ! STEP 4
160   !
170   N=200
180   OUTPUT @Ref;"CHAN1:RANG 2.0V;TIM:RANG 10US;ACQ:POIN ";N
190   OUTPUT @Chan1;"CHAN1:RANG 2.0V;TIM:RANG 10US;ACQ:POIN ";N
200   OUTPUT @Chan2;"CHAN1:RANG 3.0V;TIM:RANG 10US;ACQ:POIN ";N
210   !
220   ! STEP 5
230   !
240   ! * only used for REC or aborting trace waiting for trigger *
250   !
260   ! STEP 6
270   !
280   OUTPUT @Ref;"*CLS;MCSET;"
290   !
300   ! STEP 7
310   !
320   OUTPUT @Chans;"*CLS;MCSET;"
330   !
340   ! STEP 8
350   !
360   OUTPUT @Ref;"*CLS;MCSNC;"
370   !
380   ! STEP 9
390   !
400   OUTPUT @Chans;"*CLS;MCARM;"
410   !
420   ! STEP 10
430   !
440   OUTPUT @Ref;"*CLS;MCARM;"
450   !
460   ! STEP 11
470   !
480   Graph(0,N,N/10,0,4096,512,"REMOTE SLAVED DIGs","time","ampl",3)
490   OUTPUT @Ref ;"*WAI;"
500   OUTPUT @Ref;"WAV:DATA?"
```

```
510    ENTER @Ref USING "#,2A";A$
520    MOVE -100,0
530    FOR J=0 TO N-1
540       ENTER @Ref USING "#,W";Value
550       DRAW J,Value
560    NEXT J
570    !
580    ! STEP 12
590    !
600    ! * only used for REC *
610    !
620    ! STEP 13
630    !
640    OUTPUT @Ref;"MCTRG?"
650    ENTER @Ref;Temp
660    OUTPUT @Chans;"MCTRG";Temp
670    !
680    ! STEP 14 (repeat for each channel)
690    !
700    OUTPUT @Chan1;"*WAI;"
710    MOVE -100,0
720    OUTPUT @Chan1;"WAV:DATA?"
730    ENTER @Chan1 USING "#,2A";A$
740    FOR J=0 TO N-1
750       ENTER @Chan1 USING "#,W";Value
760       DRAW J,Value
770    NEXT J
780    !
790    OUTPUT @Chan2;"*WAI;"
800    MOVE -100,0
810    OUTPUT @Chan2;"WAV:DATA?"
820    ENTER Khan2 USING "#,2A";A$
830    FOR J=0 TO N-1
840       ENTER Khan2 USING "#,W";Value
850       DRAW J,Value
860    NEXT J
870    !
880    END
890    !
900    ! This portion of the program example listed below is used to set
910    ! up the display screen and is not part of the slaving process.
920    !
930    SUB   Graph(Xmin,Xmax,Xstep,Ymin,Ymax,Ystep,Title$,Xaxis$,Yaxis,Plotdev)
940       GINIT
950       IF Plotdev<>3 THEN
960          PLOTTER IS Plotdev,"HPGL"
970       END IF
980       PEN 4
990       GRAPHICS ON
1000      Dx=Xmax-Xmin
1010      Dy=Ymax-Ymin
1020      WINDOW Xmin-.1*Dx,Xmax+.1*Dx,Ymin-.15*Dy,Ymax+.1*Dy
1030      LINE TYPE 1
```

```
1040          CSIZE 3
1050          LORG 5
1060          LDIR -PI/2
1070          MOVE Xmax+.025*Dx,Ymin+Dy/2
1080          LABEL Yaxis$
1090          LDIR 0
1100          MOVE Xmin+Dx/2,Ymin-.075*Dy
1110          LABEL Xaxis$
1120          MOVE Xmin+Dx/2,Ymax+.05*Dy
1130          LABEL Title$
1140          X=Xmin
1150          WHILE X<Xmax
1160            MOVE X,Ymin-.025*Dy
1170            LABEL X
1180            X=DROUND(X+Xstep,8)
1190          END WHILE
1200          Y=Ymin
1210          WHILE Y<Ymax
1220            MOVE Xmin-.025*Dx,Y
1230            LABEL Y
1240            Y=DROUND(Y+Ystep,8)
1250          END WHILE
1260          CLIP Xmin,Xmax,Ymin,Ymax
1270          IF Plotdev=3 THEN LINE TYPE 4
1280          GRID Xstep,Ystep,Xmin,Ymin
1290          LINE TYPE 1
1300          PEN 1
1310      SUBEND
```

REMOTE SLAVING PROCEDURE FOR RANDOM EVENT CAPTURE

This section describes how the Random Event Capture (REC) mode can be used in the Multiple Digitizer Remote Slaving Procedure. The procedure shown below parallels the procedure in the previous section, but this procedure must be used when using REC. The two major differences in the procedures are that I) an additional command, MCREC, has been added; and 2) steps 6 and 7 of the previous procedure are reversed. A Remote Slaving Diagram using REC is provided, as well as another HP BASIC example program.

1. Configure the address switches of the modules and install the cabling properly.

2. Use the following command sequence:

```
REFERENCE CHANNEL:  *RST;STOP;TIM:MODE ASIN|SING;
   OTHER CHANNELS:  *RST;STOP;TIM:MODE SING;
```

3. Use the following command sequence:

```
OTHER CHANNELS: TRIG:SOUR SLAV;
```

4. Use the following command sequence:

```
REFERENCE CHANNEL/CHANNELS:  set up measurement parameters.
```

5. Use this command sequence only if aborting a trace that is waiting for a trigger.

REFERENCE CHANNEL/CHANNELS: *CLS;MCABT;

NOTE

The order of steps 6 and 7 is reversed in this procedure.

6. All channels set their hardware to take a measurement and are ready to be synchronized with the reference channel:

OTHER CHANNELS: *CLS;MCSET;

7. The reference channel sets its hardware to take a measurement:

REFERENCE CHANNEL: *CLS;MCSET;

8. This command sequence is not required when using REC, but may be used if desired.

REFERENCE CHANNEL: *CLS;MCSNC;

9. Use the following command sequence:

OTHER CHANNELS: *CLS;MCARM;

10. Use the following command sequence:

REFERENCE CHANNEL: *CLS;MCARM;

11. In this step, the trigger event is selected and the REC algorithm is applied to the reference channel.

```
REFERENCE CHANNEL: TIM:EVEN Event
                   MCREC;
```

The *OPC or ***WAI** commands may be used at this time.

Program line examples:

```
OUTPUT @Ref;"TIM:EVEN";Event
OUTPUT @Ref;"MCREC;"
```

12. Use the following command sequence:

REFERENCE CHANNEL: WAV:DATA?

13. Use the following command sequence:

```
REFERENCE CHANNEL: MCINF?
                   ENTER @Ref;Datl,Dat2,Dat3,Dat4

   OTHER CHANNELS: *CLS;MCINF Datl,Dat2,Dat3,Dat4
```

NOTE

Interpolation is not applicable for **REC** mode at this time; therefore, the MCTRG command should not be used.

14. Use the following command sequence:

OTHER CHANNELS: WAV:DATA?

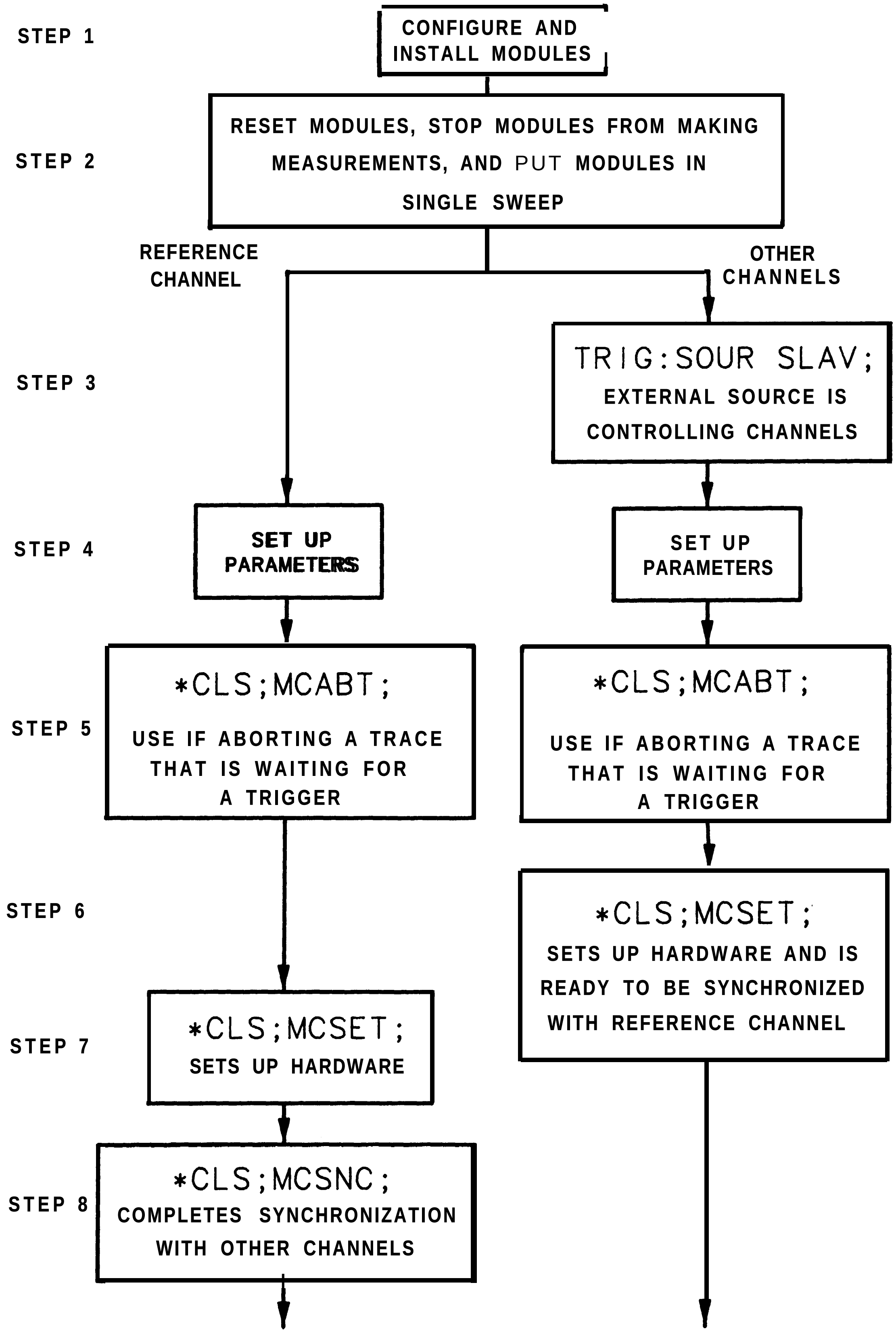


Figure 2-1 1. Remote Slaving Procedure Diagram for Random Event Capture (1 of 2)

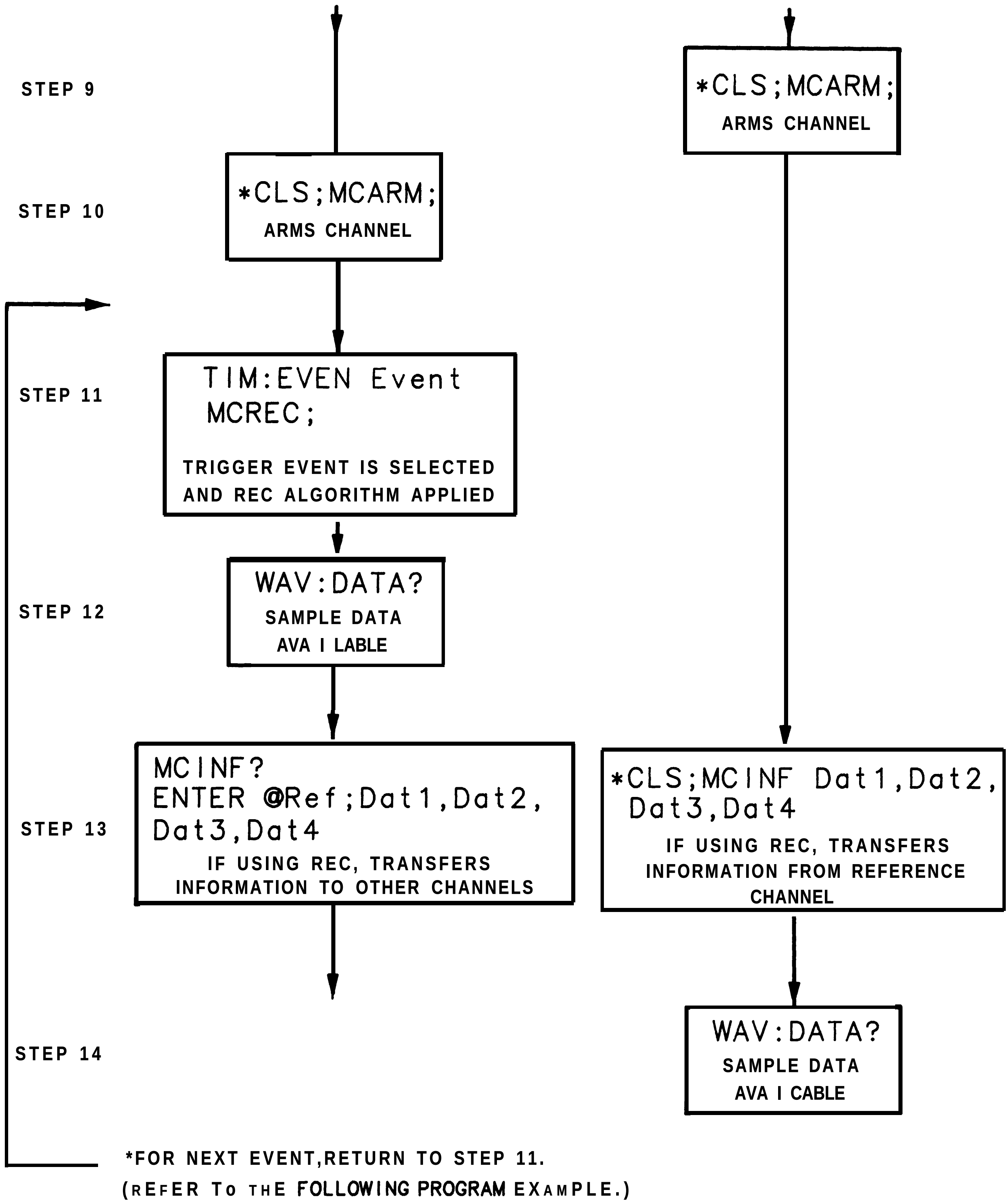


Figure 2-11. Remote Slaving Procedure Diagram for Random Event Capture (2 of 2)

. EXAMPLE PROGRAM FOR RANDOM EVENT CAPTURE

```
10   ASSIGN @Ref TO 715
20   ASSIGN @Chans TO 708
30   ASSIGN @Chanl TO 708
40   !
50   ! STEP 2
60   !
70   OUTPUT @Ref;"*RST;STOP;TIM:MODE ASIN;SAMP REC;*WAI;*ESE 255;*SRE 255"
80   OUTPUT @Chans;"*RST;STOP;TIM:MODE SING;SAMP REC;*WAI;*ESE 255;*SRE 255"
90   !
100  ! STEP 3
110  !
120  OUTPUT @Chans;"TRIG:SOUR SLAV;"
130  !
140  ! STEP 4
150  !
160  N=200
170  OUTPUT @Ref;"CHAN1:RANG 2.0V;TIM:RANG 100US;ACQ:POINT ";N
180  OUTPUT @Chanl;"CHAN1:RAN 2.0V;TIM:RANG 100US;ACQ:POIN ";N
190  !
200  ! STEP 5
210  !
220  !  * use only for aborting a trace waiting for a trigger *
230  !
240  ! STEP 6
250  !
260  OUTPUT @Chans;"*CLS;MCSET;"
270  !
280  ! STEP 7
290  !
300  OUTPUT @Ref;"*CLS;MCSET;"
310  !
320  ! STEP 8
330  !
340  !  * not required when using REC *
350  !
360  ! STEP 9
370  !
380  OUTPUT @Chans;"*CLS;MCARM;"
380  !
390  ! STEP 10
400  !
410  OUTPUT @Ref;"*CLS;MCARM;"
420  !
430  ! STEP 11
440  FOR Event= 1 TO 5
450     OUTPUT @Ref;"TIM:EVEN ";Event
460     OUTPUT @Ref;"MCREC;"
470  !
480  ! STEP 12
490  !
500  Graph(0,N,N/10,0,4096,512,"REMOTE SLAVED DIGs","time","ampl",3)
```

```
510   OUTPUT @Ref;"*WAI;"
520   OUTPUT @Ref;"WAV:DATA?"
530   ENTER @Ref USING "#,2A";A$
540   MOVE -100,0
550   FOR J=0 TO N-1
560     ENTER @Ref USING "#,W";Value
570     DRAW J,Value
580   NEXT J
590   !
600   ! STEP 13
610   !
620   ! * only used for REC *
630   OUTPUT @Ref;"MCINF?"
640   ENTER @Ref;Dat1,Dat2,Dat3,Dat4
650   OUTPUT @Chans;"MCINF ";Dat1,Dat2,Dat3,Dat4
660   OUTPUT @Chans;"MCREC;"
670   !
680   ! * Interpolation is not applicable for REC mode at this time. *
690   !
700   !  STEP 14 (repeat for each channel)
710   !
720   OUTPUT @Chan1;"*WAI;"
730   MOVE -100,0
740   OUTPUT @Chan1;"WAV:DATA?"
750   ENTER @Chan1 USING "#,2A";A$
760   FOR J=0 TO N-1
770     ENTER @Chan1 USING "#,W";Value
780     DRAW J,Value
790   NEXT J
800   NEXT Event
810   !
820   END
830   !
840   ! This portion of the program example listed below is used to set up
850   ! the display screen and is not part of the slaving process.
860   !
870   SUB Graph(Xmin,Xmax,Xstep,Ymin,Ymax,Ystep,Title$,Xaxis$,Yaxis$,Plotdev)
880     GINIT
890     IF Plotdev<>3 THEN
900       PLOTTER IS Plotdev,"HPGL"
910     END IF
920     PEN 4
930     GRAPHICS ON
940     Dx=Xmax-Xmin
950     Dy=Ymax-Ymin
960     WINDOW Xmin-.1*Dx,Xmax+.1*Dx,Ymin-.15*Dy,Ymax+.1*Dy
970     LINE TYPE 1
980     CSIZE 3
990     LORG 5
1000    LDIR -PI/2
1010    MOVE Xmax+.025*Dx,Ymin+Dy/2
1020    LABEL Yaxis$
1030    LDIR 0
```

```
1040          MOVE Xmin+Dx/2,Ymin-.075*Dy
1050          LABEL Xaxis$
1060          MOVE Xmin+Dx/2,Ymax+.05*Dy
1070          LABEL Title$
1080          X=Xmin
1090          WHILE X<Xmax
1100             MOVE X,Ymin-.025*Dy
1110             LABEL X
1120             X=DROUND(X+Xstep,8)
1130          END WHILE
1140          Y=Ymin
1150          WHILE Y<Ymax
1160             MOVE Xmin-.025*Dx,Y
1170             LABEL Y
1180             Y=DROUND(Y+Ystep,8)
1190          END WHILE
1200          CLIP Xmin,Xmax,Ymin,Ymax
1210          IF Plotdev=3 THEN LINE TYPE 4
1220          GRID Xstep,Ystep,Xmin,Ymin
1230          LINE TYPE 1
1240          PEN 1
1250      SUBEND
```

Chapter 3
LANGUAGE REFERENCE

This chapter contains complete information for the commands available to operate a digitizer system by remote computer control. The commands are divided into the functional subsystems listed below, and are listed in alphabetical order within each subsystem.

Common Command Set
Digitizer Top-Level Command Set
Acquire Subsystem
Calibration Subsystem
Channel Subsystem
Display Subsystem
Domain Subsystem
Function Subsystem
Measure Subsystem
Timebase Subsystem
Trigger Subsystem
Waveform Subsystem
Window Subsystem

A syntax diagram. for each subsystem is shown at the beginning of each subsystem.

The commands and/or queries of each subsystem are listed in alphabetical order according to their mnemonics. A functional description is provided for each; however, if more detailed information regarding a command function is necessary, refer to the manual operation softkey descriptions in Chapter 2 of the HP 70700A Digitizer Operation Manual, Digitizer Functions.

NOTE

Only the three- or four-character mnemonic may be used in programming.

For your convenience, four cross-reference listings for the digitizer remote commands are supplied in the Appendices.

Appendix A, Command Listing By Subsystem
Appendix B, Alphabetical Mnemonic Listing
Appendix C, Alphabetical Command Description Listing
Appendix D, Alphabetical Command Summary

NOTATION CONVENTIONS

Pictorial

The following general guidelines refer to the pictorial syntax diagrams for each functional subsystem in this manual.

- All items enclosed by a rounded envelope must be entered exactly as shown.
- Items enclosed by a rectangular box indicate parameters used in the command sequence. A description of each parameter is given in the respective command descriptions.
- Command sequence items are connected by lines. Each line can be followed in only one direction, as indicated by an arrow at the end of each line.
- Any combination of command sequence items that can be generated by following the lines in the proper direction is syntactically correct.
- A command sequence item is optional if there is a valid path around it.

Textual Notation

CAPITAL LETTERS	Capital letters are used to indicate program mnemonics of a command program header.
< >	Angle brackets are used to enclose elements of the language being defined. Refer to Functional Syntax Conventions, below, for an explanation of information contained within these brackets.
[ ]	Square brackets are used to enclose optional information not required for execution of the command sequence.
{ }	Braces are used to enclose a descriptive comment referring to the preceding item in the command sequence.
|	The vertical line indicates a choice of exactly one element from a list.

FUNCTIONAL SYNTAX CONVENTIONS

Functional syntax is required to create program messages that are transmitted to a device. Program messages are composed of sequences of program message units, each unit representing a program command or query. Each program command or query is composed of a sequence of functional syntactic elements. Legal program commands and queries are created from functional element sequences generated by using the functional syntax diagrams.

Terminated program messages are complete “controller-to-device” messages. They are sequences of zero or more <PROGRAM MESSAGE UNIT> elements. The <PROGRAM MESSAGE UNIT> element represents a programming command or data sent to the device from the controller. See Figure 3-1.

A <PROGRAM MESSAGE UNIT> element is defined in more detail as either a <COMMAND MESSAGE UNIT> or <QUERY MESSAGE UNIT>. See Figure 3-2.

Each <COMMAND/QUERY MESSAGE UNIT> contains a <PROGRAM HEADER SEPARATOR> and may be optionally followed by a <PROGRAM DATA> clement and a <PROGRAM DATA SEPARATOR> See Figure 3-3.

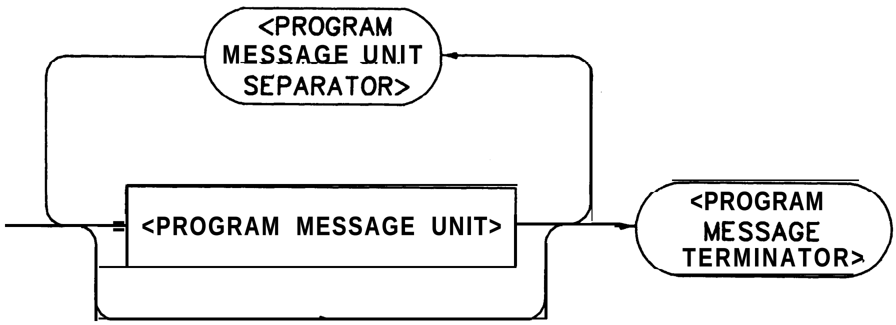


Figure 3-1. Terminated Program Message Functional Element Syntax

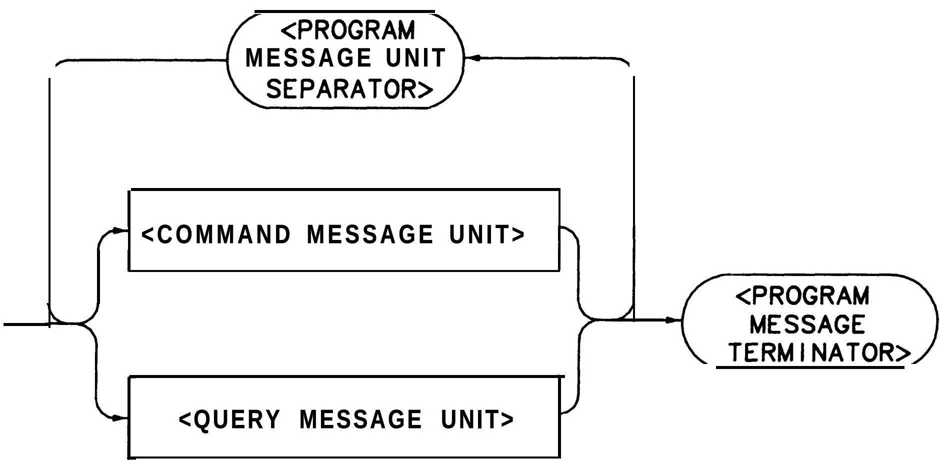


Figure 3-2. <PROGRAM MESSAGE UNIT> Functional Element Syntax

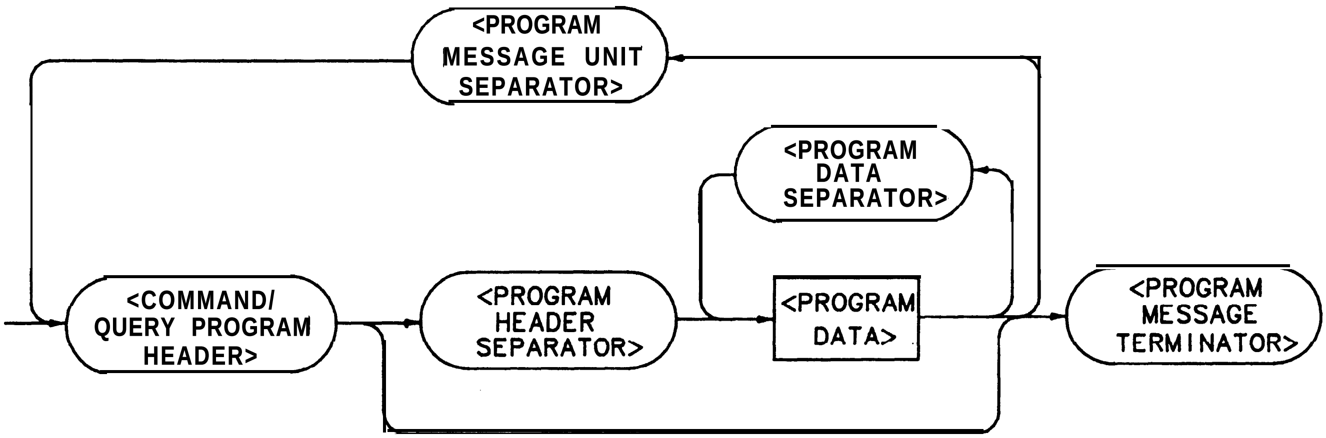
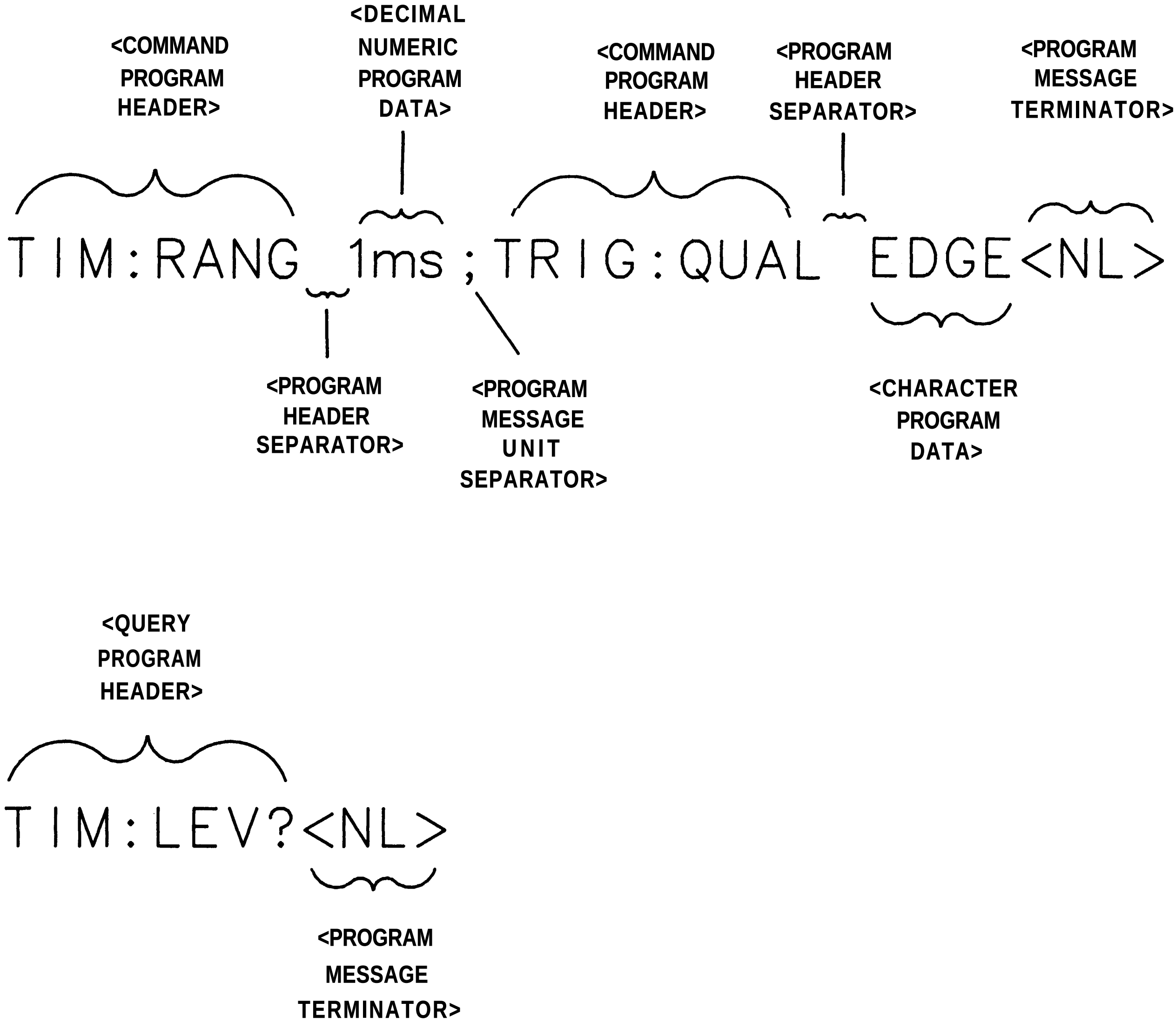


Figure 3-3. <COMMAND/QUERY MESSAGE UNIT> Functional Element Syntax

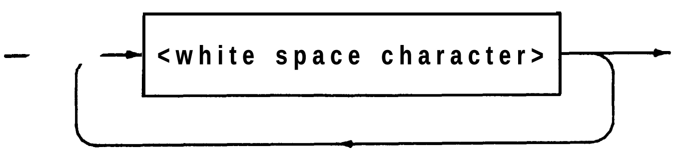
The syntax for typical command and query sequences is shown below:



SEPARATOR FUNCTIONAL ELEMENTS

The three separator functional elements that are used in the functional **syntax** of the digitizer system are defined in more detail below.

Since **<white** space> is an integral part of each separator functional element, it is defined as follows:



where <white-space character> is defined as a single ASCII-encoded byte in the range 00-09, OB-20 hexadecimal (O-9, 1 l-32 decimal). This range includes the ASCII-control characters and the space, but excludes the new line (NL).

<PROGRAM MESSAGE UNIT SEPARATOR> separates sequential <PROGRAM MESSAGE UNIT> elements from one another within a program message. See Figure 3-4.

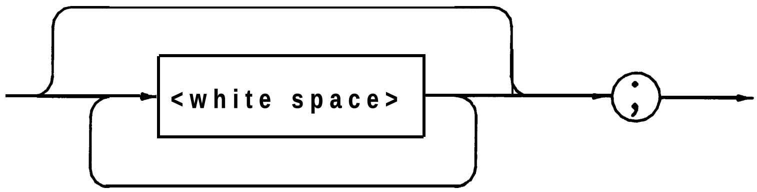


Figure 3-4. <PROGRAM MESSAGE UNIT SEPARATOR> Syntax

<PROGRAM HEADER SEPARATOR> separates the <COMMAND PROGRAM HEADER> or <QUERY PROGRAM HEADER> from the <PROGRAM DATA> elements. See Figure 3-5.

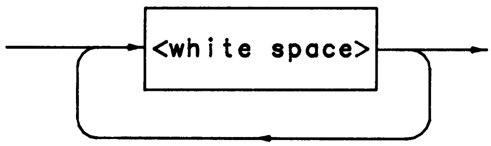


Figure 3-5.<PROGRAM HEADER SEPARATOR> Syntax

<PROGRAM DATA SEPARATOR> separates sequential <PROGRAM DATA> elements from one another after a <COMMAND PROGRAM HEADER> or <QUERY PROGRAM HEADER>. It is used when a <COMMAND PROGRAM HEADER> or <QUERY PROGRAM HEADER> has multiple parameters. See Figure 3-6.

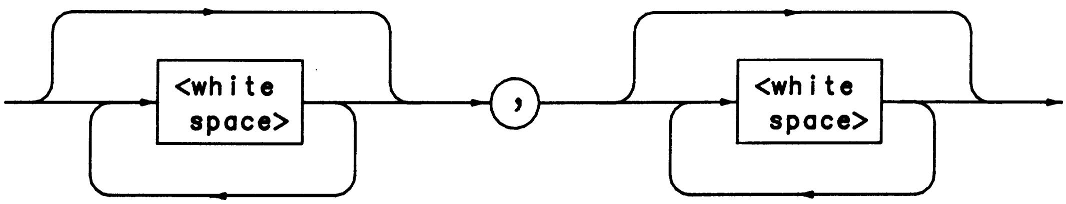
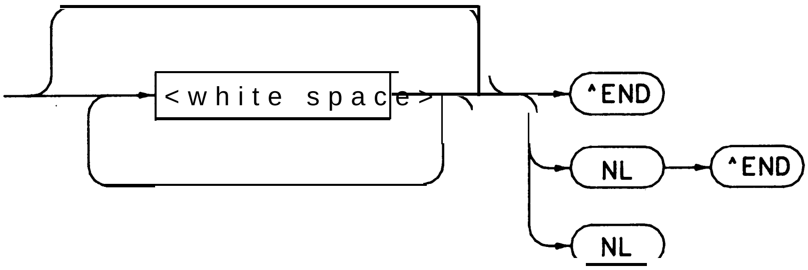


Figure 3-6. <PROGRAM DATA SEPARATOR, Syntax

PROGRAM MESSAGE TERMINATOR

A **<PROGRAM** MESSAGE TERMINATOR> terminates a program message which is a sequence of one or more definite-length **<PROGRAM** MESSAGE UNIT> elements. The program message terminator syntax used in the functional syntax of the digitizer system is defined as follows:



where ^END is used to indicate concurrent transmission of the END message with the last preceding data byte;

where NL (new line) is defined as a single ASCII-encoded byte OA (IO decimal).

FUNCTIONAL ELEMENT SUMMARY

<PROGRAM MESSAGE UNIT>	Represents a single command, programming data, or query received by the device.
<COMMAND MESSAGE UNIT>	Represents a single command or programming data received by the device.
<COMMAND PROGRAM HEADER>	Specifies the function or operation to be performed in the device and may be optionally followed by associated parameters encoded as <PROGRAM DATA> elements. A <COMMAND PROGRAM HEADER> is further defined as either a <simple command program header>, a <compound command program header>, or a <common command program header>.
<QUERY MESSAGE UNIT>	Represents a single query sent from the controller to the device.
<QUERY PROGRAM HEADER>	Similar to <COMMAND PROGRAM HEADER>, except a query indicator (i.e., ?) shows that a response is expected from the device. A <QUERY PROGRAM HEADER> is further defined as either a <simple query program header>, a <compound query program header>, or a <common query program header>.
<PROGRAM DATA>	Represents any of the four different program data types: <CHARACTER PROGRAM DATA> is a data type suitable for sending short mnemonic data, generally where a numeric data type is not suitable. <DECIMAL NUMERIC PROGRAM DATA> is a data type suitable for sending decimal integers or decimal fractions with or without exponents. <SUFFIX PROGRAM DATA> is an optional field that may follow and is used to indicate associated multipliers and units. <STRING PROGRAM DATA> is a data type suitable for sending seven-bit ASCII character strings where the content needs to be “hidden” by delimiters. <ARBITRARY BLOCK PROGRAM DATA> is a data type suitable for sending blocks of arbitrary eight-bit information.

<PROGRAM MESSAGE UNIT SEPARATOR>	Separates sequential <PROGRAM MESSAGE UNIT> elements from one another within a program message.
<PROGRAM HEADER SEPARATOR	Separates the command/query header from any associated <PROGRAM DATA> elements.
<PROGRAM DATA SEPARATOR>	Separates sequential <PROGRAM DATA> elements, that are related to the same header, from one another.
<PROGRAM MESSAGE TERMINATOR>	Terminates a program message.
<nrf>	Represents flexible numeric representation in the syntax diagram and is defined in each appropriate mnemonic description.
<NR1>	Numeric response data consisting of a set of implicit point representations of numeric values.
<NR3>	Numeric response data that are representations of scaled explicit radix point numeric values, together with an exponent notation.

COMMON COMMAND SET

The commands of the Common Command Set are available at any time during remote control of a digitizer system. Refer to Figure 3-7 for a syntax diagram of the Common Command Set commands.

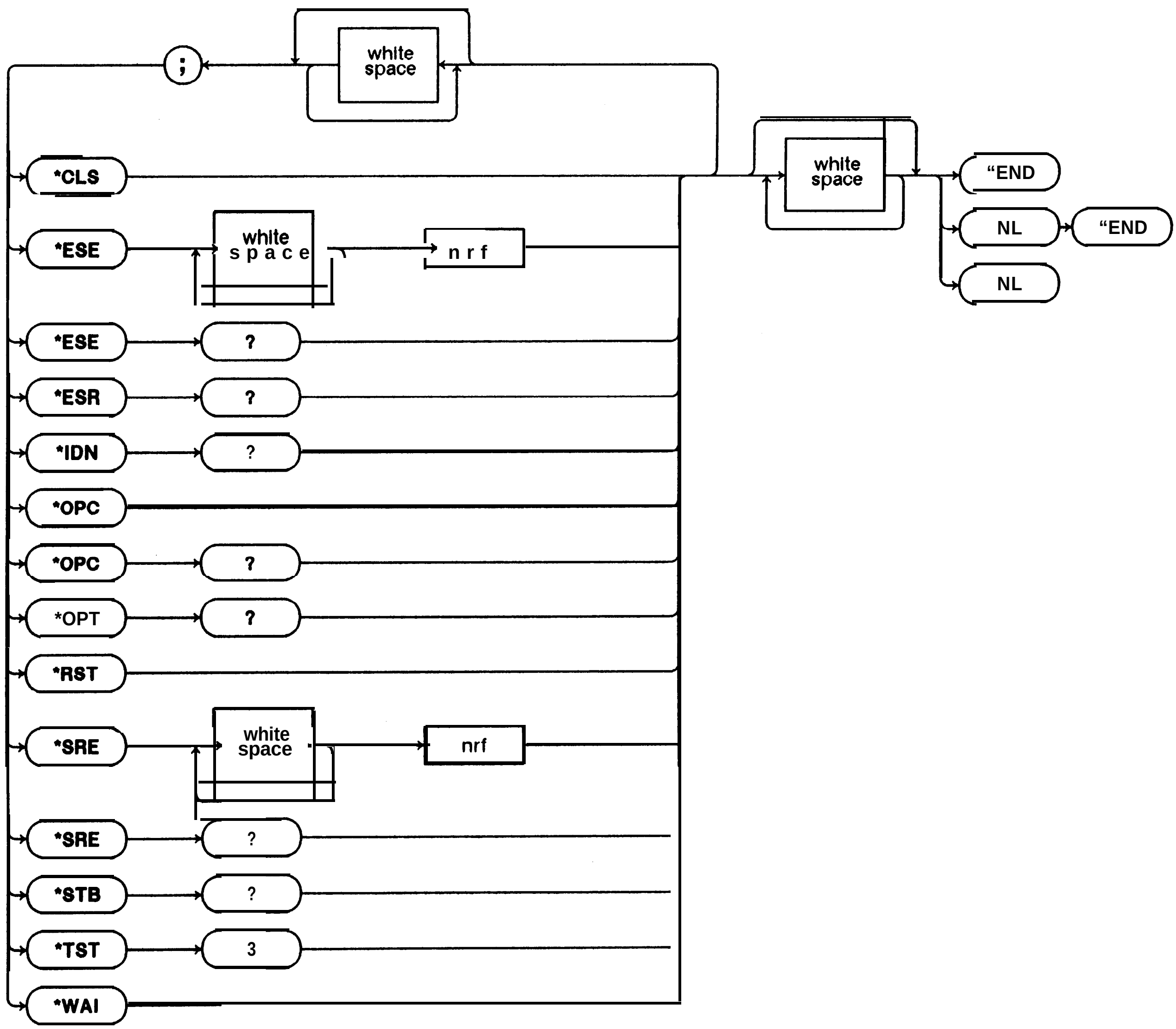


Figure 3-7. Common Command Set Commands

*CLS

CLEAR STATUS command

The CLEAR STATUS command clears status data structures and the Request-for-OK (Operation Complete) flag. (Refer to the OPERATION COMPLETE command in this section.)

If the CLEAR STATUS command immediately follows a <PROGRAM MESSAGE TERMINATOR>, the Output Queue and the MAV (Message Available) bit will be cleared. Any new <PROGRAM MESSAGE> after a <PROGRAM MESSAGE TERMINATOR> clears the Output Queue.

Command Syntax: *CLS <terminator>

Example: OUTPUT 707;"*CLS"

*ESE

STANDARD EVENT STATUS ENABLE command/query

The STANDARD EVENT STATUS ENABLE command sets the Standard Event Status Enable Register bits. The query allows the programmer to determine the current contents of the Standard Event Status Enable Register. Refer to the Standard Status Data Structure section in Chapter 2.

Command Syntax: *ESE <mask> <terminator>

Example: OUTPUT 707;"*ESE 32"

Preset State: 0

Parameter Range: 0 through 255

Query Syntax *ESE? <terminator>

Query Response: <NR1> <NL>

*ESR

STANDARD EVENT STATUS REGISTER query

The STANDARD EVENT STATUS REGISTER query allows the programmer to determine the current contents of the Standard Event Status Register. Reading the Standard Event Status Register will clear it.

Query Syntax *ESR? <terminator>

Query Response: <NR1> <NL>

*IDN

IDENTIFICATION

query

The IDENTIFICATION query is for identifying devices over the system interface. The response is organized into four fields separated by commas. The field definitions are as follows:

Field 1	Manufacturer	required
Field 2	Model	required
Field 3	Serial Number	ASCII character 0 if not available
Field 4	Firmware Datecode	ASCII character 0 if not available

An example of a query response is: . HEWLETT PACKARD,70700A,24 19A00256,8 703 17

Query Syntax *IDN? <terminator>

Query Response: HEWLETT PACKARD,70700A,<serial number>,
<firmware datecode (yymmdd)> <NL>

*OPC

OPERATION COMPLETE

command/query

The OPERATION COMPLETE command sets the request for the Operation Complete flag. When all pending device operations have been finished, the OPC bit in the Standard Event Status Register is set. The query places an ASCII character "1" in the Output Queue when all pending device operations have been finished.

NOTE

Because reading trace data is an operation, the synchronization commands should not be used after the request of trace data (e.g., WAV:DATA?; or DIG **CHAN1;)** until all data has been read. If the synchronization commands are used, the system may be placed in a state where it is "waiting" indefinitely. In this state, the digitizer system waits for the data to be read and the computer waits for all operations to be completed.

Command Syntax: *OPC <terminator>

Example: OUTPUT 707;"*OPC"

Query Syntax: *OPC? <terminator>

Query Response: <NR1> <NL>

*OPT

OPTION IDENTIFICATION **query**

The OPTION IDENTIFICATION query is for identifying reportable device options over the system interface. The response is organized into five fields separated by commas. The field definitions are as follows:

Field 1	Memory Size
Field 2	ID of RAM program 0
Field 3	ID of RAM program 1
Field 4	ID of RAM program 2
	ID of RAM program 3

For example: **65536,86** 12 **16,0,0,0** will be returned as a five-field option.

Query Syntax: *OPT? <terminator>

Query Response: <options (five fields)> <NL>

*RST

RESET command

The RESET command performs a device reset and the following actions:

- sets device-dependent functions to a known state
- aborts pending operations
- clears the Output Queue
- clears the Request-for-Operation Complete flag

The RESET command will not:

- affect the hardware interface
- modify the Standard Status Register Enable setting
- modify the Standard Event Status Enable setting
- modify the power-on-clear flag
- modify calibration data.

Command Syntax: *RST <terminator>

Example: OUTPUT 707;"*RST"

*SRE

SERVICE REQUEST ENABLE

command/query

The SERVICE REQUEST ENABLE command sets the Service Request Enable Register bits. The query returns the current contents of the Service Request Enable Register. Refer to the Standard Status Data Structure section in Chapter 2.

Command Syntax: *SRE <mask> <terminator>

Example: OUTPUT 707;"*SRE 150"

Preset State: 0

Parameter Range: 0 through 255

Query Syntax: *SRE? <terminator>

Query Response: <NR1> <NL>

*STB

READ STATUS BYTE

query

The READ STATUS BYTE query allows the programmer to read the Status Byte Register and the Master Summary Status bit. Refer to the Standard Status Data Structure section in Chapter 2.

Query Syntax: *STB? <terminator>

Query Response: <NR1> <NL>

*TST

SELF-TEST

query

The SELF-TEST query executes an internal self test and places the pass/fail code in the Output Queue. Pass is equal to 0; fail is less than or greater than 0.

Query Syntax: *TST? <terminator>

Query Response: <NR1> <NL>

*WAI

WAIT-TO-CONTINUE command

The WAIT-TO-CONTINUE command stops the device from executing any further commands or queries until the No-Operation-Pending flag, Power-On (PON), or Device Clear Active State Message (DCAS) is true.

NOTE

Because reading trace data is an operation, the synchronization commands should not be used after the request of trace data (e.g., WAV:DATA?; or DIG **CHAN1;)** until all data has been read. If the synchronization commands are used, the system may be placed in a state where it is "waiting" indefinitely. In this state, the digitizer system waits for the data to be read and the computer waits for all operations to be completed.

Command Syntax: *WAI <terminator>

Example: OUTPUT 707;"*WAI"

DIGITIZER TOP-LEVEL COMMAND SET

The commands of the Digitizer Top-Level Command Set are general digitizer commands which do not. reside in any particular subsystem and are available at any time. Refer to Figure 3-8 for a syntax diagram of the Digitizer Top-Level Command Set commands.

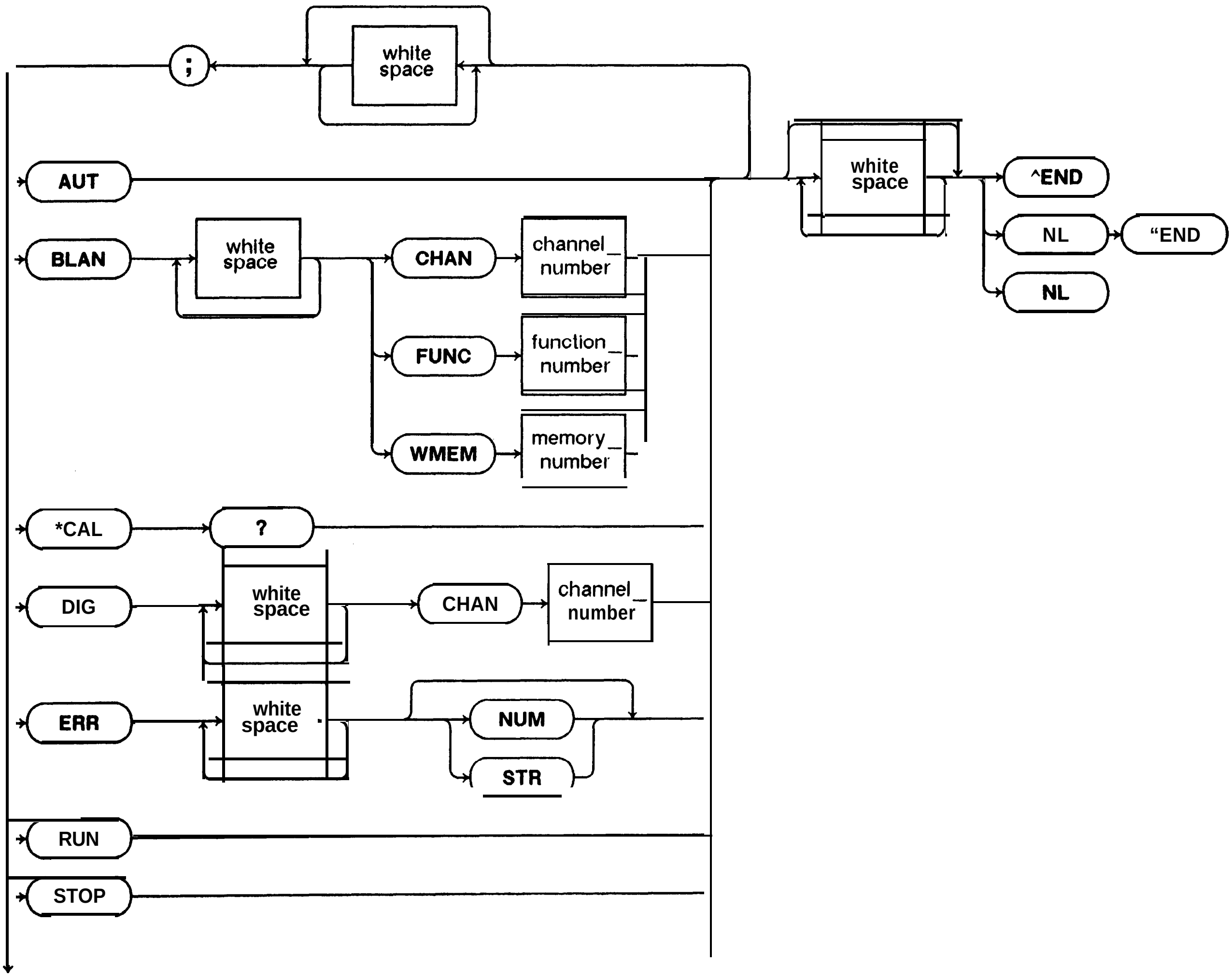


Figure 3-8. Digitizer 'Top-Level Command Set Commands (1 of 2)

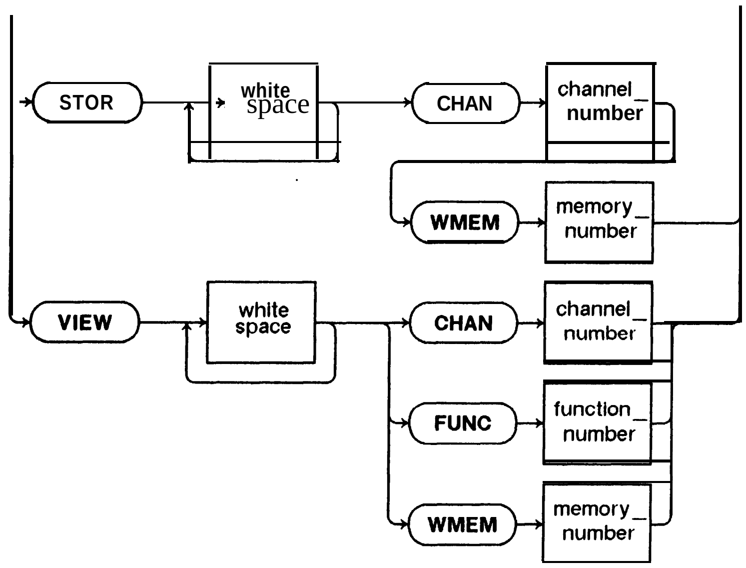


Figure 3-8. Digitizer Top-Level Command Set Commands (2 of 2)

AUT

AUTOSCALE

command

The AUTOSCALE command performs the autoscale function which automatically selects the vertical sensitivity, vertical offset, trigger level, and sweep speed for a display of the input signal. If the function fails, an error is declared.

Command Syntax: AUT <terminator>

Example: OUTPUT 707;"AUT"

BLAN

BLANK

command

The BLANK command causes the instrument to turn off (i.e., to stop displaying on the display screen) the specified channel, function, or waveform memory.

Command Syntax: BLAN {CHANX|FUNCX|WMEMX} <terminator>

Example: OUTPUT 707;"BLAN WMEM2"

*CAL

CALIBRATION **query**

The CALIBRATION query executes an internal self-calibration routine and generates a response in the Output Queue that indicates whether or not the device completed the self-calibration routine without error. Pass is equal to 0; fail is less than or greater than 0.

Query Syntax: *CAL? <terminator>

Query Response: <NR1> <NL>

DIG

DIGITIZE command

The DIGITIZE command is used to request that waveform data from a specified channel be returned to the controller via HP-IB. Once the necessary data is received to satisfy the acquisition criteria, all other measurement acquisitions are stopped. If the display is on, the new data will also be displayed.

Command Syntax: DIG (CHANX) <terminator>

Example: OUTPUT 707;"DIG CHAN2"

ERR

ERROR **query**

The ERROR query returns the next error in the Error Queue. If there are no errors, 0 (no errors) is returned. If there has been an error, the instrument should respond with the first one. Subsequent responses to the ERROR query should continue with the error list until there is no error remaining. The optional parameters NUMBER and STRING indicate whether a numeric or text description of the error is to be returned.

Query Syntax: ERR? [NUM|STR] <terminator>

Query Response: if NUMBER: <NR1> <NL>
if STRING: <NR1> , <error text (message)> <NL>

RUN command

The RUN command directs the instrument to acquire data for the active waveform display. If the instrument is in SINGLE SWEEP, one trigger is enabled which in turn generates one measurement. If the instrument is in CONTINUOUS SWEEP, triggers are enabled repeatedly and the instrument displays the data it acquires continuously.

Command Syntax: RUN <terminator>

Example: OUTPUT 707;"RUN"

STOP
command

The STOP command directs the instrument to stop acquiring data for the active waveform display. Note that this does not affect the SINGLE SWEEP or CONTINUOUS SWEEP mode.

Command Syntax: STOP <terminator>

Example: OUTPUT 707;"STOP"

STOR
STORE
command

The STORE command directs the instrument to move the current waveform in a specified channel to the specified waveform memory. The channel is always the source, and the waveform memory is the destination.

Command Syntax: STOR {CHANX,WMEMX} <terminator>

Example: OUTPUT 707;"STOR CHAN1,WMEM3"

VIEW
command

The VIEW command causes the instrument to turn on (i.e., to start displaying on the display screen) the specified channel, function, or waveform memory.

Command Syntax: VIEW{CHANX|FUNCX|WMEMX} <terminator>

Example: OUTPUT 707;"VIEW FUNC1"

ACQUIRE SUBSYSTEM

The Acquire Subsystem commands are used to set up conditions when a system command is executed. This subsystem is used to select the number of trace averages, the number of points desired, trace length and timebase coupling, and the type of data. Refer to Figure 3-9 for a syntax diagram of the Acquire Subsystem commands.

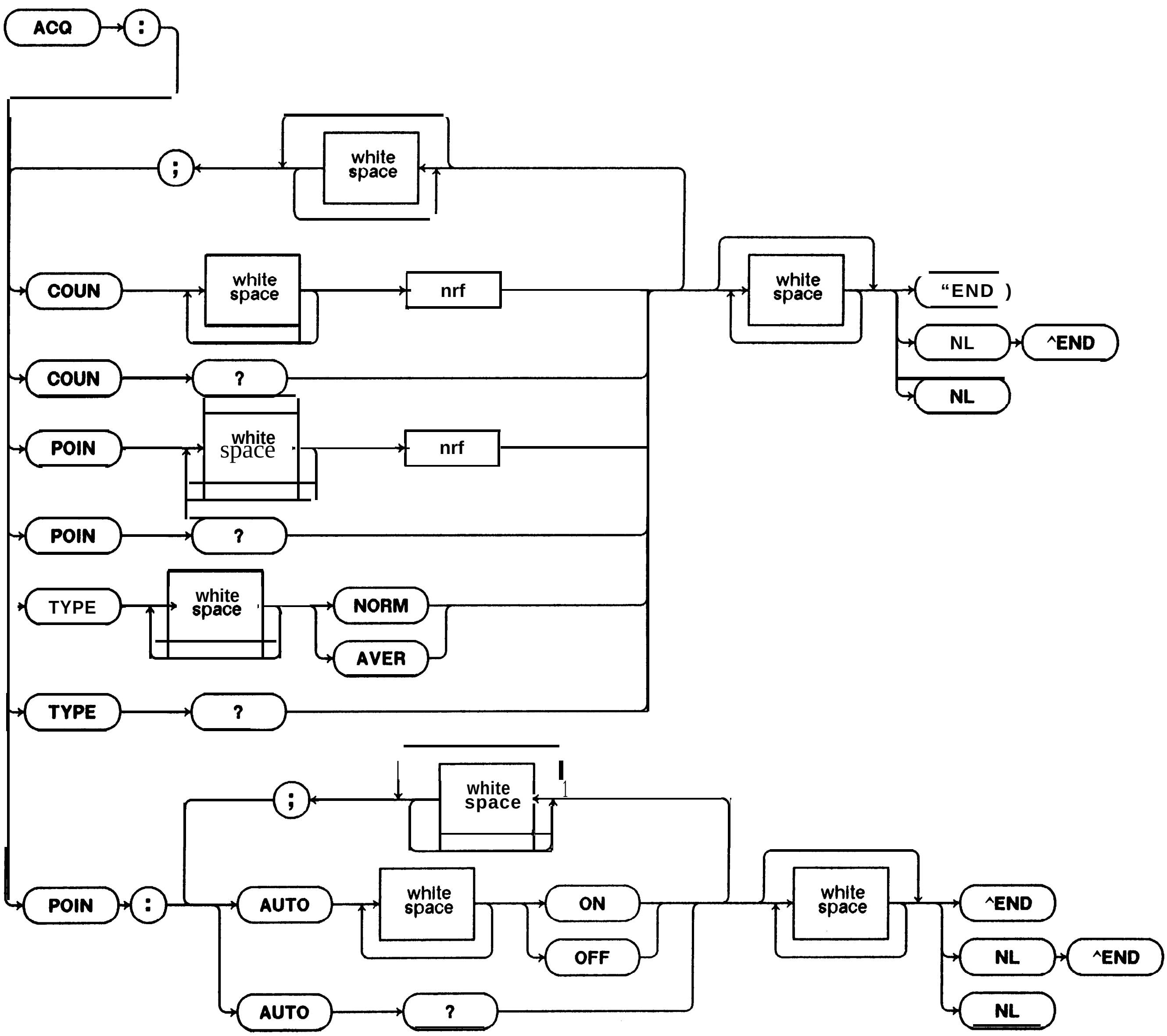


Figure 3-9. **Acquire Subsystem Commands**

COUN

COUNT

command/query

The COUNT command sets the number of averaged traces that may be selected. The traces averaged are always the most recent traces (or sweeps) taken. The query returns the current number of averaged traces.

Command Syntax: ACQ:COUN <number> <terminator>

Example: OUTPUT 707;"ACQ:COUN 55"

Preset State: 1

Parameter Range: 1 to 1024

Query Syntax: ACQ:COUN? <terminator>

Query Response: <NR1> <NL>

POIN

POINTS

command/query

The POINTS command sets the number of points in a trace. It determines the number of data points to be sampled and saved, which can be up to a maximum of the available memory. (The available memory is 256K words.) The query returns the current number of points in the trace.

NOTE

Because the trace length functions in conjunction with the SECONDS/DIVISION function, the minimum/maximum for a given time range will vary. If a value that conflicts with the current time range is requested (e.g., 10000 points with a time range of 1 **µs,** which would require a 10 **GHz** sampling rate), a range error is reported and the TIME RANGE will be adjusted. For example, **"*RST;ACQ:POIN** 10000;" causes a range error. However, the trace is still 10000 points but with a time range of 31.2 **µs.**

Command Syntax: ACQ:POIN <number of points> <terminator>

Example: OUTPUT 707;"ACQ:POIN 500"

Preset State: 200

Parameter Range: 20 to (256K words – 256 data points)
available memory: 256K words

Query Syntax: ACQ:POIN? <terminator>

Query Response: <NR1> <NL>

The AUTO ON command allows the digitizer module to choose the trace length that is most suitable. Refer to the AUTO command below for more information.

AUTO command/query

The AUTO command sets the status of the Timebase and Trace Length coupling. When ON is selected, the SECONDS/DIVISION function and the TRACE LENGTH function are directly coupled. When these two functions are coupled, the digitizer attempts to set the trace length points to 500 ≤ number of points ≤ 1000. If this cannot be achieved, then the digitizer sets the number of points between a minimum of 20 and a maximum of the available memory.

When OFF is selected, the SECONDS/DIVISION function can be changed without affecting the trace length, The query returns the current status of the Timebase and Trace Length coupling.

Command Syntax: ACQ:POIN:AUTO {ON|OFF} <terminator>

Example: OUTPUT 707;'ACQ:POIN:AUTO ON'

Preset State: ON

Query Syntax: ACQ:POIN:AUTO? <terminator>

Query Response: ON|OFF <NL>

TYPE command/query

The TYPE command allows selection of the type of acquisition that is to take place when a digitizer system command is executed.

When NORMAL is selected, evenly-spaced sequential data points in a trace are measured and returned.

When AVERAGE is selected, the number of averaged traces may be selected in which the most recent evenly-spaced sequential data points in a trace are averaged with the previous trace. The traces averaged are always the most recent traces (or sweeps) taken.

Command Syntax: ACQ:TYPE {NORMIAVER) <terminator>

Example: OUTPUT 707;'ACQ:TYPE AVER'

Preset State: NORMAL

Query Syntax: ACQ:TYPE? <terminator>

Query Response: NORM|AVER <NL>

CALIBRATION SUBSYSTEM

The Calibration Subsystem allows the user to invoke the internal calibration routine remotely, or to load previously-acquired calibration data. Refer to Figure 3-10 for a syntax diagram of the Calibration Subsystem commands.

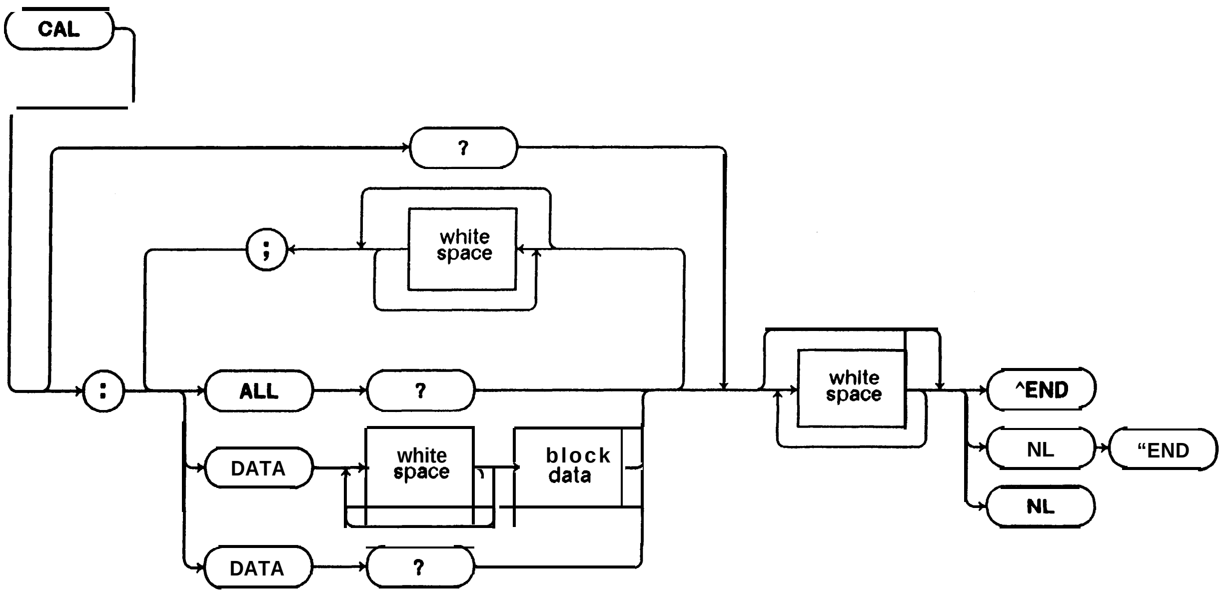


Figure 3-1 0. Calibration Subsystem Commands

ALL query

The ALL query invokes the internal calibration, and returns a pass or fail code.

Query Syntax: CAL:ALL? <terminator> or CAL? <terminator>

Query Response: 0|-1 <NL>
where 0 indicates "pass" and -1 indicates "fail"

DATA

command/query

The DATA command provides a means for the user to load calibration data. The query returns the calibration data of the module as a definite block (#3xxxdata). The data resides in the internal format. of the module.

Command Syntax: CAL:DATA <block data>

Example:

```
DIM A$[500]                      ! Get calibration data.
OUTPUT 707;"CAL:DATA?"           ! Request calibration data.
ENTER 707 USING "#,-K";A$ ! Enter data up to <END>.
      .                            (# ignores CRLF; K gets all
      .                            text up to <END>)
      .
      .

SIZE = VAL(A$[3,5])              ! Restore calibration cycle.
                                 ! (# suppresses CRLF; K sends
                                 ! text)
OUTPUT 707 USING "#,K";"CAL:DATA "&A$[1,SIZE+5]
```

Query Syntax: CAL:DATA?

Query Response:

```
#3xxx <binary byte>...<binary byte> <NL>
  /  \ /
binary 3 digits
```

CHANNEL SUBSYSTEM

The Channel Subsystem allows the user to control all vertical or Y-axis functions of the digitizer system. Refer to Figure 3-11 for a syntax diagram of the Channel Subsystem commands.

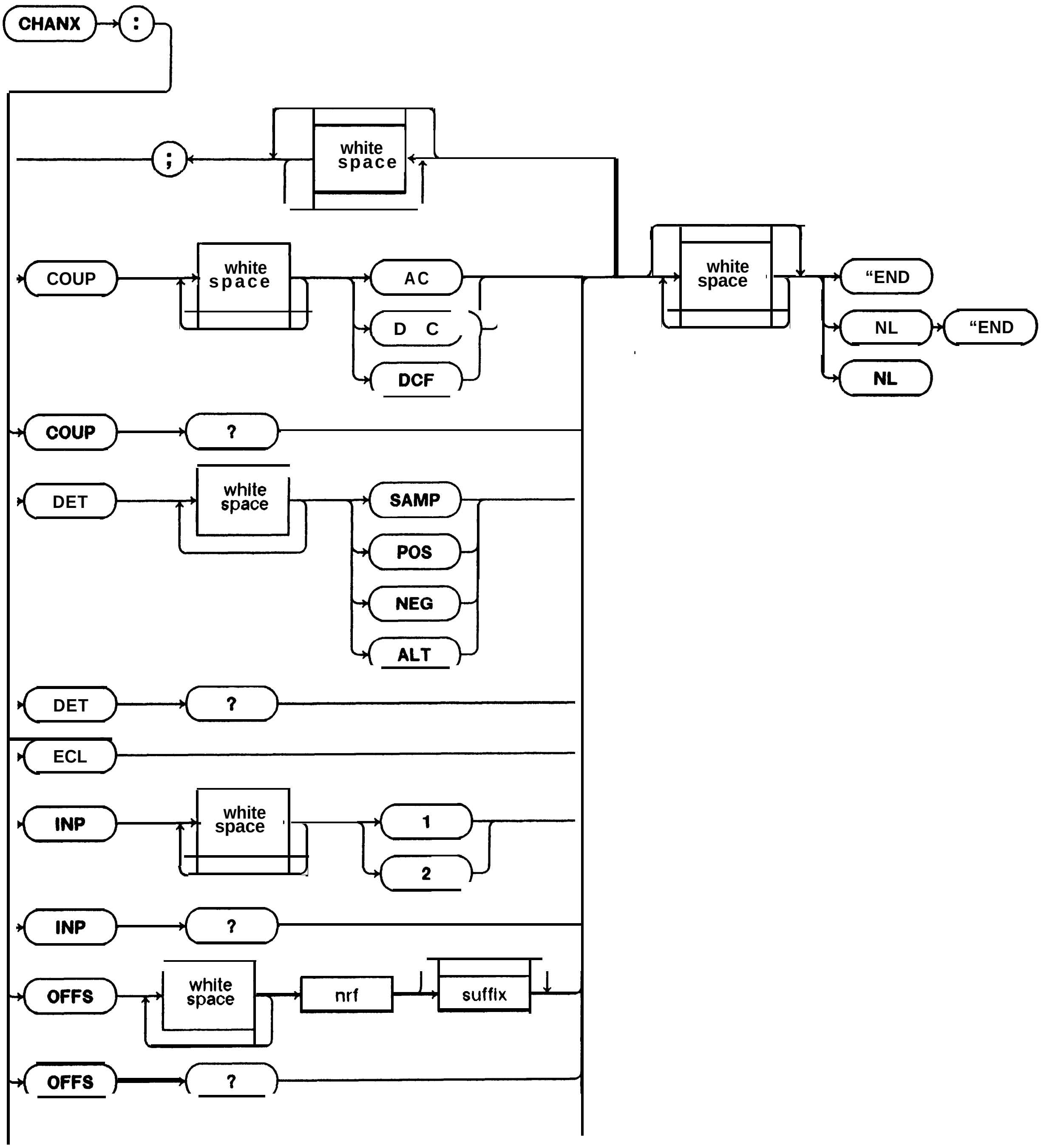


Figure 3-11. Channel Subsystem Commands (1 of 2)

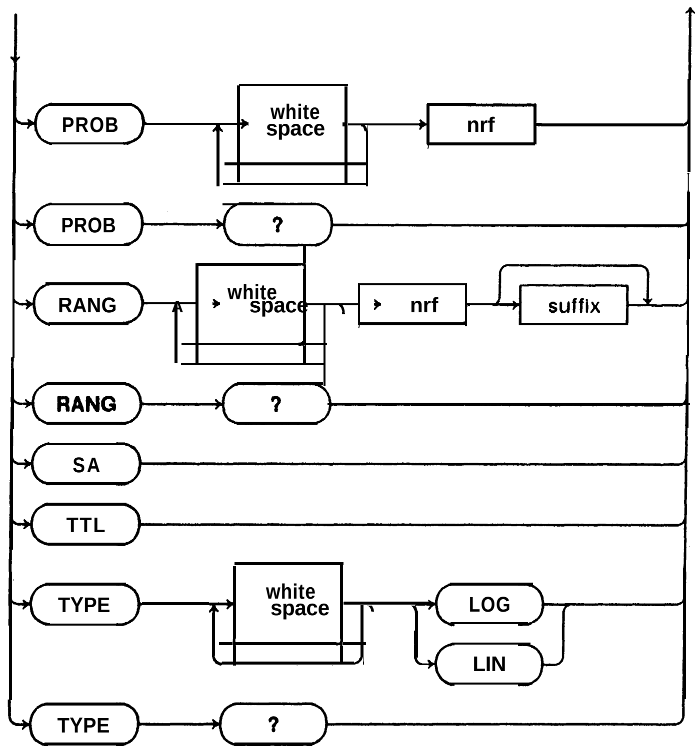


Figure 3-11. Channel Subsystem Commands (2 of 2)

COUP

COUPLING

command/query

The COUPLING command sets the signal coupling to the digitizer for the indicated channel (CHANX) to be ac-coupled into 1 megohm (AC), dc-coupled into 1 megohm (DC), or dc-coupled into 50 ohms (DCF). The query returns the current signal coupling.

Command Syntax: CHANX:COUP {AC|DC|DCF} <terminator>

Example: OUTPUT 707; "CHAN1:COUP AC"

Preset State: DC

Query Syntax: CHANX:COUP? <terminator>

Query Response: AC|DC|DCF <NL>

DET

DETECTOR

command/query

The DETECTOR command sets the detector mode used by the indicated channel (CHANX). The four detector modes used for sampling a waveform are: SAMPLE, POSITIVE PEAK, NEGATIVE PEAK, and ALTERNATE PEAK. The query returns the current detector mode.

The detector modes help display the frequency and number of occurrences of a waveform when using slower sweep rates. To determine the sample interval when using any of the detector modes, the waveform is divided into regularly-spaced intervals based on the TRACE LENGTH. When the timebase has been set up so that more than one 20 MHz conversion (digitized value) occurs in a sample interval, the detector mode may be selected to specify which conversion is stored to memory.

SAMPLE retains the last conversion in each sample interval.

POSITIVE PEAK retains only the highest value of data in each sample interval.

NEGATIVE PEAK retains only the lowest value of data in each sample interval.

ALTERNATE PEAK alternately retains the highest and lowest values of data within the sample interval. The highest value consists of the highest value since the last POSITIVE PEAK was kept; likewise, the lowest value consists of the lowest value since the last NEGATIVE PEAK was kept.

The sample interval for the ALTERNATE PEAK mode is twice as large as the POSITIVE and NEGATIVE PEAK sample interval. The sample rate is the same though, since each interval overlaps the preceding interval by half. This is to ensure that no information is overlooked as the POSITIVE and NEGATIVE PEAK values are retained alternately.

Command Syntax: CHANX:DET {SAMP|POS|NEG|ALT} <terminator>

Example: OUTPUT 707;"CHAN1:DET NEG"

Preset State: SAMPLE

Query Syntax: CHANX:DET? <terminator>

Query Response: SAMP|POS|NEG|ALT <NL>

ECL

EMITTER-COUPLED LOGIC

command

The ECL command sets the voltage range, offset, and trigger level for the indicated channel (CHANX) to values appropriate for examining ECL signals. The voltage range is 1.6V, the offset is -l.OV, and the trigger level is -1.OV.

Command Syntax: CHANX:ECL <terminator>

Example: OUTPUT 707;"CHAN1:ECL"

Preset State: Inactive

INP

INPUT

command/query

The INPUT command sets the input connection for the indicated channel (CHANX) to be either INPUT 1 or INPUT 2. The query returns the current input.

Command Syntax: CHANX:INP {1|2} <terminator>

Example: OUTPUT 707;"CHAN1:INP 2"

Preset State: INPUT 1

Query Syntax: CHANX:INP? <terminator>

Query Response: 1|2 <NL>

OFFS

OFFSET

command/query

The OFFSET command sets the level (amplitude reference) of the display midscreen in volts for the indicated channel (CHANX). The input range for a signal will be: ± IOV × PROBE. The query returns the current offset.

Command Syntax: CHANX:OFFS <offset> [xV] <terminator>

Example: OUTPUT 707;"CHAN1:OFFS 500 mV"

Preset State: OV

Parameter Range: -lOV x PROBE to 10V x PROBE

Fundamental Unit: Volts

Query Syntax: CHANX:OFFS? <terminator>

Query Response: <NR3> <NL>

PROB

PROBES

command/query

The PROBES command sets the value of the probe multiplier for the indicated channel (CHANX). The query returns the current probe value.

Command Syntax: CHANX:PROB <multiplier> <terminator>

Example: OUTPUT 707;"CHANX:PROB 10"

Preset State: 1

Parameter Range: 1 x 10E-6 to 1 x 10E6

Query Syntax: CHANX:PROB? <terminator>

Query Response: <NR3> <NL>

RANG

RANGE

command/query

The RANGE command sets the voltage range (amplitude) for the indicated channel (CHANX). The input range for a signal will be: 0.1V to 20V x PROBE. The query returns the current voltage range.

Command Syntax: CHANX:RANG <range> [xV] <terminator>

Example: OUTPUT 707;"CHAN2:RANG 2.5V"

Preset State: 2.0V

Parameter Range: 0.1V x PROBE to 20.0V x PROBE

Fundamental Unit: Volts

Query Syntax: CHANX:RANG? <terminator>

Query Response: <NR3> <NL>

SA

SPECTRUM ANALYZER

command

The SPECTRUM ANALYZER command presets the voltage range, offset, trigger level, and input coupling of the digitizer to view the video output of a spectrum analyzer. The voltage range is set to 2.0V, the offset is 1.0V, the trigger level is 1.0V, and the input coupling is set to dc into I megohm.

Command Syntax: CHANX:SA <terminator>

Example: OUTPUT 707;"CHAN1:SA"

Preset State: Inactive

TTL

TRANSISTOR-TRANSISTOR LOGIC

command

The TTL command sets the voltage range, offset, and trigger level for the indicated channel (CHANX) to values appropriate for examining TTL signals. The voltage range is 8V (1.0 V/div), the offset is 1.6V, and the trigger level is 1.6V.

Command Syntax: CHANX:TTL <terminator>

Example: OUTPUT 707;"CHAN1:TTL"

Preset State: Inactive

TYPE

command/query

The TYPE command specifies trace data to be in either logarithmic or linear units and is only applicable in the SPECTRUM ANALYZER mode. The query returns the logarithmic/linear status.

When the SPECTRUM ANALYZER preset mode is selected, trace data is displayed as LOGged data in which the top graticule is the spectrum analyzer reference level and the bottom graticule is equal to -100 dB. In LINEAR units (unlogged data), the top graticule remains the reference level, but the bottom graticule becomes OV.

In LOGARITHMIC mode, the trace data is passed unchanged assuming a logged input. In LINEAR mode, the trace data is unlogged.

Command Syntax: CHANX:TYPE {LOG|LIN} <terminator>

Example: OUTPUT 707;"CHAN1:TYPE LIN"

Preset State: LOGARITHMIC

Query Syntax: CHANX:TYPE? <terminator>

Query Response: LOG|LIN <NL>

DISPLAY SUBSYSTEM

The Display Subsystem is used to control the display of data, markers, text, graticule, and screen format. Refer to Figure 3-12 for a syntax diagram of the Display Subsystem commands.

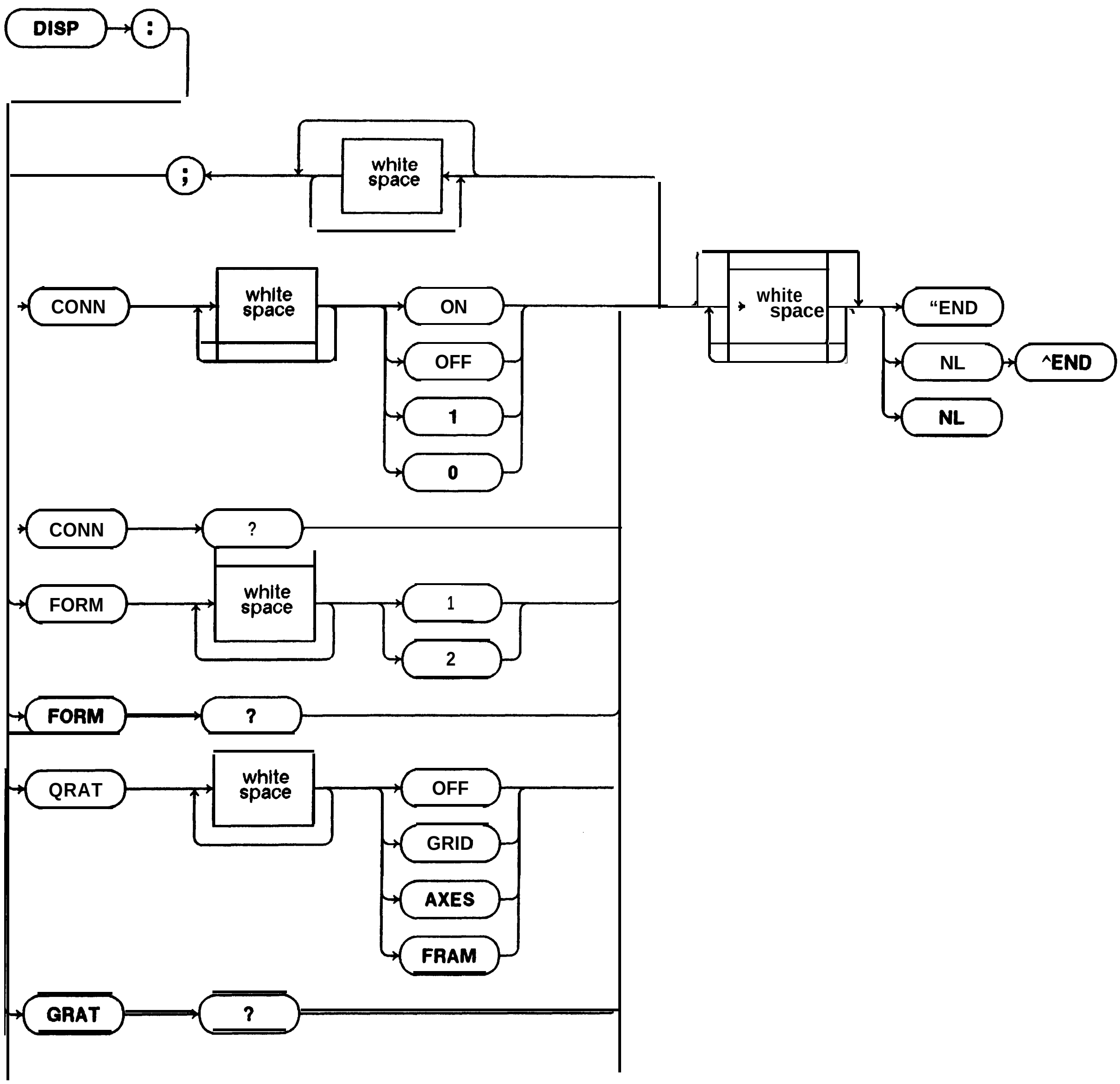


Figure 3-12. Display Subsystem Commands (1 of 2)

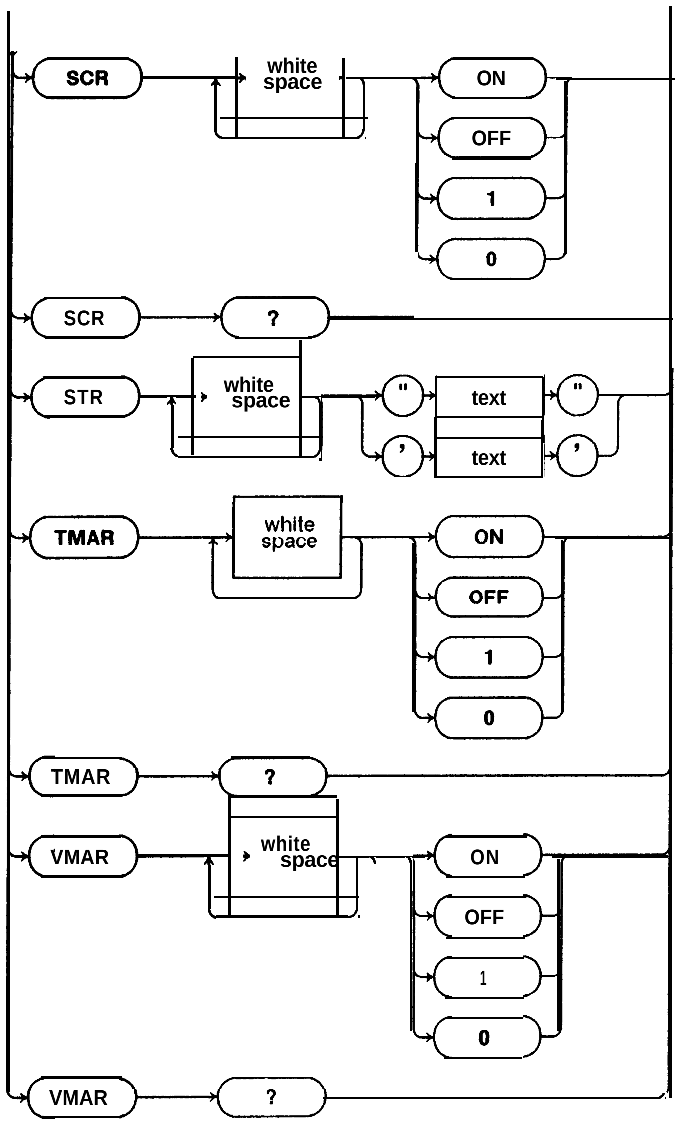


Figure 3-12. Display Subsystem Commands (2 of 2)

CONN

CONNECT

command/query

The CONNECT command turns the dots mode on or off. When ON (or 1) is selected, the waveform is displayed as a continuous solid trace. When OFF (or 0) is selected, the dots mode is enabled. The dotted line represents actual data points that were sampled, while the solid trace results from joining the dots with straight line segments. The query returns the current connect mode.

A maximum number of available memory data points may be sampled, but only 1024 data points or less may be displayed at any time. Each of the displayed data points corresponds directly to a data point that has actually been sampled. Selecting the number of data points for sampling may be achieved by using the POINTS command in the Acquire Subsystem. Refer to the Acquire Subsystem for more information on the POINTS command.

NOTE

Display Dot Generator release 3.2 or later must be installed in the display instrument when using the CONNECT OFF command.

Command Syntax: DISP:CONN {ON|OFF|1|0}<terminator>

Example: OUTPUT 707;"DISP:CONN 0"

Preset State: ON

Query Syntax: DISP:CONN? <terminator>

Query Response: ON|OFF <NL>

FORM

FORMAT

command/query

The FORMAT command sets the split-screen format. When 1 is selected, the normal full-screen format is displayed. When 2 is selected, a split-screen format is displayed. The query returns the current status of the screen format.

When the split-screen format is enabled, odd-numbered channel/memory/functions are displayed in the top portion of the screen and even-numbered channel/memory/functions are displayed in the bottom portion of the screen. The channel/memory/functions are only displayed if they have been previously turned on by the Top-Level Subsystem VIEW command.

Command Syntax: DISP:FORM {1|2} <terminator>

Example: OUTPUT 707;"DISP:FORM 2"

Preset State: 1

Parameter Range: 1 to 2
1 = full-screen format, 2 = split-screen format

Query Syntax: DISP:FORM? <terminator>

Query Response: <NR1> <NL>

GRAT

GRATICULE

command/query

The GRATICULE command sets the display graticule with one of three sets of vertical and horizontal lines that format the display screen. AXES superimposes one set of vertical and horizontal lines on the display screen. FRAME superimposes lines that border the edges of the display screen. GRID superimposes evenly-spaced vertical and horizontal lines on the display screen. The OFF parameter blanks all graticule format. The query returns the current display graticule format.

Command Syntax: DISP:GRAT {OFF|GRID|AXES|FRAM} <terminator>

Example: OUTPUT 707;"DISP:GRAT GRID"

Preset State: AXES

Query Syntax: DISP:GRAT? <terminator>

Query Response: OFF|GRID|AXES|FRAM <NL>

SCR

SCREEN

command/query

The SCREEN command sets the display screen status. When OFF, most of the display screen is blanked. The query returns the current screen setting.

Command Syntax: DISP:SCR{ON|OFF|1|0}<'terminator>

Example: OUTPUT 707;"DISP:SCR OFF"

Preset State: ON

Query Syntax: DISP:SCR? <terminator>

Query Response: ON|OFF <NL>

STR

STRING

command

The STRING command displays the input string parameter on the display screen.

Command Syntax: **DISP:STR** {"text"|'text'} **<terminator>**

Example: **OUTPUT** 707;"DISP:STR **'Label '"**
or
OUTPUT 707;"DISP:**STR** *Label*"

TMAR

TIME MARKERS

command/query

The TIME MARKERS command turns on and off the display of the time markers. The annotation related to the time markers is indicated by T(1) and T(2) at the bottom of the display screen, and is also turned on and off by this command. The query returns the current stat us of the time markers.

The time markers are turned on independently of the voltage markers, but their display on a channel, memory, or function is dependent on the Display Subsystem VMAR (VOLTAGE MARKER) command and the Measure Subsystem SOUR (SOURCE) command.

When the voltage markers are either off or displayed on a channel, the time markers are displayed on a channel. However, when the voltage markers are displayed on a memory, the time markers can only be displayed on the same memory. Refer to the VMAR command below for more detailed information on the function of the VOLTAGE MARKERS.

When the time markers are displayed, the TSTA (START MARKER) and TSTO (STOP MARKER) commands must be used to reposition the time markers. Refer to the TSTA and TSTO commands in the Measure Subsystem.

Command Syntax: **DISP:TMAR** {ON|OFF|1|0}**<terminator>**

Example: **OUTPUT** 707;"DISP:TMAR 1"

Preset State: **OFF**

Query Syntax: DISP:TMAR? **<terminator>**

Query Response: ON|OFF <NL>

VMAR

VOLTAGE MARKERS

command/query

The VOLTAGE MARKERS command turns on and off the display of the voltage markers. The annotation related to the voltage markers is indicated by V(J) and V(2) at the bottom of the display screen and is also turned on and off by this command. The query returns the current voltage marker status.

This command functions in conjunction with the Measure Subsystem SOURCE command to determine whether the voltage markers are displayed on a channel, memory, or function display. Refer to the SOUR command in the Measure Subsystem.

When the voltage markers are displayed, the VSTA (VOLTAGE MARKER 1) and VSTO (VOLTAGE MARKER 2) commands must be used to reposition the voltage markers. Refer to the VSTA and VSTO commands in the Measure Subsystem.

Command Syntax: DISP:VMAR {ON|OFF|1|0}<terminator>

Example: OUTPUT 707;"DISP:VMAR ON"

Preset State: OFF

Query Syntax: DISP:VMAR? <terminator>

Query Response: ON|OFF <NL>

DOMAIN SUBSYSTEM

The Domain Subsystem commands define the display domain to be either time or frequency. Refer to Figure 3-13 for a syntax diagram of the Domain Subsystem commands,

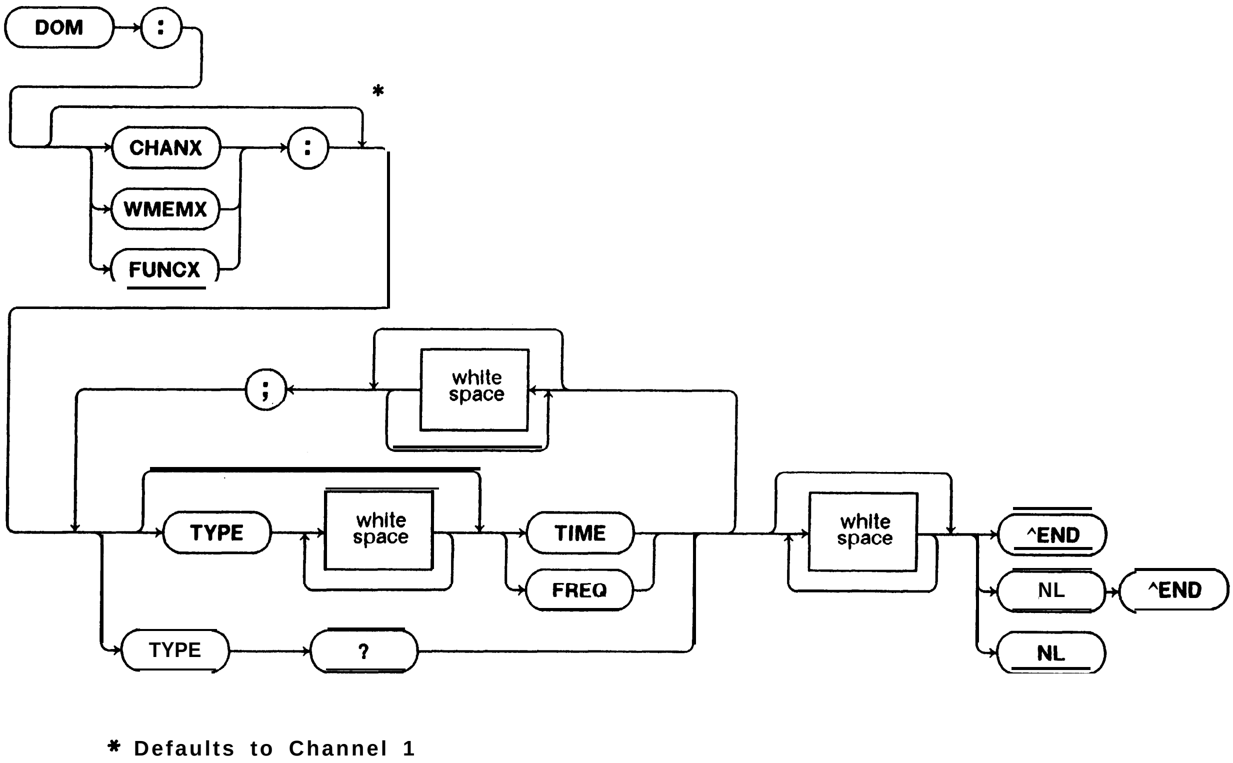


Figure 3- 13. Domain Subsystem Commands

TYPE

command/query

The TYPE command sets the domain status of the specified trace to be in either the time or frequency domain. The time domain displays data as amplitude versus time and the frequency domain displays data as a logarithmic amplitude versus frequency. The query returns the current status of the display domain.

NOTE

The trace specifier defaults to Channel 1 if a trace has not been specified.

When TIME is selected, all references to the horizontal axis are related to time.

When FREQUENCY is selected, a fast Fourier transform (FFT) is performed to translate the data from the time domain to the frequency domain. Also, all references to the horizontal axis are related to frequency. The range of frequencies displayed is the frequency span and is indicated at the lower right graticule edge. The grid display graticule of the FREQUENCY domain has a vertical range of 100 dB; the sensitivity per division is annotated in the lower-left portion of the display screen.

Command Syntax: DOM:[CHANX|WMEMX|FUNCX:] [TYPE] {TIME|FREQ} <terminator>

Example: OUTPUT 707;"DOM:FREQ" or OUTPUT 707;"DOM:CHAN4 TYPE FREQ"

Preset State: TIME

Query Syntax: DOM:TYPE? <terminator>

Query Response: TIME|FREQ <NL>

FUNCTION SUBSYSTEM

The Function Subsystem allows a trace math operation to be performed using the available channels and waveform memories as operands. The trace math operators are: ADD, INVERT, MULTIPLY, OFFSET, ONLY, RANGE, SUBTRACTION, and VERSUS. Refer to Figure 3-14 for a syntax diagram of the Function Subsystem commands.

NOTE

If the add, subtract, multiply, or versus operation is performed on two traces that are not equal in length, then the function length is set to the smaller trace size and the larger trace is "compressed" to the size of the smaller trace before the function is performed.

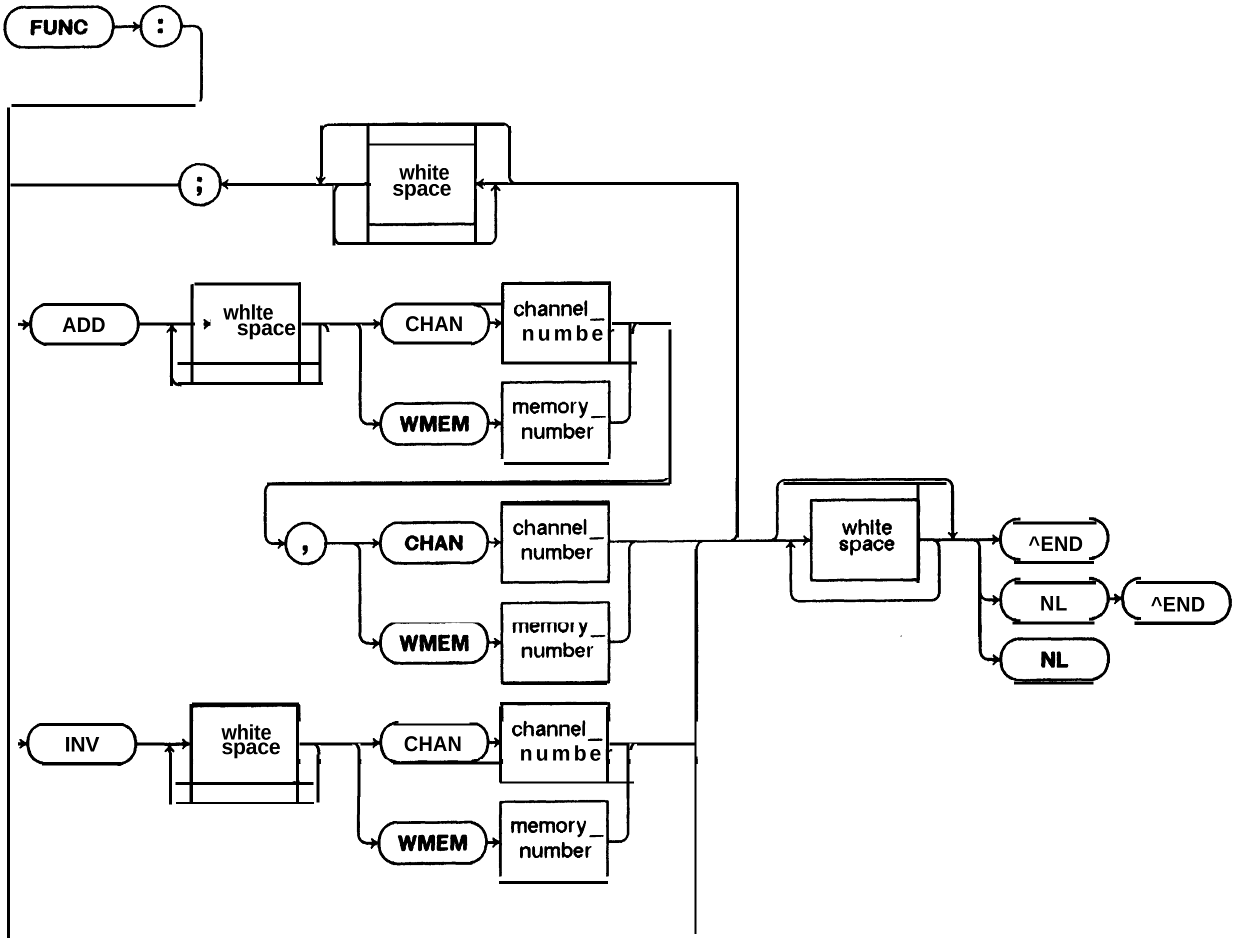


Figure 3-14. Function Subsystem Commands (1 of 3)

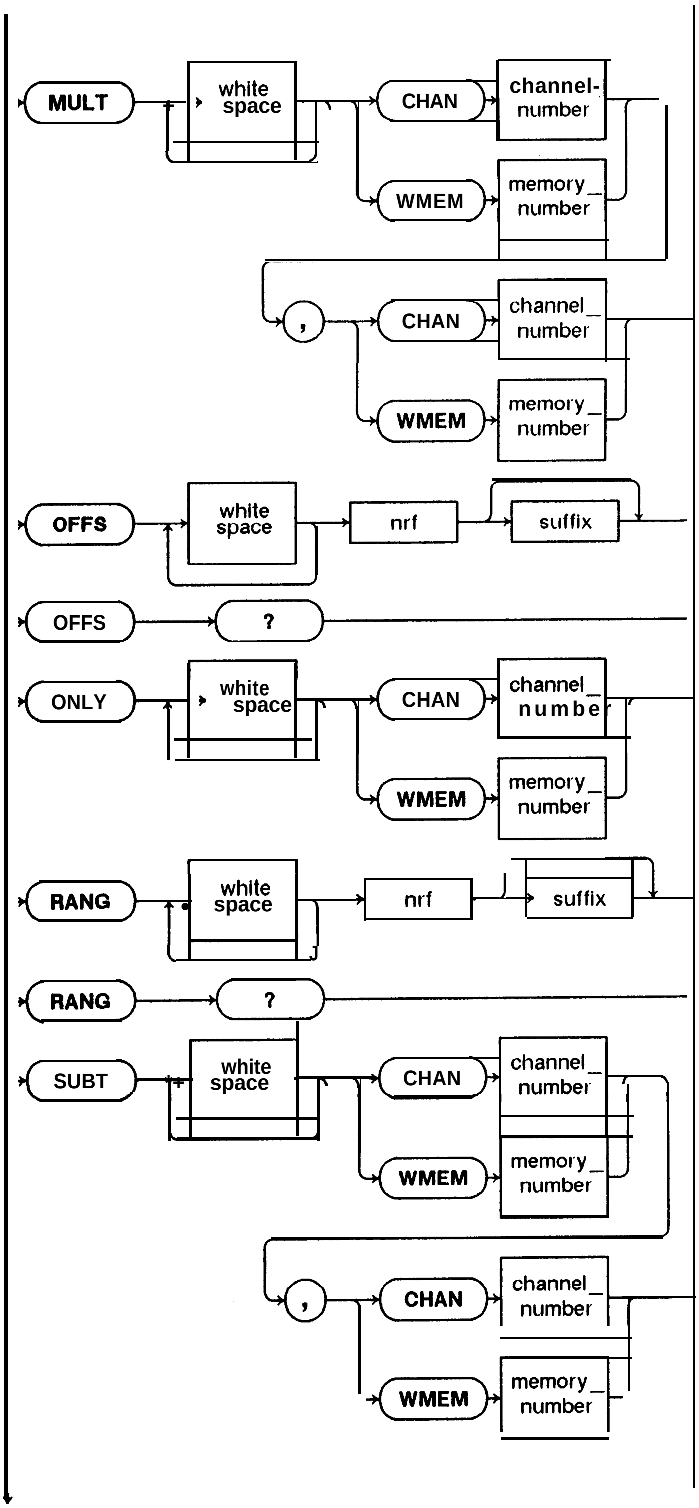


Figure 3-14. Function Subsystem Commands (2 of 3)

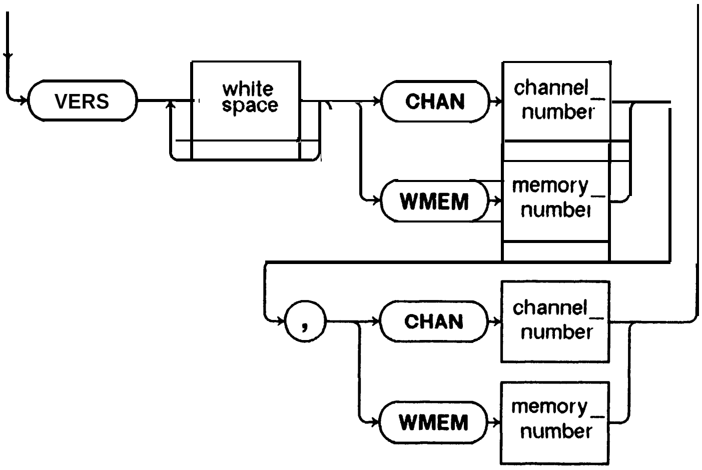


Figure 3-14. Function Subsystem Commands (3 of 3)

ADD

ADDITION command

The ADDITION command allows the two defined operands to be summed together algebraically.

Command Syntax: FUNC:ADD {CHAN<number>|WMEM<number>}, {CHAN<number>|WMEM<number>} <terminator>

Example: OUTPUT 707;"FUNC:ADD CHAN1,WMEM3"

Preset State: ADDITION of CHANNEL 1, CHANNEL 1

INV

INVERT command

The INVERT command allows the defined operand to be inverted.

Command Syntax: FUNC:INV {CHAN<number>|WMEM<number>} <terminator>

Example: OUTPUT 707;"FUNC:INV WMEM3"

Preset State: ADDITION of CHANNEL 1, CHANNEL 1

MULT

MULTIPLY

command

The MULTIPLY command sets the current trace math function to MULTIPLY with the parameters indicating the two operands. The two operands are algebraically multiplied together.

Command Syntax: FUNC:MULT {CHAN<number>|WMEM<number>},
{CHAN<number>|WMEM<number>} <terminator>

Example: OUTPUT 707;"FUNC:MULT CHAN2, WMEM1"

Preset State: ADDITION of CHANNEL 1, CHANNEL 1

OFFS

OFFSET

command/query

The OFFSET command sets the current voltage offset of the function. The query returns the current voltage offset.

Whenever a new operator or source is defined, the offset is set. as follows:

ONLY	offset of operand
INVERT	-offset of operand
ADDITION	first operand offset + second operand offset
SUBTRACTION	first operand offset + second operand offset

Command Syntax: FUNC:OFFS <offset> [xV] <terminator>

Example: OUTPUT 707;"FUNC:OFFS 2V"

Preset State: OV

Parameter Range: -20 x probe to +20 x probe
(where probe is the larger value of the probe for each operand)

Fundamental Unit: Volts

Query Syntax: FUNC:OFFS? <terminator>

Query Response: <NR3> <NL>

ONLY

command

The ONLY command allows the function to be defined as any available channel or waveform memory without any change.

Command Syntax: FUNC:ONLY {CHAN<number>|WMEM<number>} <terminator>

Example: OUTPUT 707;"FUNC:ONLY WMEM2"

Preset State: ADDITION of CHANNEL 1, CHANNEL 1

RANG

RANGE

command/query

The RANGE command allows the full-scale vertical axis of a function's display to be defined. The query returns the current range of the function.

Whenever a new operator or source is defined, the range is set as follows:

ONLY	range of operand
INVERT	range of operand
ADDITION	first operand range + second operand range
SUBTRACTION	first operand range + second operand range

Command Syntax: FUNC:RANG <range> [xV] <terminator>

Example: OUTPUT 707;"FUNC:RANG 2V"

Preset State: 4V

Parameter Range: .1V x probe to 20V x probe (where probe is the larger value of the probe for each operand)

Query Syntax: FUNC:RANG? <terminator>

Query Response: <NR3> <NL>

SUBT

SUBTRACTION

command

The SUBTRACTION command sets the current trace math function to SUBTRACTION with the parameters indicating the two operands. The second operand is subtracted from the first,

Command Syntax: FUNC:SUBT {CHAN<number>|WMEM<number>},
{CHAN<number>|WMEM<number>} **<terminator>**

Example: **OUTPUT 707;"FUNC:SUBT CHAN2, WMEM3"**

Preset State: **ADDITION of CHANNEL 1, CHANNEL 1**

VERS

VERSUS

command

The VERSUS command allows the two defined operands to be plotted with respect to each other on the X and Y axes. The first operand defines the X axis and the second operand defines the Y axis.

Command Syntax: FUNC:VERS {CHAN<number>|WMEM<number>},
{CHAN<number>|WMEM<number>} **<terminator>**

Example: **OUTPUT 707;"FUNC:VERS CHAN2, CHAN4"**

Preset State: **ADDITION of CHANNEL 1, CHANNEL 1**

MEASURE SUBSYSTEM

The commands in the Measure Subsystem allow various measurements to be made on a waveform, such as pulse parameter and voltage measurements. These measurements may also be customized by commands within the subsystem for particular applications.

NOTE

When a measurement cannot be made and a value is requested, a value of **1.0E38** will be returned.

Refer to Figure 3-15 for a syntax diagram of the Measure Subsystem commands.

Two terms that are frequently used in the Measure Subsystem command descriptions are defined below.

threshold refers to a level which is used as a reference in making a measurement. For example, if "lower threshold" is used, it refers to the point where the waveform crosses that threshold. The thresholds may be defined in terms of absolute voltages, or by referring to a percentage on the waveform (such as **10%**, or 90% referenced to TOP and BASE).

histogram refers to using a histogram to determine a top/base voltage value. A histogram of the waveform's voltage values is constructed. Next, the waveform is scanned to find the voltage values with the largest number of data points. If the maximum number of data points is greater than the limit criteria (approximately 5%) of the maximum number of data points in the record, that voltage level is used for the top or the base. If the limit criteria is not satisfied, the absolute minimum and maximum values are used as the base and the top.

NOTE

Due to various limitations, the Measure Subsystem is limited to a trace length of 1024 points or less. If a trace greater than 1024 points is measured, the results will not be exact because all of the data is not used. For example, if the trace length is 2048 points and the SAMPLE detector is **being** used, every other point is skipped. The measurement is then made on the resulting **1024-point** trace.

NOTE

Two percent hysteresis is used for any edge-searching done in the Measure Subsystem.

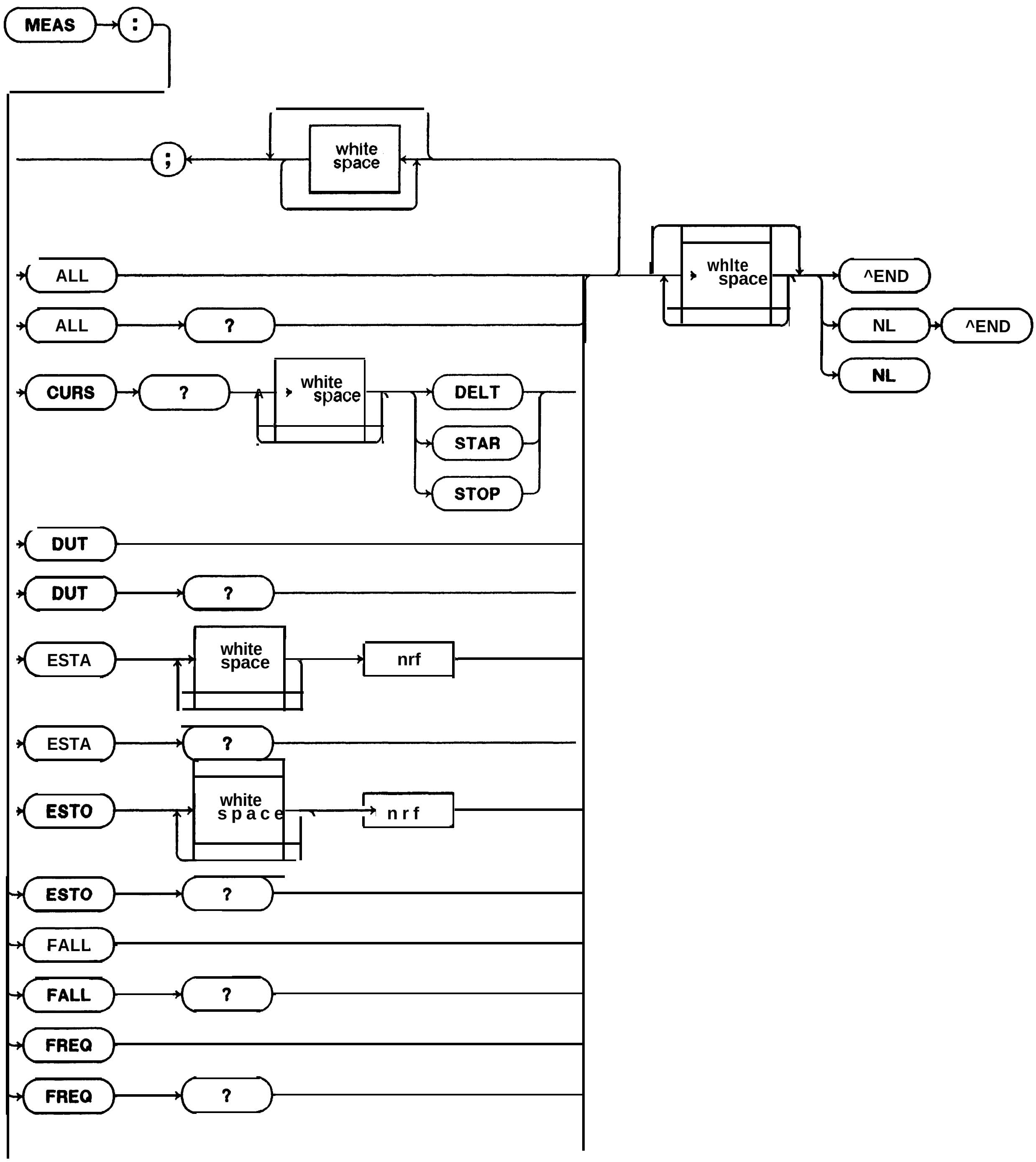


Figure 3-15. Measure Subsystem Commands (1 of 5)

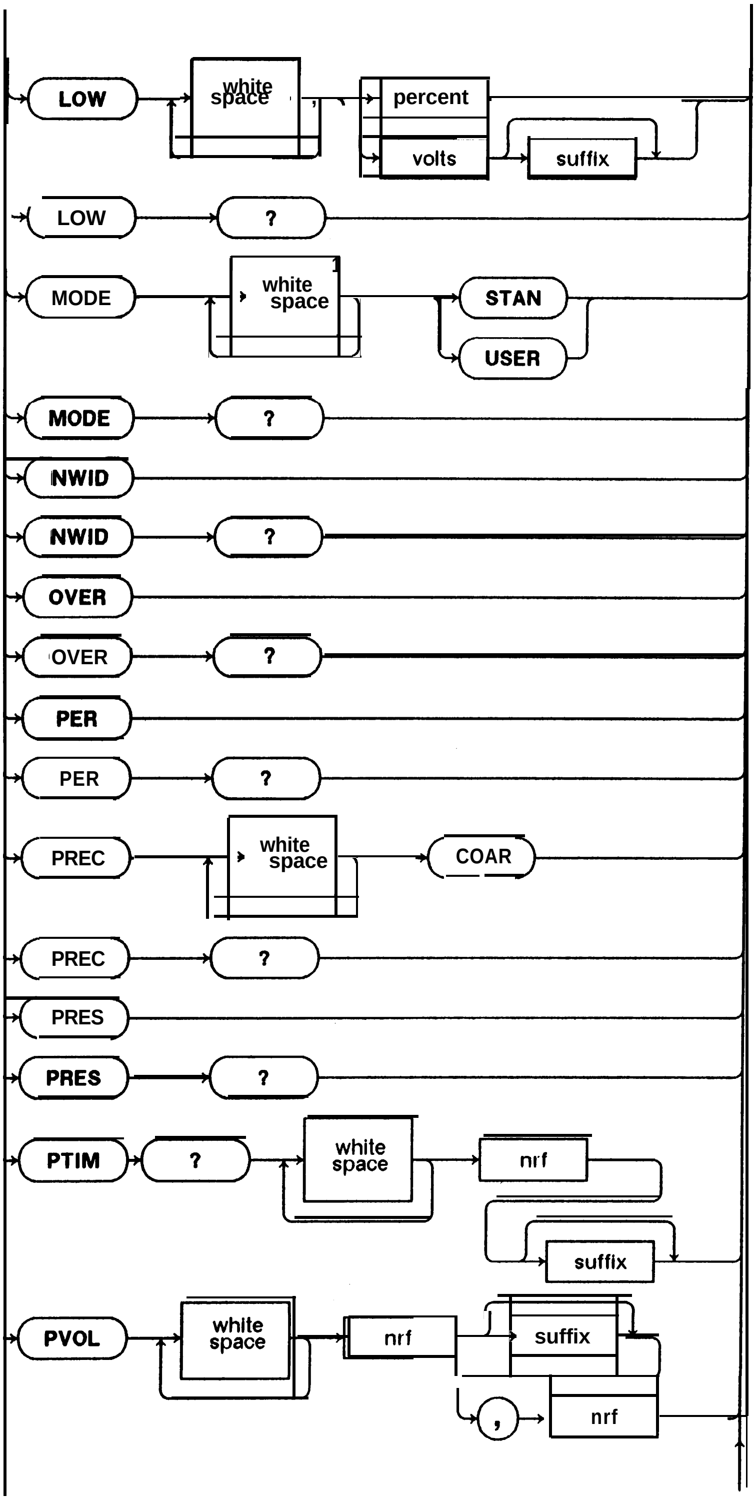


Figure 3-15. Measure Subsystem Commands (2 of 5)

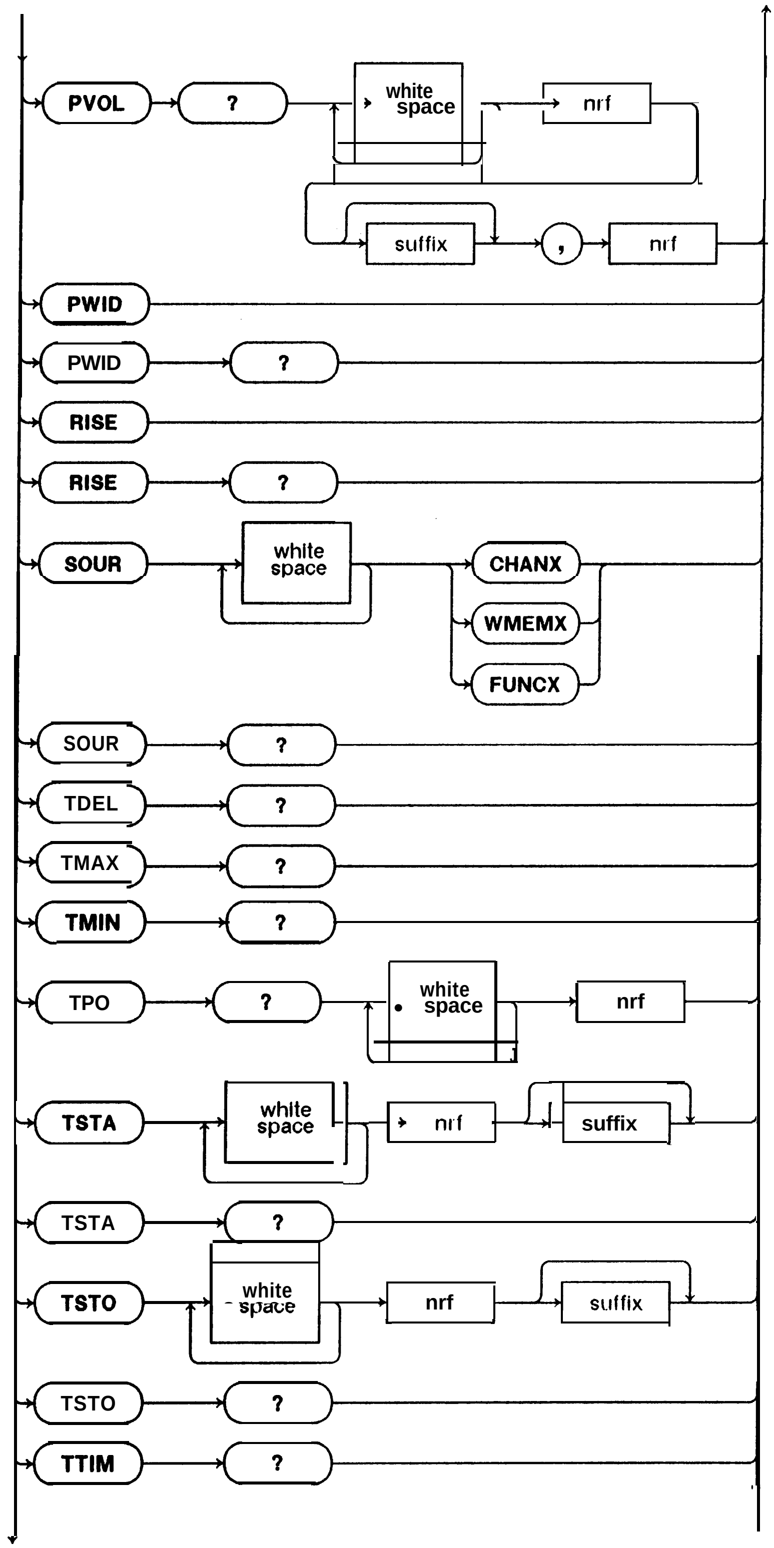


Figure 3-15. Measure Subsystem Commands (3 of 5)

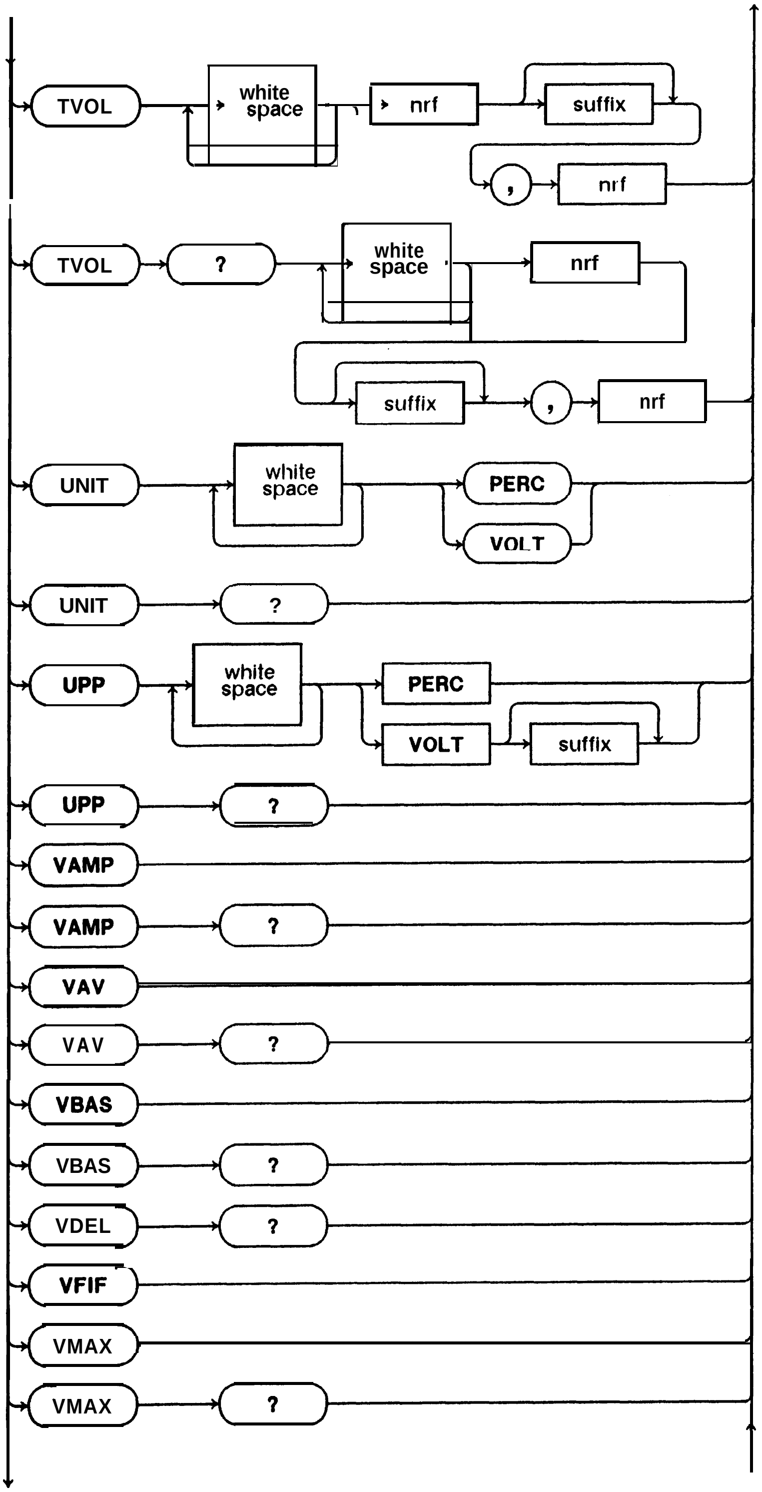


Figure 3-15. Measure Subsystem Commands (4 **of 5)**

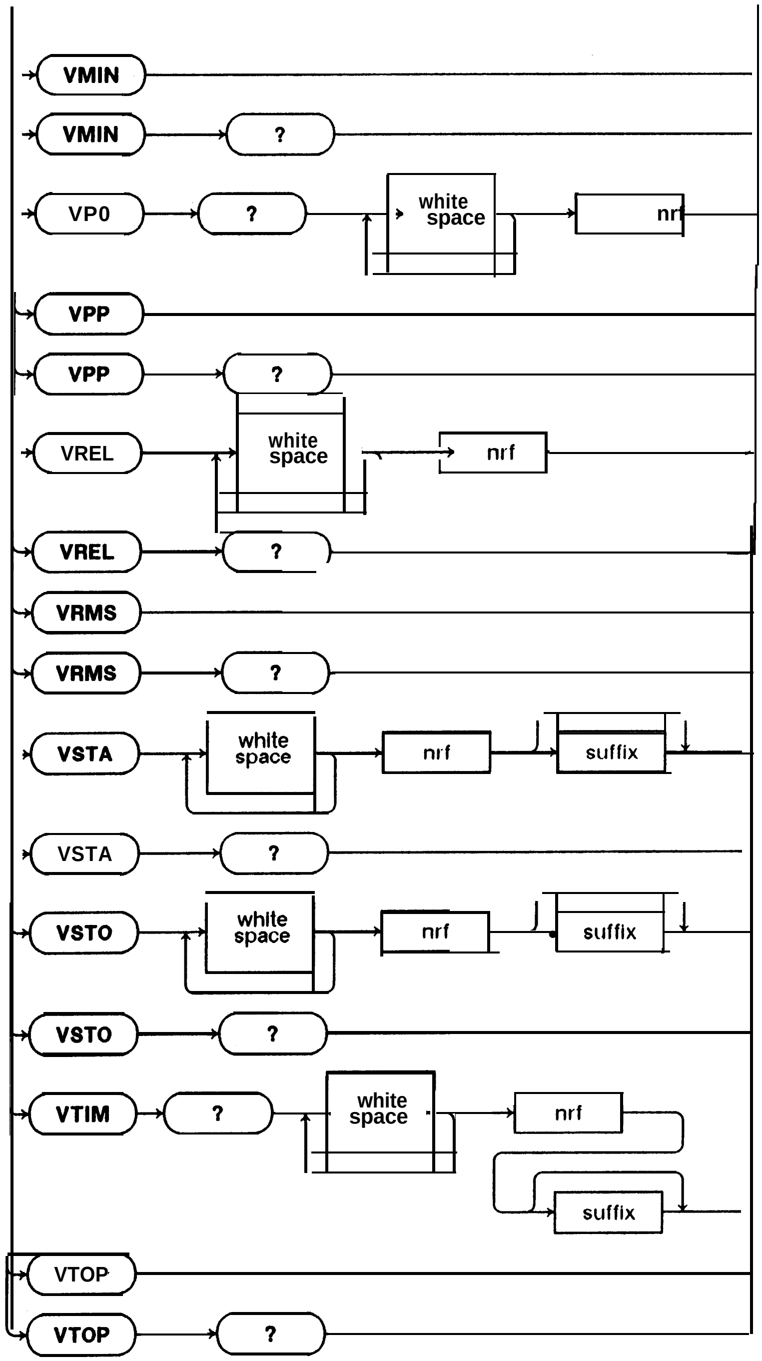


Figure 3-15. Measure Subsystem Commands (5 of 5)

ALL

command/query

The ALL command makes as many measurements as possible on the defined source waveform. The voltage and time markers are displayed at the TOP/BASE values and (edge 1)/(edge 2) (of one period), respectively. The query returns as many measurements as possible.

Command Syntax: MEAS:ALL <terminator>

Example: OUTPUT 707;"MEAS:ALL"

Query Syntax: MEAS:ALL? <terminator>

Query Response: The query response returns the following parameters as <NR3> values in the following sequence: <frequency>, <period>, <positive width>, <negative width>, <rise time>, <fall time>, <amplitude>, <peak-to-peak voltage>, <preshoot>, (overshoot>, <duty cycle>, <rms voltage>, <maximum voltage>, <minimum voltage>, <top voltage>, and <base voltage> <NL>

CURS

CURSOR

query

The CURSOR query returns the specified markers as an ordered pair of time and voltage values. DELTA specifies the difference between the two time markers and the two voltage markers. START specifies the positions of voltage marker 1 and the start marker. STOP specifies the positions of voltage marker 2 and the stop marker.

Query Syntax: MEAS:CURS? {DELT|STAR|STOP) <terminator>

Query Response: <NR3> (time), <NR3> (voltage) <NL>

DUT

DUTY CYCLE

command/query

The DUTY CYCLE command measures the duty cycle of the signal. Using the pulse width for this measurement, the markers are displayed at the 50% voltage point and (edge 1)/(edge 2) (of one period) of the waveform. The query returns the duty cycle of the signal.

The duty cycle is measured from 0 to 100% and is based on the following:

duty cycle (%) = (positive pulse width / period) x 100

Command Syntax: MEAS:DUT <terminator>

Example: OUTPUT 707;"MEAS: DUT"

Query Syntax: MEAS:DUT? <terminator>

Query Response: <NR3> <NL>

ESTA

START MARKER EDGE NUMBER

command/query

The ESTA command positions the start marker at a specified intersection of voltage marker 1 and the waveform. The sign of the parameter indicates the slope and the value indicates the intersection number, For example, -3 specifies the third negative slope intersection of the waveform and voltage marker 1. The query returns the currently-specified edge number of the start marker.

Command Syntax: MEAS:ESTA <edge number> <terminator>

Example: OUTPUT 707;"MEAS:ESTA -2"

Preset State: 1

Parameter Range: 1 to number of edges on display screen

Query Syntax: MEAS:ESTA? <terminator>

Query Response: <NR1> <NL>

ESTO

STOP MARKER EDGE NUMBER

command/query

The ESTO command positions the stop marker at a specified intersection of voltage marker 2 and the waveform. The sign of the parameter indicates the slope and the value indicates the intersection number. For example, -2 specifies the second negative slope intersection of the waveform and voltage marker 2. The query returns the currently-specified edge number of the stop marker.

Command Syntax: MEAS:ESTO <edge number> <terminator>

Example: OUTPUT 707;"MEAS:ESTO 3"

Preset State: 1

Parameter Range: 1 to number of edges on display screen

Query Syntax: MEAS:ESTO? <terminator>

Query Response: <NR1> <NL>

FALL

FALL TIME — command/query

The FALL command measures the fall time of the first falling edge whose defined thresholds are on the waveform. (The defined thresholds are dependent on the MODE command.) The voltage and time markers are displayed at the upper/lower thresholds and intersections of the waveform and the upper/lower thresholds, respectively. The query returns the current fall time of the waveform.

fall time = time at lower threshold – time at upper threshold

Command Syntax: MEAS:FALL <terminator>

Example: OUTPUT 707;"MEAS:FALL"

Query Syntax: MEAS:FALL? <terminator>

Query Response: <NR3> <NL>

FREQ

FREQUENCY — command/query

The FREQ command measures the frequency of the first complete period of the waveform. The markers are displayed at the 50% voltage point and (edge 1)/(edge 2) (of one period) of the waveform. The query returns the current frequency of the waveform.

The frequency measurement is based on:

frequency = 1 / period.

Command Syntax: MEAS:FREQ <terminator>

Example: OUTPUT 707;"MEAS:FREQ"

Query Syntax: MEAS:FREQ? <terminator>

Query Response: <NR3> <NL>

LOW

LOWER THRESHOLD — command/query

The LOW command sets the lower measurement threshold in either percent or volt units. This point is used for the lower threshold for rise and fall time measurements when the MODE command of the Measure Subsystem is specified as USER. When the lower threshold value is entered in volts, the range checking is done when a measurement is performed. The query returns the current value of the lower measurement threshold.

The LOWER THRESHOLD command functions in conjunction with the MODE, UNITS, and UPPER THRESHOLD commands of the Measure Subsystem. Refer to the respective command description in this section for more detailed information.

Command Syntax: MEAS:LOW {<percent>|<volts> [xV]} <terminator>

Example: For a 20 — 80% rise time measurement

```
OUTPUT 707;"MEAS:MODE USER";    ! Specifies how measurement
                                  is to be made.
OUTPUT 707;"UNIT PERC";         ! Defines units to be in percent.
OUTPUT 707;"LOW 20";            ! Sets the lower threshold at 20%.
OUTPUT 707;"UPP 80";            ! Sets the upper threshold at 80%.
OUTPUT 707;"RISE?"              ! Requests the rise time of the
                                  waveform.
ENTER 707;"RISE"                ! Retrieves the rise time value.
```

Preset State: 10%

Parameter Range: 0% to 50% for percent; BASE to UPPER for volts

Fundamental Unit: Volts (if specified in volt UNITS)

Query Syntax: MEAS:LOW? <terminator>

Query Response: <NR3> <NL>

MODE

command/query

The MODE command determines how rise and fall time measurements are to be made, STANDARD specifies the upper and lower thresholds as 10% and 90%. USER specifies the upper and lower threshold values defined by the LOWER THRESHOLD, UPPER THRESHOLD, and UNITS commands. The query returns the current measurement mode.

Command Syntax: MEAS:MODE {STAN|USER} <terminator>

Example: OUTPUT 707;"MEAS:MODE USER"

Preset State: STANDARD

Query Syntax: MEAS:MODE? <terminator>

Query Response: STAN|USER <NL>

NWID

NEGATIVE PULSE WIDTH

command/query

The NWID command measures the negative pulse width by determining the change in time from the first falling edge threshold to the next rising edge threshold. The same threshold is used for both edges. The markers are displayed at the 50% voltage point and (edge 1)/(edge 2) (of one period) of the waveform. The query returns the current negative pulse width of the waveform.

negative pulse width = -(time of first falling edge) + (time of next rising edge)

Command Syntax: MEAS:NWID <terminator>

Example: OUTPUT 707;"MEAS: NWID"

Query Syntax: MEAS:NWID? <terminator>

Query Response: <NR3> <NL>

OVER

OVERSHOOT

command/query

The OVERSHOOT command measures the overshoot of the defined waveform. The overshoot measurement uses the following algorithm:

if the first edge is positive, then overshoot = Vmax − TOP
if the first edge is negative, then overshoot = BASE − Vmin

The voltage and time markers are displayed at the TOP/BASE values and (edge 1)/(edge 2) (of one period), respectively. The query returns the current overshoot measurement value.

Command Syntax: MEAS:OVER <terminator>

Example: OUTPUT 707;"MEAS:OVER"

Query Syntax: MEAS:OVER? <terminator>

Query Response: <NR3> <NL>

PER

PERIOD

command/query

The PERIOD command measures the period of the waveform by determining the change in time between the first edge and the following "like" edge. The markers are displayed at the 50% voltage point and (edge 1)/(edge 2) (of one complete cycle) of the waveform. The query returns the current period of the waveform.

period = – (time of first edge) + (time of following "like" edge)

Command Syntax: MEAS:PER <terminator>

Example: OUTPUT 707;"MEAS:PER"

Query Syntax: MEAS:PER? <terminator>

Query Response: <NR3> <NL>

PREC

PRECISION

command/query

The PRECISION command specifies the precision of the measurements. Only COARSE precision is available at this time. The query returns the current precision status of the digitizer.

Command Syntax: MEAS:PREC (COAR) <terminator>

Example: OUTPUT 707;"MEAS:PREC COAR"

Query Syntax: MEAS:PREC? <terminator>

Query Response: COAR <NL>

PRES

PRESHOOT

command/query

The PRESHOOT command measures the preshoot of the defined waveform. The preshoot measurement uses the following algorithm:

if the first edge is positive, then preshoot = BASE – Vmin
if the first edge is negative, then preshoot = Vmax – TOP

The voltage and time markers are displayed at the TOP/BASE values and (edge 1)/(edge 2) (of one period), respectively. The query returns the current preshoot measurement value.

Command Syntax: MEAS:PRES <terminator>

Example: OUTPUT 707;"MEAS:PRES"

Query Syntax: MEAS:PRES? <terminator>

Query Response: <NR3> <NL>

PTIM

PRECEDING POINT OF REQUESTED TIME — query

The PTIM command returns the nearest point preceding the requested time. (The first point is the zero point.)

Query Syntax: MEAS:PTIM? <time> [xs] <terminator>

Query Response: <NR3> <NL>

Parameter Range: Left to right graticule edges of display screen

PVOL

POINT OF SPECIFIED VOLTAGE/INTERSECTION — command/query

The PVOL command sets voltage marker 1 to a defined voltage level and tries to find the edge intersection of voltage marker I and the waveform. If the edge intersection is negative, then a negative-going intersection is required; otherwise, a positive-going intersection will be searched for. The query returns the point number of the specified edge intersection in which the first point is the zero point.

Command Syntax: MEAS:PVOL <volt> [xV], <edge number> <terminator>

Example: OUTPUT 707;"MEAS:PVOL 1V, -3"

Parameter Range: Bottom of display screen to top of display screen,
1 to number of edges on display screen

Fundamental Unit: Volts

Query Syntax: MEAS:PVOL? <volt> [xV], <edge number> <terminator>

Query Response: <NR3> <NL>

PWID

POSITIVE PULSE WIDTH
command/query

The **PWID** command measures the positive pulse width by determining the change in time from the first rising edge threshold to the next falling edge threshold. The same threshold is used for both edges. The markers are displayed at the 50% voltage point and (edge 1)/(edge 2) (of one period) of the waveform. The query returns the current positive pulse width of the waveform.

positive pulse width = -(time of first rising edge) + (time of next falling edge)

Command Syntax: `MEAS:PWID <terminator>`

Example: `OUTPUT 707;"MEAS:PWID"`

Query Syntax: `MEAS:PWID? <terminator>`

Query Response: <NR3> <NL>

RISE

RISE. TIME
command/query

The RISE command measures the rise time of the first rising edge whose defined thresholds are on the waveform. (The defined thresholds are dependent on the MODE command.) The voltage and time markers are displayed at the upper/lower thresholds and intersections of the waveform and the upper/lower thresholds, respectively. The query returns the current rise time of the waveform.

rise time = time at upper threshold – time at lower threshold

Command Syntax: `MEAS:RISE <terminator>`

Example: `OUTPUT 707;"MEAS:RISE"`

Query Syntax: `MEAS:RISE? Germinator>`

Query Response: <NR3> <NL>

SOUR

SOURCE

command/query

The SOURCE command selects the source of data for subsequent operations in the Measure Subsystem. The source may be selected from any available channel, waveform memory, or function. The. query returns the current source.

Command Syntax: MEAS:SOUR {CHANX|WMEMX|FUNCX} <terminator>

Example: OUTPUT 707;'MEAS:SOUR WMEM2'

Preset State: CHANNEL 1

Query Syntax: MEAS:SOUR?

Query Response: CHANX|WMEMX|FUNCX <NL>

TDEL

TIME MARKER DELTA

query

The TDEL query returns the time difference between the start and stop time markers.

Query Syntax: MEAS:TDEL? <terminator>

Query Response: <NR3> <NL>

TMAX

TIME OF FIRST OCCURRENCE OF MAXIMUM VOLTAGE

query

The TMAX query returns the time at which the first occurrence of the current absolute maximum voltage (VMAX) occurred.

Query Syntax: MEAS:TMAX? <terminator>

Query Response: <NR3> <NL>

TMIN

TIME OF FIRST OCCURRENCE OF MINIMUM VOLTAGE — query

The TMIN query returns the time at which the first occurrence of the current absolute minimum voltage (VMIN) occurred.

Query Syntax: MEAS:TMIN? <terminator>

Query Response: <NR3> <NL>

TPO

TIME OF SPECIFIED POINT — query

The TPO query returns the time of the specified point (zero is the first point.)

Query Syntax: MEAS:TPO? <point number> <terminator>

Query Response: <NR3> <NL>

Parameter Range: Left to right graticule edges

TSTA

TIME OF START MARKER — command/query

The TSTA command sets the position of the start time marker. The query returns the position of the start marker.

Command Syntax: MEAS:TSTA <time> [xs] <terminator>

Example: OUTPUT 707;"MEAS:TSTA 10 ms"

Preset State: -2.5 us

Parameter Range: Left graticule edge to right graticule edge

Fundamental Unit: seconds

Query Syntax: MEAS:TSTA?

Query Response: <NR3> <NL>

TSTO

TIME OF STOP MARKER

command/query

The TSTO command sets the position of the stop time marker. The query returns the position of the stop marker.

Command Syntax: MEAS:TSTO <time> [xs] <terminator>

Example: OUTPUT 707;"MEAS:TSTO 20 ms"

Preset State: +2.5 us

Parameter Range: Left graticule edge to right graticule edge

Fundamental Unit: seconds

Query Syntax: MEAS:TSTO?

Query Response: <NR3> <NL>

TTIM

TIME OF REC TRIGGER EVENT

query

The TTIM query returns the time of the Random Event Capture (REC) trigger event with respect to the previous trigger event or a specific trigger event. (Refer to the DELT command in the **Timebase** Subsystem.) This query functions in conjunction with the REC Delta Time Status (DELT) command in the **Timebase** Subsystem.

Query Syntax: MEAS:TTIM? <terminator>

Query Response: <NR3> <NL>

TVOL

TIME OF SPECIFIED VOLTAGE/INTERSECTION

command/query

The TVOL command sets voltage marker 1 to a defined voltage level and tries to find the edge intersection of voltage marker 1 and the waveform. If the edge intersection is negative, then a negative-going intersection is required; otherwise, a positive-going intersection will be searched for. The query returns the time of the specified edge intersection with respect to the trigger in which the first point is the zero point.

Command Syntax: MEAS:TVOL <volt> [xV], <edge number> <terminator>

Example: OUTPUT 707;"MEAS:TVOL 1V, -3"

Parameter Range: Bottom graticule edge to top graticule edge,
1 to number of edges on display screen

Fundamental Unit: Volts

Query Syntax: MEAS:TVOL? <volt> [XV], <edge number>

Query Response: <NR3> <NL>

UNIT

UNITS

command/query

The UNITS command sets the measurement threshold units for marker positioning when a measurement is made. Units may be specified in either PERCENT or VOLTS. The query returns the currently-specified threshold units.

The UNITS command functions in conjunction with the MODE, UPPER THRESHOLD, and LOWER THRESHOLD commands of the Measure Subsystem. Refer to the respective command descriptions in this section for more detailed information.

Command Syntax: MEAS:UNIT {PERC|VOLT} <terminator>

Example: For a 20 — 80% rise time measurement

```
OUTPUT 707;"MEAS:MODE USER";    ! Specifies how measurement
                                  is to be made.
OUTPUT 707;"UNIT PERC";         ! Defines units to be in percent.
OUTPUT 707;"LOW 20";            ! Sets the lower threshold at 20%.
OUTPUT 707;"UPP 80";            ! Sets the upper threshold at 80%.
OUTPUT 707;"RISE?";             ! Requests the rise time of the
                                  waveform.
ENTER 707;"RISE"                ! Retrieves the rise time value.
```

Preset State: PERCENT

Query Syntax: MEAS:UNIT?

Query Response: PERC|VOLT <NL>

UPP

UPPER THRESHOLD

command/query

The UPP command sets the upper measurement threshold in either percent or volt units. This point is used for the upper threshold for rise and fall time measurements when the MODE command of the Measure Subsystem is specified as USER. When the upper threshold value is entered in volts, the range is checked when a measurement is performed. The query returns the current value of the upper measurement threshold.

The UPPER THRESHOLD command functions in conjunction with the MODE, UNITS, and LOWER THRESHOLD commands of the Measure Subsystem. Refer to the respective command description in this section for more detailed information.

Command Syntax: MEAS:UPP {<percent>|<volts> [xV]} <terminator>

Example: For a 20 – 80% rise time measurement

```
OUTPUT 707;"MEAS:MODE USER";    ! Specifies how measurement
                                  is to be made.
OUTPUT 707; "UNIT PERC";        ! Defines units to be in percent.
OUTPUT 707;"LOW 20";            ! Sets the lower threshold at 20%.
OUTPUT 707;"UPP 80";            ! Sets the upper threshold at 80%.
OUTPUT 707;"RISE?";             ! Requests the rise time of the
                                  waveform.
ENTER 707; "RISE"               ! Retrieves the rise time value.
```

Preset State: 90%

Parameter Range: 50% to 100% for percent; LOWER to TOP for volts

Fundamental Unit: Volts (if specified in volt UNITS)

Query Syntax: MEAS:UPP? <terminator>

Query Response: <NR3> <NL>

VAMP

SIGNAL AMPLITUDE

command/query

The VAMP command measures the signal amplitude in volts and is based on:

amplitude = Vtop – Vbase.

The query returns the current signal amplitude value.

Command Syntax: MEAS:VAMP <terminator>

Example: OUTPUT 707;"MEAS:VAMP"

Query Syntax: MEAS:VAMP? <terminator>

Query Response: <NR3> <NL>

VAV

SIGNAL AVERAGE — command/query

The VAV command measures the average voltage of the waveform. The query returns the current average voltage value.

Command Syntax: MEAS:VAV <terminator>

Example: OUTPUT 707;"MEAS:VAV"

Query Syntax: MEAS:VAV? <terminator>

Query Response: <NR3> <NL>

VBAS

WAVEFORM BASE VOLTAGE LEVEL — command/query

The VBAS command measures the voltage level of the base of the waveform. The query returns the current waveform base voltage level.

Command Syntax: MEAS:VBAS <terminator>

Example: OUTPUT 707;"MEAS:VBAS"

Query Syntax: MEAS:VBAS? <terminator>

Query Response: <NR3> <NL>

VDEL

VOLTAGE MARKER DELTA — query

The VDEL query returns the difference in voltage between voltage markers 1 and 2.

Query Syntax: MEAS:VDEL? <terminator>

Query Response: <NR3> <NL>

VFIF

VOLTAGE MARKERS TO 50% **command**

The VFIF command sets the voltage markers at the 50% level.

Command Syntax: MEAS:VFIF <terminator>

Example: OUTPUT 707;"MEAS:VFIF"

VMAX

MAXIMUM VOLTAGE OF SIGNAL **command/query**

The VMAX command measures the absolute maximum voltage of the waveform. The query returns the current maximum voltage of the waveform.

Command Syntax: MEAS:VMAX <terminator>

Example: OUTPUT 707;"MEAS:VMAX"

Query Syntax: MEAS:VMAX? <terminator>

Query Response: <NR3> <NL>

VMIN

MINIMUM VOLTAGE OF SIGNAL **command/query**

The VMIN command measures the absolute minimum voltage of the waveform. The query returns the current minimum voltage of the waveform.

Command Syntax: MEAS:VMIN <terminator>

Example: OUTPUT 707;"MEAS: VMIN"

Query Syntax: MEAS:VMIN? <terminator>

Query Response: <NR3> <NL>

VP0

VOLTAGE OF SPECIFIED POINT **query**

The **VPO** query returns the value (in volts) of the specified point. (The first point is the zero point.)

Query Syntax: MEAS:VPO? <point> <terminator>

Query Response: <NR3> <NL>

VPP

SIGNAL VOLTAGE PEAK-TO-PEAK **command/query**

The VPP command measures the peak-to-peak voltage of the waveform and is based on the following:

Vpk-pk = Vmax — Vmin.

The query returns the current peak-to-peak voltage value.

Command Syntax: MEAS:VPP <terminator>

Example: OUTPUT 707;"MEAS:VPP"

Query Syntax: MEAS:VPP? <terminator>

Query Response: <NR3> <NL>

VREL

RELATIVE VOLTAGE MARKER POSITIONING **command/query**

The VREL command sets the voltage marker positions as a function of the last established values. For example, after executing the VAMP command to position the voltage markers at the TOP and BASE of the waveform, "VREL 10" will position voltage marker 1 at 10%, and voltage marker 2 at 90%, of the original positions. The query returns the current relative voltage marker positioning.

Command Syntax: MEAS:VREL <percent> <terminator>

Example: OUTPUT 707;"MEAS:VREL 20"

Preset State: 100

Parameter Range: 10, 20, 30, 40, 50, 100

Query Syntax: MEAS:VREL? <terminator>

Query Response: 10|20|30|40|50|100 <NL>

VRMS

RMS VOLTAGE

command/query

The VRMS command measures the rms voltage of one complete period of the waveform. The query returns the current rms voltage.

Command Syntax: MEAS:VRMS <terminator>

Example: OUTPUT 707;"MEAS:VRMS"

Query Syntax: MEAS:VRMS? <terminator>

Query Response: <NR3> <NL>

VSTA

POSITION OF VOLTAGE MARKER 1

command/query

The VSTA command sets the position of voltage marker 1. The query returns the current position.

Command Syntax: MEAS:VSTA <volts> [xV] <terminator>

Example: OUTPUT 707;"MEAS:VSTA 5.0V"

Preset State: –0.5V

Parameter Range: Extends 12.5% above and below the top and bottom graticule edges

Fundamental Unit: Volts

Query Syntax: MEAS:VSTA? <terminator>

Query Response: <NR3> <NL>

VSTO

POSITION OF VOLTAGE MARKER 2 — command/query

The VSTO command sets the position of voltage marker 2. The query returns the current position.

Command Syntax: MEAS:VSTO <volts> [xV] <terminator>

Example: OUTPUT 707;"MEAS:VSTO 1.5V"

Preset State: 0.5V

Parameter Range: Extends 12.5% above and below the top and bottom graticule edges

Fundamental Unit: Volts

Query Syntax: MEAS:VSTO? <terminator>

Query Response: <NR3> <NL>

VTIM

VOLTAGE AT SPECIFIED TIME — query

The VTIM query returns the value (in volts) of the waveform at the specified time. If the specified time does not lie on top of a measured point, the result is interpolated from the two closest points.

Query Syntax: MEAS:VTIM? <time> [xs] <terminator>

Query Response: <NR3> <NL>

Fundamental Unit: seconds

VTOP

TOP VOLTAGE LEVEL — command/query

The VTOP command measures the voltage level of the TOP of the waveform. The query returns the current TOP voltage value.

Command Syntax: MEAS:VTOP Germinator>

Example: OUTPUT 707;"MEAS:VTOP"

Query Syntax: MEAS:VTOP? <terminator>

Query Response: <NR3> <NL>

TIMEBASE SUBSYSTEM

The Timebase Subsystem commands control the digitizer horizontal axis functions. Refer to Figure 3-16 for a syntax diagram of the Timebase Subsystem commands.

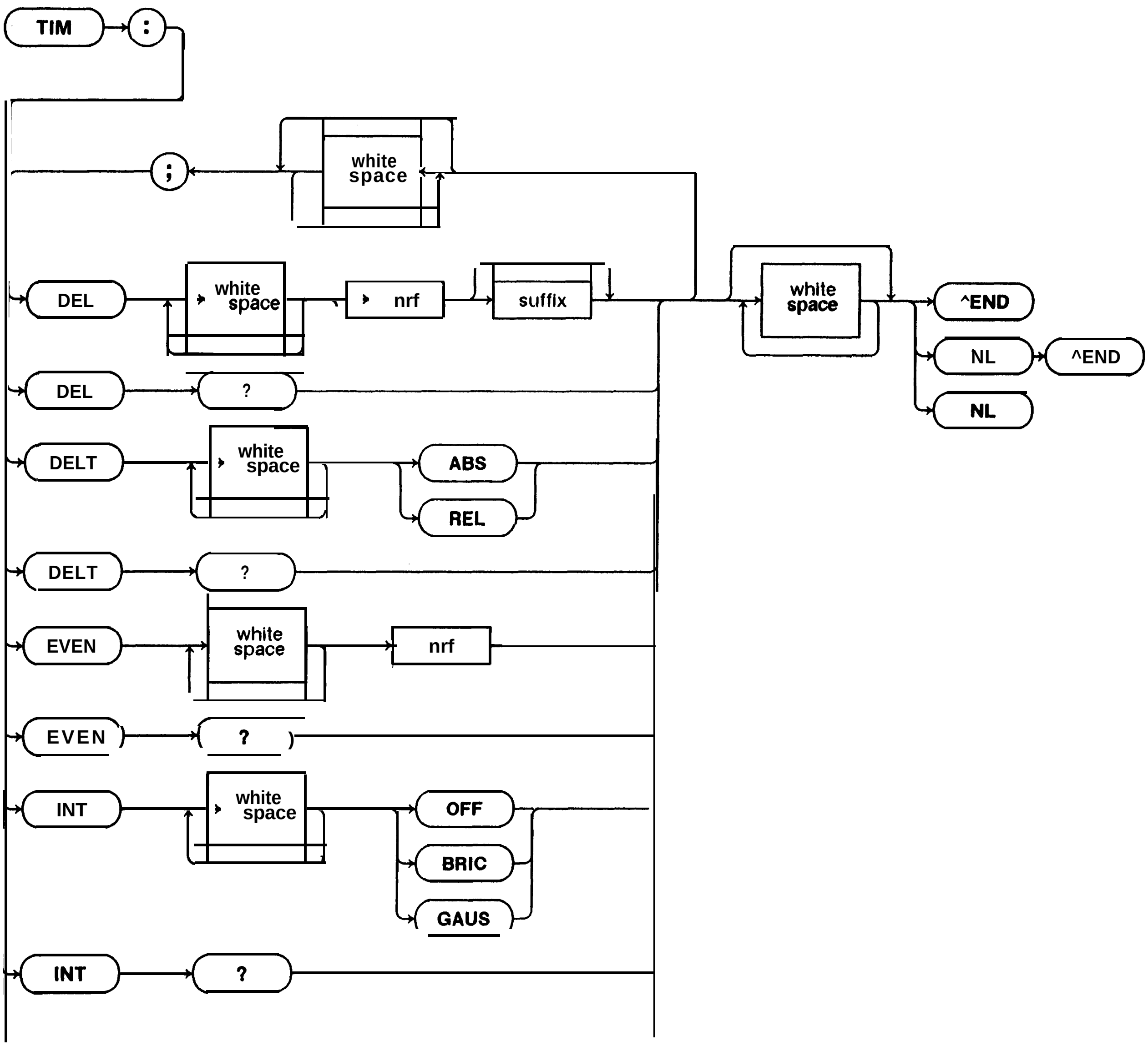


Figure 3-l 6. Timebase Subsystem Commands (1 of 3)

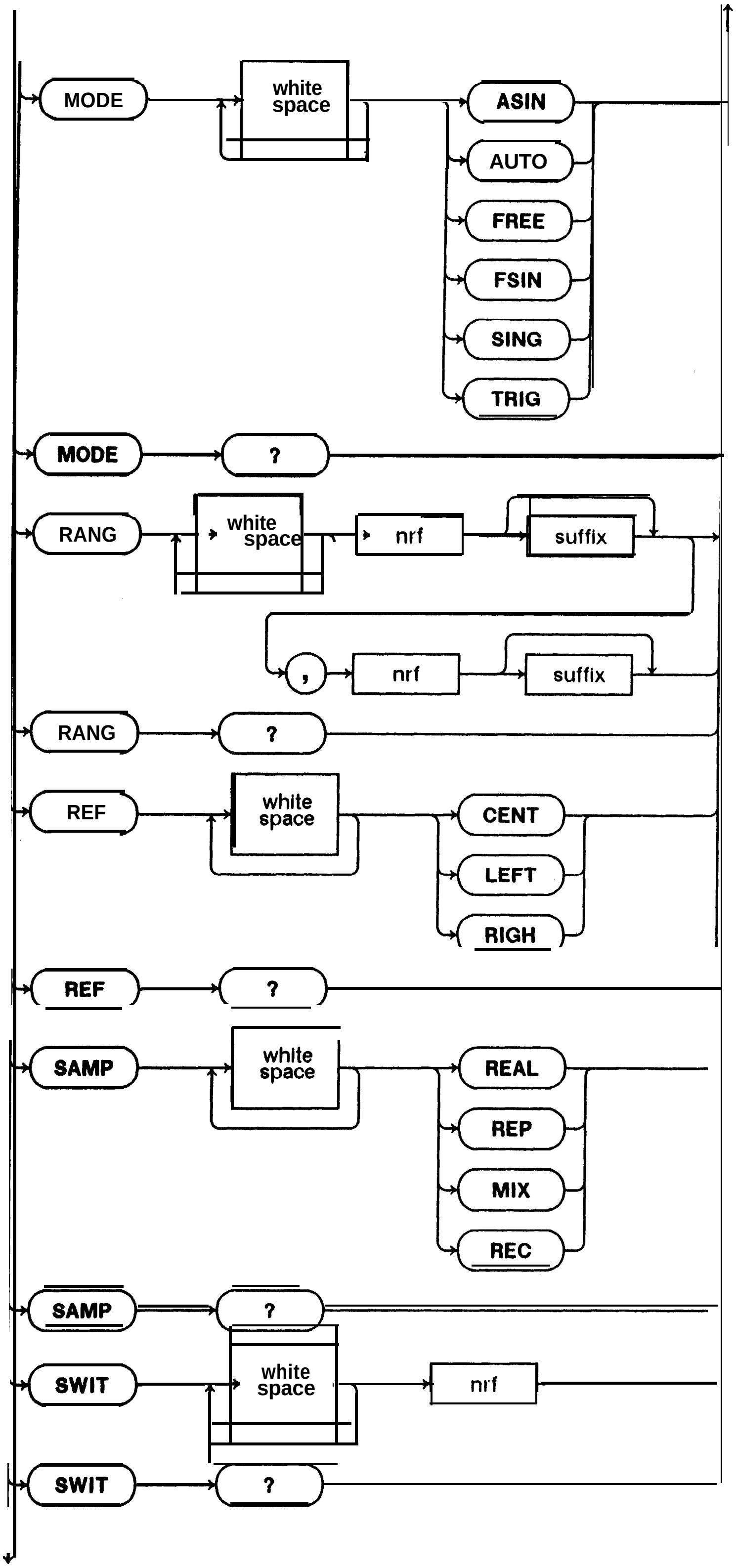


Figure 3-16. Timebase Subsystem Commands (2 of 3)

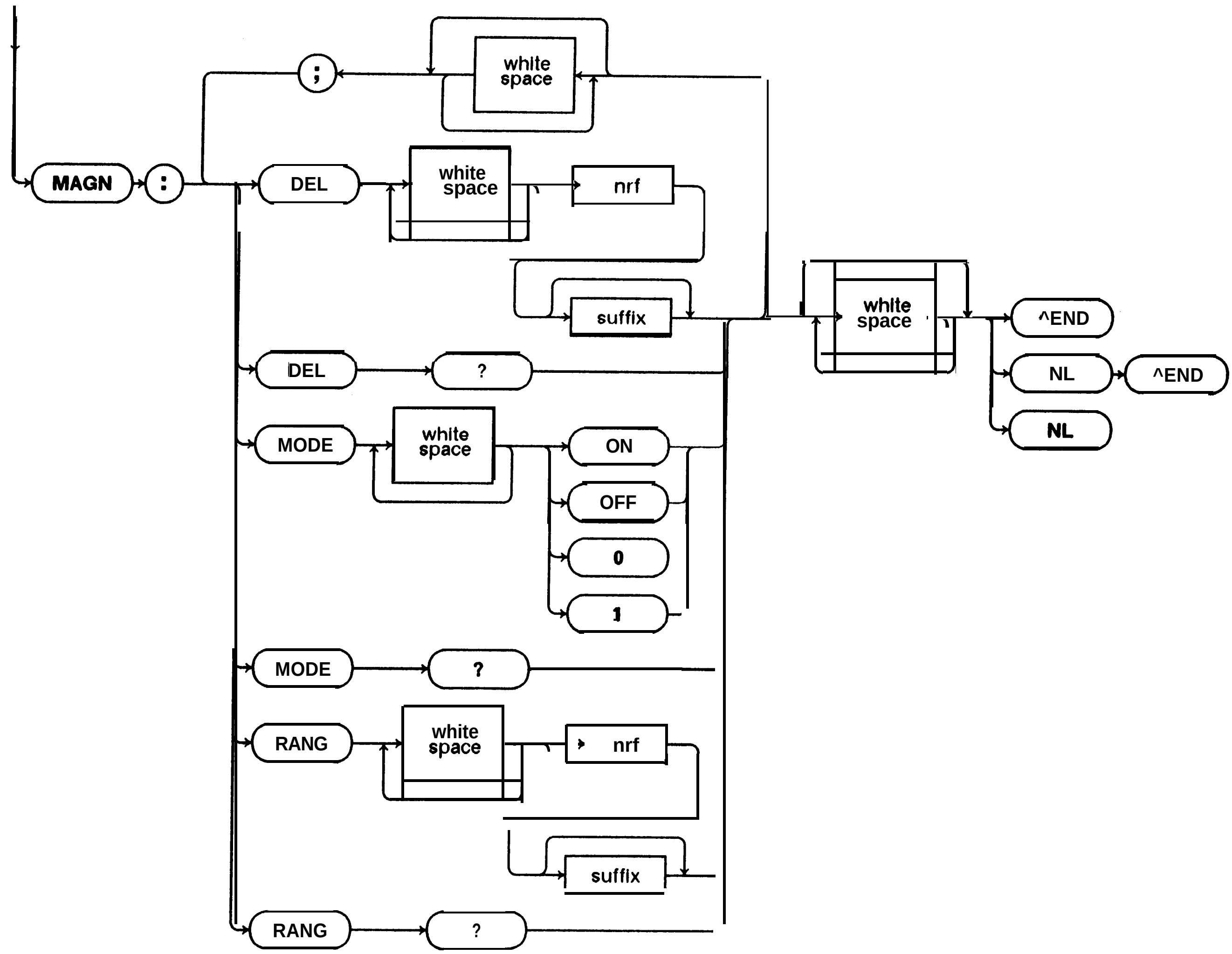


Figure 3-l 6. Timebase Subsystem Commands (3 of 3)

DEL

DELAY **command/query**

The DELAY command sets the timebase delay. This delay is the time interval between the trigger event and the onscreen delay reference point. The query returns the current delay time value.

Command Syntax: TIM:DEL <delay> [xs] <terminator>

Example: OUTPUT 707;"TIM:DEL 500 us"

Preset State: 0s

Parameter Range: The parameter range is dependent on the following three parameters: delay reference, delay, and seconds/division.

Delay Reference	Minimum	Maximum
left	-(time range)	(2E20 – length – 1) x time range/length
center	-(time range)/2	(2E20 – length/2 – 1) x time range/length
right	-(time range)/2	(2E20 – 1) x time range/length

Fundamental Unit: seconds

Query Syntax: TIM:DEL? <terminator>

Query Response: <NR3> <NL>

DELT

REC DELTA TIME STATUS **command/query**

The DELT command sets the delta time status for the Random Event Capture mode. When RELATIVE is specified, all times are referenced to the previous trigger event. When ABSOLUTE is specified, all times are referenced to the trigger event selected at the time the DELT command was executed. The query returns the current delta time status for the Random Event Capture mode.

RELATIVE time measures the elapsed time between the currently indicated trigger event and its previous trigger event.

ABSOLUTE time indicates the elapsed time as either positive or negative with respect to a specific trigger event.

NOTE

Whenever a Random Event Capture measurement is executed, the DELT command is reset to RELATIVE.

Command Syntax: TIM:DELT {ABS|REL} <terminator>

Example: OUTPUT 707;"TIM:DELT ABS"

Preset State: RELATIVE

Query Syntax: TIM:DELT? <terminator>

Query Response: ABS|REL <NL>

EVEN

EVENT

command/query

The EVENT command can specify any trigger event that was saved in memory in the Random Event Capture mode. Note that the digitizer must already be in the REC mode. (Refer to the SAMPLING command in the Timebase Subsystem.) The query returns the current trigger event.

Command Syntax: TIM:EVEN <event number> <terminator>

Example: OUTPUT 707;"TIM:EVEN 3"

Preset State: 1

Parameter Range: 1 to 32,767 (depending on how many trigger events were recorded)

Query Syntax: TIM:EVEN? <terminator>

Query Response: <NR1> <NL>

INT

INTERPOLATION

command/query

The INTERPOLATION command specifies the interpolation method used. At faster sweep rates, the interpolation functions, BRICK and GAUSSIAN, provide a more accurate graphic representation of the displayed waveform. This is achieved by up-sampling (increasing the sampling density of) the digitized waveform by zero-insertion, then passing the up-sampled waveform through a filter. The filtering serves to "interpolate" the inserted samples to appropriate values. BRICK selects an interpolation algorithm that uses a brick-wall filter for reconstructing the displayed waveform. GAUSSIAN selects an interpolation algorithm that uses a Gaussian filter. The query returns the current interpolation method used.

Command Syntax: TIM:INT {OFF|BRIC|GAUS} <terminator>

Example: OUTPUT 707;"TIM:INT GAUS"

Preset State: BRICK

Query Syntax: TIM:INT? <terminator>

Query Response: OFF|BRIC|GAUS <NL>

MODE

command/query

The MODE command specifies the timebase operating mode.

ASIN: auto-triggered, single sweep
AUTO: auto-triggered, continuous sweep
FREE: free-running, continuous sweep
FSIN: free-running, single sweep
SING: triggered single sweep
TRIG: triggered continuous sweep

The query returns the current timebase operating mode.

Command Syntax: TIM:MODE {ASIN|AUTO|FREE|FSIN|SING|TRIG}<terminator>

Example: OUTPUT 707;"TIM:MODE FSIN"

Preset State: AUTO

Query Syntax: TIM:MODE? <terminator>

Query Response: ASIN|AUTO|FREE|FSIN|SING|TRIG <NL>

RANG

RANGE

command/query

The RANGE command sets the time range. When the SAMPLING command specifies the MIXED operating mode, a second time range may be specified. (Refer to the SAMPLING command in the Timebase Subsystem.) The query returns the current time range(s).

When the MIXED operating mode is selected in the SAMPLING command, dual time ranges may be specified. The first time range indicates the sweep rate of the first portion of the waveform, and the second time range indicates the sweep rate of the second portion.

Command Syntax: TIM:RANG <time 1> [xs] [, <time 2> [xs]] <terminator>

Example: OUTPUT 707;"TIM:RANG 50 us"

Preset State: 10 us [, 10 us]

Fundamental Unit: seconds

Query Syntax: TIM:RANG? <terminator>

Query Response: <NR3> [, <NR3>]<NL>

REF

REFERENCE

command/query

The REFERENCE command sets a trigger event delay reference with respect to the left, center, or right graticule edge of the display screen. The query returns the current display screen position that the trigger delay is referenced to.

Command Syntax: TIM:REF {CENT|LEFT|RIGH} <terminator>

Example: OUTPUT 707;"TIM: REF LEFT"

Preset State: CENTER

Query Syntax: TIM:REF? <terminator>

Query Response: CENT|LEFT|RIGH <NL>

SAMP

SAMPLE

command/query

The SAMPLE command sets the operating mode of the digitizer. REAL TIME (REAL) implies “single-shot” capture of data in which a complete data record is collected on one trigger event. EQUIVALENT TIME SAMPLING (REP) implies that the input signal is stable, periodic, and repetitive; a data record is collected using multiple trigger events to form a composite waveform. MIXED mode (MIX) specifies a dual timebase function. RANDOM EVENT CAPTURE (REC) “captures” every successive trigger event of a measurement, sequentially, as long as predefined parameters are met. The query returns the current operating mode.

Command Syntax: TIM:SAMP {REAL|REP|MIX|REC} <terminator>

Example: OUTPUT 707;"TIM:SAMP MIX"

Preset State: REAL TIME

Query Syntax: TIM:SAMP? <terminator>

Query Response: REAL|REP|MIX|REC <NL>

SWIT

SWITCH

command/query

The SWITCH command sets the position on the display screen at which the time range switches **when** operating **the** instrument in the **MIXED timebase** mode (i.e., the point at which the first time range switches **to** the second time range in the **MIXED timebase** mode). The query returns the current position for switching time ranges.

Command Syntax: TIM:SWIT **<switch time> <terminator>**

Example: OUTPUT 707;"TIM:SWIT .40"

Preset State: 0.50

Parameter Range: 8/trace length to 1.0
(values are constrained to 8/trace length)

Query Syntax: TIM:SWIT? **<terminator>**

Query Response: <NR3> <NL>

MAGNIFY SUBSYSTEM (SUBSET OF THE TIMEBASE SUBSYSTEM)

The Magnify Subsystem allows a user to expand a portion of a **waveform and can only be activated** through the **Timebase** Subsystem. Refer to Figure 3-16 at the beginning of the **Timebase Subsystem section** for a syntax diagram of the Magnify Subsystem commands.

Single sweep mode is generally used for the Magnify Subsystem, since the **DELAY** and RANGE commands of the **Timebase** Subsystem may be used to perform the same function in the continuous sweep mode. Using a trace length with a large number of points is recommended to make the original measurement, since only the points used by the **RANGE** command of the Magnify Subsystem will be displayed. For example, for a **1000-point** trace, if the **RANGE** of the Magnify Subsystem is **10 μs** and the RANGE of the **Timebase** Subsystem is 1 ms (MAGNIFY:RANGE / **TIMEBASE:RANGE** = O.Ol), then only 10 points of the original trace would be displayed.

NOTE

When using the Waveform Subsystem to retrieve data, the displayed number of points will vary depending on the RANGE command of the Magnify Subsystem. The POINT query of the Waveform Subsystem may be used to determine the number of points in the magnified trace.

DEL

DELAY command/query

The DELAY command sets the delay for a magnified trace and is always referenced to the center of the display screen, regardless of the REFERENCE setting of the Timebase Subsystem. Once the size of the area to be magnified has been established by the RANGE command (Magnify Subsystem), the DELAY command (Magnify Subsystem) may be used (when repositioned) to view other portions of the displayed waveform of that particular size. The query returns the current delay value.

Command Syntax: TIM:MAGN:DEL <delay> [xs] <terminator>

Example: OUTPUT 707;"TIM:MAGN:DEL 100 us"

Preset State: 0s

Parameter Range: (original measurement start time + MAGN:RANG/2) to (original measurement stop time – MAGN:RANG/2) (i.e., the ends of the magnified trace must lie within the original measurement time)

Fundamental Unit: seconds

Query Syntax: TIM:MAGN:DEL? <terminator>

Query Response: <NR3> <NL>

MODE command/query

The MODE command sets the on/off status of the magnify function. The query returns the current magnify status.

Command Syntax: TIM:MAGN:MODE {ON|OFF|0|1} <terminator>

Example: OUTPUT 707;"TIM:MAGN:MODE 1"

Preset State: OFF

Query Syntax: TIM:MAGN:MODE? <terminator>

Query Response: ON|OFF <NL>

RANG

RANGE command/query

The RANGE command sets the time range of the area to be magnified for the magnify function. The query returns the current time range.

Command Syntax: TIM:MAGN:RANG <range> [xs] <terminator>

Example:. OUTPUT 707;"TIM:MAGN:RANG 50 us"

Preset State: 5 us

Parameter Range: (5 x original time range / original length) to (time range of the original measurement) (i.e., the ends of the magnified trace must lie within the original measurement)

Query Syntax: TIM:MAGN:RANG? <terminator>

Query Response: <NR3> <NL>

TRIGGER SUBSYSTEM

The Trigger Subsystem commands are used to define the conditions for a trigger. Refer to Figure 3-17 for a syntax diagram of the Trigger Subsystem commands.

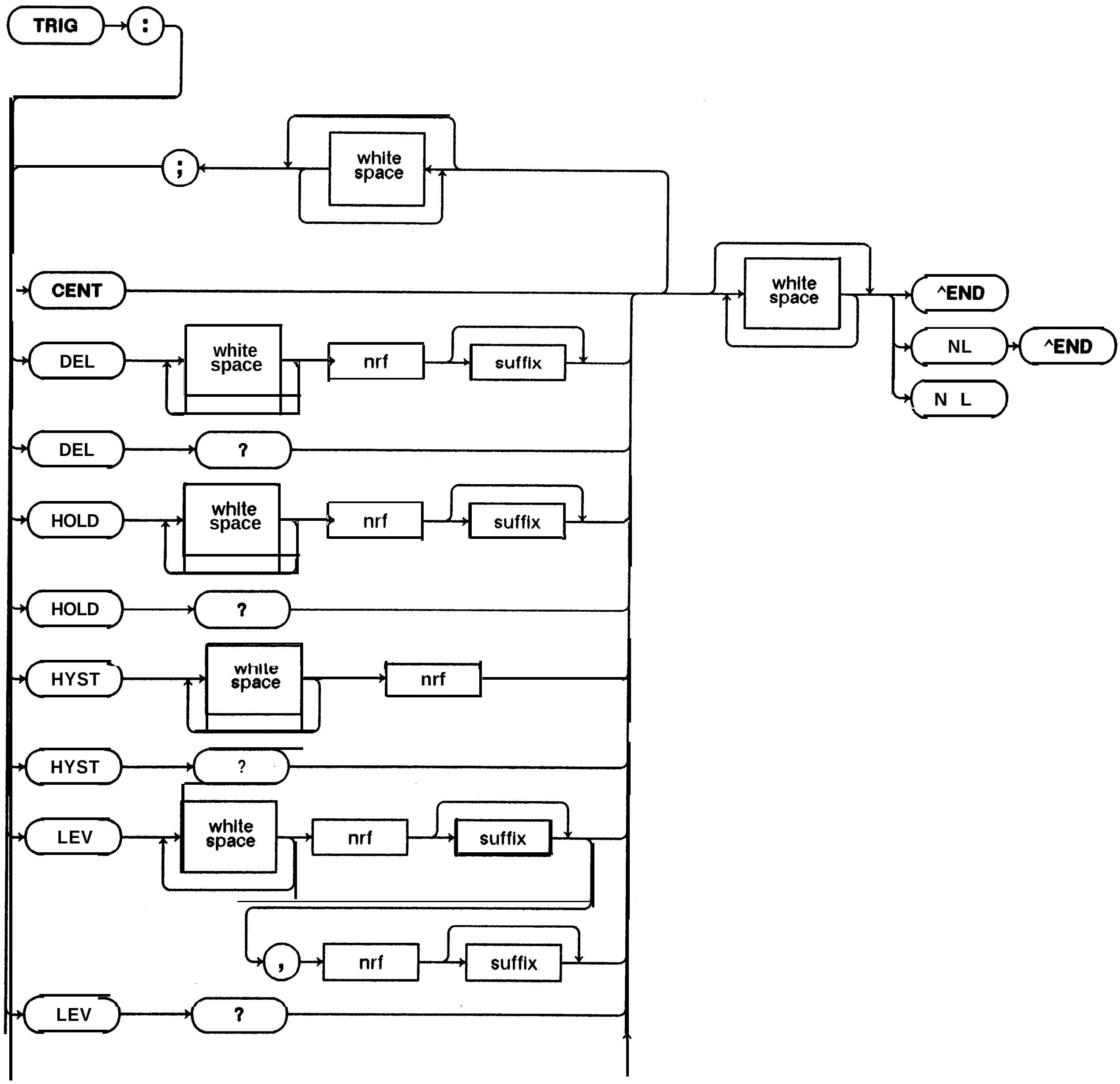


Figure 3-17. Trigger Subsystem Commands (1 of 2)

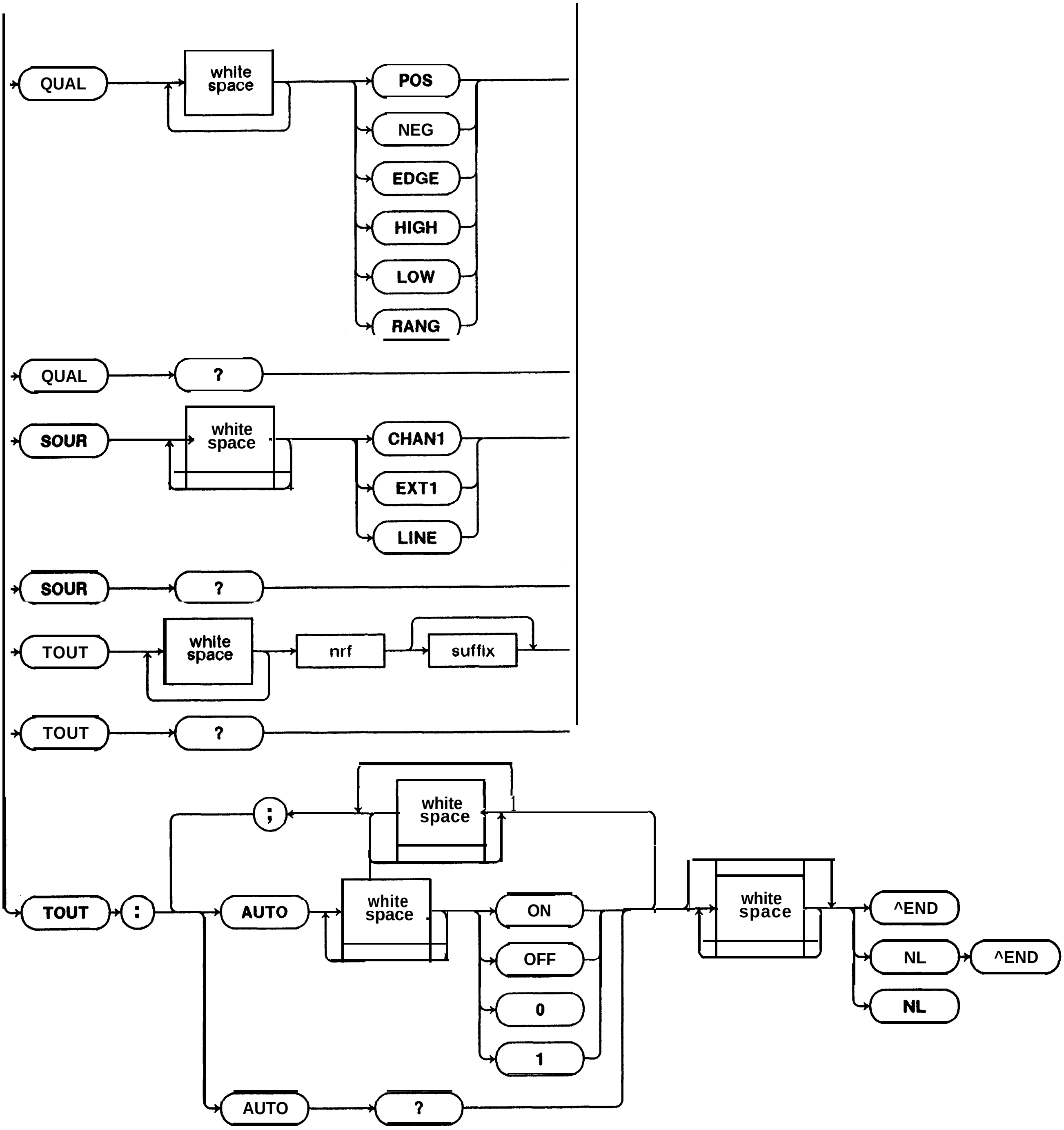


Figure 3-17. Trigger Subsystem Commands (2 of 2)

CENT

CENTER

command

The CENTER command sets the trigger level to the current midpoint of the display screen for internal triggering. (Also see the OFFSET command in the Channel Subsystem.) If RANGE of the QUALIFIER command is enabled, LEVEL 1 – LEVEL 2 is unchanged, but (LEVEL I + LEVEL 2 / 2) is set to the midpoint of the display screen.

Command Syntax: TRIG:CENT <terminator>

Example: OUTPUT 707;"TRIG:CENT"

DEL

DELAY

command/query

The DELAY command sets the timebase delay. This delay is the time interval between the trigger event and the onscreen delay reference point. The query returns the current delay time value.

Command Syntax: TRIG:DEL <delay> [xs] <terminator>

Example: OUTPUT 707;"TRIG:DEL 500 us"

Preset State: 0s

Parameter Range: The parameter range is dependent on the following three parameters: delay reference, delay, and seconds/division.

Delay Reference	Minimum	Maximum
left	-(time range)	(2E20 – length – 1) x time range/length
center	-(time range)/2	(2E20 – length/2 – 1) x time range/length
right	-(time range)/2	(2E20 – 1) x time range/length

Fundamental Unit: seconds

Query Syntax: TRIG:DEL? <terminator>

Query Response: <NR3> <NL>

HOLD

HOLDOFF

command/query

The **HOLDOFF** command sets the value for trigger **holdoff** time. The query returns the **current trigger holdof** f time.

Command Syntax: TRIG:HOLD <holdoff time> [xs] <terminator>

Example: OUTPUT 707;"TRIG:HOLD 100 us"

Preset State: 0s

Parameter Range: 0 to 10,000 seconds (resolution: 10 ms)

Fundamental Unit: seconds

Query Syntax: TRIG:HOLD? <terminator>

Query Response: <NR3> <NL>

HYST

HYSTERESIS

command/query

The HYSTERESIS command sets the value (in percent) for trigger hysteresis for internal triggering. The query returns the current percentage of hysteresis.

Hysteresis may be used as a safeguard to prevent false triggering. It is used to create a "window" that must be passed through before a trigger event may occur. To cause a trigger, the signal must first pass through the hysteresis level that has been set up. Therefore, if the hysteresis window exceeds the amplitude of the noise, false trigger events are prevented.

The hysteresis window is determined **by** a percentage of full-scale amplitude on the display screen. For example, if POSITIVE EDGE is selected as the QUALIFIER for triggering, the hysteresis window is below the trigger level. This is to ensure that the hysteresis level has been passed through prior to arming the trigger to occur at the next sample exceeding the set trigger level.

Command Syntax: TRIG:HYST <percent> <terminator>

Example: OUTPUT 707;"TRIG:HYST 15"

Preset State: 2%

Parameter Range: 1 to 50

Query Syntax: TRIG:HYST? <terminator>

Query Response: <NR3> <NL>

LEV

LEVEL

command/query

The LEVEL command sets the value (in volts) of the trigger level for internal triggering. The second trigger level is used only when the trigger QUALIFIER command is specified to be RANGE. When this condition is established, the signal must then be above LEVEL 1 and below LEVEL 2 to trigger. The query returns the current value of the trigger level.

Command Syntax: TRIG:LEV <level 1> [xV] [, <level 2> [xV]] <terminator>

Example: OUTPUT 707;"TRIG:LEV 500 mV" or
OUTPUT 707;"TRIG:LEV 250 mV, -250 mV"

Preset State: OV [, OV]

Parameter Range: Bottom graticule to top graticule

Fundamental Unit: Volts

Query Syntax: TRIG:LEV? <terminator>

Query Response: <NR3> [, <NR3>] <NL>

QUAL

QUALIFIER

command/query

The QUALIFIER command specifies the trigger qualifier. The query returns the current **trigger qualifier** status.

NOTE

For external or line triggering, only the edge qualifiers are available.

Positive Edge (POS) selects the positive slope of the trigger source for triggering.

Negative Edge (NEG) selects the negative slope of the trigger source for triggering.

Either Edge (EDGE) can select either the positive or negative slope of the trigger source for triggering.

Higher (HIGH) allows a measurement to be triggered at any value above the set trigger level.

Lower (LOW) allows a measurement to be triggered at any value below the set trigger level.

Range (RANG) allows a measurement to be triggered at any value outside of the specified ranges of trigger LEVEL 1 and trigger LEVEL 2.

Command Syntax: TRIG:QUAL {POS|NEG|EDGE|HIGH|LOW|RANGE} <terminator>

Example: OUTPUT 707;"TRIG: QUAL EDGE"

Preset State: Positive Edge

Query Syntax: TRIG:QUAL? <terminator>

Query Response: POS|NEG|EDGE|HIGH|LOW|RANG <NL>

SOUR

SOURCE **command/query**

The SOURCE command specifies the trigger source. The query returns the current trigger source status.

Channel 1 **(CHAN1)** selects the input **signal from Channel 1 as the trigger source,**

External **1 (EXT1) selects the signal from the external trigger connector as the trigger source,**

Line (LINE) selects the ac line as the trigger source.

Slave **(SLAV)** Refer to the Multiple Digitizer Remote Slaving Procedure at the end of Chapter 2, Programming Fundamentals, for use of this argument.

Command Syntax: TRIG:SOUR {CHAN1|EXT1|LINE|SLAV} <terminator>

Example: OUTPUT 707;"TRIG:SOUR LINE"

Preset State: CHANNEL 1

Query Syntax: TRIG:SOUR? <terminator>

Query Response: CHAN1|EXT1|LINE <NL>

TOUT

TIMEOUT

command/query

The TIMEOUT command sets the amount of time the instrument will wait for a trigger in **AUTO trigger** mode. Setting any value automatically turns the auto-coupling status to OFF (refer to the AUTO command, below). The query returns the current amount of time the instrument will wait for a trigger.

Command Syntax: TRIG:TOUT <time> [xs] <terminator>

Example: OUTPUT 707;"TRIG:TOUT 200 ms"

Preset State: 50 ms

Parameter Range: 0 to 1 x 10E6 seconds

Fundamental Unit: seconds

Query Syntax: TRIG:TOUT? <terminator>

Query Response: <NR3> <NL>

AUTO

command/query

The AUTO command sets the auto-coupling status of the TIMEOUT command in the Trigger Subsystem and the RANGE command in the Timebase Subsystem. The query returns the current auto-coupling status.

Command Syntax: TRIG:TOUT:AUTO {ON|OFF|0|1} <terminator>

Example: OUTPUT 707;"TRIG:TOUT:AUTO 1"

Preset State: ON

Query Syntax: TRIG:TOUT:AUTO? <terminator>

Query Response: ON|OFF <NL>

WAVEFORM SUBSYSTEM

The Waveform Subsystem commands are used to access trace data, and to transfer waveforms to and from the four waveform memories. Waveform data consists of a preamble and a data record. The preamble contains scaling information useful for interpreting the data record while the data record contains the actual waveform data values.

The actual values set in the preamble are determined when the DIGITIZE or RUN command is executed and are based on the settings of variables in the Acquire Subsystem. The following commands may be used to read values from a preamble in which each command will act on the preamble of the currently-specified source.

FORMAT	returns the data transmission mode
POINTS	returns the points value
PREAMBLE	returns the entire preamble
TYPE	returns the data type
XINC	returns the x increment
XOR	returns the x origin
XREF	returns the x reference
YINC	returns the y increment
YOR	returns the y origin
YREF	returns the y reference

Refer to Figure 3-18 for a syntax diagram of the Waveform Subsystem commands.

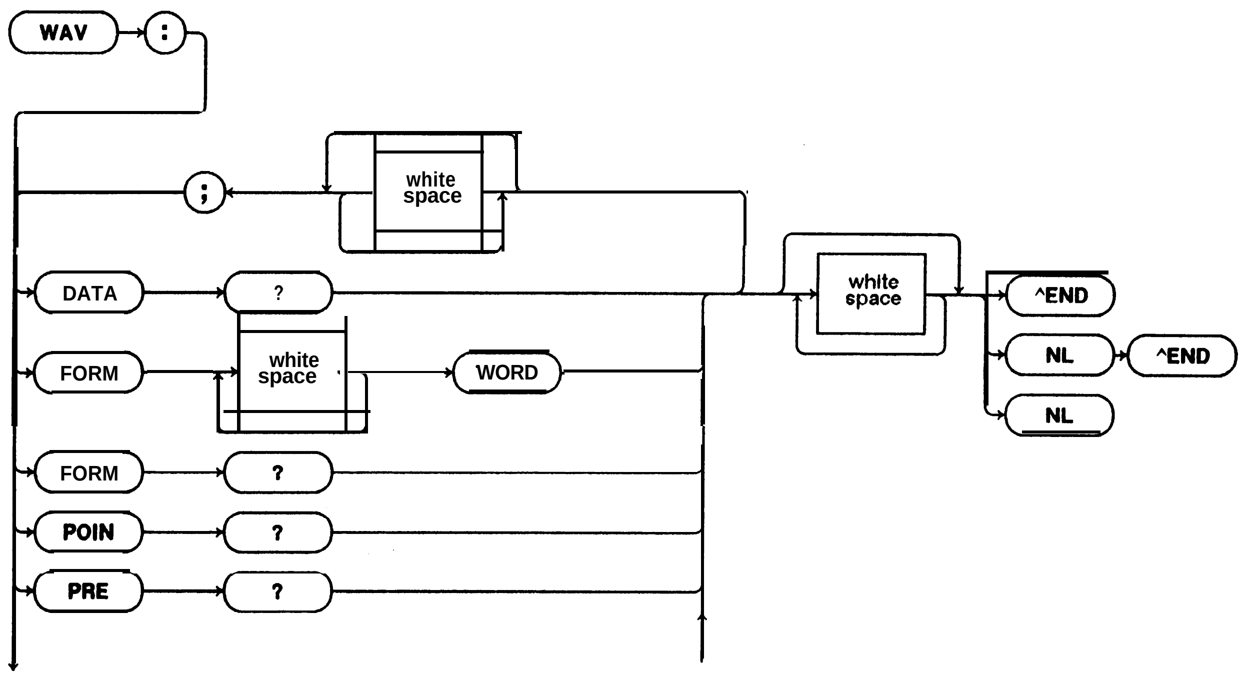


Figure 3-18. Waveform Subsystem Commands (1 of 2)

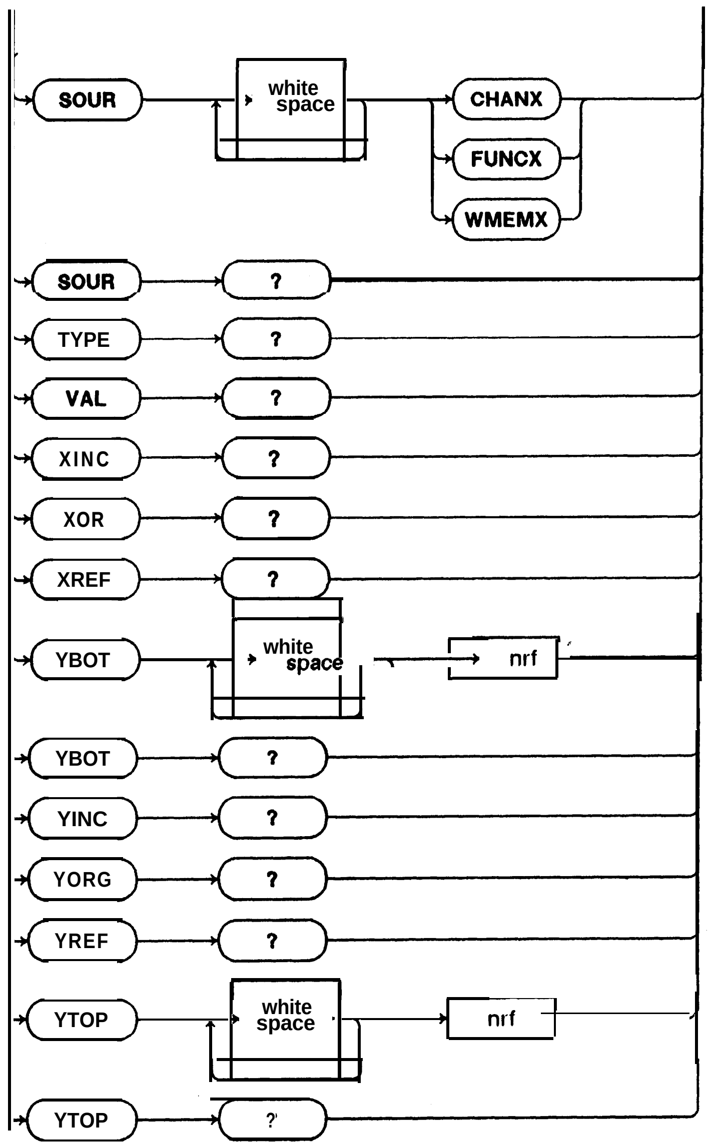


Figure 3-18. Waveform Subsystem Commands (2 of 2)

DATA **query**

The DATA query returns the data in the source trace. The data is an arbitrary length binary sequence (e.g., preceded by #0 and terminated with EOI).

The time of a given point is: XOR + XINC x point number, first. point is 0.

The voltage value of a point is: YOR + YINC x (value − YREF).

Query Syntax: WAV:DATA? <terminator>

Query Response: FORMAT = WORD:
#0 data END (where data is 16 bits per point)

FORM

FORMAT **command/query**

The FORMAT command selects the data transmission mode for the waveform data to be wide. WORD data is two bytes wide, with the most significant byte of each word's being transmitted first. The query returns the currently specified format.

Command Syntax: WAV:FORM (WORD} <terminator>

Example: OUTPUT 707;"WAV:FORM WORD"

Preset State: WORD

Query Syntax: WAV:FORM? <terminator>

Query Response: WORD <NL>

POIN

POINTS **query**

The POINTS query returns the number of data points in the source trace.

Query Syntax: WAV:POIN? <terminator>

Query Response: <NR1> <NL>

PRE

PREAMBLE

query

The PREAMBLE query returns a description of the source trace. The data returned is:

FORM	FORMAT
TYPE	TYPE
POIN	POINTS
XINC	X INCREMENT
XOR	X ORIGIN
XREF	X REFERENCE
YINC	Y INCREMENT
YOR	Y ORIGIN
YREF	Y REFERENCE

Query Syntax: WAV:PRE? <terminator>

Query Response: <FORM>,<TYPE>,<POIN>,<XINC>,<XOR>,<XREF>,<YINC>,<YOR>,
<YREF> <NL>

SOUR

SOURCE

command/query

The SOURCE command selects the channel, waveform memory, or function that is to be used as the source trace in subsequent WAVEFORM commands. The query returns the currently-specified source trace.

Command Syntax: WAV:SOUR {CHANX|WMEMX|FUNCX} <terminator>

Example: OUTPUT 707;"WAV:SOUR WMEM3"

Preset State: CHANNEL 1

Query Syntax: WAV:SOUR? <terminator>

Query Response: CHANX|WMEMX|FUNCX <NL>

TYPE **query**

The TYPE query returns the data type for the currently specified source trace. The two data types that may be returned are NORMAL and AVERAGE.

NORMAL data consists of the last data point in each time bucket. This data is transmitted over HP-IB in a linear fashion starting with time bucket 0 and proceeding through time bucket $n-1$, where n is the number returned by the WAVEFORM POINTS query. Only the voltage values of each data point are transmitted; the time values correspond to the position in the data array. The first voltage value corresponds to the first time bucket on the left side of the display screen, and the last value corresponds to the next-to-the-last time bucket on the right side of the display screen.

AVERAGE data consists of a moving average of the last m traces, where m is determined by the value returned by the ACQUIRE COUNT query when the measurement is made. This data is transmitted over the HP-IB in a linear fashion starting with time bucket 0 and proceeding through time bucket $n-1$, where n is the number returned by the WAVEFORM POINTS query. The first value corresponds to a point at the left side of the display screen, and the last value is one point away from the right side of the display screen.

Query Syntax: WAV:TYPE? <terminator>

Example: NORM|AVER <NL>

VAL

VALID **query**

The VALID query returns a 0 if the source trace does not have valid data. If there is valid data in the previously-selected source trace, the response will be 1.

Query Syntax: WAV:VAL? <terminator>

Query Response: 0|1 <NL>

XINC

X INCREMENT **query**

The XINC query returns the x-increment value currently in the preamble. This value is the time difference between adjacent data points in the source trace.

Query Syntax: WAV:XINC? <terminator>

Query Response: <NR3> <NL>

XOR

X ORIGIN

query

The XOR query returns the x-origin value currently in the preamble. This value is the time of the first point in the source trace.

Query Syntax: WAV:XOR? <terminator>

Query Response: <NR3> <NL>

XREF

X REFERENCE

query

The XREF query returns the x-reference value currently in the preamble. This value specifies the data point that is associated with the x-origin data values and is always 0.

Query Syntax: WAV:XREF? <terminator>

Query Response: 0 <NL>

YBOT

BOTTOM REFERENCE VALUE

command/query

The YBOT command sets the bottom reference value (i.e., the value corresponding to the bottom of the display screen). The query returns the current bottom reference value.

Command Syntax: WAV:YBOT <value> <terminator>

Example: OUTPUT 707;"WAV:YBOT 585"

Preset State: 0

Parameter Range: -32,767 to +32,767

Query Syntax: WAV:YBOT? <terminator>

Query Response: <NR1> <NL>

YINC

Y INCREMENT **query**

The YINC query returns the y-increment value currently in the preamble. This value is the voltage difference between adjacent data values.

Preset State: 0.488281250 mV (2V / 4096)

Query Syntax: WAV:YINC? <terminator>

Query Response: <NR3> <NL>

YOR

Y ORIGIN **query**

The YOR query returns the y-origin value currently in the preamble. This value is the voltage of the trace at center screen.

Preset State: 0.0V

Query Syntax: WAV:YOR? <terminator>

Query Response: <NR3> <NL>

YREF

Y REFERENCE **query**

The YREF query returns the y-reference value currently in the preamble. This value specifies the data point where the y origin occurs.

Preset State: 2048

Query Syntax: WAV:YREF? <terminator>

Query Response: <NR3> <NL>

YTOP

TOP REFERENCE VALUE

command/query

The YTOP command sets the top reference value (i.e., the value corresponding to the top of the display screen). The query returns the current top reference value.

Command Syntax: WAV:YTOP <value> <terminator>

Example: OUTPUT 707;"WAV:YTOP 766"

Preset State: 4095

Parameter Range: -32,767 to +32,767

Query Syntax: WAV:YTOP? <terminator>

Query Response: <NR1> <NL>

WINDOW SUBSYSTEM

The Window Subsystem commands are used to specify the window to be used by the fast Fourier transform (FFT) algorithm when translating data from the time domain to the frequency domain, Refer to Figure 3-19 for a syntax diagram of the Window Subsystem commands.

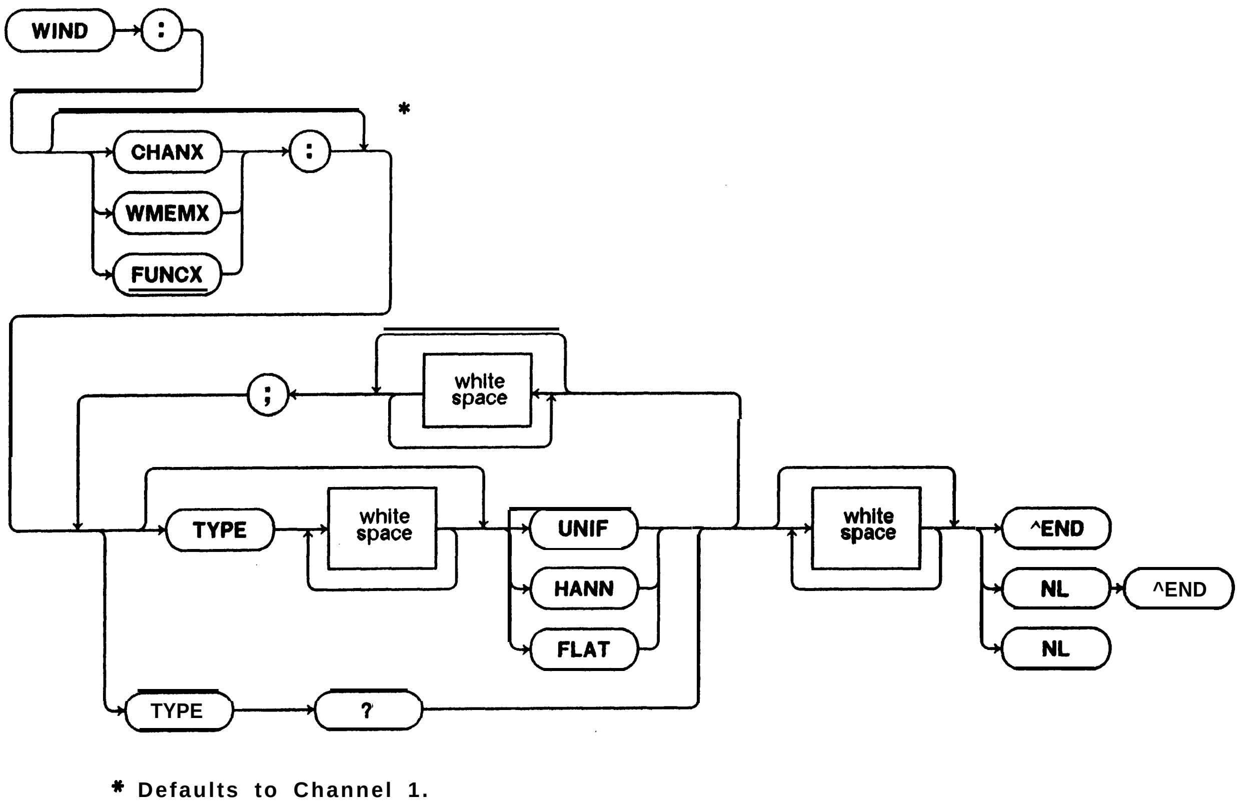


Figure. 3-19. Window Subsystem Commands

TYPE

command/query

The TYPE command selects the window to be used on the specified source trace when displaying data in the frequency domain. The three window selections are UNIFORM, HANNING, and FLATTOP. The query returns the currently specified window.

Command Syntax: WIND:[CHANX|WMEMX|FUNCX:][TYPE] {UNIF|HANN|FLAT} <terminator>

Example: OUTPUT 707;"WIND:WMEM2:TYPE UNIF"

Preset State: HANNING

Default: CHANNEL 1 for source trace
(i.e., WIND:TYPE = WIND:CHAN1:TYPE)

Query Syntax: WIND:TYPE? <terminator>

Query Response: UNIF|HANN|FLAT <NL>

APPENDIX A
COMMAND LISTING BY SUBSYSTEM

SUBSYSTEM	DESCRIPTION	MNEMONIC
COMMON COMMAND SET	CLEAR STATUS	*CLS
	STANDARD EVENT STATUS ENABLE	*ESE
	STANDARD EVENT STATUS REGISTER	*ESR
	IDENTIFICATION	*IDN
	OPERATION COMPLETE	*OPC
	OPTION IDENTIFICATION	*OPT
	RESET	*RST
	SERVICE REQUEST ENABLE	*SRE
	READ STATUS BYTE	*STB
	SELF-TEST	*TST
	WAIT-TO-CONTINUE	*WAI
DIGITIZER TOP-LEVEL	AUTOSCALE	AUT
	BLANK	BLAN
	CALIBRATION	*CAL
	DIGITIZE	DIG
	ERROR	ERR
	MULTIPLE CHANNEL TRACE ABORT	MCABT
	MULTIPLE CHANNEL TRIGGER ARM	MCARM
	MULTIPLE CHANNEL INFORMATION TRANSFER	MCINF
	MULTIPLE CHANNEL SETUP	MCSET
	MULTIPLE CHANNEL RANDOM EVENT CAPTURE	MCREC
	MULTIPLE CHANNEL SYNCHRONIZE	MCSNC
	MULTIPLE CHANNEL TRIGGER POINT TRANSFER	MCTRG
	RUN	RUN
	STOP	STOP
	STORE	STOR
	VIEW	VIEW
ACQUIRE	COUNT	COUN
	POINTS	POIN
	TYPE	TYPE
	AUTO	AUTO
CALIBRATION	ALL	ALL
	DATA	DATA
CHANNEL	COUPLING	COUP
	DETECTOR	DET
	EMITTER-COUPLED LOGIC	ECL

	INPUT	INP
	OFFSET	OFFS
	PROBES	PROB
	RANGE	RANG
	SPECTRUM ANALYZER	SA
	TRANSISTOR-TRANSISTOR LOGIC	TTL
	TYPE	TYPE
DISPLAY	CONNECT	CONN
	FORMAT	FORM
	GRATICULE	GRAT
	SCREEN	SCR
	STRING	STR
	TIME MARKERS	TMAR
	VOLTAGE MARKERS	VMAR
DOMAIN	TYPE	TYPE
FUNCTION	ADDITION	ADD
	INVERT	INV
	MULTIPLY	MULT
	OFFSET	OFFS
	ONLY	ONLY
	RANGE	RANG
	SUBTRACTION	SUBT
	VERSUS	VERS
MEASURE	ALL	ALL
	CURSOR	CURS
	DUTY CYCLE	DUT
	START MARKER EDGE NUMBER	ESTA
	STOP MARKER EDGE NUMBER	ESTO
	FALL TIME	FALL
	FREQUENCY	FREQ
	LOWER THRESHOLD	LOW
	MODE	MODE
	NEGATIVE PULSE WIDTH	NWID
	OVERSHOOT	OVER
	PERIOD	PER
	PRECISION	PREC
	PRESHOOT	PRES
	PRECEDING POINT OF REQUESTED TIME	PTIM
	POINT OF SPECIFIED VOLTAGE/ INTERSECTION	PVOL
	POSITIVE PULSE WIDTH	PWID
	RISE TIME	RISE
	SOURCE	SOUR
	TIME MARKER DELTA	TDEL
	TIME OF FIRST OCCURRENCE OF MAXIMUM VOLTAGE	TMAX
	TIME OF FIRST OCCURRENCE OF MINIMUM VOLTAGE	TMIN
	TIME OF SPECIFIED POINT	TPO

	TIME OF START MARKER	TSTA
	TIME OF STOP MARKER	TSTO
	TIME OF REC TRIGGER EVENT	TTIM
	TIME OF SPECIFIED VOLTAGE/ INTERSECTION	TVOL
	UNITS	UNIT
	UPPER THRESHOLD	UPP
	SIGNAL AMPLITUDE	VAMP
	SIGNAL AVERAGE	VAV
	WAVEFORM BASE VOLTAGE LEVEL	VBAS
	VOLTAGE MARKER DELTA	VDEL
	VOLTAGE MARKERS TO 50%	VFIF
	MAXIMUM VOLTAGE OF SIGNAL	VMAX
	MINIMUM VOLTAGE OF SIGNAL	VMIN
	VOLTAGE OF SPECIFIED POINT	VP0
	SIGNAL VOLTAGE PEAK-TO-PEAK	VPP
	RELATIVE VOLTAGE MARKER POSITIONING	VREL
	RMS VOLTAGE	VRMS
	POSITION OF VOLTAGE MARKER 1	VSTA
	POSITION OF VOLTAGE MARKER 2	VSTO
	VOLTAGE AT SPECIFIED TIME	VTIM
	TOP VOLTAGE LEVEL	VTOP
TIMEBASE	DELAY	DEL
	REC DELTA TIME STATUS	DELT
	EVENT	EVEN
	INTERPOLATION	INT
	MODE	MODE
	RANGE	RANG
	REFERENCE	REF
	SAMPLE	SAMP
	SWITCH	SWIT
	MAGNIFY SUBSYSTEM	
	DELAY	DEL
	MODE	MODE
	RANGE	RANG
TRIGGER	CENTER	CENT
	DELAY	DEL
	HOLDOFF	HOLD
	HYSTERESIS	HYST
	LEVEL	LEV
	QUALIFIER	QUAL
	SOURCE	SOUR
	TIMEOUT	TOUT
	AUTO .	AUTO
WAVEFORM	DATA	DATA
	FORMAT	FORM
	POINTS	POIN
	PREAMBLE	PRE
	SOURCE	SOUR

	TYPE	TYPE
	VALID	VAL
	X INCREMENT	XINC
	X ORIGIN	XOR
	X REFERENCE	XREF
	BOTTOM REFERENCE VALUE	YBOT
	Y INCREMENT	YINC
	Y ORIGIN	YOR
	Y REFERENCE	YREF
	TOP REFERENCE VALUE	YTOP
WINDOW	TYPE	TYPE

APPENDIX B
ALPHABETICAL MNEMONIC LISTING

MNEMONIC	DESCRIPTION	SUBSYSTEM
*CAL	CALIBRATION	DIGITIZER TOP-LEVEL
*CLS	CLEAR STATUS	COMMON COMMAND SET
*ESE	STANDARD EVENT STATUS ENABLE	COMMON COMMAND SET
*ESR	STANDARD EVENT STATUS REGISTER	COMMON COMMAND SET
*IDN	IDENTIFICATION	COMMON COMMAND SET
*OPC	OPERATION COMPLETE	COMMON COMMAND SET
*OPT	OPTION IDENTIFICATION	COMMON COMMAND SET
*RST	RESET	COMMON COMMAND SET
*SRE	SERVICE REQUEST ENABLE	COMMON COMMAND SET
*STB	READ STATUS BYTE	COMMON COMMAND SET
*TST	SELF-TEST	COMMON COMMAND SET
*WAI	WAIT-TO-CONTINUE	COMMON COMMAND SET

– A –

ADD	ADDITION	FUNCTION
ALL	ALL	CALIBRATION
ALL	ALL	MEASURE
AUT	AUTOSCALE	DIGITIZER TOP-LEVEL
AUTO	AUTO	ACQUIRE
AUTO	AUTO	TRIGGER

– B –

BLAN	BLANK	DIGITIZER TOP-LEVEL

– C –

CENT	CENTER	TRIGGER
CONN	CONNECT	DISPLAY
COUN	COUNT	ACQUIRE
COUP	COUPLING	CHANNEL
CURS	CURSOR	MEASURE

– D –

DATA	DATA	CALIBRATION
DATA	DATA	WAVEFORM
DEL	DELAY	MAGNIFY
DEL	DELAY	TIMEBASE
DEL	DELAY	TRIGGER
DELT	REC DELTA TIME STATUS	TIMEBASE
DET	DETECTOR	CHANNEL

DIG	DIGITIZE	DIGITIZER TOP-LEVEL
DUT	DUTY CYCLE	MEASURE

– E –

ECL	EMITTER-COUPLED LOGIC	CHANNEL
ERR	ERROR	DIGITIZER TOP-LEVEL
ESTA	START MARKER EDGE NUMBER	MEASURE
ESTO	STOP MARKER EDGE NUMBER	MEASURE
EVEN	EVENT	TIMEBASE

– F –

FALL	FALL TIME	MEASURE
FORM	FORMAT	DISPLAY
FORM	FORMAT	WAVEFORM
FREQ	FREQUENCY	MEASURE

– G –

GRAT	GRATICULE	DISPLAY

– H –

HOLD	HOLDOFF	TRIGGER
HYST	HYSTERESIS	TRIGGER

– I –

INP	INPUT	CHANNEL
INT	INTERPOLATION	TIMEBASE
INV	INVERT	FUNCTION

– L –

LEV	LEVEL	TRIGGER
LOW	LOWER THRESHOLD	MEASURE

– M –

MCABT	MULTIPLE CHANNEL TRACE ABORT	DIGITIZER TOP-LEVEL
MCARM	MULTIPLE CHANNEL TRIGGER ARM	DIGITIZER TOP-LEVEL
MCINF	MULTIPLE CHANNEL INFORMATION TRANSFER	DIGITIZER TOP-LEVEL
MCREC	MULTIPLE CHANNEL RANDOM EVENT CAPTURE	DIGITIZER TOP-LEVEL
MCSET	MULTIPLE CHANNEL SETUP	DIGITIZER TOP-LEVEL
MCSNC	MULTIPLE CHANNEL SYNCHRONIZE	DIGITIZER TOP-LEVEL
MCTRG	MULTIPLE CHANNEL TRIGGER POINT TRANSFER	DIGITIZER TOP-LEVEL
MODE	MODE	MEASURE
MODE	MODE	TIMEBASE

MODE	MODE	MAGNIFY
MULT	MULTIPLY	FUNCTION

– N –

NWID	NEGATIVE PULSE WIDTH	MEASURE

– O –

OFFS	OFFSET	CHANNEL
OFFS	OFFSET	FUNCTION
ONLY	ONLY	FUNCTION
OVER	OVERSHOOT	MEASURE

– P –

PER	PERIOD	MEASURE
POIN	POINTS	ACQUIRE
POIN	POINTS	WAVEFORM
PRE	PREAMBLE	WAVEFORM
PREC	PRECISION	MEASURE
PRES	PRESHOOT	MEASURE
PROB	PROBES	CHANNEL
PTIM	PRECEDING POINT OF REQUESTED TIME	MEASURE
PVOL	POINT OF SPECIFIED VOLTAGE/ INTERSECTION	MEASURE
PWID	POSITIVE PULSE WIDTH	MEASURE

– Q –

QUAL	QUALIFIER	TRIGGER

– R –

RANG	RANGE	CHANNEL
RANG	RANGE	FUNCTION
RANG	RANGE	MAGNIFY
RANG	RANGE	TIMEBASE
REF	REFERENCE	TIMEBASE
RISE	RISE TIME	MEASURE
RUN	RUN	DIGITIZER TOP-LEVEL

– S –

SA	SPECTRUM ANALYZER	CHANNEL
SAMP	SAMPLE	TIMEBASE
SCR	SCREEN	DISPLAY
SOUR	SOURCE	MEASURE
SOUR	SOURCE	TRIGGER
SOUR	SOURCE	WAVEFORM
STOP	STOP	DIGITIZER TOP-LEVEL
STOR	STORE	DIGITIZER TOP-LEVEL
STR	STRING	DISPLAY

SUBT	SUBTRACTION	FUNCTION
SWIT	SWITCH	TIMEBASE

– T –

TDEL	TIME MARKER DELTA	MEASURE
TMAR	TIME MARKERS	DISPLAY
TMAX	TIME OF FIRST OCCURRENCE OF MAXIMUM VOLTAGE	MEASURE
TMIN	TIME OF FIRST OCCURRENCE OF MINIMUM VOLTAGE	MEASURE
TPO	TIME OF SPECIFIED POINT	MEASURE
TSTA	TIME OF START MARKER	MEASURE
TSTO	TIME OF STOP MARKER	MEASURE
TTIM	TIME OF REC TRIGGER EVENT	MEASURE
TOUT	TIMEOUT	TRIGGER
TTL	TRANSISTOR-TRANSISTOR LOGIC	CHANNEL
TVOL	TIME OF SPECIFIED VOLTAGE/ INTERSECTION	MEASURE
TYPE	TYPE	ACQUIRE
TYPE	TYPE	CHANNEL
TYPE	TYPE	DOMAIN
TYPE	TYPE	WAVEFORM
TYPE	TYPE	WINDOW

– U –

UNIT	UNITS	MEASURE
UPP	UPPER THRESHOLD	MEASURE

– V –

VAL	VALID	WAVEFORM
VAMP	SIGNAL AMPLITUDE	MEASURE
VAV	SIGNAL AVERAGE	MEASURE
VBAS	WAVEFORM BASE VOLTAGE LEVEL	MEASURE
VDEL	VOLTAGE MARKER DELTA	MEASURE
VERS	VERSUS	FUNCTION
VFIF	VOLTAGE MARKERS TO 50%	MEASURE
VIEW	VIEW	DIGITIZER TOP-LEVEL
VMAR	VOLTAGE MARKERS	DISPLAY
VMAX	MAXIMUM VOLTAGE OF SIGNAL	MEASURE
VMIN	MINIMUM VOLTAGE OF SIGNAL	MEASURE
VPO	VOLTAGE OF SPECIFIED POINT	MEASURE
VPP	SIGNAL VOLTAGE PEAK-TO-PEAK	MEASURE
VREL	RELATIVE VOLTAGE MARKER POSITIONING	MEASURE
VRMS	RMS VOLTAGE	MEASURE
VSTA	POSITION OF VOLTAGE MARKER 1	MEASURE
VSTO	POSITION OF VOLTAGE MARKER 2	MEASURE
VTIM	VOLTAGE AT SPECIFIED TIME	MEASURE
VTOP	TOP VOLTAGE LEVEL	MEASURE

– X –

XINC	X INCREMENT	WAVEFORM
XOR	X ORIGIN	WAVEFORM
XREF	X REFERENCE	WAVEFORM

– Y –

YBOT	BOTTOM REFERENCE VALUE	WAVEFORM
YINC	Y INCREMENT	WAVEFORM
YOR	Y ORIGIN	WAVEFORM
YREF	Y REFERENCE	WAVEFORM
YTOP	TOP REFERENCE VALUE	WAVEFORM

APPENDIX C
ALPHABETICAL COMMAND DESCRIPTION LISTING

DESCRIPTION	MNEMONIC	SUBSYSTEM
– A –		
ADDITION	ADD	FUNCTION
ALL	ALL	CALIBRATION
ALL	ALL	MEASURE
AUTO	AUTO	ACQUIRE
AUTO	AUTO	TRIGGER
AUTOSCALE	AUT	DIGITIZER TOP-LEVEL
– B –		
BLANK	BLAN	DIGITIZER TOP-LEVEL
BOTTOM REFERENCE VALUE	YBOT	WAVEFORM
– C –		
CALIBRATION	*CAL	DIGITIZER TOP-LEVEL
CENTER	CENT	TRIGGER
CLEAR STATUS	*CLS	COMMON COMMAND SET
CONNECT	CONN	DISPLAY
COUNT	COUN	ACQUIRE
COUPLING	COUP	CHANNEL
CURSOR	CURS	MEASURE
– D –		
DATA	DATA	CALIBRATION
DATA	DATA	WAVEFORM
DELAY	DEL	MAGNIFY
DELAY	DEL	TIMEBASE
DELAY	DEL	TRIGGER
DETECTOR	DET	CHANNEL
DIGITIZE	DIG	DIGITIZER TOP-LEVEL
DUTY CYCLE	DUT	MEASURE
– E –		
EMITTER-COUPLED LOGIC	ECL	CHANNEL
ERROR	ERR	DIGITIZER TOP-LEVEL
EVENT	EVEN	TIMEBASE

– F –

FALL TIME	FALL	MEASURE
FORMAT	FORM	DISPLAY
FORMAT	FORM	WAVEFORM
FREQUENCY	FREQ	MEASURE

– G –

GRATICULE	GRAT	DISPLAY

- H –

HOLDOFF	HOLD	TRIGGER
HYSTERESIS	HYST	TRIGGER

– I –

IDENTIFICATION	*IDN	COMMON COMMAND SET
INPUT	INP	CHANNEL
INTERPOLATION	INT	TIMEBASE
INVERT	INV	FUNCTION

– L –

LEVEL	LEV	TRIGGER
LOWER THRESHOLD	LOW	MEASURE

– M –

MAXIMUM VOLTAGE OF SIGNAL	VMAX	MEASURE
MINIMUM VOLTAGE OF SIGNAL	VMIN	MEASURE
MODE	MODE	MAGNIFY
MODE	MODE	MEASURE
MODE	MODE	TIMEBASE
MULTIPLE CHANNEL INFORMATION TRANSFER	MCINF	DIGITIZER TOP-LEVEL
MULTIPLE CHANNEL RANDOM EVENT CAPTURE	MCREC	DIGITIZER TOP-LEVEL
MULTIPLE CHANNEL SETUP	MCSET	DIGITIZER TOP-LEVEL
MULTIPLE CHANNEL SYNCHRONIZE	MCSNC	DIGITIZER TOP-LEVEL
MULTIPLE CHANNEL TRACE ABORT	MCABT	DIGITIZER TOP-LEVEL
MULTIPLE CHANNEL TRIGGER ARM	MCARM	DIGITIZER TOP-LEVEL
MULTIPLE CHANNEL TRIGGER POINT TRANSFER	MCTRG	DIGITIZER TOP-LEVEL
MULTIPLY	MULT	FUNCTION

– N –

NEGATIVE PULSE WIDTH	NWID	MEASURE

– O –

OFFSET	OFFS	CHANNEL
OFFSET	OFFS	FUNCTION
ONLY	ONLY	FUNCTION
OPERATION COMPLETE	*OPC	COMMON COMMAND SET
OPTION IDENTIFICATION	*OPT	COMMON COMMAND SET
OVERSHOOT	OVER	MEASURE

– P –

PERIOD	PER	MEASURE
POINT OF SPECIFIED VOLTAGE/ INTERSECTION	PVOL	MEASURE
POINTS	POIN	ACQUIRE
POINTS	POIN	WAVEFORM
POSITION OF VOLTAGE MARKER 1	VSTA	MEASURE
POSITION OF VOLTAGE MARKER 2	VSTO	MEASURE
POSITIVE PULSE WIDTH	PWID	MEASURE
PREAMBLE	PRE	WAVEFORM
PRECEDING POINT OF REQUESTED TIME	PTIM	MEASURE
PRECISION	PREC	MEASURE
PRESHOOT	PRES	MEASURE
PROBES	PROB	CHANNEL

– Q –

QUALIFIER	QUAL	TRIGGER

– R –

RANGE	RANG	CHANNEL
RANGE	RANG	FUNCTION
RANGE	RANG	MAGNIFY
RANGE	RANG	TIMEBASE
READ STATUS BYTE	*STB	COMMON COMMAND SET
REC DELTA TIME STATUS	DELT	TIMEBASE
REFERENCE	REF	TIMEBASE
RELATIVE VOLTAGE MARKER POSITIONING	VREL	MEASURE
RESET	*RST	COMMON COMMAND SET
RISE TIME	RISE	MEASURE
RMS VOLTAGE	VRMS	MEASURE
RUN	RUN	DIGITIZER TOP-LEVEL

– S –

SAMPLE	SAMP	TIMEBASE
SCREEN	SCR	DISPLAY
SELF-TEST	*TST	COMMON COMMAND SET
SERVICE REQUEST ENABLE	*SRE	COMMON COMMAND SET
SIGNAL AMPLITUDE	VAMP	MEASURE
SIGNAL AVERAGE	VAV	MEASURE
SIGNAL VOLTAGE PEAK-TO-PEAK	VPP	MEASURE

SOURCE	SOUR	MEASURE
SOURCE	SOUR	TRIGGER
SOURCE	SOUR	WAVEFORM
SPECTRUM ANALYZER	SA	CHANNEL
STANDARD EVENT STATUS ENABLE	*ESE	COMMON COMMAND SET
STANDARD EVENT STATUS REGISTER	*ESR	COMMON COMMAND SET
START MARKER EDGE NUMBER	ESTA	MEASURE
STOP	STOP	DIGITIZER TOP-LEVEL
STOP MARKER EDGE NUMBER	ESTO	MEASURE
STORE	STOR	DIGITIZER TOP-LEVEL
STRING	STR	DISPLAY
SUBTRACTION	SUBT	FUNCTION
SWITCH	SWIT	TIMEBASE

– T –

TIME MARKER DELTA	TDEL	MEASURE
TIME MARKERS	TMAR	DISPLAY
TIME OF FIRST OCCURRENCE OF MAXIMUM VOLTAGE	TMAX	MEASURE
TIME OF FIRST OCCURRENCE OF MINIMUM VOLTAGE	TMIN	MEASURE
TIME OF REC TRIGGER EVENT	TTIM	MEASURE
TIME OF SPECIFIED POINT	TPO	MEASURE
TIME OF SPECIFIED VOLTAGE/ INTERSECTION	TVOL	MEASURE
TIME OF START MARKER	TSTA	MEASURE
TIME OF STOP MARKER	TSTO	MEASURE
TIMEOUT	TOUT	TRIGGER
TOP REFERENCE VALUE	YTOP	WAVEFORM
TOP VOLTAGE LEVEL	VTOP	MEASURE
TRANSISTOR-TRANSISTOR LOGIC	TTL	CHANNEL
TYPE	TYPE	ACQUIRE
TYPE	TYPE	CHANNEL
TYPE	TYPE	DOMAIN
TYPE	TYPE	WAVEFORM
TYPE	TYPE	WINDOW

– U –

UNITS	UNIT	MEASURE
UPPER THRESHOLD	UPP	MEASURE

– V –

VALID	VAL	WAVEFORM
VERSUS	VERS	FUNCTION
VIEW	VIEW	DIGITIZER TOP-LEVEL
VOLTAGE AT SPECIFIED TIME	VTIM	MEASURE
VOLTAGE MARKER DELTA	VDEL	MEASURE
VOLTAGE MARKERS	VMAR	DISPLAY
VOLTAGE MARKERS TO 50%	VFIF	MEASURE
VOLTAGE OF SPECIFIED POINT	VPO	MEASURE

– W –

WAIT-TO-CONTINUE	*WAI	COMMON COMMAND SET
WAVEFORM BASE VOLTAGE LEVEL	VBAS	MEASURE

– X –

X INCREMENT	XINC	WAVEFORM
X ORIGIN	XOR	WAVEFORM
X REFERENCE	XREF	WAVEFORM

– Y –

Y INCREMENT	YINC	WAVEFORM
Y ORIGIN	YOR	WAVEFORM
Y REFERENCE	YREF	WAVEFORM

APPENDIX D
ALPHABETICAL COMMAND SUMMARY

*CAL?	*CAL? Executes an internal self-calibration routine.	query
*CLS	*CLS Clears status data structures and the Request-for-OPC flag.	command
*ESE	*ESE Sets the Standard Event Status Enable Register bits.	command/query
*ESR?	*ESR? Determines the current contents of the Standard Event Status Register.	query
*IDN?	*IDN? Identifies devices over the system interface and is organized into four fields.	query
*OPC	*OPC Sets the request for the Operation complete flag.	command/query
*OPT?	*OPT? Identifies reportable device options over the system interface and is organized into five fields.	query
*RST	*RST Performs a device reset.	command
*SRE	*SRE Sets the Service Request Enable Register bits.	command/query
*STB?	*STB? Reads the Status Byte Register and the Master Summary Status bit.	query
*TST?	*TST? Executes an internal self-test routine.	query

***WAI** *WAI — command

Stops the device from executing any further commands or queries until the No-Operation-Pending flag, Power-On (PON), or Device Clear Active State Message (DCAS) is true.

– A –

ADD FUNC:ADD {CHAN <number>|WMEM <number>}, (CHAN <number>|WMEM <number>) — command

Sums the two defined operands together algebraically.

ALL? CAL:ALL? or CAL? — query

Invokes the internal calibration routine and returns a pass or fail code.

ALL MEAS:ALL — command/query

Makes as many measurements as possible on the defined source waveform.

AUT AUT — command

Performs the autoscale function which automatically selects the vertical sensitivity, vertical offset, trigger level, and sweep speed for a display of the input signal.

AUTO ACQ:POIN:AUTO {ON|OFF} — command/query

Sets the status of the Timebase and Trace Length coupling.

TRIG:TOUT:AUTO {ON|OFF|0|1} — command/query

Sets the auto-coupling status of the TIMEOUT command in the Trigger Subsystem and the RANGE command in the Timebase Subsystem.

– B–

BLAN BLAN {CHANX|FUNCX|WMEMX} — command

Turns off the display of the specified channel, function, or waveform memory.

– C –

CENT

TRIG:CENT — command

Sets the trigger level to the current mid-point of the display screen for internal triggering.

CONN

DISP:CONN {ON|OFF|1|0} — command/query

Turns the dots mode on or off.

COUN

ACQ:COUN <number> — command/query

Sets the number of averaged traces that may be selected.

COUP

CHANX:COUP {AC|DC|DCF} — command/query

Sets the signal coupling for the indicated channel to be ac coupled into 1 megohm, dc coupled innto 1 megohm, or dc coupled into 50 ohms.

CURS?

MEAS:CURS? {DELT|STAR|STOP} — query

Returns the specified markers as an ordered pair of time and voltage values. DELT specifies the difference between the two time markers and the two voltage markers. STAR specifies the positions of voltage marker 1 and the start marker. STOP specifies the positions of voltage marker 2 and the stop marker,

– D –

DATA

CAL:DATA <block data> — command/query

Provides a means to load calibration data.

DATA?

WAV:DATA? — query

Returns the data in the source trace.

DEL

TIM:DEL <delay> [xs] — command/query

Sets the timebase delay which is the time interval between the trigger event and the on-screen delay reference point.

TIM:MAGN:DEL <delay> [xs] — command/query

Sets the delay for a magnified trace and is always referenced to the center of the display screen.

TRIG:DEL <delay> [xs] — command/query

Sets the timebase delay which is the time interval between the trigger event and the on-screen delay reference point.

DELT TIM:DELT {ABS|REL} — command/query

Sets the delta time status for the Random Event Capture mode. ABS specifies that all times are referenced to the trigger event selected at the time the DELT command was executed. REL specifies that all times are referenced to the previous trigger event.

DET CHANX:DET {SAMP|POS|NEG|ALT} — command/query

Selects and sets the detector mode used by the indicated channel.

DIG DIG (CHANX) — command

Requests that waveform data from a specified channel be returned to the controller via HP-IB.

DUT MEAS:DUT — command/query

Measures the duty cycle of the signal.

– E –

ECL CHANX:ECL — command

Sets the voltage range, offset, and trigger level for the indicated channel to values appropriate for examining ECL signals. The voltage range is 1.6V, the offset is –1.0V, and the trigger level is –1.0V.

ERR? ERR? [NUM|STR} — query

Returns the next error in the Error Queue.

ESTA MEAS:ESTA <edge number> — command/query

Positions the start marker at a specified intersection of voltage marker 1 and the waveform.

ESTO MEAS:ESTO <edge number> — command/query

Positions the stop marker at a specified intersection of voltage marker 2 and the waveform.

EVEN TIM:EVEN <event number> — command/query

Specifies any trigger event that was saved in memory in the Random Event Capture mode. The digitizer must already be in the REC mode. (Refer to the SAMPLING command in the Timebase Subsystem.)

– F –

FALL MEAS:FALL — command/query
Measures the fall time of the first falling edge whose defined thresholds are on the waveform.

FORM DISP:FORM {1|2} — command/query
Sets the split-screen display format. 1 specifies the full-screen format; 2 specifies the split-screen format.

WAV:FORM (WORD} — command/query
Selects the data transmission mode for the waveform data to be WORD.

FREQ MEAS:FREQ — command/query
Measures the frequency of the first complete period of the waveform.

– G –

GRAT DISP:GRAT {OFF|GRID|AXES|FRAM} — command/query
Sets the display graticule with one of three sets of vertical and horizontal lines that format the display screen.

– H –

HOLD TRIG:HOLD <holdoff time> [xs] — command/query
Sets the value for trigger holdoff time.

HYST TRIG:HYST <percent> — command/query
Sets the value (in percent) for trigger hysteresis for internal triggering.

– I –

INP CHANX:INP{1|2} — command/query
Sets the input connector for the indicated channel to be either INPUT 1 or INPUT 2.

INT TIM: INT (OFF |BRIC|GAUS) — command/query
Specifies the interpolation method used.

INV FUNC: INV {CHAN<number>|WMEM<number>} — command
Allows the defined operand to be inverted.

– L –

LEV TRIG:LEV <level 1> [xV] [, <level 2> [xV]] **command/query**

Sets the value (in volts) of the trigger level for internal triggering. The second trigger level is used only when the trigger QUALIFIER command is specified to be RANGE.

LOW MEAS:LOW {<percent>|<volts> [xV]} **command/query**

Sets the lower measurement threshold in either percent or volt units. This point is used for the lower threshold for rise and fall time measurements when the MODE command of the Measure Subsystem is specified as USER.

– M –

MODE MEAS:MODE {STAN|USER} **command/query**

Determines how rise and fall time measurements are to be made. STAN specifies the upper and lower thresholds as 10% and 90%. USER specifies the upper and lower threshold values defined by the LOWER THRESHOLD, UPPER THRESHOLD, and UNITS commands.

TIM:MODE {ASIN|AUTO|FREE|FSIN|SING|TRIG} **command/query**

Specifies the timebase operating mode.
ASIN: auto-triggered, single sweep
AUTO: auto-triggered, continuous sweep
FREE: free-running, continuous sweep
FSIN: free-running, single sweep
SING: triggered single sweep
TRIG: triggered continuous sweep

MAGN TIM:MAGN:MODE{ON|OFF|0|1} **command/query**

Sets the on/off status of the magnify function.

MULT FUNC:MULT {CHAN<number>|WMEM<number>}, {CHAN<number>|WMEM<number>} **command**

Sets the current trace math function to MULTIPLY with the parameters indicating the two operands.

– N –

NWID MEAS:NWID — command/query

Measures the negative pulse width by determining the change in time from the first falling edge threshold to the next rising edge threshold.

– O –

OFFS CHANX:OFFS <offset> [xV] — command/query

Sets the level (amplitude reference) of the display midscreen in volts for the indicated channel,

FUNC:OFFS <offset> [xV] — command/query

Sets the current voltage offset of the function. Whenever a new operator or source is defined, the offset is set as follows:

ONLY	offset of operand
INVERT	-offset of operand
ADDITION	1st operand offset t 2nd operand offset
SUBTRACTION	1st operand offset t 2nd operand offset

ONLY FUNC:ONLY {CHAN<number>|WMEM<number>} — command

Allows the function to be defined as any available channel or waveform memory without any change.

OVER MEAS:OVER — command/query

Measures the overshoot of the defined waveform.

– P –

PER MEAS:PER — command/query

Measures the period of the waveform by determining the change in time between the first edge and the following "like" edge.

POIN ACQ:POIN <number of points> — command/query

Sets the number of points in a trace to determine the number of data points to be sampled and saved.

POIN? WAV:POIN? — query

Returns the number of data points in the source trace.

PRE? WAV:PRE? — query

Returns a description of the source trace. The data returned is:

FORM	FORMAT
TYPE	TYPE
POIN	POINTS
XINC	X INCREMENT
XOR	X ORIGIN
XREF	X REFERENCE
YINC	Y INCREMENT
YOR	Y ORIGIN
YREF	Y REFERENCE

PREC MEAS:PREC (COAR) — command/query

Specifies the precision of the measurements to be COARSE.

PRES MEAS:PRES — command/query

Measures the preshoot of the defined waveform.

PROB CHANX:PROB <multiplier> — command/query

Sets the value of the probe multiplier for the indicated channel.

PTIM? MEAS:PTIM? <time> [xs] — query

Returns the nearest point preceding the requested time.

PVOL MEAS:PVOL <volt> [xV], <edge number> — command/query

Sets voltage marker 1 to a defined voltage level and tries to find the edge intersection of voltage marker 1 and the waveform.

PWID MEAS:PWID — command/query

Measures the positive pulse width by determining the change in time frotn the first rising edge threshold to the next falling edge threshold.

– Q –

QUAL TRIG:QUAL {POS|NEG|EDGE|HIGH|LOW|RANG} — command/query

Specifies the trigger qualifier.

– R –

RANG

CHANX:RANG <range> [xV] — command/query
Sets the voltage range (amplitude) for the indicated channel,

FUNC:RANG <range> [xV] — command/query
Allows the full-scale vertical axis of a function's display to be defined. Whenever a new operator or source is defined, the range is set as follows:

ONLY	range of operand
INVERT	range of operand
ADDITION	1st operand range + 2nd operand range
SUBTRACTION	1st operand range + 2nd operand range

TIM:RANG <time 1> [xs] [, <time 2> [xs]] — command/query
Sets the time range. When the SAMPLING command specifies the MIXED operating mode, a second time range may be specified. (Refer to the SAMPLING command in the Timebase Subsystem.)

TIM:MAGN:RANG <range> [xs] — command/query
Sets the time range of the area to be magnified for the magnify function.

REF

TIM:REF {CENT|LEFT|RIGH} — command/query
Sets a trigger event delay reference with respect to the left, center, or right graticule edge of the display screen.

RISE

MEAS:RISE — command/query
Measures the rise time of the first rising edge whose defined thresholds are on the waveform.

RUN

RUN — command
Directs the instrument to acquire data for the active waveform -display.

– S –

SA	CHANX:SA Presets the voltage range, offset, trigger level, and input coupling to view the video output of a spectrum analyzer.	command
SAMP	TIM:SAMP {REAL|REP|MIX|REC} Sets the operating mode of the digitizer.	command/query
SCR	DISP:SCR {ON|OFF|1|0} Sets the display screen status.	command/query
SOUR	MEAS:SOUR {CHANX|WMEMX|FUNCX} Selects the source of data for subsequent operations in the Measure Subsystem.	command/query
	TRIG:SOUR {CHAN1|EXT1|LINE} Specifies the trigger source,	command/query
	WAV:SOUR {CHANX|WMEMX|FUNCX} Selects the source trace in subsequent operations in the Waveform Subsystem.	command/query
STOP	STOP Directs the instrument to stop acquiring data from the active waveform display.	command
STOR	STOR (CHANX,WMEMX) Directs the instrument to move the current waveform in a specified channel to the specified waveform memory,	command
STR	DISP:STR {"text"|'text'} Displays the input string parameter on the display screen.	command
SUBT	FUNC:SUBT {CHAN<number>|WMEM<number>}, {CHAN<number>|WMEM<number>} Sets the current trace math function to SUBTRACTION with the parameters indicating the two operands.	command
SWIT	TIM:SWIT <switch time> Sets the position at which the time range switches when operating the instrument in the MIXED timebase mode.	command/query

– T –

TDEL? MEAS:TDEL? — query
Returns the time difference between the start and stop time markers.

TMAR DISP:TMAR {ON|OFF|1|0} — command/query
Turns on and off the display of the time markers.

TMAX? MEAS:TMAX? — query
Returns the time at which the first occurrence of the current absolute maximum voltage occurred.

TMIN? MEAS:TMIN? — query
Returns the time at which the first occurrence of the current absolute minimum voltage occurred.

TPO? MEAS:TPO? <point number> — query
Returns the time of the specified point.

TSTA MEAS:TSTA <time> [xs] — command/query
Sets the position of the start time marker.

TSTO MEAS:TSTO <time> [xs] — command/query
Sets the position of the stop time marker.

TTIM? MEAS:TTIM? — query
Returns the time of the Random Event Capture trigger event with respect to the previous trigger event or a specific trigger event. (Refer to the DELT command in the Timebase Subsystem.)

TOUT TRIG:TOUT <time> [xs] — command/query
Sets the amount of time the instrument will wait for a trigger in AUTO trigger mode.

TTL CHANX:TTL — command
Sets the voltage range, offset, and trigger level for the indicated channel to values appropriate for examining TTL signals. The voltage range is 8V, the offset is 1.6V, and the trigger level is 1.6V.

TVOL MEAS:TVOL <volt> [xV], <edge number> — command/query
Sets voltage marker 1 to a defined voltage level and tries to find the edge intersection of voltage marker 1 and the waveform.

TYPE

ACQ:TYPE {NORM|AVER} — command/query

Allows selection of the type of acquisition that is to take place when a command is executed.

CHANX:TYPE {LOG|LIN} — command/query

Specifies trace data to be in either linear or logarithmic units and is only applicable in the SPECTRUM ANALYZER mode.

DOM:[CHANX|WMEMX|FUNCX:] [TYPE] {TIME|FREQ} — command/query

Sets the domain status of the specified trace to be in either the time or frequency domain.

TYPE?

WAV:TYPE? — query

Returns the data type for the currently specified source trace.

WIND:[CHANX|WMEMX|FUNCX:] [TYPE] {UNIF|HANN|FLAT} — command/query

Selects the window to be used on the specified source trace when displaying data in the frequency domain.

- U -

UNIT

MEAS:UNIT {PERC|VOLT} — command/query

Sets the measurement threshold units for marker positioning when a measurement is made.

UPP

MEAS:UPP {<percent>|<volts> [xV]} — command/query

Sets the upper measurement threshold in either percent or volt units. This point is used for the upper threshold for rise and fall time measurements when the MODE command of the Measure Subsystem is specified as USER.

– V –

VAL?

WAV:VAL? — query

Returns a 0 if the source trace does not have valid data. A response of 1 is returned for valid data in the previously selected source trace.

VAMP

MEAS:VAMP — command/query

Measures the signal amplitude in volts.

VAV

MEAS:VAV — command/query

Measures the average voltage of the waveform.

VBAS	MEAS:VBAS Measures the voltage level of the base of the waveform.	command/query
VDEL?	MEAS:VDEL? Returns the difference in voltage between voltage markers 1 and 2.	query
VERS	FUNC:VERS {CHAN<number>|WMEM<number>}, {CHAN<number>|WMEM<number>} Allows the two defined operands to be plotted with respect to each other on the X and Y axes. The first operand defines the X axis and the second operand defines the Y axis.	command
VFIF	MEAS:VFIF Sets the voltage markers at the 50% level.	command
VIEW	VIEW {CHANX|FUNCX|WMEMX} Turns on the specified channel, function, or waveform memory.	command
VMAR	DISP:VMAR {ON|OFF|1|0} Turns on and off the display of the voltage markers.	command/query
VMAX	MEAS:VMAX Measures the absolute maximum voltage of the waveform.	command/query
VMIN	MEAS:VMIN Measures the absolute minimum voltage of the waveform.	command/query
VPO?	MEAS:VPO? <point> Returns the value (in volts) of the specified point.	query
VPP	MEAS:VPP Measures the peak-to-peak voltage of the waveform.	command/query
VREL	MEAS:VREL <percent> Sets the voltage marker positions as a function of the last established values.	command/query
VRMS	MEAS:VRMS Measures the rms voltage of one complete period of the waveform.	command/query
VSTA	MEAS:VSTA <volts> [xV] Sets the position of voltage marker 1.	command/query

VSTO MEAS:VSTO <volts> [xV] — command/query

Sets the position of voltage marker 2.

VTIM? MEAS:VTIM? <time> [xs] — query

Returns the value (in volts) of the waveform at the specified time.

VTOP MEAS:VTOP — command/query

Measures the voltage level of the TOP of the waveform.

– X –

XINC? WAV:XINC? — query

Returns the x-increment value currently in the preamble. This value is the time difference between adjacent data points in the source trace.

XOR? WAV:XOR? — query

Returns the x-origin value currently in the preamble. This value is the time of the first point in the source trace.

XREF? WAV:XREF? — query

Returns the x-reference value currently in the preamble. This value specifies the data point that is associated with the x-origin data values and is always 0.

– Y –

YBOT WAV:YBOT <value> — command/query

Sets the bottom reference value.

YINC? WAV:YINC? — query

Returns the y-increment value currently in the preamble. This value is the voltage difference between adjacent data values.

YOR? WAV:YOR? — query

Returns the y-origin value currently in the preamble. This value is the voltage of the trace at center screen.

YREF? WAV:YREF? — query

Returns the y-reference value currently in the preamble. This value specifies the data point where the y origin occurs.

YTOP WAV:YTOP <value> — command/query

Sets the top reference value.

APPENDIX E
HP-IB REVIEW

BUS DESCRIPTION

The Hewlett-Packard Interface Bus (HP-LB) is HP's implementation of the IEEE-488 communication interface. It is used by a variety of instruments, disc drives, and peripherals manufactured by Hewlett-Packard and other companies. The HP-IB is a 16-line bus that connects up to 15 devices in parallel on a communication link. Figure E-l illustrates the HP-LB connector.

Of the sixteen signal lines, eight are data lines, three are for handshake purposes, and the remaining five are control lines. Information is transferred across the eight data lines in a bit-parallel, byte-serial fashion. Briefly, the eight control and handshake lines are used as follows:

ATN (Attention) is used primarily to differentiate between Command mode and Data mode. When ATN is true, information on the data lines is interpreted as a bus command; when ATN is false, the information is treated as a data byte.

EOI (End or Identify) has two uses. In the first, EOI is asserted on the last byte of a data transfer. This signals all listening devices that no more data should be expected on the transfer. The second use is in combination with ATN and is used in performing a parallel poll.

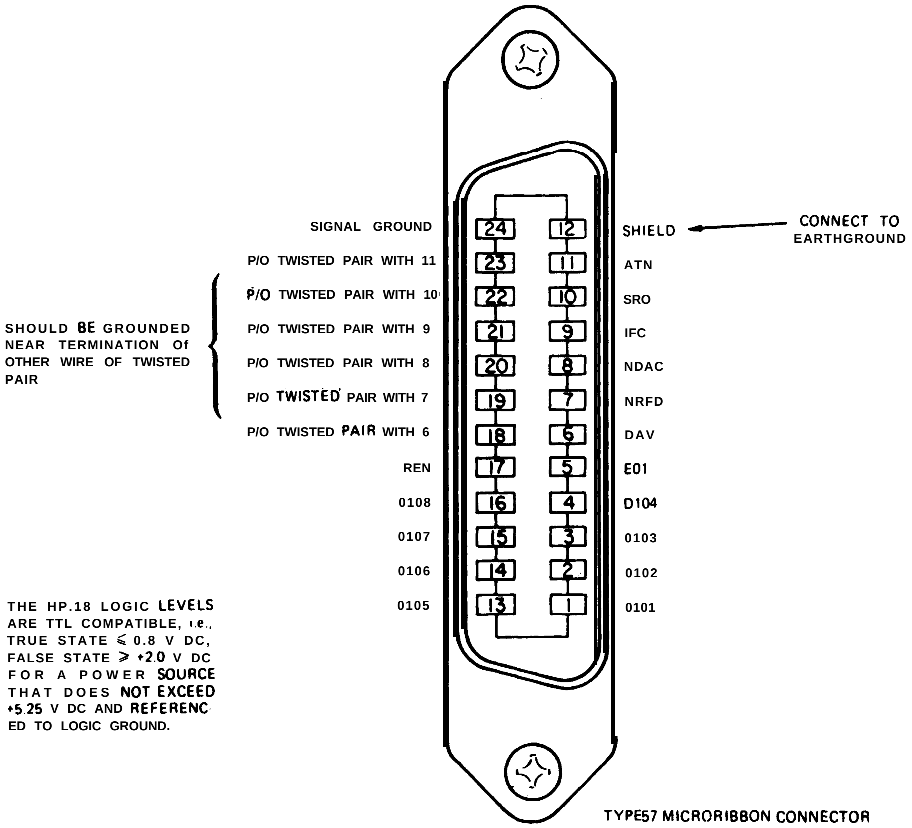


Figure E- 1. HP-IB Connector

IFC (Interface Clear) is under the exclusive control of the system controller. When it is pulsed true, all device interfaces are returned to an idle state and the state of the bus is cleared.

REN (Remote Enable) may be set by the system controller to permit devices to operate in Remote mode; that is, to operate under programmed HP-IB control instead of via the device's front panel.

SRQ (Service Request) can be set by a device on the interface to indicate that it is in need of service. Examples where SRQ might be set are: completion of a task such as taking a measurement, an error detected during device operation, or a request to be active controller.

DAV (Data Valid) this handshake line indicates that the active talker has placed data on the data lines (DIO1 – D108).

NRFD (Not Ready for Data) this handshake line indicates that one or more active listeners is not ready for more data, and the active talker should wait before sending new data on the bus.

NDAC (Not Data Accepted) this handshake line indicates that one or more active listeners has not accepted the current data byte, and the active talker should leave the current byte asserted on the data lines.

Figure E-2 illustrates the HP-IB structure.

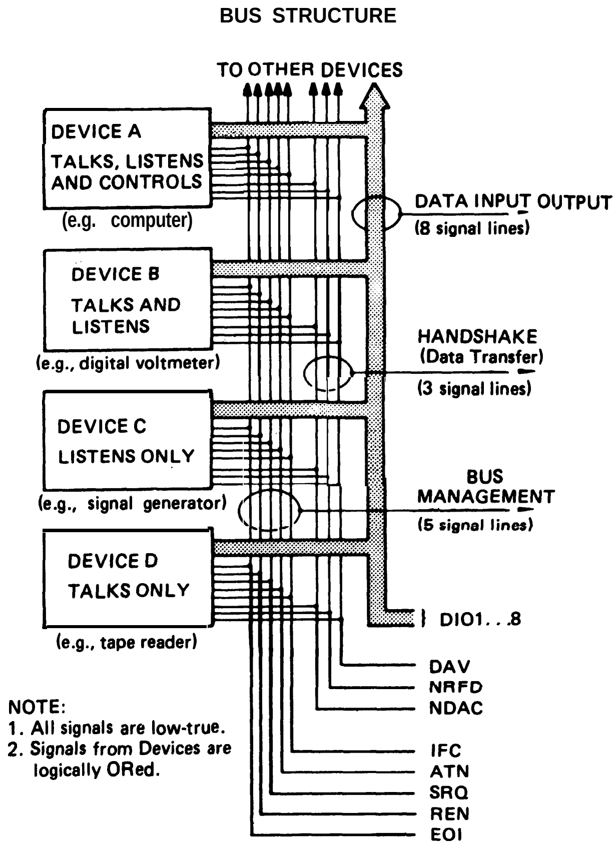


Figure E-2. HP-IB Structure

COMMANDS AND DATA

There are two modes of communication on an HP-IB: Command mode and Data mode.

In Command mode, information transmitted across the eight data lines is interpreted as talk or listen addresses, or universal, address or unaddress commands (explained later). In this mode, only seven of the data lines are used. Some devices use the eighth line as a parity check for certain protocols.

In Data mode, any eight-bit value can be transmitted. The HP-IB can therefore **be used for transmission of** binary data as well as ASCII characters.

The three-line handshake scheme has several advantages. First, data transfer is asynchronous: the data rate is limited **only** by the speed of the devices actively involved in the transfer. A second, related advantage is that devices with different I/O speeds can be interconnected without need for other synchronization mechanisms. Also, multiple devices can be addressed concurrently.

CONTROLLERS, TALKERS, AND LISTENERS

Controller

To understand communication among devices, you should be familiar with the concepts of controller, talker, and listener.

There are two types of controller within an HP-IB system: **system controller and active controller.**

There must be a single “system controller” capable **of taking control of the interface at any time. The** system controller has exclusive control over the IFC and REN lines.

Each system also has one or more devices capable of being **“active controller” (sometimes referred to as** controller-in-charge), although there may only be one active controller at **any given time. The active** controller has the ability to establish listeners and talkers, send bus commands, **perform serial polls, etc.**

In most systems, a single computer will be both the system controller and the only active controller. Some non-system controller devices may request service, indicating their desire to be active controller in order to perform some operation such as plotting data or directly accessing disc drives. The current active controller “passes control” to a requesting device to make it the active controller. In other systems, a system controller may not be capable of operating as non-active controller, and therefore no pass-control capabilities will exist. Note that system controller capabilities may not be transferred.

Talker

In each system, there can be at most one device addressed as talker at any given time. A device becomes addressed as talker by receiving its talk address from the active controller. Each device on the bus must have a unique bus address, usually set at the manufacturing site or settable by switches on the instrument.

The addresses are in the range 0 to 30. A talk address is formed by adding the primary bus address to the talk address base of 64 and transmitting **that** value across the data lines while ATN is asserted. For example, talk address 9 would be formed by taking 64 + 9 = 73, asserting ATN and transmitting a byte whose value is 73 (ASCII character “I”).

Listener

Listen addresses are formed in a similar manner to talk addresses, except that listen addresses use a base of 32 (e.g., listen address 9 is formed as 32 + 9 = 41 = ASCII character ")") transmitted with ATN true,

Multiple devices may be addressed to listen at any time and data bytes will be received by all listeners in parallel. However, most devices cannot be addressed to both talk and listen at the same time. (Refer to Table E-3 for talk and listen address codes.)

Extended Addressing

The descriptions of talk address and listen address refer to a device's primary address. Some devices also have extended talker or extended listener capabilities, sometimes referred to as secondary addresses, With extended addressing, talk and listen addresses are represented by two command bytes. The first byte is the primary talk or listen address as described above. The second byte is a secondary address command.

Secondary addresses may be in the range 0 to 31. The secondary commands transmitted are formed by adding the secondary address to the base of 96 and transmitting the byte with ATN true.

Extended addresses can be used, for example, to access a specific I/O card within an instrument that allows multiple I/O cards.

BUS COMMANDS

There are five types of information transmitted when the bus is operating in Command mode (i.e., when ATN is asserted):

a. talk address
b. listen address
c. universal commands
d. addressed commands
e. unaddress commands

Talk address and listen address are discussed above.

Universal Commands

Universal commands are received by all responding devices on the bus whether addressed to listen or not. Table E-l lists the universal commands.

Addressed Commands

Addressed commands are executed only by those devices that are currently addressed as listeners. They allow the controller to initiate a simultaneous action by a selected group of devices on the bus, such as triggering them to take readings at the same time. Table E-2 lists the addressed commands.

Table E- 1. Universal Commands

MNEMONIC	COMMAND	DESCRIPTION
LLO	Local Lockout	Disables the front panel of the responding device. The REN line must be asserted in order for LLO to have any effect. If the instrument is already in Remote mode, the lockout will be immediate. Otherwise, the lockout will commence when the device receives its listen address.
DCL	Universal Device Clear	All devices capable of responding are returned to some known, device-dependent state. In some cases a device will perform a self-test in response to a Universal Device Clear.
PPU	Parallel Poll Unconfigure	Directs all devices on the HP-IB that have parallel poll configure capabilities to not respond to a parallel poll.
SPE	Serial Poll Enable	Enables Serial Poll mode on the interface.
SPD	Serial Poll Disable	Disables Serial Poll mode on the interface.

Table E-2. Addressed Commands

MNEMONIC	COMMAND	DESCRIPTION
SDC	Selected Device Clear	Similar to a Universal Device Clear (DCL) with only those devices addressed to listen responding.
GTL	Go To Local	Returns devices that are addressed to listen to Local mode (i.e. re-enables front panel programming). REN stays asserted when a GTL is sent, and devices will be returned to Remote upon receipt of their listen address.
GET	Group Execute Trigger	Initiates some pre-programmed action by responding devices. This may be used to simultaneously start action in a group of devices that are addressed to listen.
PPC	Parallel Poll Configure	Configures a device to respond to a parallel poll on a specified data line with either a positive or negative signal. A secondary command sent after PPC contains the data that configures the device.
TCT	Take Control	Transfers active controller status to another device on the bus. (Note: This command is not permitted in the HP-IB Command Library for MS-DOS)

Unaddress Commands

The two unaddress commands can be considered as extensions of talk and listen addresses. The first, UNL (Unlisten), causes all devices on the bus (except those that have a built-in switch set to Listen Only) to stop being listeners.

Similarly, UNT (Untalk), directs any device on the interface to no longer be addressed as talker. Since there may only be one device addressed to talk at any time, receipt of another device's talk address is equivalent to receiving an UNT. UNL and UNT are logically equivalent to listen address 31 and talk address 31, respectively.

SERVICE REQUESTS

Some devices that operate on the interface have the ability to request service from the system controller. A device may request service when it has completed a measurement, when it has detected a critical condition, or under many other circumstances.

A service request (SRQ) is initiated when the **device sets that SRQ line true. The controller, sensing SRQ** has been set (typically either by polling **the status of the line or by enabling an SRQ interrupt), can poll** devices in one of two ways: serial poll or parallel poll.

Serial Poll

A typical sequence of events in performing a serial poll is to establish a device as a talker, send SPE to set up Serial Poll mode, wait for the addressed device to send its serial poll response byte, and then send an SPD and UNT to disable the Serial Poll mode.

The meaning of the serial poll response byte depends upon the individual device. However, if bit 6 of the response byte (bit value 64) is **1,** the device is indicating it has requested service. If bit 6 is 0, the polled device was not the one that requested service. Individual device manuals provide additional information on the meanings of serial poll response bytes.

Parallel Poll

Parallel polling permits the status of multiple devices on the HP-IB to be checked simultaneously. Each device is assigned a data line **(DIO1** through D108) that is set true by the device during the parallel poll routine if it requires service.

More than one device can be assigned to a particular data line. If a shared line is sensed true, a serial poll can typically be performed to determine which device requested service. A parallel poll is started when the controller asserts ATN and EOI together. After a short period of time, the controller reads the poll byte and begins its interpretation thereof.

Some devices can be configured (by the PPC command) to respond on specific data lines. Other devices may respond on lines selected by switches or jumpers in the devices. Some devices do not have parallel poll capability.

SYSTEM CONFIGURATION

HP-IB systems can be configured in three ways:

No Controller This mode of data transfer is limited to a direct transfer between one device manually set to talk only, and one or more devices manually set to Listen Only.

Single Controller In this configuration, data transfer can be:

- from controller to device(s) (Command or Data mode).
- from device to controller (Data mode only).
- from a device to other device(s) (Data mode).

Multiple Controllers This mode of data transfer is similar to that of a single controller, with the requirement that active controller status be passable from one controller to another. In this configuration, one controller must be designated as the system controller. This controller is the only one that can control the IFC and REN lines.

Control is passed to another controller by addressing it as a talker, and commanding it to "take control" (TCT).

Note that the HP-IB Command Library does not support more than one controller. Therefore, control may not be passed.

Table E-3. ASCII Codes

B7 B6 B5 BITS / B4 B3 B2 B1	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
	CONTROL		NUMBERS SYMBOLS		UPPER CASE		LOWER CASE	
0 0 0 0	0 NUL 0 0	20 DLE 10 16	40 L0 SP 20 32	60 L16 0 30 48	100 T0 @ 40 84	120 T16 P 50 80	140 \ 60 96	160 p 70 112
0 0 0 1	1 GTL SOH 1 1	21 LLO DC1 11 17	41 L1 ! 21 33	61 L17 1 31 49	101 T1 A 41 85	121 T17 Q 51 81	141 a 61 97	161 q 71 113
0 0 1 0	2 STX 2 2	22 DC2 12 18	42 L2 " 22 34	82 L18 2 32 50	102 T2 B 42 88	122 T18 R 52 82	142 b 62 98	182 r 72 114
0 0 1 1	3 ETX 3 3	23 DC3 13 19	43 L3 # 23 35	83 L19 3 33 51	103 T3 C 43 87	123 T19 S 53 83	143 c 63 99	183 s 73 115
0 1 0 0	4 SDC EOT 4 4	24 DCL DC4 14 20	44 L4 $ 24 38	84 L20 4 34 52	104 T4 D 44 88	124 T20 T 54 84	144 d 64 100	184 t 74 118
0 1 0 1	5 PPC ENQ 5 5	25 PPU NAK 15 21	45 L5 % 25 37	85 L21 5 35 53	105 T5 E 45 89	125 T21 U 55 85	145 e 65 101	185 u 75 117
0 1 1 0	6 ACK 6 6	26 SYN 16 22	46 L8 & 26 38	88 L22 6 38 54	108 T8 F 48 70	128 T22 V 58 88	146 f 66 102	166 v 76 118
0 1 1 1	7 BEL 7 7	27 ETB 17 23	47 L7 ' 27 39	67 L23 7 37 55	107 T7 G 47 71	127 T23 W 57 87	147 g 67 103	167 w 77 119
1 0 0 0	10 GET BS 8 8	30 SPE CAN 18 24	50 L8 (28 40	70 L24 8 38 56	110 T8 H 48 72	130 T24 x 58 88	150 h 68 104	170 x 78 120
1 0 0 1	11 TCT HT 9 9	31 SPD EM 19 25	51 L9) 29 41	71 L25 9 39 57	111 T9 I 49 73	131 T25 Y 59 89	151 i 69 105	171 y 79 121
1 0 1 0	12 LF A 10	32 SUB 1A 28	52 L10 ★ 2A 42	72 L26 : 3A 58	112 T10 J 4A 74	132 T28 Z 5A 90	152 j 6A 108	172 z 7A 122
1 0 1 1	13 VT B 11	33 ESC 1B 27	53 L11 + 2B 43	73 L27 ; 3B 59	113 T11 K 4B 75	133 T27 [5B 91	153 k 6B 107	173 { 7B 123
1 1 0 0	14 FF C 12	34 FS 1C 28	54 L12 , 2C 44	74 L28 < 3C 60	114 T12 L 4c 78	134 T28 \ 5C 92	154 l 6C 108	174 ¦ 7C 124
1 1 0 1	15 CR D 13	35 GS 1D 29	55 L13 – 2D 45	75 L29 = 3D 61	115 T13 M 4D 77	135 T29 1 5D 93	155 m 6D 109	175 } 7D 125
1 1 1 0	16 SO E 14	36 RS 1E 30	56 L14 . 2E 46	76 L30 > 3E 62	116 T14 N 4E 78	138 T30 A 5E 94	156 n 6E 110	176 ~ 7E 126
1 1 1 1	17 SI F 15	37 US 1F 31	57 L15 / 2F 47	77 UNL ? 3F 63	117 T15 0 4F 79	137 UNT — 5F 95	157 o 6F 111	177 RUBOUT (DEL) 7F 127
	ADDRESSED COMMANDS	UNIVERSAL COMMANDS	LISTEN ADDRESSES		TALK ADDRESSES		SECONDARY ADDRESSES OR COMMANDS	

KEY

Octal 25	PPU	Message
	NAK	ASCII/ISO character
hex 15	21 decimal	

INDEX

– Q –

– R –

– S –

– T –

HEWLETT-PACKARD SALES AND SERVICE OFFICES

To obtain servicing information or to order replacement parts, contact the nearest Hewlett-Packard Sales and Service Office listed in the HP Catalog, or contact the nearest regional office listed below:

IN THE UNITED STATES

CALIFORNIA
P.O. Box 4230
1421 South Manhattan Avenue
Fullerton 92631

GEORGIA
P.O. Box 105005
2000 South Park Place
Atlanta 30339

ILLINOIS
5201 Tollview Drive
Rolling Meadows 60008

NEW JERSEY
W. 120 Century Road
Paramus 07652

IN CANADA

Hewlett-Packard (Canada) Ltd.
17500 South Service Road
Trans-Canada Highway
Kirkland, Quebec H9J 2M5

IN FRANCE

Hewlett-Packard France
F-91947 Les Ulis Cedex
Orsay

IN GERMAN FEDERAL REPUBLIC

Hewlett-Packard GmbH
Vertriebszentrale Frankfurt
Berner Strasse 117
Postfach 560 140
D-6000 Frankfurt 56

IN GREAT BRITAIN

Hewlett-Packard Ltd.
King Street Lane
Winnersh, Wokingham
Berkshire RG1 1 5AR

IN OTHER EUROPEAN COUNTRIES

SWITZERLAND
Hewlett-Packard (Schweiz) AG
29 Chemin Chateau Bloc
CH-1219 LeLignon-Geneva

IN ALL OTHER LOCATIONS

Hewlett-Packard Inter-Americas
3200 Hillview Avenue
Palo Alto, California 94304

HEWLETT
PACKARD

HP MANUAL PART NUMBER 70700-90021

PRINTED IN U.S.A.