Scans
By
Outsource-Options
(NR1DX)

Outsource-Options
Suite 2M
187 Depot Road,
Hollis, NH 03049

nr1dx@arrl.net or Henderson@outsource-options.com

"High resolution scans of obsolete technical manuals"

If you purchased this manual from us (typically through our Ebay name of NR1DX) thank you very much. If you received this from a well-meaning "friend" for free we would appreciate your treating this much like you would "share ware". By that we mean a donation of at least $5-10 per manual is appreciated in recognition of the time, energy and quality of effort that went into preserving this manual. Donations via PayPal go to: nr1dx@arrl.net or can be mailed to us the address above. This is not a hobby for us but a semi-retirement business used to augment our diminished income in the face of the current degraded world economy.

Typically the scans in our manuals are done as follows;
1) Typed text pages are typically scanned in black and white at 300 dpi.
2) Photo pages are typically scanned in gray scale mode at 600 dpi
3) Schematic diagram pages are typically scanned in black and white at 600 dpi unless the original manual had colored high lighting (as is the case for some 70's vintage Tektronix manuals).

Original pages, which exceed the 8.5 x 11 format, are scanned with enough overlap so that they can be printed and pieced. This is done for primarily for two reasons, the time involved in post processing these to their original size is not trivial and most people do not have the ability to print 11 X 17 pages anyway.

If your looking for a quality scanned technical manual in PDF format please drop us a line at nr1dx@arrl.net. and we will be happy to email you a current list of the manuals we have available. If you don't see the manual you need on the list drop us a line anyway we may still be able to point you to other sources. If you have an existing manual you would like scanned please write for details, This can often be done very reasonably in consideration for adding your manual to our library.

Thanks

Dave Henderson
Outsource-Options

Programming Reference

HP 54111D Digitizing Oscilloscope

HEWLETT PACKARD

Product Warranty

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

For warranty service or repair, this product must be returned to a service facility designated by Hewlett-Packard. However, warranty service for products installed by Hewlett-Packard and certain other products designated by Hewlett-Packard will be performed at the Buyer's facility at no charge within the Hewlett-Packard service travel area. Outside Hewlett-Packard service travel areas, warranty service will be performed at the Buyer's facility only upon HP's prior agreement and the Buyer shall pay Hewlett-Packard's round trip travel expenses.

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

Limitation of Warranty

The foregoing warranty shall not apply to defects resulting from improper or inadequate maintenance by the Buyer, Buyer-supplied software of 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, HEWLETT-PACKARD SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Exclusive Remedies

THE REMEDIES PROVIDED HEREIN ARE THE BUYER'S SOLE AND EXCLUSIVE REMEDIES. HEWLETT-PACKARD 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.

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.

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

Table of Contents

Table of Contents (Continued)

Chapter 3:   Acquire Subsystem

Table of Contents (Continued)

Chapter 4:   **Channel Subsystem**

Chapter 5:   **Display Subsystem**

Table of Contents (Continued)

Chapter 6:   **Function Subsystem**

Table of Contents (Continued)

Table of Contents (Continued)

Chapter 9:

Timebase Subsystem

Chapter 10:

Trigger Subsystem

Table of Contents (Continued)

Chapter 11:   **Waveform Subsystem**

Table of Contents (Continued)

Chapter 12:   **Remote Operation**

Appendix A: **HP-IB Data Transfer Tips**

Table of Contents (Continued)

Appendix B: **Program Examples**

Appendix C: **Format Statement Descriptions**

1 Introduction To Programming

Introduction

With the exception of the line switch, all the front panel controls as well as some instrument features that are remote only operations can be controlled by sending the appropriate commands over the HP-IB. Prior to operating this instrument over HP-IB review the *"Setting Up the Oscilloscope"* booklet for power and HP-IB connection information.

HP-IB Addressing

Before you get started programming your instrument, the instrument's HP-IB address must be properly set. This is done by selecting the Utility Menu from the front panel of the instrument. Once the Utility Menu is selected, select the HP-IB function menu. You can now set the HP-IB mode to Talk Only, Listen Only, or to Talk/Listen by pressing the function key. When you select the Talk/Listen mode the Address selection is available. The HP-IB address can be changed by rotating the knob, pressing the increment/decrement keys, or entering the desired address using the numeric key pad. For more information on setting the HP-IB address refer to the Utility Menu in the *Front Panel Reference Manual*.

Programming Manual Organization

This manual is organized into several chapters. This first chapter contains some general information and instructions to help you get started, as well as some specific programming rules and data formats for this instrument. This chapter also contains an alphabetic command to subsystem table and a command set organization chart. You should review this chapter very carefully to become familiar with the data formats and abbreviations used throughout this manual.

Some of the major topics in this chapter are:
- Command Structure
- Command Abbreviations
- Notation Conventions and Definitions
- Program Code Parameters
- Data Messages
- Default Settings
- Instrument Status
- Service Request
- Program Examples
- Command Set Organization

The next ten chapters contain the actual instruction set for the HP 54111D. The instruction set has been broken down into ten chapters because the instrument is logically organized into system commands and nine sets of subsystem commands. Each of the chapters contain a tab with the subsystem name for easy access to the commands.

The final chapter contains more information about operating over the HP-IB, and information about special operations.

Command Order Considerations

Commands are interpreted and setups are changed in the HP 54111D as they are received and found to be syntactically correct. If an error is found in a multi-command message, all commands are executed up to the point where the error is detected. This provides consistent operation whether commands are sent one per message or several per message.

When a query is executed the reply is placed in the output buffer of the HP 54111D. Multiple queries on one line result in the last reply overwriting the previous replies. The exception to this is when multiple parameter measurement queries are sent on one command line. In this case the replies to the measurement queries are buffered in the order that the queries occurred in the command line.

Command Structure

The HP 54111D programming commands are divided into three types: system commands, subsystem selectors, and subsystem commands. A programming command tree is shown in figure 1-1, and a programming command cross-reference is shown in table 1-7.

System Commands

The system commands control general oscilloscope functions and may be issued at any time. They do not change the subsystem selection.

Subsystem Selectors

The subsystem selectors are special system commands that address portions of the instrument. Only one subsystem may be selected at any given time. At power on, no subsystem is selected.

The nine subsystems are:

Acquire: allows the parameters for acquiring and storing data to be set.

Channel: controls all Y-axis oscilloscope functions, and channel input coupling.

Display: controls how waveforms, voltage and time markers, graticules, and text are displayed and written on the screen.

Function: controls the waveform math functions of the oscilloscope.

Hardcopy: controls the parameters used during the plotting or printing of waveforms.

Measure: selects the automatic measurements to be made.

Timebase: controls all X-axis oscilloscope functions.

Trigger: controls the trigger modes and parameters for each trigger mode.

Waveform: provides access to waveform data, including active data from channels and functions as well as static data from waveform memories.

Subsystem Commands

The subsystem commands may be issued once a subsystem has been selected. For example, once the timebase subsystem has been selected, any timebase command may be issued.

Command Abbreviations

Every command and alpha parameter has at least two forms, a short form and a long form. In some cases they will be the same. The short form is obtained by using the following rule:

> If the long form has four characters or less,
> then the long form and short form are the same.
> If the long form has more than 4 characters,
>> then if the 4th character is a vowel or is the same as the 3rd character,
>> then truncate to 3 characters,
>> else truncate to 4 characters.

> EXAMPLE: LONGFORM abbreviates to LONG.
>> SERIAL abbreviates to SER.
>> YOFFSET abbreviates to YOF.

In the case where two short forms would be identical, one of them will be changed slightly to differentiate between the two. In the command descriptions that follow, each command is given in both long and short forms.

As an example, COLOR and COLUMN would be abbreviated to COL using the above rule. In this case the COLOR command short form has been changed to COLO.

Some commands also have industry standard forms and these have been included in the instruction set. This means that some commands will have three forms. The examples below show three forms of the command. The first form is the long form, the second form is the short form, and the third form is an industry form. Each of the three forms work the same way, therefore it is left to you to use the form you like best.

> READ | READ | RDY
> REQUEST | REQ | RQS
> RESET | RES | RST

Notation Conventions and Definitions

The following conventions are used in this manual in descriptions of remote (HP-IB) operation:

< >   Angular brackets enclose words or characters that are used to symbolize a program code parameter or an HP-IB command.

::=   "is defined as." For example, <A> ::= <B> indicates that <A> can be replaced by <B> in any statement containing <A>.

| "or": Indicates a choice of one element from a list. For example, <A>|<B> indicates <A> or <B> but not both.

...   An ellipsis (trailing dots) is used to indicate that the preceding element may be repeated one or more times.

[ ]   Square brackets indicate that the enclosed items are optional.

{ }   When several items are enclosed by braces, one and only one of these elements must be selected.

The following definitions are used:

d ::=A single ASCII numeric character, 0-9.

n ::=A single ASCII non-zero, numeric character, 1-9.

<LF>::= ASCII line-feed (decimal 10).

<CR>::= ASCII carriage return (decimal 13).

<sp>::=ASCII space (decimal 32).

Syntax Diagrams

At the beginning of the system command section and each subsystem command section are syntax diagrams showing the proper syntax for each command. All characters contained in a circle or oblong are literals, and must be entered exactly as shown. Words and phrases contained in rectangles are names of items used with the command and are described in the accompanying text of each command. Each line can only be entered from one direction as indicated by the arrow on the entry line. Any combination of commands and arguments that can be generated by following the lines in the proper direction is syntactically correct. An argument is optional if there is a path around it. Alpha arguments must be separated with a space; queries (?) do not require a space.

Program Code Parameters

Program code parameters consist of a header followed by a parameter field. Headers can be of a long form or short form. The long form allows easier understanding of the program code and the short form allows more efficient use of the computer. When receiving a query response, the header is optional.

Example: { CHANNEL | CHAN | CH } { 1 | 2 }

Long form   Industry Standard

Short form   Program command parameters
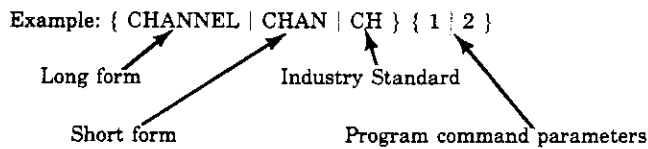
Program code parameters (arguments) may be of four types:

Strings - Any group of ASCII characters, excluding quotation marks (decimal 34), surrounded by quotation marks.

Numeric - Any integer, floating point, or exponential value. The ASCII characters <E> or <e> are used to delimit the mantissa of exponential parameters. Spaces are allowed between +, −, or <E> and the digits, but not between digits or the decimal point (.) and digits. The three types of numeric data are defined in IEEE 728-1982. The HP 54111D responds with two types of numeric data formats. They are NR1, and NR3 (refer to the syntax diagrams below).

Integer Format NR1
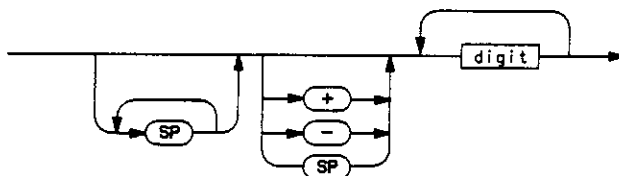
Example of NR1 format
{<sp> | + | −}ddd

Example of NR1
−5555

Exponential Format NR3
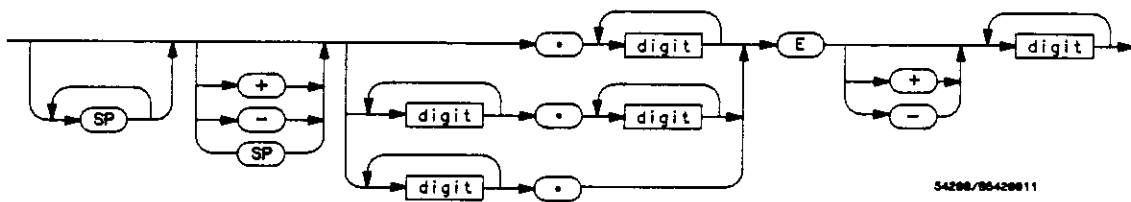
Example of NR3 format:

[<sp> | + | −]n.dd...{E | e}[+ | −]dd

Example of NR3
3.0E − 4

Alpha - Some commands require or allow alpha arguments such as "ON," or "OFF." These arguments are ASCII strings that start with an alpha character and are followed by a printable character except space (SP), semicolon (;), comma (,), number symbol (#), quote ("), or underline (_).

The general rules for program command formatting are:

- The HP 54111D sends and receives data messages in standard ASCII code.

- The HP 54111D sends and receives waveform data in binary or ASCII using the #A format as defined in IEEE 728-1982.

- The instrument responds equally to upper and lower case characters.

- The instrument responds equally to long form and short form command headers and alpha arguments. Both forms are shown throughout this manual. Long forms are shown using upper and lower case letters, and the short form is shown using the upper case letters of the commands.

- Parameter fields containing multiple parameters require a comma (,) to delimit individual parameters.

- Alpha arguments must be separated from the command header with a space.

Syntax errors in data messages are trapped and can be reported via HP-IB. Refer to "System Commands" Chapter 2, for a discussion of the key words "STATUS?" and "ERROR?".

Blocks - A block of binary data in the #A format as defined in IEEE 728-1982. This format is a binary block with the format:

```
<#><A><length word><DAB...DAB>
```

ASCII #

ASCII A

16 bit binary integer

even number of data bytes
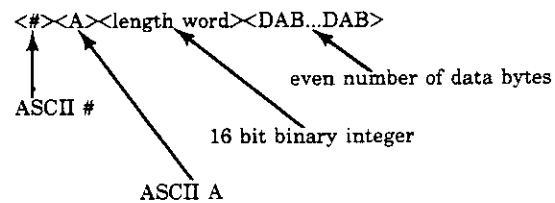
DABs are binary data bytes. Length word is a 16 bit integer representing the number of DABs (data bytes) to be sent in this block. The number of data bytes (DABs) transmitted will always be an even number of bytes except when transmitting a memory 5-8 in BYTE format. <#> and <A> are ASCII bytes. <#> defines the transmission as a block mode data transmission, and <A> defines the data to be binary format. When data is to be sent in the block mode this format must be used to ensure that the oscilloscope and the controller expect the same data types and formats.

The length word is two binary bytes. The most significant byte is transmitted first, then the least significant byte is transmitted.

	Length Byte 1							
HP-IB Data Line (DIO)	8	7	6	5	4	3	2	1
Length bit	L15	L14	L13	L12	L11	L10	L9	L8

	Length Byte 2							
HP-IB Data Line (DIO)	8	7	6	5	4	3	2	1
Length bit	L7	L6	L5	L4	L3	L2	L1	L0

Data Output (Query) Format

When a query command (command followed by a "?") is sent to the HP 54111D, a response message is generated and sent back to the controller. The HP 54111D should be addressed to talk as soon as the query has been sent in order to receive the response. If the response is not read prior to another query being sent, the first query response will be overwritten in the HP-IB output buffer. Therefore the response to the first query is lost.

The command headers and subsystem selectors will be returned if the HEADER command has been set ON and will not be returned if set to OFF.

Headers and alpha arguments will be returned in the long form if the LONGFORM command is set to ON and will be returned in the short form if the LONGFORM command is set to OFF.

There are three types of output arguments from the HP 54111D. They are:
- alpha arguments
- integers
- real numbers.

All output fields are an even number of bytes long except when transmsitting a memory 5-8 in BYTE format. Arguments may or may not be preceded by a header, depending on whether or not the HEADER command is set to ON or OFF.

Data Messages

The HP 54111D can send data messages in local or remote mode, when addressed to talk, or when in the talk-only mode (refer to "HP-IB Addressing," page 1-1). Data messages are normally sent in response to a query command.

Note

Before the instrument is addressed to talk, the desired output data must be specified with the appropriate input data message (query). Otherwise, the instrument outputs the over range value "1E38" by default to complete the bus transaction. If the ERR (status bit 5) service request is enabled, a service request will be generated with the "Output Buffer Empty" error in the ERRor queue.

Output data messages include the settings of individual functions, instrument status information, and binary learn string or cal string data. All output data messages contain either a leading space (<SP>) or minus sign (<->) followed by the function value or status data. <CR> and <LF> are sent as the end of line (EOL) message for all output data.

Characters transmitted over the HP-IB are the same whether they are a string, real numbers, or integers. For information exchange to be useful, the controller and the HP 54111D must be set up for the same data formats.

When exchanging data with the HP 54111D, there are four data types used. The four types of output available are:
- headers and alpha arguments
- integer numbers
- real numbers
- waveform data

These formats shown below refer only to the output of the HP 54111D. All output fields transmitted from the HP 54111D contain an even number of bytes.

When transmitting information to the HP 54111D, virtually any format can be used as long as the values are legal for the command.

Headers and alpha arguments are of varying length when LONGFORM is ON, and they are in the form:

    <alpha>...<alpha><SP>[SP]

A trailing space is automatically added if needed to make the output an even number of bytes. When the LONGFORM is OFF, the abbreviations are in the form:

    <alpha>...<alpha><SP>[SP...SP]

Trailing spaces are used to adjust the string length to six characters. Six characters is the minimun number of characters sent for both LONGFORM ON and LONGFORM OFF.

Integers (NR1 format) can represent values in the range − 32,768 to 32,767 and are transmitted in the form:

    [SP...SP]<SP or − ><d>...<d>

Leading spaces are added to make a total of six characters transmitted.

When querying TRIGGER HOLDOFF EVENT, the response is an integer that is 12 characters long.

Real numbers (NR3 format) are transmitted in scientific notation with an "E" separating the mantissa and exponent in the form:

<SP or −><d><.><d><d><d><d><d><d><E><+ or-><d><d>

WAVEFORM data is exchanged in three different TYPES:
- WORD
- BYTE
- ASCII

WORD and BYTE type data are transmitted using the IEEE standard 728-1982 #A binary block format. The character string "#A" is sent first, indicating that binary data will follow. Next a two-byte length value (16-bit integer value) is sent; it specifies the number of data bytes to follow. ASCII type data is transmitted one value at a time separated by <CR><LF>. DISPLAY data is exchanged in the same binary method as WORD and BYTE WAVEFORM data.

Learn and Cal Strings

If a "SETUP?" command is sent to the HP 54111D, the oscilloscope will output a learn string. The HP 54111D must be addressed to talk immediately after the SETUP? command is sent. The learn string contains information about the current front panel setup of the oscilloscope. The learn string consists of 296, 8-bit bytes of binary data. This binary data can be stored in the controller's memory, then this same information can be returned to the oscilloscope with a SETUP command. When the learn string is returned to the oscilloscope, the instrument's front panel will return to the original configuration. The learn string includes only those parameters that determine the front panel setup of the instrument.

If a "CALIBRATE?" command is sent to the HP 54111D, the oscilloscope will output a cal string. The HP 54111D must be addressed to talk immediately after the CALIBRATE? command is sent. The cal string consists 630, 8-bit bytes containing, the delay cal factors. This binary data can be stored in the controller's memory for future use.

The learn string data comprise the same information that is in the instrument's SAVE/RECALL registers. Refer to the *Front Panel Reference Manual* for additional information concerning these registers. Refer to Chapter 2, "System Commands," of this manual for more information on the commands SETUP and CALIBRATE.

Default Settings

When power is applied to the instrument, several interface parameters are put in the reset condition. Specifically the request mask (RQS mask) is set to 32546 (bits 1,5,8-14 set).

If you hold a front panel key down at the same time the unit is energized (key down power-up), the HP 54111D will initialize a more complete set of parameters. See table 1-1 and table 1-2 for the reset conditions.

Table 1-1. Reset Conditions for the HP 54111D.

Channel 1/Channel 2 Display	— On
Channel 1/Channel 2 VOLTS/DIV	— 2.00 volts/div
Channel 1/Channel 2 OFFSET	— 0.00 volts
Channel 1/Channel 2 Preset	— None
Channel 1/Channel 2 Input Coupling	— dc
Channel 1/Channel 2 Input Impedance	— 1 MΩ
Time/DIV	— 1 μs/div
DELAY	— 0.00 s
Delay Ref at	— Center
Auto/Triggered Sweep	— Auto
Trigger Mode	— Edge
Trigger Source	— Channel 1
TRIGGER LEVEL (all sources)	— 0.00 volts
Trigger Slope (all sources)	— Positive
Holdoff (Edge mode)	— Time
HOLDOFF TIME (Edge mode)	— 70.000 ns
HOLDOFF EVENTS (Edge mode)	— 2 Events
Trigger Pattern (pattern mode)	— Channel 1 - High
	— Channel 2 - Don't Care
	— Trigger 3 - Don't Care
	— Trigger 4 - Don't Care
When Entered/Exited (pattern mode)	— Entered
Holdoff (pattern mode)	—70.000 ns

Table 1-1. Reset Conditions for the HP 54111D (continued).

HOLDOFF TIME (pattern mode)	— 70.000 ns
HOLDOFF EVENTS (pattern mode)	— 2 Events
Trigger On Edge (state mode)	— Positive
Edge source (state mode)	— Channel 1
Pattern (state mode)	— Ch1 Trig = Clock
	— Channel 2 - High
	— Trigger 3 - Don't Care
	— Trigger 4 - Don't Care
Is Present/Not Present (state mode)	— Present
Holdoff Mode (state mode)	— TIME
HOLDOFF TIME (state mode)	— 70.000 ns
Arming Slope (time mode)	— Negative
Arming Source (time mode)	— Channel 1
DELAY (time mode)	— 20.00 ns
Trigger Slope (time mode)	— Positive
Trigger Channel (time mode)	— Channel 1
Arming Slope (events mode)	— Negative
Arming Source (events mode)	— Channel 1
NUMBER OF EVENTS (events mode)	— 1
Trigger Slope (events mode)	— Positive
Trigger Source (events mode)	— Channel 1
Display Mode	— Repetitive
Averaging (Repetitive mode)	— On
NUMBER OF AVERAGES (Repetitive mode)	— 2
Screen	— Dual
Graticule	— Axes
Resolution (Real Time mode)	— 6 bits
Completion Criteria (for HP-IB DIGitize command)	— 100%
Voltage Markers	— Off
V Marker 1 Position	— −2.0 volts
V Marker 2 Position	— +2.0 volts
Preset Levels	— 10 - 90%
Display Mode	— Repetitive
Averaging (Repetitive mode)	— On
NUMBER OF AVERAGES (Repetitive mode)	— 2
Screen	— Dual
Graticule	— Axes
Resolution (Real Time mode)	— 6 bits
Completion Criteria (for HP-IB DIGitize command)	— 100%

Table 1-1. Reset Conditions for the HP 54111D (continued).

Voltage Markers	— Off
V MARKER 1 POSITION	— −2.5 volts
V MARKER 2 POSITION	— +2.5 volts
V Marker 1 Source	— Channel 1
V Marker 2 Source	— Channel 1, 2
Preset Levels	— 0 - 100%
Time Markers	— Off
START MARKER POSITION	— 2.5 μs
STOP MARKER POSITION	— −2.5 μs
Start Marker Edge Slope	— Positive
START MARKER NUMBER	— 1
Stop Marker Edge Slope	— Negative
STOP MARKER NUMBER	— 1
Waveform Memories	— Off
Source for Store	— Channel 1
Selected Memory	— Waveform Memory 5
Waveform Data	— 0 volts
Memory Scaling	
TIME/DIV	— 500 ns/div
VOLTS/DIV	— 2.00 V/div
Function 1 and 2	— Off
Functions Definition	— Chan 1 − Chan 2
Measure Source	— Channel 1

Table 1-2. HP-IB Reset Conditions for the HP 54111D

Service Request Mode	— Disabled (RQS OFF)
Service Request Mask	— Decimal 32546
	(bits 1,5, 8-14 set)
Serial Poll Status Byte	— Clear
Error Queue	— Empty
WAVeform Format	— Word
EOI	— On
Long form	— Off
Header	— Off

Instrument Status

The instrument status indicates what the instrument is doing at the time it is checked. In the HP 54111D, 11 different operations have been defined as part of the instrument status. Some of these operations are very specific while others are of a more general nature. The more specific operations are defined as the ready byte, while the more general operations are defined as the status byte. As an example, the ready byte contains information about whether the instrument is running a self test, if the instrument is currently doing a data acquisition, a hardcopy output, or if it is triggered, etc.

The status byte contains general information such as is the instrument ready? - has a front panel key been pressed? - was a message displayed on the CRT? - or has a service request been issued? In each of these cases you are told that something has happened, but you don't know the specifics; i.e., a key has been pressed - which key?, a message has been displayed - which message?, or the instrument has requested service - what does it need?

Of special interest in the status byte is the READY bit, which indicates whether the instrument is ready or not ready to receive commands. Not ready indicates that the instrument is busy at this time. The things that determine the ready condition are contained in the ready byte (see above). If the status is checked and it is determined that the instrument is not ready, you could then read the ready byte to find out why it isn't ready. In the same manner, by checking the status byte you could find if a key has been pressed, or a message has been displayed on the CRT. Then you could read the key code register or message code register to find out exactly what happened.

There is one more very special bit contained in the the status byte. This bit is the Service Request bit and is described on page 1-23.

Refer to table 1-3 (page 1-18) for the meaning of each bit in the status word.

Checking the Status

Several different methods are available in the HP 54111D to check the status of the instrument. These methods are each discussed below.

The Status Word — The status word is a combination of the status byte and the ready byte. The status word is read using the command "STATUS?". The information in the status word is dynamic, meaning that it reflects the actual condition of the instrument at any instant in time. The status word is a 16-bit word (two 8-bit bytes).

The status word contains the information shown in the table below. The table shows each bit in the status word and the bit weight. When you read the status word, the value returned is the total bit weights of all bits that are high.

When the status word is read, the returned value is usually a large number since most of the bits in the upper half of the word are usually high. They are high because in most cases, a signal is not being acquired (bit 10), a parse is not being done (bit 8), a hardcopy output is not in progress (bit 12), etc.

As indicated, in table 1-3, bit 15 in the status word is used to indicate that the trigger is armed and no trigger has yet been received. This causes a potential problem in programs written in BASIC. While BASIC will properly handle most math operations (DIV, MOD, etc) on integers greater than 32757 (Bit 15 is set), some operations such as BIT test and SHIFT will fail with an INTEGER OVERFLOW error.

This anomaly will potentially affect currently written test programs. The solution is to treat the status word as two separate bytes. This can be done by using the RDY? query to obtain the upper byte and using the STB? query to obtain the status byte. Alternately the following code can be used to separate the result of the STA? query.

```
OUTPUT @Scope;"sta?"
ENTER @Scope;Status
Ready=Status DIV 256
Status=Status MOD 256
```

The desired operations can now be performed on the two variables (Ready) and (Status).

Refer to the system command STATUS? in Chapter 2, "System Commands."

Table 1-3. The Status Word.

BIT	BIT WEIGHT	BIT NAME	CONDITION
15	32768	Arm	0 = trigger not armed
			1 = trigger armed
14	16384	Cal	0 = calibration in progress
			1 = calibration not in progress
13	8192	Test	0 = self test in progress
			1 = self test not in progress
12	4096	Hard	0 = hardcopy output in progress
			1 = hardcopy output not in progress
11	2048	Data	0 = data buffer empty
			1 = data in buffer to be read
10	1024	Acq	0 = waveform acquisition in progress
			1 = waveform acquisition not in progress
9	512	Trig	0 = instrument is not receiving triggers
			1 = instrument is receiving triggers
8	256	Parse	0 = command is being parsed
			1 = command is not being parsed
7	128	MSG	0 = no message has been displayed
			1 = message has been displayed
6	64	RQS	0 = instrument is not requesting service
			1 = instrument is requesting service
5	32	ERR	0 = no error has occurred
			1 = error has occurred
4	16	RDY	0 = not ready (instrument is busy)
			1 = instrument ready
3	8	LCL	1 = local switch pressed or power cycle (transitions high then back low)
2	4	FPS	0 = front panel key has not been pressed
			1 = front panel key has been pressed
1	2	PWR	Power, NOT used, always 1
0	1	RQC	Request control, not used, always 0

0 = False = Low   1 = True = High

The Status Byte — The status byte is the lower byte of the status word. The status byte is read using the command SPOLL?. Do not confuse the SPOLL? command with an HP-IB serial poll (SPOLL(707)) discussed later in this section, they are NOT the same. The status byte is an eight-bit byte. It, like the status word, is dynamic and reflects the status at the instant it is read. The table below shows the status byte. The table shows each bit in the status byte and the bit weight. When you read the status byte, the value returned is the total bit weights of all bits that are high at the time you read the byte.

Refer to the SPOLL? command in Chapter 2, "System Commands."

Table 1-4. The Status Byte.

BIT	BIT WEIGHT	BIT NAME	CONDITION
7	128	MSG	0 = no message has been displayed 1 = message has been displayed
6	64	RQS	0 = instrument is not requesting service 1 = instrument is requesting service
5	32	ERR	0 = no error has occurred 1 = error has occurred
4	16	RDY	0 = not ready (instrument is busy) 1 = instrument ready
3	8	LCL	1 = local switch pressed or power cycle (transitions high then back low)
2	4	FPS	0 = no front panel key has been pressed 1 = front panel key has been pressed
1	2	PWR	Power, NOT used, always 1
0	1	RQC	Request control, NOT used, always 0

0 = False = Low
1 = True = High

The Ready Byte — The ready byte contains the same information as the upper byte (bits 8 through 15) of the status word. The ready byte is an eight-bit byte. The ready byte is read using the command READY?. It, like the status word, is dynamic and reflects the status at the instant it is read. The only difference between the ready byte and the upper byte of the status word is that the bit weights have been changed in the ready byte. This makes the returned values much smaller. The table below shows the ready byte. The table shows each bit in the ready byte, and the bit weight. When you read the ready byte, the value returned is the total bit weight of all bits that are high at the time you read the byte.

Refer to the READY? command in Chapter 2, "System Commands."

Table 1-5. The Ready Byte.

BIT	BIT WEIGHT	BIT NAME	CONDITION
7	128	Arm	0 = trigger not armed 1 = trigger armed
6	64	Cal	0 = calibration in progress 1 = calibration not in progress
5	32	Test	0 = self test in progress 1 = self test not in progress
4	16	Hard	0 = hardcopy output in progress 1 = hardcopy output not in progress
3	8	Data	0 = data buffer empty 1 = data in buffer to be read
2	4	Acq	0 = waveform acquisition in progress 1 = waveform acquisition not in progress
1	2	Trig	0 = instrument is not receiving triggers 1 = instrument is receiving triggers
0	1	Parse	0 = command is being parsed 1 = command not being parsed

0 = False = Low
1 = True = High

Note

Most of the bits in the ready byte are normally high, and only go low while an instrument operation is taking place.

Request Mask — The request mask is a 16-bit word that is sent to the HP 54111D with the REQUEST command. The value of the current request mask can be read using the REQUEST? command. This command allows selection of the conditions that will produce a service request from the HP 54111D. Refer to table 1-6 for request mask information. This table shows the bit weights and what is masked with each bit.

The lower byte of the request mask is used to enable bits in the status byte that will produce a service request. The upper byte of the request mask is used to enable bits in the ready byte that will set the ready (RDY) bit in the status byte. The conditions selected to create a service request must contain a one or high in that bit position of the request mask. The bits are set high by sending a value that is the total of all the mask weight values you want to use.

The request mask is shown in table 1-6. Note that the 16 bits in the request mask have the same names as the 16 bits in the status word. Each bit in the request mask will enable (1) or disable (0) the corresponding bit in the status word.

For example, if the request mask value is set to zero, no conditions are allowed to produce a service request from the HP 54111D. If the request mask value is set to 65,535, any change in the status word will cause a service request. This is probably not the way you would want to use the Request Mask, therefore some practical examples are shown below.

Example 1

If you would like to be flagged (via service request from the HP 54111D) when a front panel key has been pressed, send the request value 68. The value is the total mask weight of the status values you want to see in this example, FPS (front panel service) mask weight 4 and RQS (request service) mask weight 64. The RQS bit must be enabled for the instrument to produce a service request. In this example, if a key is pressed a service request will be produced.

Example 2

If you wish to know when a hardcopy output is complete, set the request mask value to 4176. This value is the total of the mask weights for Hard (hardcopy) mask weight 4096, RQS (request service) mask weight 64, and RDY (ready) mask weight 16. Keep in mind that only a status byte change can produce service request and the ready byte can only produce a change in the RDY bit of the status byte. This means that the ready bit must be used as an indirect link between hardcopy in the ready byte and RQS bit in the status byte.

Refer to "Service Request" (next in this chapter) and to the REQUEST command in the System Commands chapter of this manual, for more information about service requests.

Table 1-6. The Request Mask.

LOWER BYTE									
Request Mask Bit	0	1	2	3	4	5	6	7	Status Byte Mask
Mask Weight	1	2	4	8	16	32	64	128	(High - enables service request)
What It Masks	RQC	PWR	FPS	LCL	RDY	ERR	RQS	MSG	
UPPER BYTE									
Request Mask Bit	8	9	10	11	12	13	14	15	Ready Byte Mask
Mask Weight	256	512	1024	2048	4096	8192	16384	32768	(High - enables ready bit)
What It Masks	Parse	Trig	Acq	Data	Hard	Test	Cal	Arm	

Service Request

The service request bit (RQS) is special because it is an HP-IB (IEEE 488) standard. This bit is always bit six of the status byte in any instrument that conforms to the IEEE 488 standard. The RQS bit is also associated with the SRQ (service request) control line of the HP-IB. When bit six of the status byte goes true, it causes the SRQ control line to go high. The SRQ control line can create an interrupt on controllers that support "Serial Poll" as defined by the IEEE 488 standard.

The service request is useful because it can tell the controller when the HP 54111D has finished an operation and is ready for another command. This allows the controller to be used for other processing, instead of waiting for the HP 54111D to finish an operation. When the HP 54111D is busy, it is not ready. The HP 54111D is busy when it is acquiring data, or when it is doing a hardcopy output, etc.

Another use of the service request is to enable you to write operator interactive programs. In this case the oscilloscope might display a message to tell the operator where to place the probes for the next measurement. When the operator has the probes in place, a front panel key should be pressed. You could use the service request to tell the controller when the manual operation is complete, via a front panel key press. Once the controller "sees" the service request it can make the required readings or measurements. Until the service request is produced by the HP 54111D, the controller can be processing some other data or controlling another instrument.

In order to use the service request, your program must enable the HP-IB interrupts. After the controller is informed that one of the instruments on the bus requires service, your program must take the steps necessary to find out which instrument is asking for service. You can find out which instrument requires the service by conducting a serial poll of all the instruments on the bus. Once it is known which instrument set the SRQ line, then you can read the status of that instrument to determine what to do next.

Serial Poll

This example will show how to use the service request by conducting a serial poll of all instruments on the bus. In this example, assume that there are two instruments on the bus; an oscilloscope at address 7 and a printer at address 1. These address assumptions are made

throughout this manual, and it is also assumed that we are operating on bus #7.

The serial poll operation can be conducted in the following manner.

1. Enable interrupts on the bus. This allows the controller to "see" the SRQ line.

2. If the SRQ line is high (some instrument is requesting service) then check the instrument at address 1 to see if bit 6 of its status register is high.

3. To check if bit 6 of an instrument's status register is high, use the following command line.

   IF BIT (Stat, 6) then
       (go to step 5 if true)

4. If bit 6 of the instrument at address 1 is not high, then check the instrument at address 7 to see if bit 6 of its status register is high.

5. Disable interrupts on the bus.

6. As soon as the instrument with status bit 6 high is found, check the rest of the status bits to determine what is required.

The program command for serial poll on the HP series 200 controller is "Stat=SPOLL(707)." The address 707 is the address of the oscilloscope in the example above. The command for checking the printer above would be "STAT =SPOLL(701)" because the address of that instrument is 01 on bus address 7. This command reads the contents of the HP-IB Status Register into the variable called "Stat." At that time bit 6 of the Stat variable can be tested to see if it is set (bit 6=).

The SPOLL(707) command causes much more to happen on the bus than simply reading the register. This command clears the bus, automatically addresses the talker and listener, sends SPE (serial poll enable) and SPD (serial poll disable) bus commands, and reads the data. This is not a detailed description of the command. For more information about serial poll, refer to your controller manual, and programming language reference manuals.

After the serial poll is completed, the HP-IB Status Register in the HP 54111D will be reset if the service request bit (bit 6) in that

register was set. If the service request bit in the oscilloscope was not set, the contents of the register are not changed. The HP-IB Status Register in the HP 54111D is the only status register that is NOT dynamic. This means that once a bit in this register is set, it will remain set until a serial poll is conducted with the service request bit high, or the instrument is reset in some manner.

Do not confuse this HP-IB serial poll with the System Command (SPOLl) that reads the instrument status word. Refer to the command SPOLL? in the "System Commands" chapter of this manual.

Program Examples

The program examples given for each command in the next ten chapters were written on HP 9000 Series 200/300 computers using BASIC 4.0 language. The programs always assume the oscilloscope is at address 707. If a printer is used, it is always assumed to be at address 701. If a plotter is used, it is always assumed to be at address 705.

In these examples, special attention should be paid to the ways in which the command/query can be sent. The way the instrument is set up to respond to a command/query has no bearing on how you send the command/query. That is, the command/query can be sent using the long form, short form, or the industry standard form if one exists for that command. You can send the command/query using upper case (capital) letters or lower case (small) letters, both work the same. Also, the data can be sent using almost any form you wish. If you were sending a channel 1 sensitivity value of 100 mV/division, that value could be sent using a decimal (.1), or an exponential (1e − 1 or 1.0E − 1). As an example, set the channel 1 sensitivity to 100 mV/division by sending one of the following:
  * commands in long form and using the decimal format.
        Output 707;"CHANNEL 1 SENSITIVITY .1"

  * commands in short form and using an exponential format.
        Output 707;"CHAN 1 SENS 1E − 1"

  * commands using industry standard and short form, using an exponential.
        Output 707;"ch 1 sens 1.0e − 1"

If you want to observe the headers for the queries, you must bring the returned data into a string variable. Generally, you should dimension all string variables as shown later in this section.

If you do not need to see the headers and a numeric value is returned from the HP 54111D, then you should use a numeric variable. In this case the headers are ignored, because the number builders of the controller ignore anything except 0 through 9, $+$, $-$, e, and E.

When any of the subsystem selectors are queried or the MEASURE ALL? query is sent, the HP 54111D will respond with multiple lines of output to that query. Each of the output lines are terminated by a <CR><LF>, which will cause the controller to think the output is complete, unless something is done in the program to cause the controller to ignore the <CR><LF>s. The sample programs in this manual have done this by using the statements: USING "K" and USING " – K." These statements will cause all <CR><LF>s to be ignored, therefore EOI should be used to determine the end of the data transfer. Refer to the example program below.

```
10 Dim Fun$[300]
20 OUTPUT 707;"HEADERS ON LONGFORM ON EOI ON"
30 OUTPUT 707;"FUNCTION 1?"
40 ENTER 707 USING " – K"; Fun$
50 PRINT USING "K";Fun$
60 END
```

This program will produce an output that resembles the following:

```
FUNCTION    1
SUBTRACT  CHANNEL 1, CHANNEL 2
```

If the "USING" statements are left out of this program, the output would look similiar to:

```
FUNCTION    1
```

Only one line is displayed without the "USING" statements, because the entry is terminated by the first <CR><LF>.

Below are several more examples of BASIC programs used to receive data from the HP 54111D. These examples will return the sensitivity (volts/division) value of channel 1. All of these queries

will accomplish the same thing. These programs demonstrate the various command forms and returned data formats.

The first examples show the use of headers in the long form and short form. Since we want to see the headers we will use string ($) variables. The third example shows the data being returned to string variable with the headers off, and the fourth example shows the data being returned to a numeric variable. In all cases the value is sent over HP-IB in the exponential (NR3) format, however, in the last example the controller converts this value to a decimal for the display. All controllers may not do this conversion.

Example with HEADER "on" and LONGFORM "on."
```
10 DIM Sens$[30]
20 OUTPUT 707;"ch 1 sens?"
30 ENTER 707; Sens$
40 PRINT Sens$
50 END
```

The controller will display:
SENSITIVITY 1.00000E-01

Example with HEADER "on" and LONGform "off."
```
10 DIM Sens$[30]
20 OUTPUT 707;"CHANNEL 1 SENSITIVITY?"
30 ENTER 707;Sens$
40 PRINT Sens$
50 END
```

The controller will display:
SENS 1.00000E-1

Example with HEADER "off."
```
10 DIM Sens$[30]
20 OUTPUT 707;"CHANNEL 1 SENS?"
30 ENTER 707;Sens$
40 PRINT Sens$
50 END
```

The controller will display:
1.00000E-1

Example using a numeric variable instead of a string ($) variable.
Headers will not be displayed, therefore HEADER can be "on" or "off."
10 OUTPUT 707;"CH 1 SENSITIVITY?"
30 ENTER 707;Sens
40 PRINT Sens
50 END

The controller will display:
.1

More program examples are shown in Appendix B of this manual.

Command Set Organization

The command set for the HP 54111D is divided into ten separate groups—system commands and nine sets of subsystem command. Each of the ten groups of commands is described in the following chapters. Each of the chapters contains a brief description of the subsystem, a set of syntax diagrams for those commands, and the commands for that subsystem in alphabetic order. The commands are shown in the long form and short form using upper and lowercase letters. As an example, AUToscale indicates that the long form of the command is AUTOSCALE and the short form of the command is AUT. Industry standard forms are shown for the commands that have this form. Each of the commands contains a description of the command and its arguments, the command syntax, and a programming example.

The next ten chapters of this manual are:
System Commands - These commands control the HP-IB operations as well as the basic operation of the HP 54111D.

Acquire Subsystem - The commands in the Acquire Subsystem determine the conditions for the DIGITIZE command.

Channel Subsystem - The commands in the Channel Subsystem are used to control the two vertical inputs (channels 1 and 2).

Display Subsystem - The commands in the Display Subsystem are used to control how waveform data, time and voltage markers, text, and the graticules are displayed on the CRT.

Function Subsystem - The commands in the Function Subsystem are used to control the waveform math features of the instrument.

Hardcopy Subsystem - The Hardcopy Subsystem commands control parameters used during the printing and plotting of waveforms from the HP 54111D.

Measure Subsystem - The commands in the Measure Subsystem control the automated measurements that can be made with the HP 54111D.

Timebase Subsystem - The commands in the Timebase Subsystem control the timebase section of the HP 54111D.

Trigger Subsystem - The commands in the Trigger Subsystem control the trigger modes of the HP 54111D.

Waveform Subsystem - The commands in the Waveform Subsystem control the transfer of data to and from the HP-IB buffer memories in the HP 54111D.

HP 54111D
COMMANDS

System Commands

SUBSYSTEM SELECTORS WITH COMMANDS

	ACQuire	CHANnel	DISPlay	FUNCtion	HARDcopy	MEASure	TIMebase	TRIGger	WAVeform
AUToscale									
BLANk									
CALibrate									
CLEAr									
DIGitize									
DSP									
EOI	ACQuire	CHANnel | CH	DISPlay	FUNCtion	HARDcopy	MEASure	TIMebase	TRIGger	WAVeform
ERASe	COMPlete	COUPling	ATTRibute	ADD	PAGE	ALL?	DELay | DLY	CONDition	COMPlete?
ERRor?	COUNt | CNT	ECL	BLINk	INTegrate	PEN	CURSor?	MODE	COUPling	COUNt? | CNT?
HEADer	RESOlution	OFFset	BRIGhtness	INVErt	SPEed	DUTYcycle? | DUT?	OFFset	DELay | DLY	DATA
ID?	TYPE	PROBe	COLOr	MAXImum		ESTArt	RANGe	DURation	FORMat
KEY		RANGe	COLumn	MINimum		ESTOp	REFerence	ENABle	POINts? | PNTS?
LOCal		SENSitivity	DATA	MULTiply		FALL?	SENSitivity	HOLDoff	PREamble
LONGform		TTL	FORMat	OFFset		FREQuency?		LEVel | LVL	SOURce | SRC
MENU			GRATicule	ONLY		NWIDth?		LOGic	TYPE
MERGE			INVerse	RANGe		OVERshoot?		MODE	TDElay
PLOT			LINE	SUBtract		PBASe		PATH	TSENsitivity
PRINt			MASK	VERSus		PERiod?		PROBe	VALid?
READy? | RDY?			PERSistence			PREShoot?		SENSitivity	VOFfset
RECall			PRIority			PTOP		SLOPe	VSENsitivity
REMOTe			ROW			PWIDth?		SOURce | SRC	XINCrement?
REQuest | RQS			SETColor			RISE?			XORigin?
RESet | RST			SOURce | SRC			SOURce | SRC			XREFerence?
REVision?			STRing			TOPBase?			YINCrement?
RUN			TEXT			TDELta?			YORigin?
SAVE			TMARker			TSTArt			YREFerence?
SERial?			UNDerline			TSTOp			
SETup			VMARker			TVOLt?			
SPOLl? | STB?						VAVerage			
STAtus?						VBASe?			
STOP						VDELta?			
STORe						VMArkerset			
TEST | TST						VMAX?			
TRG | GET						VMIN?			
VIEW						VPP?			
						VRMS?			
						VRELative			
						VSTArt			
						VSTOp			
						VTIMe?			
						VTOP?			

F-Gemsys (541110)

Figure 1-1. Programming Command Tree

COMMAND	WHERE USED	COMMAND	WHERE USED	COMMAND	WHERE USED
ACQuire	Acquire Subsystem	LEVel | LVL	Trigger System	SETup	System Command
ADD	Function Subsystem	LINE	Display Subsystem	SLOPe	Trigger Subsystem
ALL?	Measure Subsystem	LOCal	System Command	SOURce | SRC	Display Subsystem
ATTRibute	Display Subsystem	LOGic	Trigger Subsystem	SOURce | SRC	Measure Subsystem
AUToscale	System Command	LONGform	System Command	SOURce | SRC	Trigger Subsystem
				SOURce | SRC	Waveform Subsystem
BLANk	System Command	MASK	Display Subsystem	SPEed	Hardcopy Subsystem
BLINk	Display Subsystem	MAXimum	Function Subsystem	SPOL? | STB?	System Query
BRIGhtness	Display Subsystem	MEASure	Subsystem Selector	STATus?	System Command
		MENU	System Command	STOP	System Command
CALibrate	System Command	MERGE	System Command	STORe	System Command
CHANnel | CH	Subsystem Selector	MINimum	Function Subsystem	STRing	Display Subsystem
CLEAr	System Command	MODE	Timebase Subsystem	SUBTract	Function Subsystem
COLOr	Display Subsystem	MODE	Trigger Subsystem		
COLumn	Display Subsystem	MULTiply	Function Subsystem	TDELay	Waveform Subsystem
COMPlete	Acquire Subsystem			TDELta?	Measure Subsystem
COMPlete	Waveform Subsystem	NWIDth?	Measure Subsystem	TEST | TST	System Command
CONDition	Trigger Subsystem			TEXT	Display Subsystem
COUNt | CNT?	Acquire Subsystem	OFFSet	Function Subsystem	TIMebase	Subsystem Selector
COUNt? | CNT?	Waveform Subsystem	OFFSet	Channel Subsystem	TMARker	Display Subsystem
COUPling	Channel Subsystem	OFFSet	Timebase Subsystem	TOPBase?	Measure Subsystem
COUPling	Trigger Subsystem	ONLY	Function Subsystem	TRG | GET	System Command
CURSor?	Measure Subsystem	OVERshoot?	Measure Subsystem	TRIGger	Subsystem Selector
				TSENsitivity	Waveform Subsystem
DATA	Display Subsystem	PAGE	Hardcopy Subsystem	TSTArt	Measure Subsystem
DATA	Waveform Subsystem	PATH	Trigger Subsystem	TSTOp	Measure Subsystem
DELay | DLY	Trigger Subsystem	PBASe	Measure Subsystem	TTL	Channel Subsystem
DELay | DLY	Timebase Subsystem	PEN	Hardcopy Subsystem	TVOLt?	Measure Subsystem
DISPlay	Subsystem Selector	PERIod?	Measure Subsystem	TYPE	Acquire Subsystem
DIGitize	System Command	PERSistence	Display Subsystem	TYPE	Waveform Subsystem
DSP	System Command	PLOT	System Command		
DURation	Trigger Subsystem	POINts? | PNTS?	Waveform Subsystem	UNDerline	Display Subsystem
DUTYcycle? | DUT?	Measure Subsystem	PREamble	Waveform Subsystem		
		PREShoot?	Measure Subsystem	VALid?	Waveform Subsystem
ECL	Channel Subsystem	PRINt	System Command	VAVerage	Measure Subsystem
ENABle	Trigger Subsystem	PRIority	Display Subsystem	VBASe?	Measure Subsystem
EOI	System Command	PROBe	Channel Subsystem	VDELta?	Measure Subsystem
ERASe	System Command	PROBe	Trigger Subsystem	VERSus	Function Subsystem
ERRor?	System Command	PTOP	Measure Subsystem	VIEW	System Command
ESTArt	Measure Subsystem	PWIDth?	Measure Subsystem	VMARker	Display Subsystem
ESTop	Measure Subsystem			VMARkerset	Measure Subsystem
		RANGe	Channel Subsystem	VMAX?	Measure Subsystem
FALL?	Measure Subsystem	RANGe	Function Subsystem	VMIN?	Measure Subsystem
FORMat	Display Subsystem	RANGe	Timebase Subsystem	VOFfset	Waveform Subsystem
FORMat	Waveform Subsystem	READy? | RDY?	System Query	VPP?	Measure Subsystem
FREQuency?	Measure Subsystem	RECall	System Command	VRELative	Measure Subsystem
FUNCtion	Subsystem Selector	REFerence	Timebase Subsystem	VRMS?	Measure Subsystem
		REMote	System Command	VSENsitivity	Waveform Subsystem
GET | TRG	System Command	REQuest RQS	System Command	VSTArt	Measure Subsystem
GRATicule	Display Subsystem	RESet | RST	System Command	VSTOp	Measure Subsystem
		RESOlution	Acquire Subsystem	VTIMe?	Measure Subsystem
HEADer	System Command	REVision?	System Query		
HARDcopy	Subsystem Selector	RISE?	Measure Subsystem	WAVeform	Subsystem Selector
HOLDoff	Trigger Subsystem	ROW	Display Subsystem		
		RUN	System Command	XINCrement?	Waveform Subsystem
ID?	System Command			XORigin?	Waveform Subsystem
INVerse	Display Subsystem	SAVE	System Command	XREFerence?	Waveform Subsystem
INVErt	Function Subsystem	SENSitivity	Channel Subsystem		
INTegrate	Function Subsystem	SENSitivity	Timebase Subsystem	YINCrement?	Waveform Subsystem
		SENSitivity	Trigger Subsystem	YORigin?	Waveform Subsystem
KEY	System Command	SERial?	System Query	YREFerence?	Waveform Subsystem
		SETColor	Display Subsystem		

Table 1-7. Programming Command Cross-reference

2 System Commands

Introduction

System commands control HP-IB operations as well as the basic operation of the oscilloscope. System commands can be called from any subsystem. When the system command has been executed the HP 54111D will return to the subsystem that it was in before the system command was executed.

If a query is sent followed by another query without a controller read (ENTER) between the queries, only the last response will be received. Although this operation is syntactically correct, the HP 54111D does not queue system command query responses. Therefore, the response to the first query will be overwritten in the HP-IB output buffer.

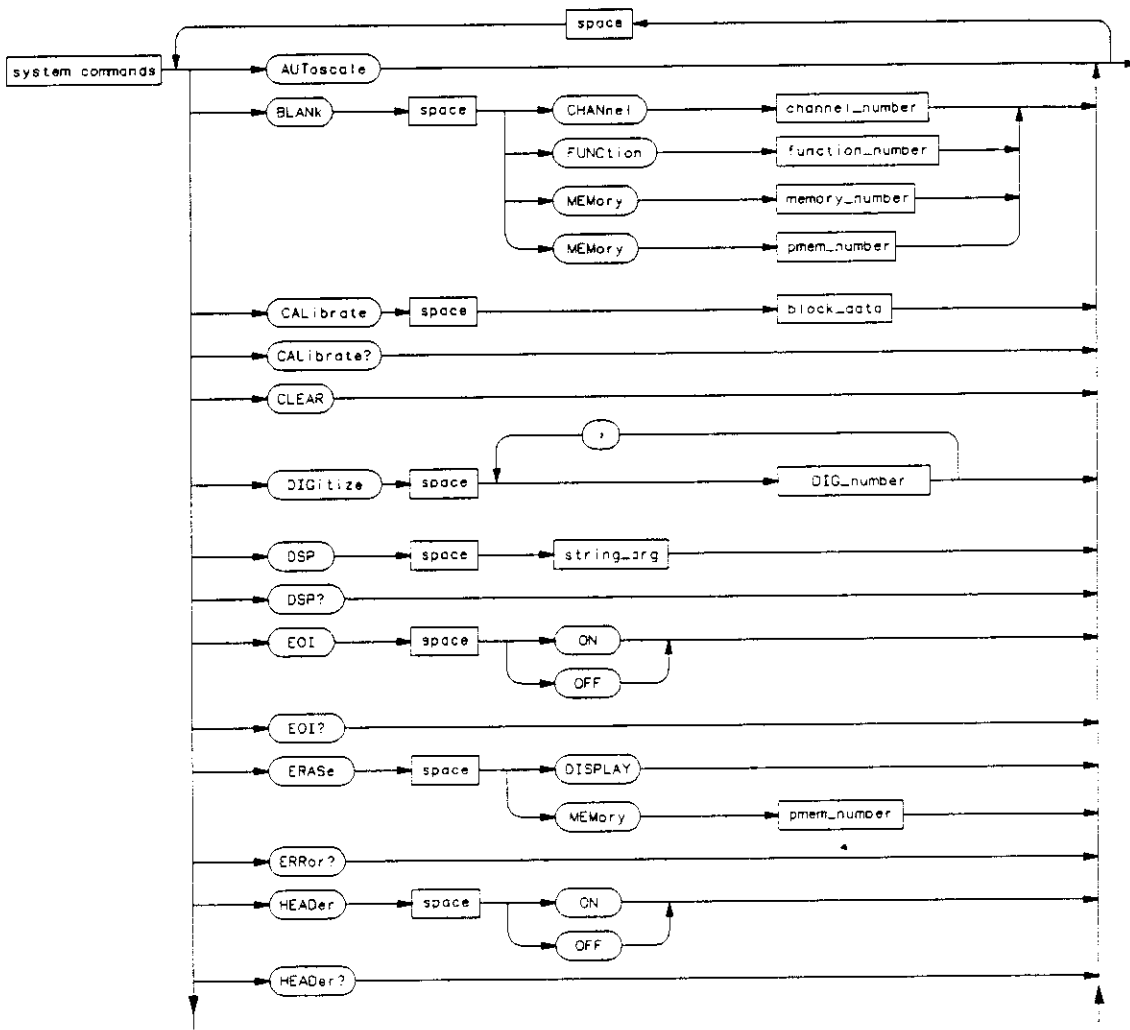Refer to figure 2-1 for system command syntax diagrams.

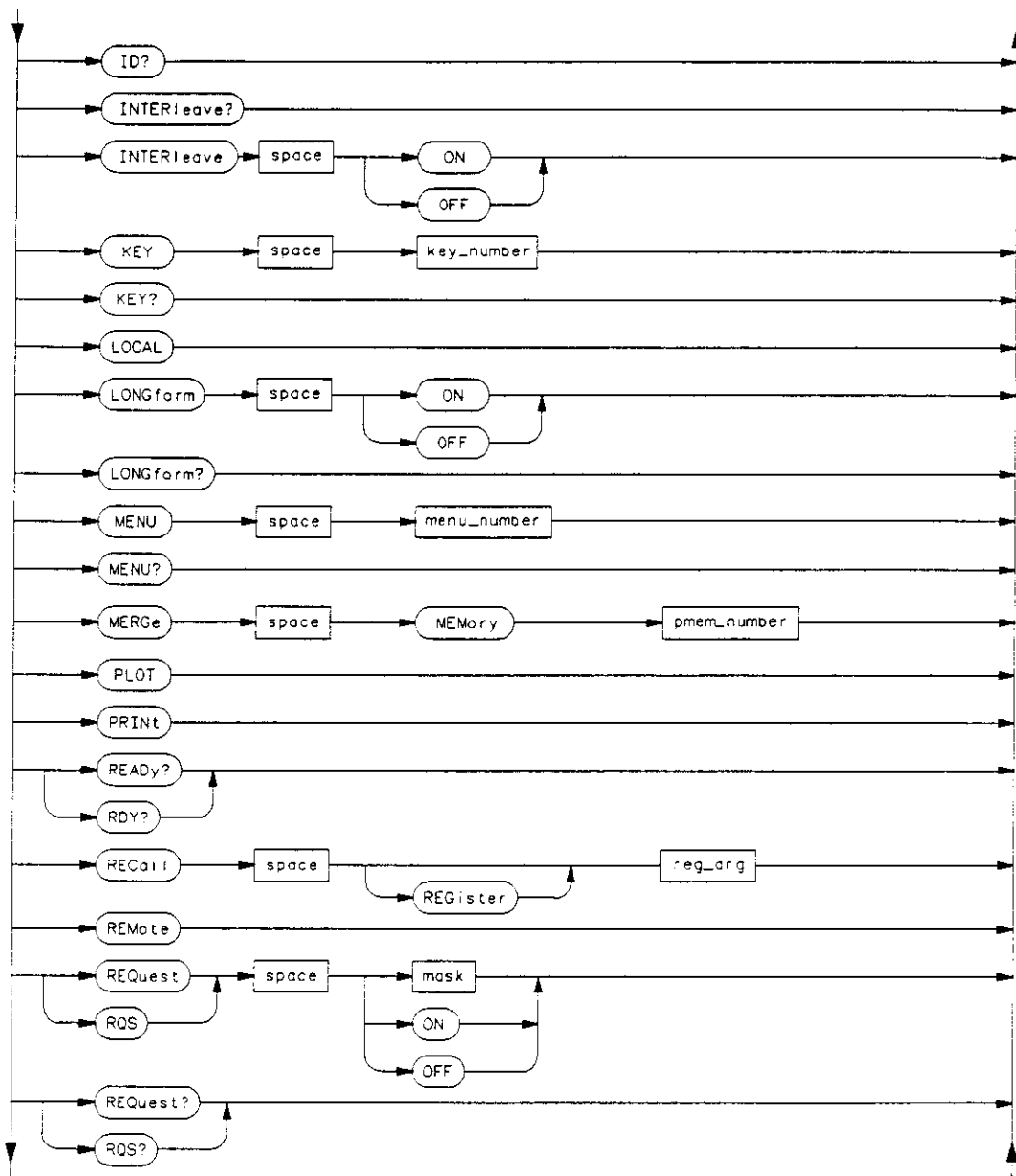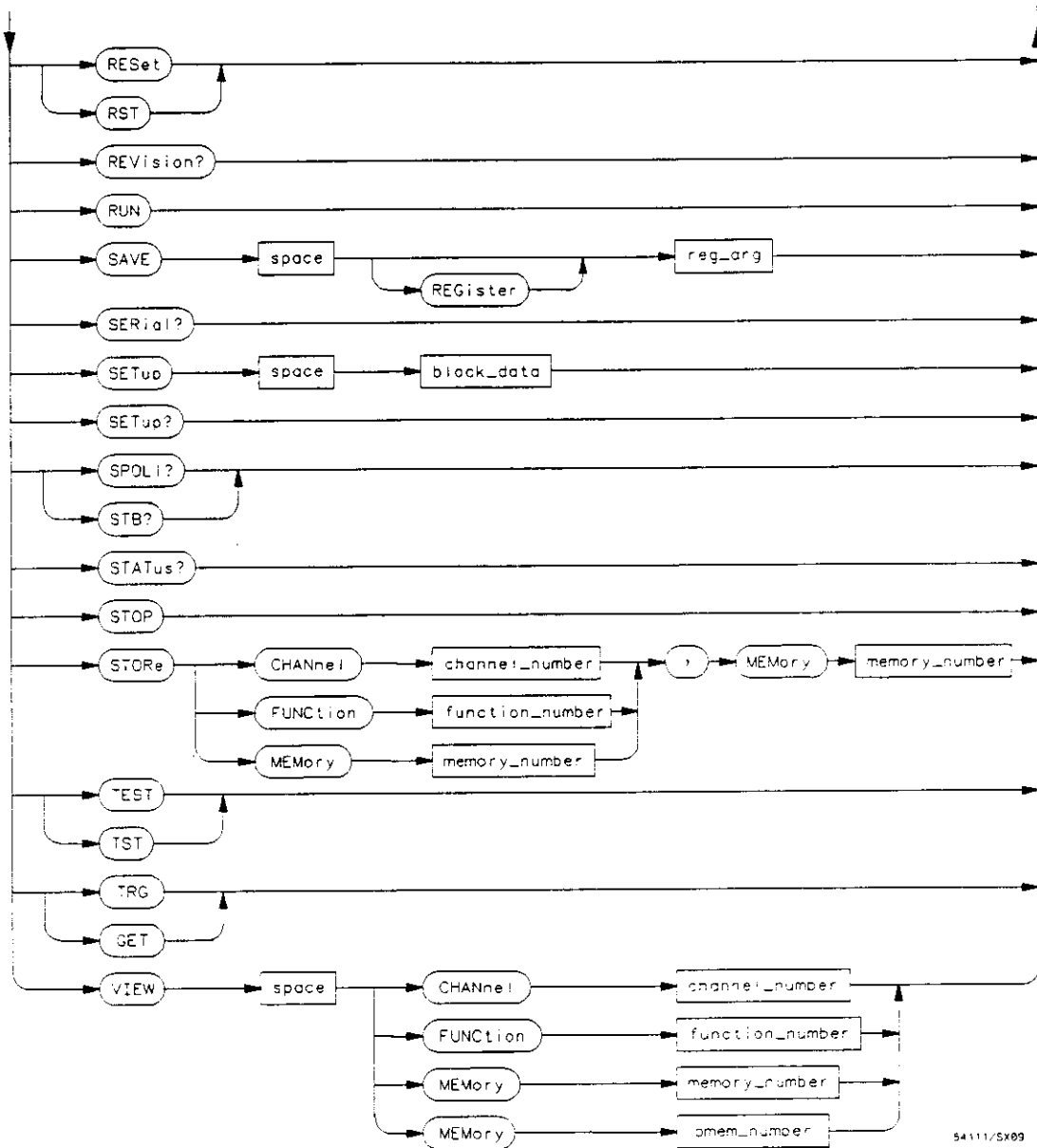*Figure 2-1. System Commands Syntax Diagrams*

Figure 2-1. System Commands Syntax Diagrams (continued)
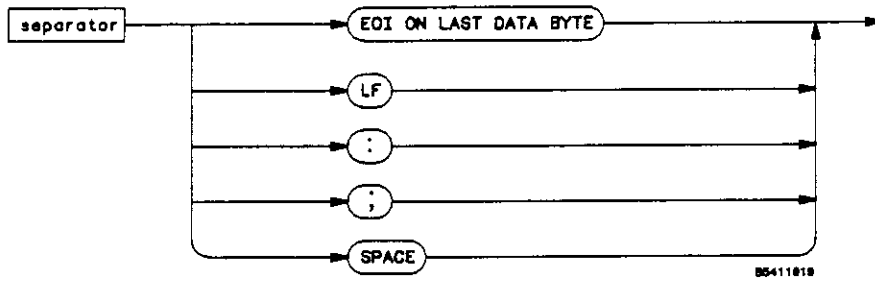
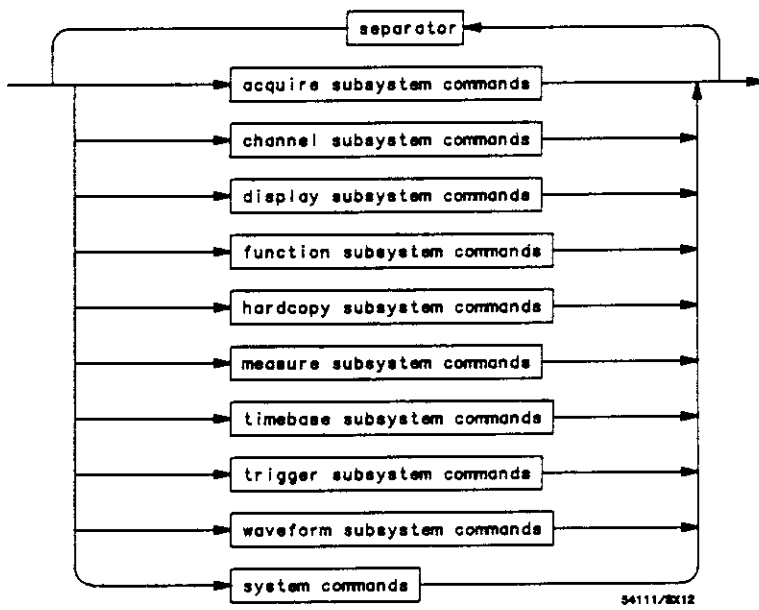*Figure 2-1. System Commands Syntax Diagrams (continued)*

Figure 2-1. System Commands Syntax Diagrams (continued).

BLOCK__DATA = A block of data in #A format as defined in IEEE Std 728-1982.

CHANNEL__NUMBER = An integer, 1 or 2.

DIG__NUMBER = An integer, 1, 2, 5, or 6.

FUNCTION__NUMBER = An integer 1 or 2.

KEY__NUMBER = An integer, 1 through 63 (see table 2-3, page 2-24, for keycodes)

MASK = An integer, 0 through 65535. This number is the sum of all the bits in the request mask corresponding to conditions that are to be enabled. See table 2-5 for the bit definitions in the request mask.

MEMORY__NUMBER = An integer from 1 to 8.

MENU__NUMBER = An integer from 1 to 10.

PMEM__NUMBER = An integer 9 or 10.

REG__ARG = An integer from 0-9.

STRING__ARG = Any collection of ASCII characters excluding quotes, surrounded by quotes.

Figure 2-1. System Commands Syntax Diagrams (continued).

AUToscale   **command**

The AUTO-SCALE command automatically selects the vertical
sensitivity, vertical offset, trigger level and sweep speed for a display
of the input signal. If input signals are present at both vertical
inputs the sweep will be triggered on Chan 1 and the display is set
to the dual screen mode. The vertical sensitivity for each channel
will be scaled appropriately. If only one of the vertical inputs has a
signal on it, the screen function is set to Single. See chapter 15 in
the *HP 54111D Front-panel Reference Manual* for input signal
requirements for proper AUTO-SCALE operation.

When the AUTO-SCALE operation is complete, the Timebase menu
will be selected, and the input devices will be assigned to the
TIME/DIV function.

Command Syntax: AUToscale

Example: OUTPUT 707;"AUTOSCALE"

BLANk   **command**

The BLANK command stops displaying (blanks) the active
display. The active display may be a channel, function, pixel
memory, or waveform memory. To blank a channel display use the
parameter CHANNEL 1|2, to blank a waveform memory display use
the parameter MEMORY 1 | ... | 8, and to blank a pixel memory
display use the parameter MEMORY 9 | 10.

Command Syntax: BLANk<display>

<display> ::= {{CHANnel 1 | 2} |
                {FUNCtion 1 | 2}|
                {MEMory 1 | ... 8} |
                {MEMory 9 | 10}}

Example: OUPUT 707;"BLANK CHANNEL1"

CALibrate

command/query

The CALibrate command sends a Cal string to the instrument. The Cal string consists of 660 8-bit bytes. These bytes contain the cal factors which are generated by the front panel cals.

The cal query "CAL?" sends the Cal string to the controller using the same format as required by the CAL command.

The skew calibration factors for the currently active configuration (i.e., 2 GSa/s On or 2GSa/s Off) will be included in the string. To fully save the cal factors it will be necessary to save a cal string for each configuration.

If the cal factors are extracted when the instrument is in the 2GSa/s On or Off mode then the instrument must be in the same mode when the factors are restored.

The cal string is check-summed and any alteration will prevent it from being accepted back into the instrument.

Command Syntax: CALibrate < cal string >
< cal string > :: = 660 8-bit bytes

Example: OUTPUT 707;"CAL"; < cal string >
< cal string > :: = 660 8-bit bytes

Query Syntax: CALibrate?

Returned Format: [CALibrate] < Cal String > < CR > < LF >

Example: DIM Cal$[1000]
OUTPUT 707;"EOI ON;HEADER OFF
OUTPUT 707;"CAL?"
ENTER 707 USING "-K"Cal$
PRINT 707 USING "K";Cal$

CALibrate

*The example above receives the calibration string then
immediately returns the same string to the oscilloscope.
Normally you would receive the string, store it until you required
that calibration again, then return it to the oscilloscope.*

CLEar   command

The CLEAR command clears the HP-IB and input/output buffers and
registers. The HP 54111D responds to the CLEAR command by:

1. Terminating all bus communications in process by setting the
instrument to untalk and unlisten.

2. Clearing all serial poll status bits.

3. Clearing the input and output buffers.

4. Clearing the error quere and key register.

5. Stoppping any measurement or acquisition processes except the
normal background acquire-display.

The HP-IB Selected Device Clear message is the preferred method of
performing steps 1 through 5 (refer to Chapter 12 for details).

Command Syntax: CLEar

Example: OUTPUT 707;"CLEAR"

DIGitize   **command**

The DIGITIZE command is used to acquire waveform data for transfer over the HP-IB. It causes an acquisition to take place on the specified channel(s) with the resulting data being stored in memory. The ACQUIRE subsystem commands are used to set up conditions such as TYPE, COMPLETION criteria, RESOLUTION, and the average COUNT for the next DIGITIZE command. See the ACQUIRE subsystem (chapter 3) for a description of these commands. To abort the DIGITIZE command, press LOCAL key on the front panel.

Digitize Source

The data sources for the DIGITIZE command are channel 1, channel 2, function 1, and function 2. When a DIGITIZE 1 command is sent, the channel 1 signal is captured and stored. When a DIGITIZE 2 command is sent, the channel 2 signal is captured and stored. When a DIGITIZE 5 command is sent, the function 1 signal is captured and stored. When a DIGITIZE 6 command is sent, the function 2 signal is captured and stored. These commands can be send in combination. For example: to digitize channel 1 and channel 2 a DIGITIZE 1,2 command is send.

Note

If an acquisition is in progress where a DIGITIZE command is send, the on going acquisition must complete before the DIGITIZE command is executed. To prevent this send a STOP before the DIGITIZE command.

Data Acquisition Types

There are five types of waveform acquisition that can be selected using the ACQUIRE subsystem and TYPE command. The five types are: FAST, NORMAL, FILTERED, AVERAGE, and VPERSISTENCE. The type of data acquisition selected and the source(s) specified for the DIGITIZE command will determine where the digitized data is placed.

For example, if ACQUIRE TYPE NORMAL and DIGITIZED 1 are specified, a data record of 8192 points is captured and stored into waveform memory 1. If DIGITIZE 2 is specified while TYPE NORMAL is selected, a data record of 8192 points is captured and stored into waveform memory 2.

Command Syntax: DIGitize [CHANnel]{1 | 2 | 5 | 6 | any combination of 1,2,5,6}

Example: OUTPUT 707;"DIGITIE CHANNEL 1,2"

TYPE NORMAL with either DIGITIZE 5 or 6 selecyted will digitize into a waveform record that is 501 points long. It is then processed into waveform memories 7 or 8, depending on the digitize source specified.

Refer to table 2-1 (page 2-15), for a condensed version of the relationships between the ACQUIRE TYPE, the DIGITIZE command, and where the acquired waveform data is placed.

Fast Data Type - When TYPE is selected in the ACQuire sybsystem, data will be processed directly from the channel buffers to internal fast buffers. DIG 1 will move da t a from channel 1 to fast buffer1. DIG 2 are allowed in FAST mode.

The data transfer from source to destination includes formatting as specified in the WAVeform subsystem. For example, if WAVeform SOURce FAST1 FORMat is BYTE when DIG 1 is executed, then data will be packed into the internal fast buffer in packer BYTe format.

In the FAST mode RESOlution is not applied to the data as it is transfered. In other words, filters are not applied in the fast mode. Raw data is moved as fast as possible while formatting.

Since the data in the FAST buffers may be in several different formats, it is not available for viewing, measuring, or post processing. The primary goal is to get data out of the 54111 as fast as possible. If viewing or processing of FAST data is required, then it must be sent back into the 54111 memories for viewing.

It is possible to change format between acquisition and data read but this would require reformating time. It is recommended that format be set as desired prior to DIGitize commands.

Normal Data Type - When the NORMAL data type is selected using the ACQUIRE subsystem, the two input channels' data and the two functions' (FUNCTION 1 and 2) data can be acquired using the system command DIGITIZE. With the NORMAL type selected and a DIGITIZE 1 command sent, the channel 1 waveform is acquired and processed into waveform memory 1. Sending a DIGITIZE 2 command acquires the channel 2 waveform and processes the data into waveform memory 2. If a DIGITIZE 5 is sent, the HP 54111D acquires the FUNCTION 1 data. If a DIGITIZE 6 is sent, the FUNCTION 2 data is acquired.

When a DIGITIZE 1 or 2 is used in the ACQUIRE TYPE NORMAL, the acquired waveform record is 8192 data points. In this mode an ACQUIRE RESOLUTION value of OFF, 6, 7, or 8 can be selected to change the vertical resolution of the acquired data. When a DIGITIZE 5 or 6 is used in the ACQUIRE TYPE NORMAL, the acquired waveform record is 501 data points and the vertical resolution can be selected.

Filtered Data Type - When the FILTERED data type is selected in the ACQUIRE subsystem, the data from the two input channels and the data from both functions can be acquired using the system command DIGITIZE. With the FILTERED type selected and a DIGITIZE 1 command sent, the channel 1 waveform is acquired and processed into waveform memory 5. Sending a DIGITIZE 2 command acquires the channel 2 waveform and processes the data in waveform memory 6. If a DIGITIZE 5 is sent, the instrument acquires the FUNCTION 1 data and processes it in waveform memory 7. Sending a DIGITIZE 6 acquires the FUNCTION 2 data and processes it in waveform memory 8.

When the ACQUIRE TYPE FILTERED is selected, all data acquired with the system command DIGITIZE is 501 data points. In this mode resolution can be selected. The major difference between

NORMAL and FILTERED is that NORMAL yields an 8k record processed through the OFF, 6, 7, or 8-bit filter only. The FILTERED mode yields a 501-point record of what is displayed on screen. This includes processing through the OFF, 6, 7, or 8-bit filter.

Average Data Type - When the AVERAGE data type is selected in the ACQUIRE subsystem, the data from the two input channels and the data from both functions can be acquired using the system command DIGITIZE. With the AVERAGE TYPE selected and a DIGITIZE 1 command sent, the channel 1 waveform is acquired and processed into waveform memory 5. Sending a DIGITIZE 2 command acquires the channel 2 waveform and processes the data in waveform memory 6. If a DIGITIZE 5 is sent, the instrument acquires the FUNCTION 1 data and processes it in waveform memory 7. Sending a DIGITIZE 6 acquires the FUNCTION 2 data and processes it in waveform memory 8.

When the ACQUIRE TYPE AVERAGE is selected, all data acquired with the system command DIGITIZE is 501 data points. In this mode the number of averages can be selected.

Vpersistence Data Type - When the VPERSISTENCE data type is selected, in the ACQUIRE subsystem, the data from the two input channels and the data from both functions can be acquired using the system command DIGITIZE. With the VPERSISTENCE type selected and a DIGITIZE 1 command sent, the channel 1 waveform is acquired and processed into waveform memory 5. Sending a DIGITIZE 2 command acquires the channel 2 waveform and processes the data in waveform memory 6. If a DIGITIZE 5 is sent, the instrument acquires the FUNCTION 1 data and processes it in waveform memory 7. Sending a DIGITIZE 6 acquires the FUNCTION 2 data and processes it in waveform memory 8.

When the ACQUIRE TYPE VPERSISTENCE is selected, all data acquired with the system command DIGITIZE is 501 data points. In this mode the persistence of the displayed can be selected.

Table 2-1 (page 2-15) shows the number of points acquired and where the digitized data is placed for each ACQUIRE TYPE and DIGITIZE command combination. This table also indicates where the data is acquired from for each of the DIGITIZE commands.

Refer to the ACQUIRE subsystem (chapter 3) for more information on data types.

Table 2-1. Digitized Waveform Locations and Points.

DIGITIZE command used	REAL TIME MODE			REPETITIVE MODE	
	FAST 8K record	NORMAL 8K record	FILTERED display data only	AVERAGE	VPERSISTENCE
DIGITIZE 1 acquires Channel 1	8192 points acquired stored in internal MEMORY FAST 1	8192 points acquired single shot, stored in MEMORY 1	501 points acquired single shot, stored in MEMORY 5	501 points acquired until complete, stored in MEMORY 5	
DIGITIZE 2 acquires Channel 2	8192 points acquired stored in internal MEMORY FAST 2	8192 points acquired single shot, stored in MEMORY 2	501 points acquired single shot, stored in MEMORY 6	501 points acquired until complete, stored in MEMORY 6	
DIGITIZE 5 acquires FUNCTION 1	illegal mode	501 points acquired until complete, stored in MEMORY 7			
DIGITIZE 6 acquires FUNCTION 2	illegal mode	501 points acquired until complete, stored in MEMORY 8			

DSP command/query

The DSP command writes a quoted string, not including quotes, to the advisory line (line 15) of the CRT.

The DSP query returns the string last written to the advisory line. This may be a string written with a DSP command or an internally generated advisory.

Command Syntax: DSP<quoted ASCII string>

Example: OUTPUT 707;"DSP""COLOR DISPLAY"""

Query Syntax: DSP?

Returned Format: [DSP]<string><CR><LF>

<string> ::= last information written on line 15

Example:
DIM Dsp$[40]
OUTPUT 707;"DSP?"
ENTER 707;Dsp$
PRINT Dsp$

EOI   **command/query**

The EOI command specifies whether or not the last byte of a reply
from the HP 54111D is to be sent with the EOI bus control line set
true or not true.

The EOI query returns the current state of EOI.

Command Syntax: EOI {ON | OFF}

Example: OUTPUT 707;"EOI OFF"

Query Syntax: EOI?

Returned Format: [EOI]{ON | OFF}<CR><LF>

Example: OUTPUT 707;"EOI?"
ENTER 707;Eoi$
PRINT Eoi$

ERASe   **command**

The ERASE command erases a specified pixel memory. Erasing
DISPlay is the same as pressing the CLEAR DISPLAY front panel
key. The ERASE DISPLAY command may also be used to clear
accumulations of data in either functions. If the scope is running
and being triggered and ERASe DISPlay is executed, the instrument
will momentarily stop acquiring data, clear the CRT, and then
continue with data acquisition.

Command Syntax: ERASe {DISPlay | MEMory9 | MEMory10}

Example: OUTPUT 707;"ERASE MEMORY 10"

ERRor?   query

The ERROR query outputs the next error number in the error queue
over HP-IB. This instrument has an error queue that is 16 errors
deep and operates on a first-in first-out basis. Successively sending
the query ERROR? returns the error numbers in the order that they
occurred until the queue is empty. Any further queries then return
0s until another error occurs. See table 2-2 (page 2-19) for the
ERROR numbers.

Query Syntax: ERRor?

Returned Format: [ERRor]<code><CR><LF>

<code> ::= integer value

Example: FOR I = 1 TO 16
    OUTPUT 707;"ERROR?"
    ENTER 707;Error
    PRINT Error
NEXT I

The example above reads and prints the entire contents of the error
queue. This example reads the error queue 16 times and prints the
contents.

Table 2-2. Error Messages.

ERROR NUMBER	DESCRIPTION
-100	Unknown command
-101	Invalid character received
-110	Command header error
-119	Command header expected
-120	Numeric argument error
-121	Numeric data expected
-123	Numeric overflow
-125	Numeric syntax error
-130	Non-numeric argument error
-131	Character data expected
-132	String data expected
-133	Block data (binary data) expected
-134	String length error
-135	Block length error
-136	Block data checksum error
-142	Too many arguments
-143	Argument delimiter error
-144	Message unit delimiter error
-149	Missing argument
-150	Query expected
-151	Query not allowed
-201	Command not executable in local mode
-202	Setting lost on power up
-211	Settings conflict
-212	Argument out of range
-222	Insufficient capability/configuration
-230	Transmission aborted
-231	Input buffer full or overflow
-233	Output buffer empty
-234	Measurement Error
-301	Interrupt fault
-302	System error
-311	RAM failure (hard error)
-312	RAM data loss (soft error)
-321	ROM checksum error
-340	Self test failed
-350	Timer error
-360	Analog hardware error
-370	Digital hardware error
-399	Power supply failure

Positive error numbers are reported after a self-test failed error (−340). These refer to the internal self test loops that failed to pass self test. Refer to the *HP 54111D Service Manual* for more information regarding self test loops.

HEADer   **command/query**

The HEADER command tells the instrument whether not to output a header for query responses. When HEADER is set to ON query responses will include the command header.

The HEADER query returns the state of the HEADER command.

Command Syntax: HEADer {ON | OFF}

Example: OUTPUT 707;"HEADER ON"

Query Syntax: HEADer?

Returned Format: [HEADer]{ON | OFF}<CR><LF>

Example: DIM Header$[10]
OUTPUT 707;"HEADER?"
ENTER 707;Header$
PRINT Header$

ID?   query

The ID query returns the instrument model number.

Query Syntax: ID?
Returned Format: [ID] HP54111D<CR> <LF>

Example: DIM Id$[10]
                       OUTPUT 707;"ID?"
                       ENTER 707;Id$
                       PRINT Id$

INTERleave   command/query

The INTERleave command tells the instrument to turn the 2 GSa/s
sample feature on or off. The commnad must be used in conjunction with
the HP 54114A Test Set.

The INTERleave data is not included in the learn string. When the
instrument is RESet, the INTERleave command resets to OFF.

Command Syntax: INTERleave {ON | OFF}
Example: OUTPUT 707;"INTERLEAVE ON"

Query Systax: INTERleave?

Returned Format: [INTERleave]{ON | OFF} < CR > < LF >

Example: DIM Interleave$[10]
                       OUTPUT 707;"INTERLEAVE?"
                       ENTER 707;Interleave$
                       PRINT Interleave$

KEY

The KEY command simulates the pressing of a specified front panel key. Key commands may be sent over the HP-IB in any order that is legal from the front panel. Use caution to ensure that the instrument is in the desired mode before executing the KEY command.

The KEY query returns the key code for the last key pressed from the front panel or the last simulated key press over the HP-IB. Key codes range from 1 to 63 with 0 representing no key (returned after power-up). Refer to table 2-3, page 2-23 for key codes.

Command Syntax: KEY<keycode>

<keycode> ::= 1 to 63

Example: OUTPUT 707;"KEY 48"

Query Syntax: KEY?

Returned Format: [key]<keycode><CR><LF>
<keycode> ::= 0 to 63
(integer - NR1 format)

Example: OUTPUT 707;"KEY?"
ENTER 707;Key
PRINT Key

Table 2-3. HP 54111D Front Panel Key Codes.

KEY	KEY CODE	KEY	KEY CODE
Menu Select 1	1	" – "(minus)	23
Menu Select 2	2	0	24
Menu Select 3	3	1	25
Menu Select 4	4	2	26
Menu Select 5	5	3	27
Menu Select 6	6	4	28
Menu Select 7	8	5	29
Menu Select 8	9	6	30
Function Select 1	15	7	31
Function Select 2	14	8	32
Function Select 3	13	9	33
Function Select 4	12	CLEAR DISPLAY	40
Function Select 5	11	RUN	41
Function Select 6	10	STOP/SINGLE	42
sec/Volt	16	SAVE	43
msec/mV	17	RECALL	44
μsec	18	LOCAL	45
nsec	19	AUTOSCALE	48
psec	20	↑ –	56
CLEAR	21	– ↓	63
"."(decimal pt.)	22	no key	0

Note

The Menu Select Keys are located at the bottom of the screen with Menu Select 1 at the lower left of the screen. The Function Select Keys are located at the right of the screen with Function Select 1 located at the upper right of the screen.

LOCAL   **command**

The LOCAL command returns the instrument to local (Front Panel) operation. The LOCAL command performs a similar operation to the Clear Lockout/Set Local HP-IB message. It is provided for controllers with limited HP-IB control capability. The HP-IB Clear Lockout/Set Local Message is the preferred method of switching the instrument from Remote to Local and clearing the Local Lockout. No short form of this command is provided.

Command Syntax: LOCAL

Example: OUTPUT 707;"LOCAL"

LONGform   **command/query**

The LONGFORM command sets the long form for the HP 54111D's responses to queries. If the LONGFORM command is set to OFF, command headers and alpha arguments are sent from the HP 54111D in the abbreviated form. If the LONGFORM command is set ON, the whole word will be output. This command does not affect the input data messages to the HP 54111D. Headers and arguments may be sent to the HP 54111D in either the long form or short form regardless of how the LONGFORM command is set.

The LONGFORM query returns the state of the LONGFORM command.

Command Syntax: LONGform {ON | OFF}

Example: OUTPUT 707;"LONG ON"

Query Syntax: LONGform?

Returned Format: [LONGform]{ON | OFF}<CR><LF>

Example:
DIM Long$[30]
OUTPUT 707;"LONGFORM?"
ENTER 707;Long$
PRINT Long$ MENU

MENU   **command/query**

The MENU command selects one of the 14 menus on the front panel.

The MENU query returns the current menu.

Command Syntax: MENU<number><CR><LF>
                     <number> ::= {1 to 14} where:

                              1 = Channel 1 menu
                              2 = Channel 2 menu
                              3 = Timebase menu
                              4 = Trigger menu
                              5 = Display menu
                              6 = Delta V menu
                              7 = Delta t menu
                              8 = Waveform Save menu
                              9 = Waveform Math menu
                             10 = Measurement menu
                             11 = Hardcopy menu
                             12 = Not used
                             13 = Not used
                             14 = Utility menu

Example: OUTPUT 707;"MENU 4"

Query Syntax: MENU?

Returned Format: [MENU] <menu #><CR><LF>
                      <menu #> ::= 1 to 14
                                         (integer - NR1 format)

Example: OUTPUT 707;"MENU?"
              ENTER 707;Menu
              PRINT Menu

MERGe   **command**

The MERGE command stores the displayed waveform into a specified pixel memory. The pixel memories are memory nine and ten.

Graticule markers and factors can not be stored.

Command Syntax: MERGe {MEMory9 | MEMory10}

Example: OUTPUT 707;"MERGe MEMory9"

PLOT

command

The PLOT command outputs a copy of the display to an HPGL compatible plotter as soon as the oscilloscope is addressed to talk.

The output includes the displayed waveforms, the graticule, time and voltage markers, trigger setup, and measurement results.

Command Syntax: PLOT

Example:	CLEAR 707	! Clear interface buffers.
	OUTPUT 707;"HARDCOPY PLOTTER"	! Select plotter as hardcopy device
	OUTPUT 707;"HARDCOPY PEN AUTOMATIC"	! Select per mode (see PEN)
		! Command, section 7).
	OUTPUT 707;"PLOT"	! Command to plot
	SEND 7;UNL UNT	! Unaddress bus, assrt ATN line
	SEND 7;LISTEN 5	! Address printer @ 705 to listen
	SEND 7;TALK 7	! Address oscilloscope to @ 707 to talk
	SEND 7;DATA	! Negates ATN live to allow data transfer
	WAIT 30	! Wait for data transfer to complete
	END	

Note

When programming the HP 54111D you should use the SRQ (Service Request) capabilities to determine if the hardcopy transfer is complete. Attempting to program the instrument while making a hardcopy transfer will cause errors.

PRINt

command

The PRINT command outputs a copy of the display when the oscilloscope is addressed to talk.

The output includes the displayed waveforms, the graticule, time and voltage markers, trigger setup, and measurement results.

Any HP GRAPHICS PRINTER may be used as the Print Hardcopy device.

Command Syntax: PRINt

Example:

CLEAR 707	!Clear interface buffers
OUTPUT 707;"HARDCOY PRINTER MONO"	!Select black and white
	!Printer as hardcopy device
OUTPUT 707;"HARDCOPY SOURCE PMEMO"	!Selects screen as info to be printed
OUTPUT 707;"HARDCOPY PAGE AUTOMATIC"	!Sends a form feed after print is
	!complete
OUTPUT 707;"PRINT"	!Command to print
SEND 7;UNL UNT	!Unaddress bus, asserts ATN line
SEND 7;LISTEN 1	!Address printer @701 to listen
SEND 7;TALK 7	!Address oscilloscope @707 to talk
SEND 7;DATA	!Negates ATN line to allow
	!data transfer
WAIT 30	!Wait for data transfer to complete
END	

Note

When you are programming the HP 54111D you should use the SRQ (Service Request) capabilities to determine if the transfer is complete. Attempting to program this instrument while making a hardcopy will cause errors.

READy? | RDY?   **query**

The READY query returns the Ready Byte (the upper byte of the status word). The value returned for the ready byte is the added weights of all bits that are set (true) in the byte.

Refer to table 2-4 for a description of the bits in the Ready Byte.

Query Syntax: {READy | RDY}?

Returned Format: [READy]<ready byte><CR><LF>

                <ready byte> ::= 0 - 255
                          (integer - NR1 format)

Example: OUTPUT 707;"READY?"
            ENTER 707;Ready
            PRINT Ready

Table 2-4. The Ready Byte.

BIT	BIT WEIGHT	BIT NAME	CONDITION
7	128	—	Not used, always 0
6	64	Cal	0 = calibration in progress 1 = calibration not in progress
5	32	Test	0 = self test in progress 1 = self test not in progress
4	16	Hard	0 = hardcopy output in progress 1 = hardcopy output not in progress
3	8	Data	0 = data buffer empty 1 = data in buffer to be read
2	4	Acq	0 = waveform acquisition in progress 1 = waveform acquisition not in progress
1	2	Trig	0 = instrument is not receiving triggers 1 = instrument is receiving triggers
0	1	Parse	0 = command is being parsed 1 = command not being parsed

Note

Most of the bits in the ready byte are normally high, and only go low while an instrument operation is taking place.

RECall   **command**

The RECALL command recalls an instrument setup and color settings from a specified save-recall register (0-9).

There is no query form of this command.

Command Syntax: RECall[REGISTER]<register number>

<register number> ::= 0 - 9

Example: OUTPUT 707;"RECALL 2"

REMote   **command**

The REMOTE command sets the instrument to the remote mode and sets the local lockout. When local lockout is set, the instrument cannot be set to local using the front panel LOCAL key. This command is provided for use by controllers that have a limited HP-IB control capability. The REMOTE command performs a similar operation as an HP-IB Remote message followed by an HP-IB Local Lockout message. The HP-IB Remote and Local Lockout messages are the preferred method of switching the instrument from local to remote and setting local lockout. If the HP-IB REN (remote enable) control line is false, the REMOTE command will have no effect.

Command Syntax: REMote

Example: OUTPUT 707;"REMote"

                                                         command/query

The REQUEST command sends an integer representing the mask weight values of the bits in the request mask.

The request mask determines what conditions are allowed to produce an HP-IB service request. A service request will be produced if any of the bits in the Status Byte change and bit 6 is enabled. The upper half of the request mask determines which bits of the Ready Byte are allowed to change the Ready Bit in the Status Byte. If the Ready Bit is allowed to change and it is not masked, it can then cause a service request to be generated.

Setting (unmasking) a bit in the Ready Byte (upper half of the Status Word) will not, by itself, cause a service request to be produced.

The service request is a special HP-IB function that allows the instrument to flag the controller when a requested operation has taken place. Some controllers may not support the service request (Serial Poll) function.

Another form of this command allows you to follow the REQUEST command with ON or OFF. This command enables or disables the ability of the HP 54111D to generate the require service message without changing the request mask. Any unmasked conditions that occur with REQUEST OFF will be saved until the REQUEST ON command is received. At that time, unmasked conditions that occurred before and after the REQUEST ON command will generate the require service message.

Refer to table 2-5, page 2-35 for mask bit weights and what they mask. A high (1) will enable the selected condition. To select more than one condition, send the total mask weight of all selected conditions.

Refer to "Service Request" and "Instrument Status" in Chapter 1 of this manual for more information.

Command Syntax: {REQuest | RQS}{0N | OFF | <mask>}

<mask> ::= 0 through 65535

Example: OUTPUT 707;"REQUEST 36"

Query Syntax: {REQuest | RQS}?

Returned Format: [REQuest]<mask><CR><LF>
  <mask> ::= 0 through 65535
    (integer - NR1 format)

Example: OUTPUT 707;"REQUEST?"
  ENTER 707;Request
  PRINT Request

Table 2-5. The Request Mask (see page 2-48 also).

LOWER BYTE									
Request Mask Bit	0	1	2	3	4	5	6	7	Status Byte Mask
Mask Weight	1	2	4	8	16	32	64	128	(High(1) - enables Service Request)
What It Masks	RQC	PWR	FPS	LCL	RDY	ERR	RQS	MSG	
UPPER BYTE									
Request Mask Bit	8	9	10	11	12	13	14	15	Ready Byte Mask
Mask Weight	256	512	1024	2048	4096	8192	16384	32768	(High(1) - enables Ready Bit)
What It Masks	Parse	Trig	Acq	Data	Hard	Test	Cal	Trig	

RESet | RST   **command**

The RESET command presets the instrument to its default settings. These settings are the same as those established during a key-down power-up. See tables 2-6 and 2-7 for the reset conditions.

Command Syntax: {RESet | RST}

Example: OUTPUT 707;"RST"

Table 2-6. Reset Conditions for the HP 54111D.

Channel 1/Channel 2 Display	— On
Channel 1/Channel 2 VOLTS/DIV	— 2.00 volts/div
Channel 1/Channel 2 OFFSET	— 0.00 volts
Channel 1/Channel 2 Preset	— None
Channel 1/Channel 2 Input Coupling	— dc
Channel 1/Channel 2 Input Impedance	— 1 MΩ
TIME/DIV	— 1 µs/div
DELAY	— 0.00 s
Delay Ref at	— Center
Auto/Triggered Sweep	— Auto
Trigger Mode Edge	— Edge
Trigger Source	— Channel 1
TRIGGER LEVEL (all sources)	— 0.00 volts
Trigger Slope (all sources)	— Positive
Holdoff (Edge mode)	— Time
HOLDOFF TIME (Edge mode)	— 70.000 ns
HOLDOFF EVENTS (Edge mode)	— 2 Events
Trigger Pattern (Pattern mode)	— Channel 1 - High
	— Channel 2 - Don't Care
	— Trigger 3 - Don't Care
	— Trigger 4 - Don't Care
When Entered/Exited (Pattern mode)	— Entered
Holdoff (Pattern mode)	— Time
HOLDOFF TIME (Pattern mode)	— 70.000 ns
HOLDOFF EVENTS (Pattern mode)	— 2 Events
Trigger On Edge (State mode)	— Positive
Edge Source (State mode)	— Channel 1
Pattern (State mode)	— Ch1 Trig = Clock
	— Channel 2 - Don't Care
	— Trigger 3 - Don't Care
	— Trigger 4 - Don't Care
Is Present/Not Present (State mode)	— Present
Holdoff (State mode)	— TIME
HOLDOFF TIME (State mode)	— 70.000 ns

Table 2-6. Reset Conditions for the HP 54111D (continued).

Arming Slope (Time mode)	— Negative
Arming Source (Time mode)	— Channel 1
DELAY (Time mode)	— 20.00 ns
Trigger Slope (Time mode)	— Positive
Trigger Channel (Time mode)	— Channel 1
Arming Slope (Events mode)	— Negative
Arming Source (Events mode)	— Channel 1
Number of Events (Events mode)	— 1
Trigger Slope (Events mode)	— Positive
Trigger Source (Events mode)	— Channel 1
Display Mode	— Repetitive
Averaging (Repetitive mode)	— On
NUMBER OF AVERAGES (Repetitive mode)	— 2
Screen	— Dual
Graticule	— Axes
Resolution (Real Time mode)	— 6 bits
Completion Criteria (for HP-IB DIGITIZE command)	— 100%
Voltage Markers	— Off
V Marker 1 Position	— 2.0 volts
V Marker 2 Position	— 2.0 volts
Preset Levels	— 10 - 90%
Time Markers	— Off
Start Marker Position	— +2.5000 μs
Stop Marker Position	— −2.5000 μs
Start Marker Edge Slope	— Positive
Start Marker Number	— 1
Stop Marker Edge Slope	— Negative
Stop Marker Number	— 1
Waveform Memories	— Off
Source for Store	— Channel 1
Selected Memory	— Waveform Memory 5
Waveform Data	— 0 volts
Memory Scaling	
Time/division	— 500 ns/div
Volts/division	— 2.00 v/div

Table 2-6. Reset Conditions for the HP 54111D (continued).

Function 1 and 2	— Off
Functions definition	— Chan 1 - Chan 2
Measure Source	— Channel 1

Table 2-7. HP-IB Reset Conditions for the HP 54111D.

Service Request Mode	— Disabled (RQS OFF)
Service Request Mask	— Decimal 32546 (bits 1, 5, 8-14 set)
Serial Poll Status Byte	— Clear
Error Queue	— Empty
WAVeform Format	— Word
EOI	— On
Longform	— Off
Header	— Off

REVision? query

The REVISION query returns an integer corresponding to the revision date of the internal firmware.

Query Syntax: REVision?

Returned Format: [REVision]<date>

<date> ::= four digit software date
(integer - NR1 format)

Example: OUTPUT 707;"REV?"
ENTER 707;Rev
PRINT Rev

RUN   **command**

The RUN command acquires data for the active waveform display. The data is acquired in the manner defined by the Timebase mode. If the Timebase mode is in SINGLE, the RUN command enables the trigger once and displays the acquired data on the CRT. This also occurs when the front panel STOP/SINGLE key is pressed when the instrument is STOPPED.

If the Timebase mode is AUTO or TRIGGERED, the RUN command will enable the trigger repeatedly and display the data it acquires continuously on the CRT. This is the same thing that happens when the front panel RUN key is pressed. See the Timebase subsystem MODE command (Chapter 9) for a description of the various modes.

Command Syntax: RUN

Example: OUTPUT 707;"RUN"

SAVE

command

The SAVE command saves the current setup and color settings in the specified save/recall register (0-9). Its action is the same as performing a SAVE operation from the front panel.

To recall a saved setup use the RECALL command.

Command Syntax: SAVE[REGISTER]<register number>

<register number> ::= 0 - 9

Example: OUTPUT 707;"SAVE 1"

SERial?

query

The SERIAL query returns the instrument serial number as a quoted string.

Query Syntax: SERial?

Returned Format: [SERial]<serial number>

<serial number> ::= 10 character quoted string

Example: DIM SER$[30]
OUTPUT 707:"SER?"
ENTER 707; Ser$
PRINT Ser$

SETup   **command/query**

The SETUP command sets the HP 54111D as defined by the data contained in the learn string sent from the controller.

The query outputs the current HP 54111D setup in the form of a learn string to the controller. The learn string (296 bytes) is sent and received as a binary block of data. The format for the data transmission is the #A format defined in IEEE 488 specification.

Refer to "Data Message" and "Learn and Cal Strings" in Chapter 1 of this manual.

Command Syntax: SETup

Example: OUTPUT 707;"SETUP";<setup>

<setup> ::= setup information in the #A format

Query Syntax: SETup?

Returned Format: [SETup]<setup>

<setup> ::= setup data in the #A format

Example: DIM Set$[302]
OUTPUT 707;"HEADER OFF EOI ON"
OUTPUT 707;"SETUP?"
ENTER 707 USING "-K";Set$
OUTPUT 707;"SETUP ";Set$

Note

The logical order for this instruction is to send the query first followed by the command. The query causes the learn string to be sent to the controller and the command causes the learn string to be returned to the HP 54111D.

SPOLl? | STB?   query

The SPOLL query returns the Status Byte (the lower byte of the Status Word). The value returned is an integer that is a sum of the bit weights of all bits that are true (1). All bits in the Status Byte returned by this query are dynamic and reflect the state of the instrument at the time of the query. This command is provided for use by controllers that have a limited HP-IB control capability. Using the HP-IB Serial Poll is the preferred method of reading the Status Byte.

Refer to table 2-8, page 2-46 for details on the Status Byte.

Refer to "Instrument Status" in Chapter 1 of this manual for more information about HP-IB Serial Poll.

Query Syntax: {SPOLl | STB}?

Returned Format: [STB]<status byte>

        <status byte> ::= 0 through 255
                      (integer - NR1 format)

Example: OUTPUT 707;"STB?"
           ENTER 707;Stb
           PRINT Stb

Table 2-8. The Status Byte.

BIT	BIT WEIGHT	BIT NAME	CONDITION
7	128	MSG	0 = no message has been received 1 = message has been received
6	64	RQS	0 = instrument is not requesting service 1 = instrument is requesting service
5	32	ERR	0 = no error has occurred 1 = error has occurred
4	16	RDY	0 = not ready (instrument is busy) 1 = instrument ready
3	8	LCL	1 = local switch pressed or power cycle (transitions high then back low)
2	4	FPS	0 = no front panel key has been pressed 1 = front panel key has been pressed
1	2	PWR	NOT used, always 0
0	1	RQC	Request control, NOT used, always 0

STAtus   **query**

The STATUS query returns the instrument Status Word. The instrument Status Word is a 16-bit word which is returned as an integer, and contains information about the instrument status. The upper 8 bits of the Status Word are known collectively as the Ready Byte, and the lower 8 bits are the Status Byte. The STATUS query is used to read the Status Word representing the current status of the HP 54111D. Unlike the response to serial poll, the conditions are dynamic, not latched. Therefore the status response reflects the current status. Refer to table 2-9 for a definition and weights of the bits in the Status Word.

Refer to "Instrument Status" in Chapter 1 of this manual.

Query Syntax: STATus?

Returned Format: [STATus]<status word>

<status word> ::= 0 to 65535
(integer · NR1 format)

Example: OUTPUT 707;"STATUS?"
ENTER 707;Stat
PRINT Stat

Table 1-3. The Status Word.

BIT	BIT WEIGHT	BIT NAME	CONDITION
15	32768	Arm	0 = trigger not armed 1 = trigger armed
14	16384	Cal	0 = calibration in progress 1 = calibration not in progress
13	8192	Test	0 = self test in progress 1 = self test not in progress
12	4096	Hard	0 = hardcopy output in progress 1 = hardcopy output not in progress
11	2048	Data	0 = data buffer empty 1 = data in buffer to be read
10	1024	Acq	0 = waveform acquisition in progress 1 = waveform acquisition not in progress
9	512	Trig	0 = instrument is not receiving triggers 1 = instrument is receiving triggers
8	256	Parse	0 = command is being parsed 1 = command is not being parsed
7	128	MSG	0 = no message has been displayed 1 = message has been displayed
6	64	RQS	0 = instrument is not requesting service 1 = instrument is requesting service
5	32	ERR	0 = no error has occurred 1 = error has occurred
4	16	RDY	0 = not ready (instrument is busy) 1 = instrument ready
3	8	LCL	1 = local switch pressed or power cycle (transitions high then back low)
2	4	FPS	0 = front panel key has not been pressed 1 = front panel key has been pressed
1	2	PWR	Power, NOT used, always 1
0	1	RQC	Request control, not used, always 0

0 = False = Low   1 = True = High

STOP command

The STOP command causes the instrument to stop acquiring data for the active display. The RUN command must be executed in order to restart the acquisition.

Command Syntax: STOP

Example: OUTPUT 707;"STOP"

STORe   command

The STORE command moves a stored waveform from one place to another internal to the instrument. This command has two parameters. The first is the source of the waveform which can be Channel 1 or 2, Function 1 or 2, or Memory 1 through 8. The second parameter is the destination of the waveform which can be Memory 1 through 8. Only real time data can be stored in Memories 1 through 4 and only repetitive data and functions can be stored in Memories 5 through 8.

Command Syntax: STORe<source>,<destination>

<source> ::= {CHANnel 1 | CHANnel 2 | FUNCtion 1 | FUNCtion 2 | MEMory {1 | ... | 8}},

<destination> ::= MEMory {1 | ... | 8}

Example: OUTPUT 707;"STORE CHANNEL2,MEMORY4"

Table 2-10. Waveform Sources and Destinations.

Sources	Destinations-Memory	
	Real-time Mode	Repetitive Mode
Channel 1,2 Function 1, 2	1 2 3 4 5 6 7 8	5 6 7 8 5 6 7 8
Memory 1, 2, 3, 4 Memory 5, 6, 7, 8	1 2 3 4 5 6 7 8	1 2 3 4 5 6 7 8

Note

The math function VESUS can not be used as a source for memory.

TEST | TST   **command**

The TEST command performs a self-test. This is the same test that is executed when the instrument is powered up. The Test bit in the Status Word (bit 13) will go to a 1 when the test is complete. If the Test bit in the Status Word and Ready Byte is already a 1, it will go to 0 while the test is in progress, then return to a 1.

Command Syntax: {TEST | TST}

Example: OUTPUT 707;"TEST"

TRG | GET   **command**

The instrument responds to this command in the same way it responds to the RUN system command and the GET HP-IB command.

The TRG command acquires data for the active waveform display based on the Timebase mode. If the Timebase mode is in SINGLE, the TRG command will cause the instrument to enable the trigger once, and display this data on the CRT. This is the same thing that happens when you press the front panel STOP/SINGLE key when the instrument is STOPPED.

If the Timebase mode is AUTO or TRIGGERED, the TRG command will cause the instrument to enable the trigger repeatedly and display the acquired data on the CRT. This is the same thing that happens when the front panel RUN key is pressed. See the Timebase subsystem MODE (Chapter 9) command for a description of the various modes.

Command Syntax: {TRG | GET}

Example: OUTPUT 707;"TRG"

VIEW   **command**

The VIEW command causes the instrument to turn on (start displaying) an active channel, function, pixel memory or waveform memory. If you want to turn on an active display use the parameter Channel {1 | 2}. If you want to turn on a pixel memory use the parameter MEMory {9 | 10}. Using the VIEW MEMory {1 | ... | 8 } command in the split screen mode causes odd numbered memories to be displayed on the upper screen and even numbered memories to be displayed on the lower.

Command Syntax: VIEW {CHANnel {1 | 2} | FUNCtion {1 | 2} |
                                            MEMory {1 | ... | 10}}

Example: OUTPUT 707;"VIEW CHANNEL 1"

3 Acquire Subsystem

Introduction

The ACQUIRE subsystem commands set uyp conditions that are used when a DIGITIZE system command is executed. This subsystem selects the type of data, the completion criteria for data acquisition, the number of averages, and the resolution value. See figure 3-1 for the ACQUIRE subsystem syntax diagrams.

The ACQUIRE subsystem is also the only HP-IB control for the Display parameters: Display Mode, Resolution, and Count. When the HP-IB parameters for ACQUIRE TYPE, RESOLUTION, or COUNT are changed, the front panel will also change. Conversely, when the front panel parameters are changed, the HP-IB parameters will change. The correspondence is described below.

Repetitive Mode with Averaging OFF (Persistence Mode)

ACQUIRE TYPE VPERSISTENCE - This sets the HP 54111D to the variable persistence mode. On the front panel this mode is selected by turning Averaging OFF. The persistence parameter is setable in the Display subsystem only and applies when the variable persistence mode is entered.

RESOLUTION - can be set in this mode, but has no impact on the current display mode or HP-IB acquisition.

COUNT - can be set in this mode, but has no impact on the current display mode or HP-IB acquisition.

COMPLETE - can be set from the HP-IB only and is used by the acquisition software to determine when a DIGITIZE channel (DIGITIZE 1 or 2) or DIGITIZE function (DIGITIZE 5 or 6) acquisition is complete. In the case of ACQUISITION TYPE VPERSISTENCE, the acquisition is complete when the percent of points in the 501-point accumulation record has the specified percent of data hits.

Front Panel Control - the front panel user activates the equivalent mode by selecting the Display menu, then setting the Display Mode to Repetitive, and finally setting Averaging to Off. Variable persistence is set with the DISPLAY TIME control. COUNT, RESOLUTION, and COMPLETE cannot be set from the front panel in this mode.

Repetitive Mode with Averaging ON (Averaging Mode)

ACQUIRE TYPE AVERAGE - This mode sets the HP 54111D to the repetitive mode with Averaging ON.

RESOLUTION - can be set in this mode, but has no impact on the current display mode or HP-IB acquisition.

COUNT - can be set in the ACQUIRE TYPE AVERAGE mode. It determines the number of averages that must be acquired.

COMPLETE - can be set in the ACQUIRE TYPE AVERAGE mode. It is used by the acquisition software to determine when a DIGITIZE channel (DIGITIZE 1 or 2) or a DIGITIZE function (DIGITIZE 5 or 6) acquisition is complete. In the ACQUIRE TYPE AVERAGE mode, the acquisition is complete when the percent of points specified in COMPLETION is the specified COUNT or average value.

Front Panel Control - to activate the Averaging mode from the front panel select the Display menu, then set the Display Mode to Repetitive, and finally set Averaging to On. This is the equivalent mode to the HP-IB AVERAGE mode. The NUMBER OF AVERAGES sets the COUNT value. The RESOLUTION and COMPLETE values cannot be set from the front panel.

Real Time (FILTERED Mode)

ACQUIRE TYPE FILTERED - When ACQUIRE TYPE FILTERED is selected, it sets the HP 54111D to the real time display mode. The purpose of this mode is to allow the HP-IB interface to have access to any possible displayed data. The FILTERED mode allows access to interpolated data, filtered data, or function data in the real time mode. In the event of a DIGITIZE command, the data is processed and placed into Memories 5 through 8. This is basically a "what you see on screen is what you get in the memory record."

RESOLUTION - can be set in this mode to OFF, 6, 7, or 8. When set to OFF, no filtering or interpolation takes place. When set to 6, 7 or 8, the appropriate filter is applied (see Chapter 5 and Appendix A for more information).

COUNT - can be set in this mode, but will not affect the display or the HP-IB acquisition.

COMPLETE - can be set from the HP-IB only. The COMPLETE value is used by the acquisition software to determine when a DIGITIZE function is complete. The function execution and acquisition (DIGITIZE 5 or 6) loops until the COMPLETE percentage of the function's 501-point buffer has data. Channel acquisition (DIGITIZE 1 or DIGITIZE 2) is not dependent on COMPLETE in this mode (single shot real time) unless RESOLUTION is OFF and the sweep speed is faster than 50 ns/div.

Front Panel Control - this mode cannot be activated from the front panel. When Display Mode Real Time is selected from the front panel the HP 54111D is always put into the ACQUIRE TYPE NORMAL mode. When either ACQUIRE TYPE FILTERED or ACQUIRE TYPE NORMAL is selected over the HP-IB, the front panel is set to Display Mode Real Time. Therefore no difference can be seen between NORMAL and FILTERED when observing the front panel.

Real Time (Fast Mode) The **ACQUIRE TYPE FAST** command will set the 54111D to the real time display mode. The purpose of this mode is to allow the HP-IB access to unprocessed acquisition data with speed as a primary objective. This will allow the fastest possible mode for acquiring data and transfering it to an external controller.

In the event of a DIGitize 1 or DIGitize 2 command, 8192 unprocessed data points are acquired and transfered into the FAST1 or FAST2 buffers respectively. The data is acquired and then formatted according to the format specification in the respective WAVeform subsystem. The data is captured and formatted as it is processed into the FAST buffers. For optimum speed, it is important that the required HP-IB format be set prior to data capture with the DIGitize command. (See Waveforms Subsystems)

Note

Waveform format BYTE will provide the fastest possible acquisition and transfer since the HP-IB data bus is 8 bits wide and 54111D's maximum resolution is 8 bits.

Data will NOT write over memories (1.4). This may be a desired mode for some users even though speed is not the prime objective.

RESOLUTION - can be set in this mode, but has no affect on data acquisition. Data is not processed.

COUNT - can be set, but has no effect on the FAST mode acquisition.

Front Panel Control - this mode cannot be activated from the front panel. When Display Mode Real Time is selected from the front panel the HP 54111D is always put into the ACQUIRE TYPE NORMAL mode. When either ACQUIRE TYPE FILTERED, FAST or ACQUIRE TYPE NORMAL is selected over the HP-IB, the front panel is set to Display Mode Real Time. Therefore no difference can be seen between NORMAL and FILTERED FAST when observing the front panel.

Real Time (NORMAL mode) **Acquire Type Normal** - The ACQUIRE TYPE NORMAL command will set the HP 54111D to the real time display mode. The purpose of this mode is to allow the HP-IB access to long (8192 point) real time data. In the event of a DIGITIZE 1 or 2 command, 8192 points of data is acquired and processed through the RESOLUTION filter and put into Memory 1 or 2. Channel 1 data (DIGITIZE 1) will be placed into Memory 1 and Channel 2 data (DIGITIZE 2) will be placed into Memory 2. This allows the HP-IB access to 8192 points of filtered or "raw" data.

In the event of a DIGITIZE 5 or 6 command, Function 1 or Function 2 is executed and the data is processed into the function buffers until a COMPLETE percentage of the data cells have data. These data records are 501 points long. A DIGITIZE 5 command executes Function 1 and places the data in Memory 7, and a DIGITIZE 6 command executes Function 2 and places the data in Memory 8.

RESOLUTION - can be set in the ACQUIRE TYPE NORMAL mode to OFF, 6, 7, or 8. The RESOLUTION selection will cause the appropriate filter to be used to process the data.

COUNT - can be set in this mode but will not affect the display or the data acquisition.

COMPLETE - can be set from the HP-IB only. The COMPLETE value is used by the acquisition software to determine when a DIGITIZE function is complete. The acquisition and function execution will loop until the COMPLETE percent of the function's buffer has data. Channel acquisition (DIGITIZE 1 or DIGITIZE 2) is not dependent on COMPLETE in this mode.

Subsystem Selection

The ACQUIRE subsystem must be selected prior to sending any ACQUIRE subsystem commands or queries. This can be done by sending a single line command "ACQUIRE" or by sending the subsystem selector prior to another command or query. As an example "ACQUIRE COMPLETE 100" will select the ACQUIRE subsystem and set the completion criteria to 100%.

System Command

A system command can be sent at any time while in this subsystem. After a system command has been sent, the instrument will return to the subsystem that was selected (in this case ACQUIRE) prior to the system command.

Sending Queries

In this subsystem if a query is sent followed by another query without a controller read (ENTER) between the queries, only the last response will be received. Although this operation is syntactically correct, the HP 54111D does not queue query responses in this subsystem. Therefore, the response to the first query will be overwritten in the HP-IB output buffer.

Note

Normally, only unprocessed data is stored in Memories 1-4 and then viewed, as the user requests, through the filters (OFF, 6, 7, 8). When the DIGITIZE command is used to filter data into Memories 1-4, the filter is invoked and data is processed as it is placed in memory. This differs from front-panel operation. To view data as it is acquired, turn RESOLUTION to OFF.

Figure 3-1. Acquire Subsystem Syntax Diagrams.

S4111/8X13
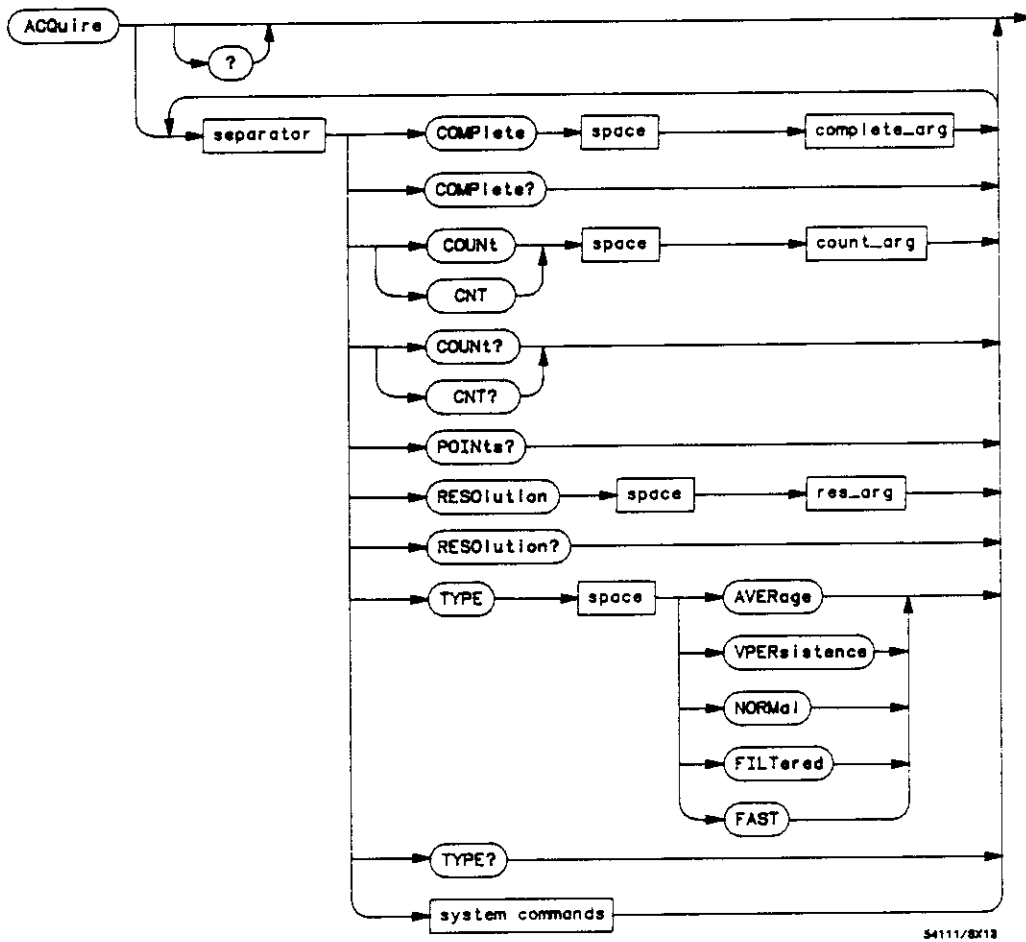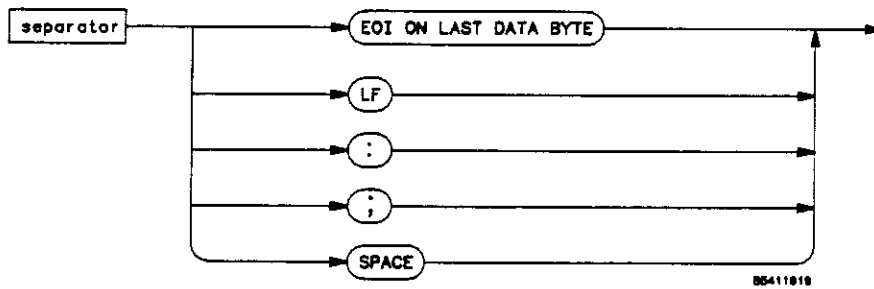
COMPLETE_ARG :: = An integer, 0 through 100, specifying, in percent, the number of time points that that must contain data before the acquisition is considered complete.

COUNT_ARG :: = An integer, 1 through 64, specifying the number of averages for each time point when in the averaged mode.

RES_ARG :: = An ASCII character (6, 7, 8) or string ("OFF") specifying the bits of vertical resolution.

Figure 3-1. Acquire Subsystem Syntax Diagrams (continued).

ACQuire command/query

The ACQUIRE command selects the acquire subsystem as the destination for the rest of the ACQUIRE subsystem commands.

The ACQUIRE query responds with the settings of the acquire subsystem. Refer to the individual ACQUIRE commands for arguments and range values.

Command Syntax: ACQuire

Example: OUTPUT 707;"ACQUIRE"

Query Syntax: ACQuire?

Returned Format:
[ACQuire <CR><LF>]
[TYPE]<argument><CR><LF>
[POINts]<NR1><CR><LF>
[COUNts]<NR1><CR><LF>
[COMPlete]<NR1><CR><LF>
[RESOlution]<argument><CR><LF>

Example:
DIM Acquire$[70]
OUTPUT 707;"EOI ON"
OUTPUT 707;"ACQUIRE?"
ENTER 707 USING " – K";Acquire$
PRINT USING "K";Acquire$

Note

For the example above, EOI (Display subsystem) must be ON.

The entire subsystem may be sent back to the HP 54111D by: OUTPUT 707; Acq$(*)

The COMPLETE command specifies the completion criteria for an acquisition. The parameter determines what percentage of the data points need to be "full" before an acquisition is considered complete.

In the ACQUIRE TYPE VPERSISTENCE mode, both channel and function data will be acquired until the specified COMPLETE percentage is reached. This means that the specified percentage of time points must contain data for the acquisition to be complete.

In the ACQUIRE TYPE AVERAGE mode, both channel and function data will be acquired until the number of points that have reached Average Count satisfies the COMPLETE value.

In the ACQUIRE TYPE FILTERED and ACQUIRE TYPE NORMAL modes, channel data (DIGITIZE 1 or 2) acquisition is not affected by the COMPLETE value unless resolution is off. In the case of a DIGITIZE 5 or 6 command, function execution and acquisition will loop until the COMPLETE percentage of the 501-point function buffer contains data.

The parameter for this command has a range of 0 to 100 and indicates the percentage of data points that must have the required number of data values. This number of data points must contain data before the acquisition is considered complete. When the completion criteria is set to 0, only a single acquisition cycle will be performed.

Command Syntax: COMPlete <comp>

                           <comp> ::= 0 to 100 percent

Example: OUTPUT 707;"COMPLETE 85"

Query Syntax: COMPlete?

Returned Format: [COMPlete] <comp><CR><LF>

                           <comp> ::= 0 to 100 percent

                                       (integer - NR1 format)

Example: OUTPUT 707;"COMPLETE?"
ENTER 707;Comp
PRINT Comp

COUNt |CNT

The COUNT command specifies the number of values to be averaged for each data point before the acquisition is considered complete for that point. The COUNT command is only used when the ACQUIRE TYPE is set to AVERAGE. This command has no effect if the TYPE is NORMAL, VPERSISTENCE, or FILTERED. However, it can be set in any mode.

The COUNT query returns the last specified count value. The COUNT parameter is an integer from 1 through 64.

Command Syntax: {COUNt | CNT} <count>

<count> ::= 1 through 64

Example: OUTPUT 707;"COUNT 25"

Query Syntax: {COUNt | CNT}?

Returned Format: [COUNt | CNT]<count><CR><LF>

<count> ::= 1 through 64
(integer - NR1 format)

Example: OUTPUT 707;"CNT?"
ENTER 707;Cnt
PRINT Cnt

POINTs? query

The POINTS query returns the number of points acquired. When the ACQUIRE TYPE is set to VPERSISTANCE, AVERAGE, or FILTERED the number of points acquired is 501. When the ACQUIRE TYPE is set to NORMAL and FAST and a DIGITIZE 1 or 2 command occurs, the number of points is 501.

Query Syntax: POINTs?

Returned Format: [POINTs]<number>

<number>::=501 or 8192

(integer - NR1 format)

Example: OUTPUT 707;"POINTS?"
ENTER 707;Points
PRINT Points

Note

POINTS may be sent as a command to the HP 54111D. This provides subsystem compatibility with other HP 541XX digitizing oscilloscopes. The actual number of points can be set only by changing ACQUIRE TYPE.

The RESOLUTION command allows selection of OFF, 6, 7, or 8 bit filtering. The RESOLUTION value can be set in any TYPE selection, however it has no effect in the Repetitive acquisition mode (TYPE VPERSISTENCE or TYPE AVERAGE).

In the Real Time acquisition mode, the RESOLUTION value selected determines how the data is processed during the acquisition. The relationships between ACQUIRE TYPE, RESOLUTION, and the acquired data are as follows:
- If TYPE FILTERED is selected and RESOLUTION is OFF then:
  - Channel 1 or 2 data (DIGITIZE 1 or 2) is acquired repetitively until the completion criteria is met. At sweep speeds of 50 ns/div and slower, a single acquisition is 100% complete.
  - Acquired data is placed in Memory 5 (channel 1) or Memory 6 (channel 2).
  - Function 1 or 2 data (DIGITIZE 5 or 6) is acquired repetitively until the completion criteria is met. Acquired data is placed in Memory 7 or 8.

- If TYPE NORMAL is selected and RESOLUTION is OFF then:
  - Channel 1 or 2 data (DIGITIZE 1 or 2) is acquired in a single acquisition to capture 8192 data points.
  - Data is not filtered, "raw" data is acquired.
  - Acquired data is placed in Memory 1 (channel 1) or Memory 2 (channel 2).
  - Function 1 or 2 data (DIGITIZE 5 or 6) is acquired repetitively until the completion criteria is met. Acquired data is placed in Memory 7 or 8.

- If TYPE NORMAL is selected and RESOLUTION is 6, 7, or 8 then:
  - Channel 1 or 2 data (DIGITIZE 1 or 2) is acquired in a single acquisition to capture 8192 data points.

- Channel data is filtered and processed into Memory 1.

Note

This is in a unique HP-IB process. Normally (front-panel store), only raw data is placed in memory with the Store keys. Then, memories are viewed through the filter specified.

- Data is captured and processed into the channel buffers. Functions execute on the processed data as defined by the Function subsystem. Data processing includes the specified filter and interpolation if it is necessary to fill in points.
- Function data is interpolated.
- Channel 1 (DIGITIZE 1) data is placed into Memory 1.
- Channel 2 (DIGITIZE 2) data is placed into Memory 2.
- Function 1 (DIGITIZE 5) data is placed in Memory 7.
- Function 2 (DIGITIZE 6) data is placed in Memory 8.

Command Syntax: RESOlution\<bits\>

                \<bits\> ::= {OFF | 6 | 7 | 8}

Example: OUTPUT 707;"RESOLUTION 8"

Query Syntax: RESOlution?

Returned Format: {RESOlution | RESO} \<bits\>

                            \<bits\> ::= 6, 7, or 8

                                  (integer · NR1 format)

Example: OUTPUT 707;"RESOLUTION?"
ENTER 707;Reso
PRINT Reso

This command selects the type of acquisition that is to take place when a DIGITIZE system command is executed. There are four acquisition types: NORMAL, FILTERED, VPERSISTENCE, and AVERAGE.

The NORMAL, FAST and FILTERED commands select the real time display mode.

The AVERAGE and VPERSISTENCE commands select the repetitive display mode. AVERAGED allows the number of averages for each time point to be selected. VPERSISTENCE selects the variable persistence display mode. The persistence of the displayed data points can be set in the DISPLAY subsystem.

The TYPE query returns the current acquisition type.

Command Syntax: TYPE {VPERSISTENCE | FILTERED | AVERAGE | NORMAL|FAST}

Example: OUTPUT 707;"ACQUIRE TYPE VPER"

Query Syntax: TYPE?

Returned Format: [TYPE]<type><CR><LF>
                            <type> ::= {VPERsistence | FILTered | AVERage | NORMal|FAST}

Example: DIM Type$[30]
OUTPUT 707;"TYPE?"
ENTER 707;Type$
PRINT Type$

4   Channel Subsystem

Introduction

The CHANNEL subsystem commands control the channel display, vertical or Y axis, and the channel input coupling functions of the HP 54111D. Channel 1 and channel 2 are independently programmable for all functions.

The channel waveform displays are turned on and off using the system commands VIEW and BLANK.

Subsystem Selection

The CHANNEL subsystem must be selected prior to sending any channel subsystem command or queries. This can be done by sending a single line command "CHANNEL 1" or "CHANNEL 2," or by sending the subsystem selector prior to another subsystem command or query. As an example, "CHANNEL 1 SENSITIVITY .1" will select the channel subsystem, select channel 1, and set the sensitivity of the selected channel (1) to 0.1 volts/division.

System Command

A system command can be sent at any time while in the subsystem. After a system command has been sent, the instrument will return to the subsystem that was selected (in this case CHANNEL 1) prior to the system command.

Sending Queries

In this subsystem if a query is sent followed by another query without a controller read (ENTER) between the queries, only the last response will be received. Although this operation is syntactically correct, the HP 54111D does not queue query responses in this subsystem. Therefore, the response to the first query will be overwritten in the HP-IB output buffer.
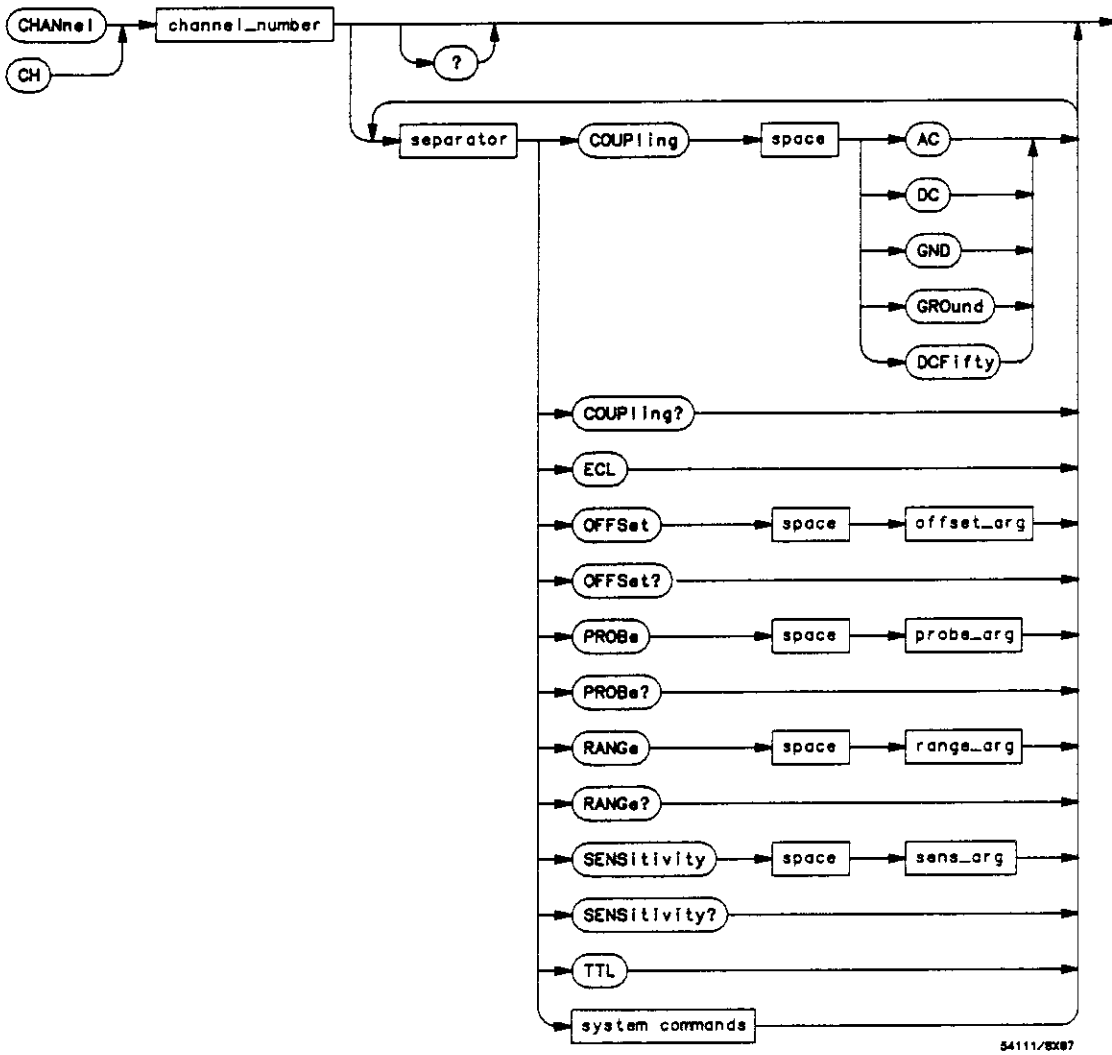
Figure 4-1. Channel Subsystem Syntax Diagrams.
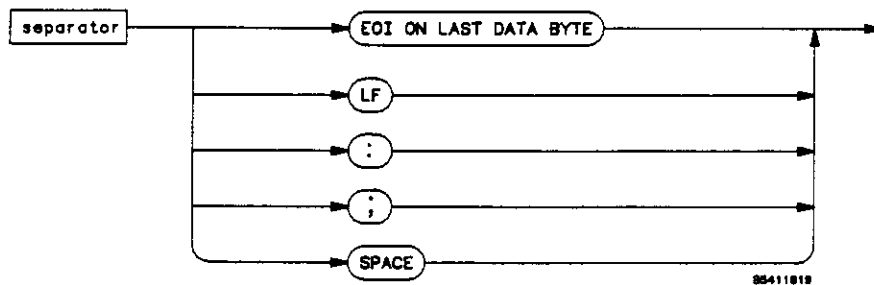
CHANNEL_NUMBER = 1 or 2

OFFSET_ARG = A real number defining the voltage at the center of the display range. Value of offset is 0 V to ±40 V and the maximum depends on the sensitivity setting. The maximum offset settings are shown below:

Sensitivity	Offset maximum voltage
1 mV to 4 mV	±200 mV
5 mV to 49 mV	±1 V
50 mV to .49 V	±10 V
.5 V to 5.0 V	±40 V

PROBE_ARG = A real number from 0.1 to 1000 specifying the probe attenuation with respect to 1.

RANGE_ARG = A real number specifying the size of the acquisition window in volts. This value is eight times the sensitivity when in the single screen display mode. In the dual screen display the range value is four times the sensitivity. In the quad screen mode, the range is two times the sensitivity value. The RANGE can be set to any value from 8 mV to 40 V when using 1:1 probe attenuation. If the probe attenuation is changed, the sensitivity and range values are multiplied by the probe attenuation factor. When the range value is changed, the sensitivity is also changed.

SENS_ARG = A real number specifying the size of the acquisition window in volts/division. Sensitivity can be set to any value from 1 mV/division to 5 V/division when using a 1:1 probe and in the single screen display mode. In the dual screen mode, sensitivity is two times the single screen value; and in the quad screen mode, the sensitivity is four times the single screen value.

Figure 4-1. Channel Subsystem Syntax Diagrams (continued).

Channel Subsystem
4-3

command/query

The CHANNEL command selects the CHANNEL subsystem with
the specified channel designated as the destination for the other
subsystem commands.

The query responds with the settings for the specified channel. Refer
to the individual commands for arguments and range values.

Command Syntax: {CHANnel | CH} {1 | 2}

Example: OUTPUT 707;"CHANNEL 1"

Query Syntax: {CHANnel | CH} {1 | 2}?

Returned Format: [CHANnel | CH]<NR1><CR><LF>
[PROBe]<NR3><CR><LF>
[RANGe]<NR3><CR><LF>
[OFFSet]<NR3><CR><LF>
[COUPling]<Argument><CR><LF>
[SENSitivity]<NR3><CR><LF>

Example: DIM Chan$[100]
OUTPUT 707;"EOI ON"
OUTPUT 707;"CHANNEL 2?"
ENTER 707 USING "-K";Chan$
PRINT USING "K";Chan$

COUPling   **command/query**

The COUPLING command sets the coupling for the input of the selected channel. Dc and ac coupling arguments provide 1 MΩ input impedance. When ac coupling is selected, the input is 1 μF. The actual ac (capacitive) coupling takes place after the attenuator. The DCFIFTY argument provides dc coupling with 50 μF input impedance. The GROUND argument places a marker on the display at the zero volt reference point for that channel.

Command Syntax: COUPling {DC | AC | GROund | GND | DCFifty}

Example: OUTPUT 707;"COUPLING AC"

Query Syntax: COUPling?

Returned Format: Coupling {DC | AC | GROund | DCFifty}

Example: DIM Coup$[30]
OUTPUT 707;"COUPLING?"
ENTER 707;Coup$
PRINT Coup$

Channel Subsystem
4-5

ECL   **command**

This ECL command sets the vertical range, offset, and trigger level of the selected channel for optimum viewing of ECL signals. The offset and trigger level are set to -1.30 volts and the range is set to 1.6 volts.

If a RANGE, SENSITIVITY, or OFFSET command is sent to the selected channel, ECL will be turned off.

There is no query form of this command.

Command Syntax: ECL

Example: OUTPUT 707;"ECL"

OFFSet   command/query

The OFFSET command sets the voltage that is represented at center screen for the selected channel. The value of offset is 0 V to ±40 V and the maximum depends on the sensitivity setting and display format and probe attenuation. The maximum offset settings are shown below:

Sensitivity	Offset maximum voltage
1 mV to 4 mV	±200 mV
5 mV to 49 mV	±1 V
50 mV to .49 V	±10 V
.5 V to 5.0 V	±40 V

The query returns the current offset value for the selected channel.

Command Syntax: OFFSet <value>
                    <value> ::= offset value (see above)

Example: OUTPUT 707;"OFFSET 650E-3"

Query Syntax: OFFSet?

Returned Format: [OFFSet] <value><CR><LF>
                    <value> ::= offset value (see above)
                              (exponential - NR3 format)

Example: OUTPUT 707;"OFFSET?"
ENTER 707;OFFSET$
PRINT OFFSET$

PROBe   **command/query**

The PROBE command specifies the probe attenuation factor for the selected channel. The range of the probe attenuation factor is from 1.0 to 1000. This command does not change the actual input sensitivity of the HP 54111D. It changes the reference constants that are used for scaling the display factors and for automatic measurements, trigger levels, etc.

The query returns the current probe attenuation factor for the selected channel.

The attenuation factor is multiplexed by ten when the HP 10033A probe is attached.

Command Syntax: PROBe <atten>
                              <atten> ::= 1 to 1000

Example: OUTPUT 707;"PROBE 15.5"

Query Syntax: PROBe?

Note

If an HP 10033A probe is attached, the number that is output has a range of 10 to 1000.

Returned Format: [PROBe]<atten><CR><LF>
                              <atten> ::= 1 to 1000
                                        (exponential - NR3 format)

Example: OUTPUT 707;"PROBE?"
              ENTER 707;Probe$
              PRINT Probe$

RANGe   **command/query**

The RANGE command defines the full scale vertical axis of the selected channel. This value is eight times the sensitivity when in the single screen display mode. In the dual screen display the range value is four times the sensitivity. In the quad screen mode, the range is two times the sensitivity value. The RANGE can be set to any value from 8 mV to 40 V, when using 1:1 probe attenuation. If the probe attenuation is changed, the sensitivity and range values are multiplied by the probe attenuation factor. When the range value is changed, the sensitivity is also changed.

The query returns the current range setting for the specified channel.

Command Syntax: RANGe <range>

<range> ::= .008 V to 40 V

when using 1:1 probe attenuation

Example: OUTPUT 707;"RANGE 4"

Query Syntax: RANGe?

Returned Format: [RANGe]<range><CR><LF>

<range> ::= .008 V to 40 V

when using 1:1 probe attenuation

(exponential - NR3 format)

Example: OUTPUT 707;"RANGE?"
ENTER 707;Range$
PRINT Range$

SENSitivity   **command/query**

> This command specifies the vertical deflection in volts/division. The sensitivity can be set to any value from 1 mV/division to 5 V/division when using a 1:1 probe and in the single screen display mode. In the dual screen mode, sensitivity is two times the single screen value. In the quad screen mode, the sensitivity is four times the single screen value. Also see the RANGE command.
>
> The query returns the current sensitivity setting for the selected channel.

Command Syntax: SENSitivity <volts/div>
                                    <volts/div> ::= .001 V/div to 5 V/div
                                                    in single screen using 1:1 probe attenuation

Example: OUTPUT 707;"SENSITIVITY 1"

Query Syntax: SENSitivity?

Returned Format: [SENSitivity] <volt/div><CR><LF>
                                    <volt/div> ::= .001 V/div to 5 V/div
                                                    with single screen and 1:1 probe attenuation.
                                                    (exponential -NR3 format)

Example: OUTPUT 707;"SENS?"
                    INPUT 707;Sens$
                    PRINT Sens$

TTL   **command**

This command sets the vertical range, offset, and trigger level for optimum viewing of TTL signals on the selected channel. Offset and trigger level is set to 1.6 volts and the range is set to 8.0 volts.

If a RANGE, SENSITIVITY, or OFFSET command is sent to the selected channel, TTL is turned off.

There is no query form of this command.

Command Syntax: TTL

Example: OUTPUT 707;"TTL"

5 Display Subsystem

Introduction   The DISPLAY subsystem is used to control the display of data, voltage and time markers, text, graticules, and the use of color. The commands which control the DISPLAY mode are TYPE and COUNT in the ACQUIRE subsystem.

Subsystem Selection   The DISPLAY subsystem must be selected prior to sending any DISPLAY subsystem commands or queries. This can be done by sending a single line command "DISPLAY" or by sending the subsystem selector prior to another command or query. As an example "DISPLAY COLUMN?" will select the display subsystem and ask for the instrument to respond with the current column number.

System Command   A system command can be sent at any time while in this subsystem. After a system command has been sent the instrument will return to the subsystem that was selected (in this case DISPLAY) prior to the system command.

Sending Queries   In this subsystem if a query is sent followed by another query without a controller read (ENTER) between the queries, only the last response will be received. Although this operation is syntactically correct, the HP 54111D does not queue query responses in this subsystem. Therefore, the response to the first query will be overwritten in the HP-IB output buffer.

See figure 5-1 for the syntax of the DISPLAY subsystem commands.
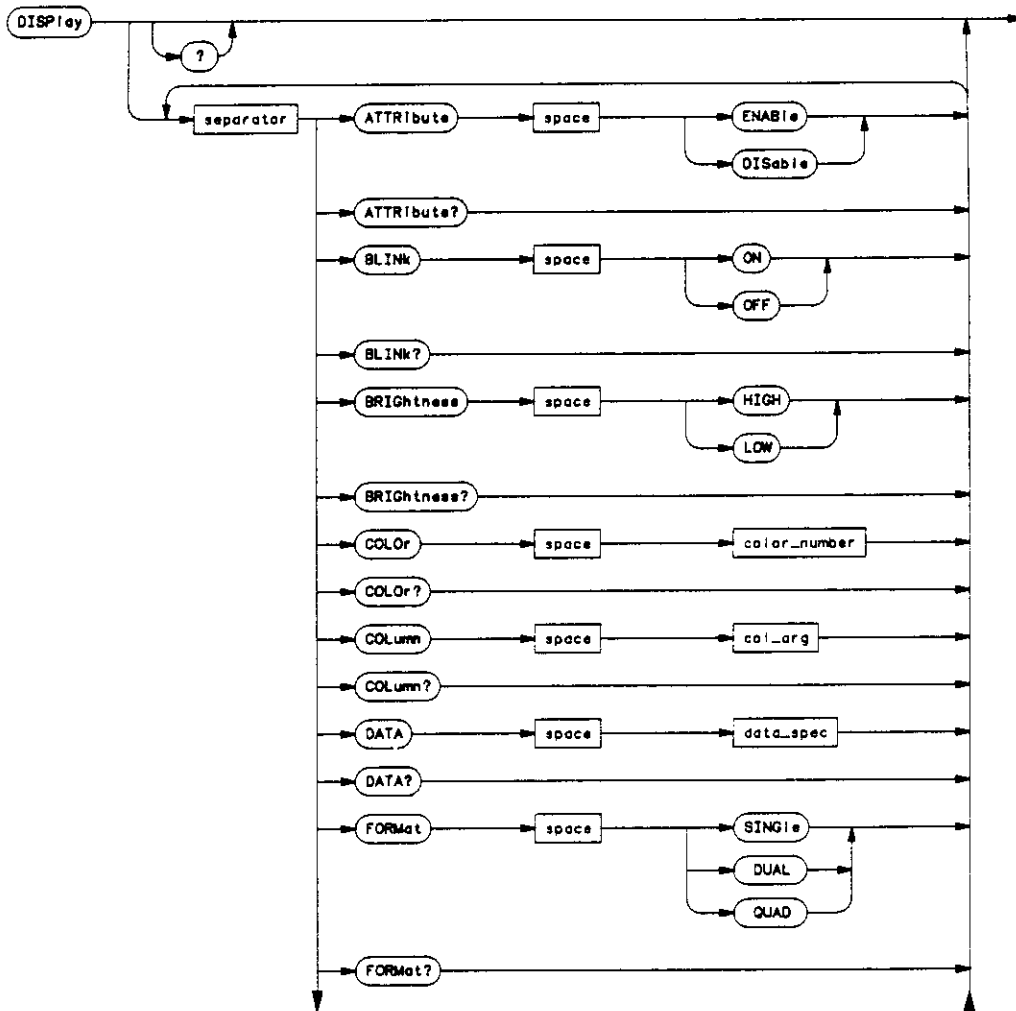
Figure 5-1. Display Subsystem Syntax Diagrams.
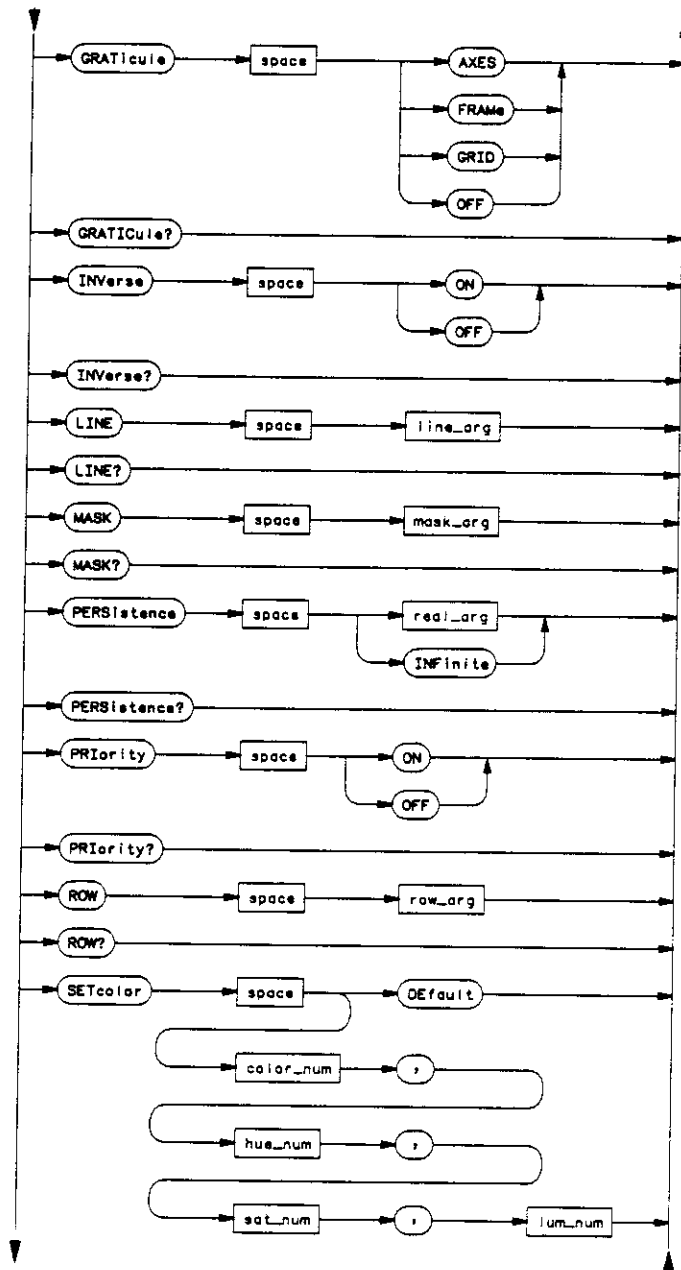
Figure 5-1. Display Subsystem Syntax Diagrams (continued).
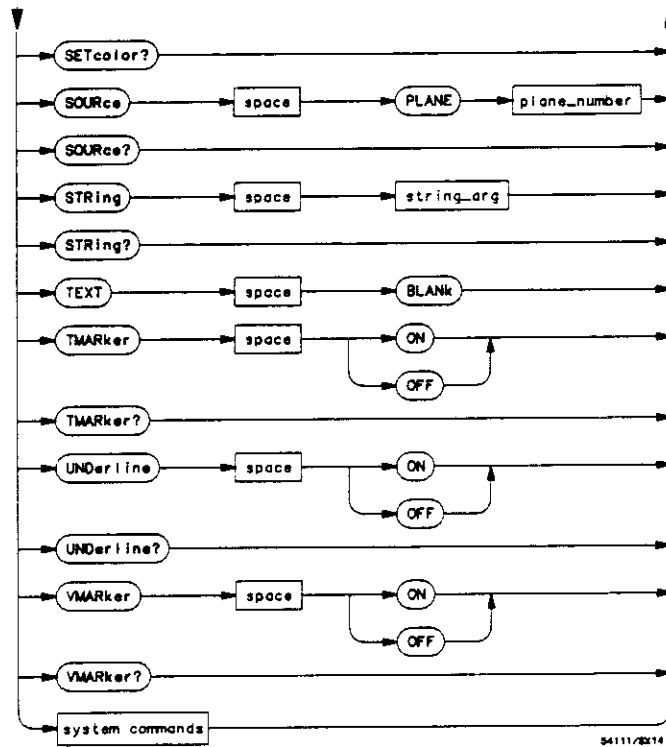
Figure 5-1. Display Subsystem Syntax Diagrams (continued).

COL_ARG = An integer from 0 to 71.

COLOR_NUMBER = An integer from 0 to 15.

DATA_SPEC = A block of data in #A format as defined in IEEE Std. 728-1982.

HUE_NUM = An integer, 0 through 100.

LINE_ARG = Any quoted string.

LUM_NUM = An integer, 0 through 100.

MASK_ARG = An integer, 0 through 255.

PLANE_NUMBER = An integer, 0 through 9.

REAL_ARG = A real number, 0.2 through 10.0 in steps of 0.1.

ROW_ARG = An integer, 0 through 22.

SAT_NUM = An integer, 0 through 100.

STRING_ARG = Any quoted string.

Figure 5-1. Display Subsystem Syntax Diagrams (continued).

DISPlay   **command/query**

The DISPLAY command selects the DISPLAY subsystem as the destination for the DISPLAY subsystem commands.

The DISPLAY query returns the parameters for this subsystem.

Command Syntax: DISPlay

Example: OUTPUT 707;"DISP"

Query Syntax: DISPlay?

Returned Format: [DISPlay]<CR><LF>
[FORMat]{ SINGLE | DUAL | QUAD }<CR><LF>
[GRATicule]{ OFF | GRID | AXES | FRAME }<CR><LF>
[ROW]<NR1><CR><LF>
[COLumn]<NR1><CR><LF>
[ATTRibute]{ DISABLE | ENABLE }<CR><LF>
[INVerse]{ OFF | ON } <CR><LF>
[BLINk]{ OFF | ON }<CR><LF>
[UNDerline]{ OFF | ON }<CR><LF>
[BRIGhtness]{ LOW | HIGH }<CR><LF>
[VMARker]{ OFF | ON }<CR><LF>
[TMARker]{ OFF | ON }<CR><LF>
[PERSistence]<NR3><CR><LF>
[COLOr]<NR1><CR><LF>
[PRIority]{ OFF | ON }
[SETColor] <NR1> <NR1> <NR1> <NR1>

Example: DIM Display$[500]
OUTPUT 707;"HEADER ON EOI ON"
OUTPUT 707;"DISPLAY?"
ENTER 707 USING "-K";Display$
PRINT USING "K";Display$

The ATTRIBUTE command controls embedded attributes in text lines that are sent with the DSP system command or the LINE or STRING commands in the DISPLAY subsystem. Attributes can be embedded in the text when any of these commands are used. The attributes control how the text will be displayed. The attributes control the color of the text display, whether or not it will blink, be in inverse video, and/or be underlined. If the ATTRIBUTE command is disabled or if no attributes are used in the text, the text will be written in color #1 (default gray), will not blink, will not be in inverse video, and will not be underlined. Refer to table 5-1 for the attribute byte information. The text attributes include:

INVerse, UNDerline, BLINk, COLOr

When this command is enabled, the embedded attribute bytes in strings sent with the LINE or STRING commands override previously set attributes. This means that the text display can be changed by sending new attributes in order to highlight key words, etc. When the ATTRIBUTE command is disabled, any attributes within text being sent are interpreted as ASCII characters. This command has no effect on text that was sent previous to the enable or disable. The ATTRIBUTE command does not affect the INVERSE, BLINK, UNDERLINE, or COLOR commands.

The ATTRIBUTE query returns the state of the command, which is either ENABLE or DISABLE.

Command Syntax: ATTRibute {DISable | ENABle}

Example: OUTPUT 707;"ATTRIBUTE ENABLE"

Query Syntax: ATTRibute?

Returned format: [ATTRibute]<state><CR><LF>
                               <state> ::= ENABle or DISable

Example: DIM Attr$[30]
OUTPUT 707;"ATTRIBUTE"
ENTER 707;Attr$
PRINT Attr$

Display Subsystem
5-7

Table 5-1. Attribute Byte.

A T T R I B U T E MSB		COLOR BITS 0 = BEIGE 8 = GRAY 16 = RED 24 = YELLOW 32 = GREEN 40 = ORANGE 48 = WHITE 56 = MAGENTA . 120 = BLACK			U N D E R L I N E	B L I N K	I N V E R S E LSB
128	64	32	16	8	4	2	1

When you want to send attributes to control a display, add the binary values of the attributes you wish to send and send them over the HP-IB to the instrument. The embedded attributes can be inhibited using the ATTRIBUTE DISABLE command, and enabled using the ATTRIBUTE ENABLE command. Attributes affect the text that is sent to the display of the instrument when you use the LINE or STRING commands.

The example below causes "HELLO" to be written in red at the current ROW and COLUMN of the display. The word "HELLO" will also be displayed in inverse video, and will be blinking.

OUTPUT 707;"DISPLAY TEXT BLANK ATTRIBUTE ENABLE"
OUTPUT 707 USING "8A,B,6A";"STRING """,128+16+2+1,"HELLO"""

In the example above the first line selects the DISPLAY subsystem (DISPLAY), blanks text displayed in user area (TEXT BLANK), then enables the attributes (ATTRIBUTE ENABLE).

The second line in the example contains some formatting. The 8A enables string data "STRING" to be sent; the B changes the attribute value (147) to an 8-bit character. The 6A enables six characters of string data "HELLO" to be sent to the instrument.

The 128 indicates that this is an attribute byte; 16 indicates the color red; 2 indicates the blink attribute; and 1 indicates the invert attribute. This could just as easily have been output to the instrument as "147."

BLINk   **command/query**

The BLINK command determines whether text sent with the DSP system command, or the LINE or STRING command in the DISPLAY subsystem is to be written with the BLINK attribute. If the BLINK attribute is turned on, text will flash on and off when it is displayed.

The BLINK query returns the state of the BLINk attribute.

Command Syntax: BLINk {OFF | ON}

Example: OUTPUT 707;"BLINK ON"

Query Syntax: BLINk?

Returned Format: [BLINk]<state><CR><LF>
<state> ::= ON or OFF

Example: OUTPUT 707;"BLINK?"
ENTER 707;Blink$
PRINT Blink$

BRIGhtness command/query

The BRIGHTNESS command specifies whether text sent with the DSP system command, or the LINE or STRING commands in the DISPlay subsystem are to be displayed in beige or gray. LOW provides gray text and HIGH provides beige text.

The BRIGHTNESS query returns the state of the BRIGHTNESS attribute. This command overrides previous COLOR commands.

Command Syntax: BRIGhtness {LOW | HIGH}

Example: OUTPUT 707;"BRIGHTNESS LOW"

Query Syntax: BRIGhtness?

Returned Format: [BRIGhtness]<state><CR><LF>
<state> ::= LOW or HIGH

Example: DIM Bright$[30]
OUTPUT 707;"BRIGHTNESS?"
ENTER 707;Bright$
PRINT Bright$

command/query

The COLOR command specifies what color the text will be when
sent with the DSP system command or the LINE or STRING
commands in the DISPLAY subsystem. This command overrides a
previous brightness command.

The COLOR query returns the color of the COLOR attribute.

Note

*The short form for this command is COLO. Convention
would have it as COL, however COL is the short form for
COLumn.*

Command Syntax: COLOr<number>

               <number> ::= 0 to 15, where

                         0 ::= beige, highlighted text
                         1 ::= gray, text fields & graticules
                         2 ::= red, advisories & errors
                         3 ::= yellow, channel 1 waveforms
                         4 ::= green, channel 2 waveforms
                         5 ::= orange, markers
                         6 ::= white, stored waveforms
                         7 ::= magenta, 2 trace overlap
                         8 ::= orange, function 1
                         9 ::= cyan, function 2
                       10 ::= magenta, memory bar
                       11 ::= magenta, 3 trace overlap
                       12 ::= magenta, 2 trace + memory overlap
                       13 ::= magenta, 3 trace + memory overlap
                       14 ::= magenta, 4 trace + memory overlap
                       15 ::= black, background

Note

*These numbers do not match the color numbers entered
through the front panel.*

Example: OUTPUT 707;"COLOR 2"

Query Syntax: COLOr?

Returned Format: [COLOr]<number><CR><LF>

              <number> ::= 1 to 15

                      (integer - NR1 format)

Example: OUTPUT 707;"COLOR?"

             ENTER 707;Color

             PRINT Color

COLumn   **command/query**

The COLUMN command specifies the starting column for subsequent STRING and LINE commands.

The COLUMN query returns the column where the next LINE or STRING will start.

Command Syntax: COLumn <number>

                                  <number> ::= 0 to 71

       Example: OUTPUT 707;"COLUMN 50"

Query Syntax: COLumn?

Returned Format: [COLumn]<number>

                                  <number> ::= 0 to 71

                                                (interger - NR1 format)

       Example: OUTPUT 707;"COLUMN?"
                    ENTER 707;Column
                    PRINT Column

The DATA command is used to write pixel data to or from one of the nine pixel planes in the HP 54111D. The pixel planes available are plane 0 through plane 8 and are specified by the SOURCE command (page 5-28).

The DATA query causes the HP 54111D to output pixel data from the specified memory plane. If plane 0 is specified, the HP 54111D will transfer the "logical or" of the channel 1, channel 2, function 1, and function 2 planes. In all other cases the specified plane will be transferred.

The DATA command is followed by a block of binary data that is transferred from the controller to a specific plane in the HP 54111D. If plane 0 is specified, that data will be transferred into the channel 1 plane. In all other cases the data will be transferred into the specified plane.

The data is in the form of 16,384 bytes with four header bytes. The header contains:

$<\#>$ ::= (decimal 35) = byte 1
$<A>$ ::= (decimal 65) = byte 2
       (decimal 64) = byte 3
       (decimal 0) = byte 4

The third and fourth bytes make up a 16-bit integer whose value is decimal 16384, or the length of the binary block. This binary format complies with the "#A" Block Data Field in IEEE 728-1982.

Pixel Format

The data that is output by this command is the displayed data from the waveform display area (inside the graticule) of the HP 54111D CRT. When using the DISPLAY DATA command, the information is output as bytes of pixel data. The CRT is 512 pixels horizontally by 256 pixels vertically. Each pixel row is divided into 64 bytes of eight bits each. Refer to figure 5-2 for the CRT pixel format.
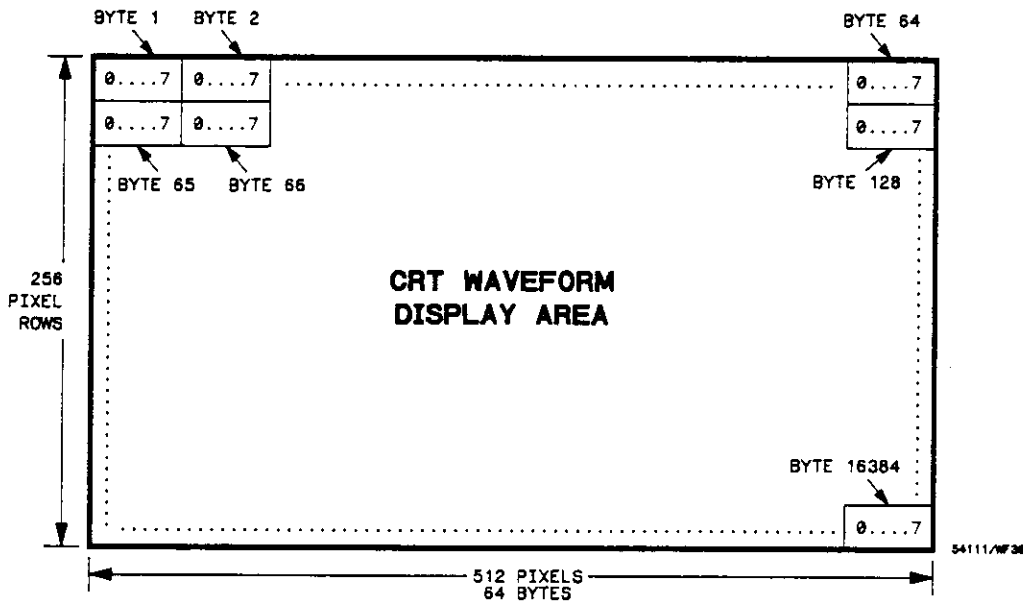
Figure 5-2. CRT Pixel Format.

Data Output Format   The pixel planes are 16,384 bytes representing the waveform display portion of the display. The waveform display portion of the display is the portion of the display from and including the top graticule line, to and including the bottom graticule line, as well as all information inside the graticule. The first 64 bytes of data that are output correspond to the top row of the waveform display (figure 5-2). The next 64 bytes that output are the second row of the waveform display, the third 64 bytes output correspond to the third row, etc. The data is output in rows until the 256th pixel row of data is finally sent. The 256th pixel row corresponds to the bottom graticule line of the waveform display area.

Each of the pixel rows contain 512 pixels that are sent in 8-bit bytes, and 256 rows of pixel data are sent. Some calculations at this point will show the relationship of the numbers. Each row is sent as 64 bytes, therefore 64 bytes x 8 bits = 512 pixels per row; 64 bytes of data are sent for each row and 256 rows or data are sent, therefore 64 bytes x 256 rows = 16,384 bytes sent for each screen of data.

Display Subsystem
5-15

Command Syntax:	DATA <binary block #A format>
Query Syntax:	DATA?
Returned Format:	[DATA]<2sp>#A<decimal 64><decimal 0><binary data>
Example:	10 ASSIGN @Fast TO 707;FORMAT OFF
	20 OUTPUT 707;"HEADER OFF EOI ON"
	30 OUTPUT 707;"DISP SOURCE PLANE0 DATA?
	40 ENTER 707 USING "#,2A,W";Header$,Length
	60 ALLOCATE INTEGER Plane0(1:Length/2)
	70 ENTER @Fast;Plane0(*)
	80 OUTPUT 707;"SOURCE PLANE1"
	90 OUTPUT 707 USING "#,6A,2A,W";"Data";Header$,Length
	100 OUTPUT @Fast;Plane0(*)
	110 END

This example transfers data from the active display memory to the
controller, then transfers the data back to pixel memory 9 in the HP
54111D.

The purpose of the #A format is so that the HP 9836 controller can
output an integer array directly to the CRT with a GLOAD
command. Do not bring it in as a string. Integer array data may be
transferred to a suitably-dimensioned integer array for use with the
HP-series 200 GLOAD command.

command/query

The FORMAT command sets the number of display areas on the
CRT. FORMAT SINGLE provides 1 display area and uses 8 divisions
for the full scale range. FORMAT DUAL sets the screen mode to
DUAL and uses 4 divisions for the full scale range. FORMAT QUAD
provides 4 display areas on the CRT and uses 2 divisions for the full
scale range.

The FORMAT query returns the current display format.

Command Syntax: FORMat {SINGle | DUAL | QUAD}

Example: OUTPUT 707;"FORMAT SINGLE"

Query Syntax: FORMat?

Returned Format: [FORMat]<mode><CR><LF>
                          <mode> ::= {SINGLE | DUAL | QUAD}

Example: DIM Format$[30]
             OUTPUT 707;"FORMAT?"
             ENTER 707;Format$
             PRINT Format$

GRATicule   command/query

The GRATICULE command selects the type of graticule that is displayed.

The GRATICULE query returns the type of graticule displayed.

Command Syntax: GRATicule {OFF | GRID | AXES | FRAMe}

Example: OUTPUT 707;"GRATICULE AXES"

Query Syntax: GRATicule?

Returned Format: [GRATicule]<type><CR><LF>
                   <type> ::= {GRID | AXES | FRAME | OFF}

Example: DIM Grat$[30]
              OUTPUT 707;"GRATICULE?"
              ENTER 707;Grat$
              PRINT Grat$

command/query

The INVERSE command sets inverse video on or off for subsequent DSP system commands or DISPLAY subsystem LINE or STRING commands.

The INVERSE query responds with the on/off state of this command.

Command Syntax: INVerse {OFF | ON}

Example: OUTPUT 707;"INVERSE OFF"

Query Syntax: INVerse?

Returned format: [INVerse]<state><CR><LF>
                            <state> ::= ON or OFF

Example: DIM Inv$[30]
            OUTPUT 707;"INVERSE?"
            ENTER 707;Inv$
            PRINT Inv$

The LINE command writes a text string to the screen. The text is displayed starting at the location of the current row and column. The row and column can be set by the ROW and COLUMN commands prior to sending the LINE command. Text may be written up to column 62 using the LINE command. If the characters in the text string do not fill the line, the rest of the line is blanked. If the text string is longer than the available space on the current line, the excess characters will be discarded. In any case, the ROW value is incremented by one and the COLUMN value remains the same. The next LINE command will write on the next line of the display starting at the same column as the previous text. After writing line 21, the last line in the display area, ROW is reset to 2.

The LINE query reads the quoted string at the ROW and COLUMN values and causes ROW to be incremented by 1.

Note

The LINE command and query can only be used in the user display area rows 2 through 21.

Command Syntax: LINE <quoted string>
                          <quoted string> ::= any series of ASCII characters
                                              enclosed in quotes.
Example: OUTPUT 707:"LINE ""ENTER PROBE ATTENUATION"""

Query Syntax: LINE?

Returned Format: [LINE?] <displayed text><CR><LF>
                          <displayed text> ::= text on current row, surrounded by quotes
Example: DIM Line$[100]
OUTPUT 707;"DISP ROW 12 COL 14 LINE ""1234"""
OUTPUT 707;"Row 12 line?"
ENTER 707;Line$
PRINT Line$

MASK   **command/query**

The MASK command inhibits the instrument from writing to selected areas of the screen. Text sent over the HP-IB using the LINE and STRING commands is not affected by this command. The purpose of the command is to allow HP-IB text to be written anywhere on screen and to prevent the instrument from overwriting the text through its normal operation.

The mask parameter is an 8-bit integer in which each bit controls writing to an area of the screen. A 0 prevents the instrument from writing to the area represented by the bit, and a 1 enables writing to the area.

The MASK query returns the current value of the MASK.

Note

The MASK command's parameters will not be reset with a RESET command.

Command Syntax: MASK <value>
                            <value> ::= 0 to 255
                                        (integer - NR1 format)

Example: OUTPUT 707;"MASK 254" ! Inhibits advisories only

Query Syntax: MASK?

Returned Format: [MASK] <value><CR><LF>
                            <value> ::= 0 to 255
                                        (integer - NR1 format)

Example: DIM Mask$[30]
                OUTPUT 707;"MASK?"
                ENTER 707;Mask$
                PRINT Mask$

Table 5-2. Display Mask Byte.

Bit	Mask Weight	Screen Area Affected
7	128	Memory bar - row 1, column 0-61
6	64	Function Softkeys - Softkey labels on the right side of the display (rows 0-17, columns 62-71)
5	32	Menu Selection Softkeys - Text on the bottom line of the display (row 22, columns 0-71).
4	16	Parameter Values - Text below the graticule (rows 18-21, columns 0-71)
3	8	Graticule Labels - Text inside the graticule (rows 2-17, columns 0-61)
2	4	Value Label - Displays value of selected knob function (row 1, columns 19-61).
1	2	Status Line - Status information on the first two lines (row 0, columns 0-71 and row 1, columns 0-18).
0	1	Advisory - Advisory and error messages appear on row 15, columns 0-61.

The PERSISTENCE command sets the display persistence. The PERSISTENCE command is only effective in the VPERSISTENCE display mode, selected in the ACQUIRE subsystem. The parameters for this command are the keyword INFINITE or a real number from 0.2 to 11.0 representing the persistence in seconds. A value of 11 seconds will set the PERSISTENCE to infinite.

The PERSISTENCE query returns the value of the current persistence value. If persistence is set to infinite the query response will be $1.1E+1$

Command Syntax: PERSistence {INFinite | 0.2 to 11.0}

Example: OUTPUT 707;"PERSISTENCE 3.0"

Query Syntax: PERSistence?

Returned Format: [PERSistence] <value><CR><LF>
                              <value> ::= 0.200E-01 to 1.100E+1
                                              where $1.100E+1$ = infinite
                                              (exponential - NR3 format)

Example: OUTPUT 707;"PERSISTENCE?"
                    ENTER 707;Pers
                    PRINT Pers

PRIority   **command/query**

The PRIORITY command sets the priority on or off for subsequent DSP system commands and LINE or STRING commands in the DISPLAY subsystem. The priority determines whether text or graphics will have priority over the other. When PRIORITY is ON, text overwrites the displayed signal(s) and graticule. When PRIORITY is OFF, the displayed signal(s) and graticule overwrite any text in the waveform display area.

The PRIORITY query returns the state of the PRIORITY command.

Command Syntax: PRIority {OFF | ON}

Example: OUTPUT 707;"PRIORITY OFF"

Query Syntax: PRIority?

Returned Format: [PRIority] <state><CR><LF>
                        <state> ::= ON or OFF

Example: DIM Pri$[30]
              OUTPUT 707;"PRIORITY?"
              ENTER 707;Pri$
              PRINT Pri$

The ROW command specifies the starting row on the CRT for
subsequent STRING and LINE commands. The ROW number
remains constant until another ROW command is received or it is
incremented by the LINE command or query. The ROW value is 2
through 21.

The ROW query returns the current value of ROW.

Command Syntax: ROW <row number>
           <row number>::= 2 through 21

Example: OUTPUT 707;"ROW 10"

Query Syntax: ROW?

Returned Format: [ROW]<row number><CR><LF>
           <row number> ::= 2 through 21
                       (integer - NR1 format)

Example: OUTPUT 707;"ROW?"
ENTER 707;Row
PRINT Row

The SETCOLOR command allows you to change any of the color selections on the CRT. All of the color selections may be changed over HP-IB one at a time. This command has four parameters: color number, hue, saturation, and luminosity.

The hue portion of this command allows you to determine the gradation of color. Hue can have a value of 0 to 100. As the hue number is increased, the selected color will cycle through the color spectrum. There is no difference between hue 0 and hue 100.

The saturation portion of this command allows you to choose the percentage of the pure color that gets mixed with white. The acceptable values for saturation are 0 to 100, where 0 is white and 100 is maximum saturation of the chosen hue.

The luminosity portion of this command determines the brightness of the chosen hue. The acceptable values for luminosity are 0 to 100, where 0 is black and 100 is maximum brightness. The SETCOLOR command followed by DEFAULT sets all colors to the default settings. (see page 5-11).

The query of this command returns the specified color number, hue, saturation, and luminosity. Refer to the *HP 54111D Front-Panel Reference Manual* (chapter 14) for more information concerning color.

Note

Color values are not reset with a RESET command.

Command Syntax: SETColor{<color>,<hue>,<sat>,<lum> | DEFault}

<color> ::= integer from 0 to 15

<hue> ::= integer from 0 to 100

<sat> ::= integer from 0 to 100

<lum> ::= integer from 0 to 100

Refer to COLOR command (page 5-11) for number assignments.

Example: OUTPUT 707;"SETColor 0,0,100,50"

Query Syntax: SETColor<color>?
                          <color> ::= color number 0 through 15

Returned Format: [SETColor] <color>,<hue>;<sat>,<lum>

Example: OUTPUT 707;"SETCOLOR 2?"
                 ENTER 707;Hue,Sat,Lum
                 PRINT Hue,Sat,Lum

SOURce | SRC   **command/query**

The SOURCE command specifies the source or destination for the DISPLAY DATA query and command. The SOURCE command has 1 parameter, PLANE 0 through PLANE 8.

The SOURCE query returns the currently specified SOURCE.

Command Syntax: {SOURce | SRC}PLANE{0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8}

      PLANE0 ::= active display
      PLANE1 ::= pixel memory 9
      PLANE2 ::= pixel memory 10
      PLANE3 ::= graticule and markers
      PLANE4 ::= displayed stored waveforms and markers
      PLANE5 ::= channel 1
      PLANE6 ::= channel 2
      PLANE7 ::= function 1
      PLANE8 ::= function 2

Example: 10   ASSIGN @Fast TO 707;FORMAT OFF
        20   OUTPUT 707;"HEADER OFF EOI ON"
        30   OUTPUT 707;"DISP SOURCE PLANE0 DATA"
        40   ENTER 707 USING "#,2A,W";Header$,Length
        60   ALLOCATE INTEGER Plane0(1:Length/2)
        70   ENTER @Fast;Plane0(*)
        80   OUTPUT 707;"SOURCE PLANE1"
        90   OUTPUT 707 USING "#,6A,2A,W";"DATA ";Header$,Length
      100 OUTPUT @Fast;Plane0(*)
      110 END

This example transfers data from the active to the controller and then back to pixel memory 9 in the HP 54111D.

The purpose of the #A format is so that the HP 9836 controller can output an integer array directly to the CRT with a GLOAD command. Do not bring it in as a string (see page 5-16).

Query Syntax: {SOURce | SRC}?

Returned Format: [SOURce]<plane><CR><LF>

                        <plane> ::= 0 through 8

                                      (integer - NR1 format)

Example: OUTPUT 707;"SRC?"
ENTER 707;Src
PRINT Src

---

                                                  command/query

The STRING command writes a text string to the CRT of the HP 54111D. The text will be written starting at the current ROW and COLUMN values. If the column limit is reached (column 71), the excess text is discarded. STRING does not increment the ROW value, however LINE does.

The STRING query returns the text on the line defined by the ROW value. The returned text starts at the column defined by the COLUMN value; all text from the current column value to column 71 is returned with the STRING query.

Command Syntax: STRing <quoted string>

Example: OUTPUT 707;"STRING ""INPUT SIGNAL TO CHANNEL 2"""

Query Syntax: STRing?

Returned Format: [STRing] <displayed text>

                <displayed text> ::= text on the defined row, starting at the defined column, surrounded by quotes.

Example: DIM Str$[90]
OUTPUT 707;"STRING?"
ENTER 707;Str$
PRINT Str$

TEXT

command

The TEXT command allows you to blank the user text area on the CRT. The user text area is rows 2 through 17, columns 0 through 62, and rows 18 through 21, columns 0 through 71. This command has only one parameter.

There is no query form of this command.

Command Syntax: TEXT BLANk

Example: OUTPUT 707;"TEXT BLANK"

TMARker   **command/query**

The TMARKER command turns the time markers on or off.

The TMARKER query returns the state of the Tmarkers.

Command Syntax: TMARker {OFF | ON}

Example: OUTPUT 707;"TMAR OFF"

Query Syntax: TMARker?

Returned Format: [TMARker] <state><CR><LF>

<state> ::= ON or OFF

Example:
```
DIM Tmar$[30]
OUTPUT 707;"TMARKER?"
ENTER 707;Tmar$
PRINT Tmar$
```

UNDerline   **command/query**

The UNDERLINE command will underline subsequent text sent
with the system command DSP or the display subsystem commands
LINE or STRING.

The UNDERLINE query returns the state of the UNDERLINE
attribute.

Command Syntax: UNDerline {OFF | ON}

Example: OUTPUT 707;"UNDERLINE ON"

Query Syntax: UNDerline?

Returned Format: [UNDerline] <state><CR><LF>
                            <state> :: = ON or OFF

Example: DIM Under$[30]
              OUTPUT 707;"UNDERLINE?"
              ENTER 707;Under$
              PRINT Under$

VMARker   **command/query**

The VMARKER command turns the voltage markers on and off.

The VMARKER query returns the state of the Vmarkers.

Command Syntax: VMARker {OFF | ON}

Example: OUTPUT 707;"VMARKER ON"

Query Suntax: VMARker?

Returned Format: [VMARker]<state><CR><LF>

<state> ::= ON or OFF

Example: DIM Vmark$[30]
OUTPUT 707;"VMARKER?"
ENTER 707;Vmark$
PRINT Vmark$

6

Function Subsystem

Introduction

The Function subsystem defines two functions using the displayed channels and/or the waveform memories as operands. The operators are: ADD, INVERT, INTEGRATE, MAXIMUM, MINIMUM, ONLY, SUBTRACT, VERSUS and MULTIPLY. See Figure 6-1 for a syntax diagram of the function subsystem commands.

The memories that can be used for function operands depend on the acquisition mode. Memories 1 through 4 are available if the data is acquired using the FILTERED or NORMAL (Real Time) modes, and memories 5 through 8 are available if the data is acquired using the AVERAGE or VPERSISTENCE mode.

Note

All mathematical functions operate on a point-to-point basis (i.e., both operands must have data in a time bucket in order for the function to have datum in that time bucket). Usually this is transparent, but it can be demonstrated under certain conditions:

1) In Real Time mode, if one function operand is a waveform memory, then memory scaling (TIME/DIV and DELAY) may be used to compress or pan a memory such that it does not fill the screen horizontally, in which case the function won't either.

2) At sweep speeds faster than 50 ns/div, fewer than 501 on-screen data are acquired per channel on each sweep. In Repetitive mode, this shows up as a function gradually filling in as differently-aligned data are acquired on successive sweeps. In Real Time mode, with RESOlution filters OFF, depending on the sweep speed and the channel-to-channel skew cal, some or all of the points in a function of channel 1 and channel 2 may be missing.

In addition, vertical scaling and offset are provided for all math functions except MAXIMUM, MINIMUM and VERSUS.

Subsystem Selection

The FUNCTION subsystem must be selected prior to sending any FUNCTION subsystem commands or queries. This can be done by sending a single line command "FUNCTION" or by sending the subsystem selector prior to another command. As an example, "FUNCTION 1 ADD CH 1, CH 2" will cause the FUNCTION subsystem to be selected and the channel 1 and channel 2 signals to be algebraically added.

System Command

A system command can be sent at any time while in this subsystem. After a system command has been sent, the instrument will return to the subsystem that was selected (in this case FUNCTION) prior to the system command.

Function Scaling

Vertical scaling of both function is provided for all mathemetical functions except versus, maximum and minimum. Scaling is a method of readjusting vertical sensitivity; therefore, you can select vertical sensitivity and reposition the function on screen.

The scaling key toggles between VERTICAL SCALING and VERTICAL OFFSET. To change vertical sensitivity or offset, use the FUNCTION # RANGE and FUNCTION # OFFSET commands.

The scaling factors are stored with the mathematical function used. This feature allows you to define another function and retain data from a previous function.

Default Scaling

Default scaling forces the results of a mathematical function to be rescaled and displayed on screen if the scalefactors have been misadjusted. Default scaling is executed every time a new operand is selected. If the function amplitude is adjusted out of range, a function may be restored to the screen by toggling through the operands.

Time Scaling

In cases where the operands are not of the same time scale, the first operand will determine the timebase scale for the function. The timebase scale for the function is displayed below the waveform display area. Time scaling cannot be changed as long as the function menu is displayed.

Vertical Scaling Units

The fundamental measuring units of an oscilloscope are VOLTS/DIV in the vertical axes and TIME/DIV in the horizontal axis. This philosophy is used regardless of the mathematical function chosen. No provisions have been made to manage units for all combinations of operands and mathematical functions available for the HP 541110.

The Multiply Apply a +2V signal to Channel 1 and a −3V signal to Channel 2.
Example Choose a function and select math function Multiply (*). Notice that
the result displayed is in fundamental units, of −6V even though
the actual units are volts squared.

The Integrate The intergral of an operand is calculated as a summation of
Example VOLTS/TIME.

Apply a +2V dc signal to channel 1. Set the timebase to 100
ms/DIV. Choose a function and select the Integrate math function.
Set the vertical scaling to 1V/DIV.

The actual units of the integrate function are VOLT Seconds/DIV,
however, the displays shows the fundamental units of VOLTS/DIV.
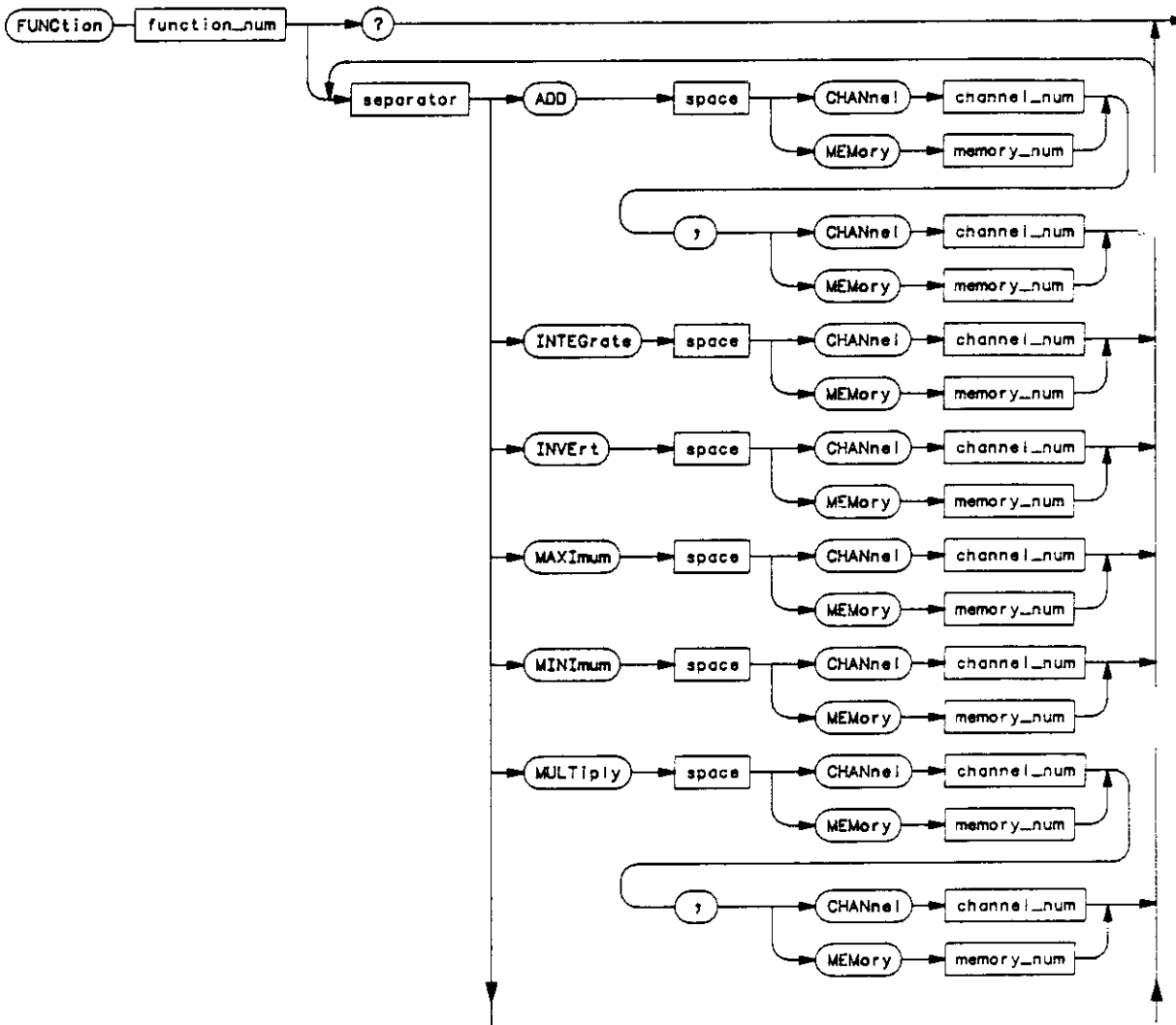
CHANNEL _ NUM = 1 or 2

FUNCTION _ NUM = 1 or 2

MEMORY _ NUM = 1 through 8

Figure 6-1. Function Subsystem Syntax Diagrams.

Function Subsystem
6-4

Figure 6-1. Function Subsystem Syntax Diagram (continued)
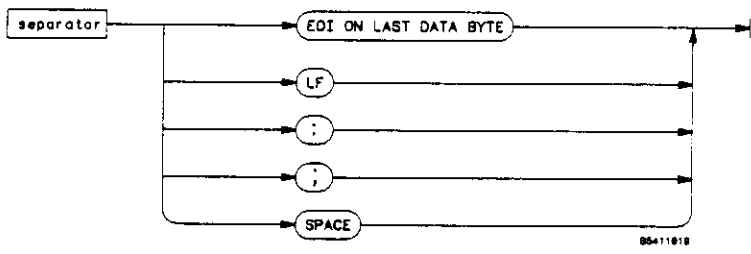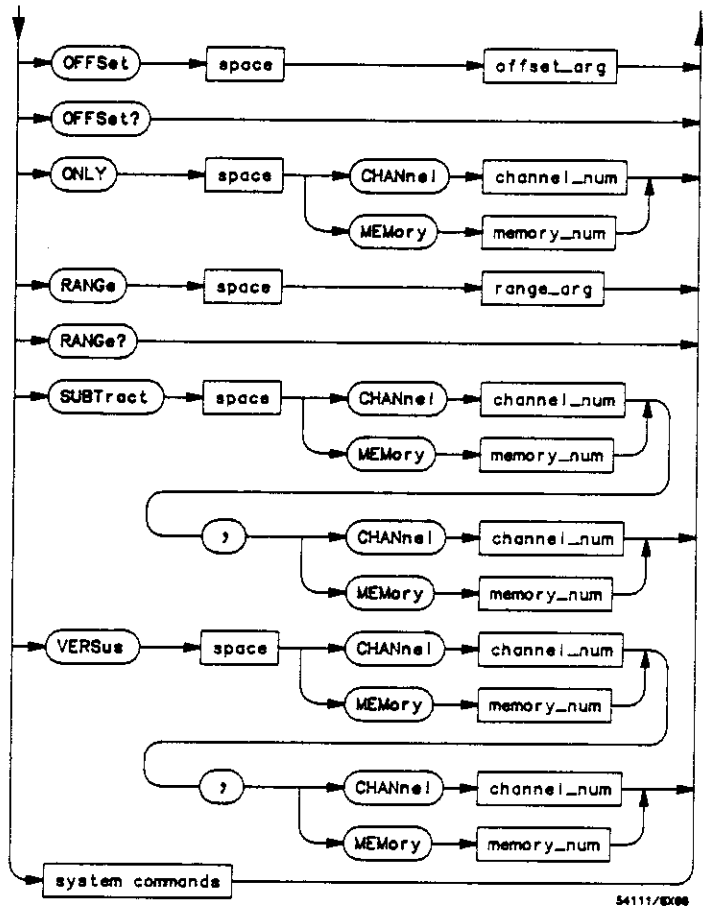
FUNCTion   **command/query**

The FUNCTION command selects the FUNCTION subsystem and the function to be used.

Function operations can only be used on data that was acquired in a like manner. That is, if channel 1 and channel 2 are being acquired in the repetitive acquisition mode, they can only be used with the repetitive memories 5-8 If channel 1 and channel 2 are being acquired in the real time mode, they can only be used with the real time memories 1-4.

The query returns the definition of the selected function.

Refer to Figure 6-1 for a syntax diagram of the FUNCTION subsystem commands.

Command Syntax: FUNCtion {1 | 2}

Example: OUTPUT 707;"FUNCTION 1 ADD CHANNEL 1, CHANNEL 2"

Query Syntax: [FUNCtion]{1 | 2}?

Returned Format: [FUNCtion]{1 | 2}<CR><LF>
{ADD | SUBtract | INVErt}
{CHANnel 1 | 2 | MEMory 1 | ... | 8 }
{CHANnel 1 | 2 | MEMory 1 | ... | 8 }<CR><LF>

Example: DIM Fun$[300]
OUTPUT 707;"EOI ON"
OUTPUT 707;"FUNCTION1?"
ENTER 707 USING "-K";Fun$

ADD   **command**

The ADD command algebraically sums the two defined operands.

Command Syntax: ADD<operand1>,<operand2>
                          <operand 1,2> ::=
                              {CHANnel1 | CHANnel2 | MEMory1 | 2 | ... | 8}

Example: OUTPUT 707;"FUNCTION1 ADD MEMORY3,MEMORY4"

INTEgrate   command

The INTEGRATE command calculates the integral of the vertical
value of the designated operand.

Command Syntax: INTEgrate<operand 1>

<operand> :: =

{channel { 1 | 2 } | Memory {1 | ...| 8}}

Example: OUTPUT 707;"FUNCTION1 INTEGRATE MEMORY1"

         command

The INVERT command inverts the operand. The possible operands are channel 1, channel 2, or memory 1 through memory 8.

Note that the short form of the command is INVE.

Command Syntax: INVErt<operand>

           <operand> ::=

           {channel 1 | 2 | Memory 1 | ...| 8}

     Example: OUTPUT 707;"FUNCTION2 INVERT MEMORY3"

MAXimum   **command**

The MAX command accumulates the maximum value of the first operand.

Command Syntax: :MAX<operand>

<operand>::= {CHANnel{1 | CHANnel2} | MEMory1 | ...| 8}}

Example: OUTPUT 707;"FUNCTION2 MAX CHANNEL1"

MINimum   **command**

The MIN command accumulates the minimum values of the
operand. This function is available only in the persistence display
mode.

Note

*For best results, make sure the persistance is not set to
infinite.*

Command Syntax: :FUNCtion<N>:MIN <operand>
                          <N> ::= 1 or 2
                      <operand> ::= {CHANnel1 | CHANnel2 |
                              WMEMory1 | WMEMory2 | WMEMory3 | WMEMory4}

Example: OUTPUT 707;"FUNCTION2 MIN CHANNEL3"

MULTiply   command

The MEMORY command calculates the voltage product of the two specified operands.

Command Syntax: MULTiply<operand 1>, <operand 2>

<operand 1,2> :: =
{CHANnel {1 | 2} | MEMory {1 | ... | 8}}

Example: OUTPUT 707;"FUNCTION1 MULTIPLY CHANNEL1,CHANNEL2"

OFFset   **command/query**

The OFFSET command set the voltage at center screen for the
selected function. The offset limit is 200 multiplied by the current
range setting.

Command Syntax: OFFSET <offset>
                     <N> :: = 1 or 2
                     offset:: = 0 to 200 × current range

Example: OUTPUT 707;"FUNCTION1 OFFSET 1"

Query Syntax: FUNCTION <N> OFFSET?
                   <N> :: = 1 or 2

Returned Format: [OFFSET] <offset><CR><Lf>
                      <offset> :: = 0 to 200 × current range

Example: DIM A$ [100]
              OUTPUT 707; "FUNCTION1 OFFSET?"
              ENTER 707; A$
              PRINT A$

ONLY   **command**

The ONLY command allows you to display the first operand of the
fuction. Vertical scalig is provided for all operands.

Command Syntax: ONLY <operand>
                        <operand> :: ={CHANnel1 | 2 |} MEMory {1 | ...| 8}}

Example: OUTPUT 707:"ONLY MEMORY1"

RANGe   **command/query**

The RANGe command defines the full-scale vertical axis of the selected function. The value is eight times the sensitivity when in the single screen display mode. In the dual screen display the range value is four times the sensitivity. In the quad screen mode, the range is two times the sensitivity value.

The RANGE query returns the current range setting to the specified function. Once the waveform signal is acquired, the function may be scaled to the following lists:

MATH FUNCTION	RANGE LIMITS
Only Add Subtract Invert	±8 mV to ±800 V
Multiply	80 pV to 40 kV
Integrate	80 pV to 20 kV

The range limit refers to the limit that a function can be scaled to. The input to channels 1 and 2 of the oscilloscope must be limited to 40 V when set to 1 MΩ input impedqance, and 5 V rms in 50 Ω. Trig 3 and 4 input must not exceed 10 V in 1 MΩ and 5 V rms in 50 Ω. If a larger input signal is to be viewed, add appropriate divider probes to the system.

Command Syntax: RANGe <range>
                  range :: = 80 pV to 40 kV depending on which math function is used.

Example: OUTPUT 707; "FUNCTION1 RANGE8"
(At RANGE 8, the sensitivity is set to 1 V/DIV if in single screen).

Query Syntax: RANGe <range>

Returned Format: [RANGe] <range><CR><LF>
                  <range> :: = 8 pV to 40 kV
                      range returned depends on math function used.

Example: — 10 DIM A$ [100]
— 20 OUTPUT 707; "FUNCTION1 RANGE?"
— 30 ENTER 707; A$
— 40 PRINT A$

Function Subsystem
6-15

SUBTract   **command**

The SUBTRACT command algebraically subtracts operand 2 from operand 1.

Command Syntax: SUBTract <operand1>,<operand2>

<operand 1,2> ::=

{CHANnel1 | CHANnel2 | MEMory 1 | ... | 8}

Example: OUTPUT 707;"FUNCTION2 SUBTRACT MEMORY8,MEMORY6"

In this example memory 6 would be algebraically subtracted from memory 8.

VERSus command

The VERSUS command draws a volt versus volts display of the two selected operands. Vertical scaling is ot available for this math functions. Measurement and markes are not useable in VERSUS.

Command Syntax: VERSus <operand1>, <operand2>
<operand1,2> :: =
{CHANel } 1 | ...| 8 }}

Example: OUTPUT 707;"FUNCTION1 VERSUS CHANNEL1,CHANNEL2"

Note

The versus math function can not be stored in memory.

7

Hardcopy Sybsystem

Introduction

The commands in the HARDCOPY subsystem set various parameters used during the plotting and printing of waveforms from the HP 54111D. Displayed waveforms, graticules, voltage and time markers, and waveform factors can be copied to an output device. The portion of waveform that is to be copied must be placed on the display.

To make the hardcopy print or plot, refer to the system commands PRINT and PLOT.

Subsystem Selection

The HARDCOPY subsystem must be selected prior to sending any HARDCOPY subsystem command or queries. This can be done by sending a single line command "HARDCOPY," or by sending the subsystem selector prior to another subsystem command or query. As an example, "HARDCOPY PAGE?" will select the HARDCOPY subsystem, and request the HP 54111D to indicate the status of the page command.

System Commands

A system command can be sent at any time while in the subsystem. After a system command has been sent, the instrument will return to the subsystem that was selected (in this case HARDCOPY) prior to the system command.

In this subsystem if a query is sent followed by another query without a controller read (ENTER) between the queries, only the last response will be received. Although this operation is syntactically correct, the HP 54111D does not queue query responses in this subsystem. Therefore, the response to the first query will be overwritten in the HP-IB output buffer.

Refer to figure 7-1 for the syntax diagrams of the HARDCOPY subsystem commands.
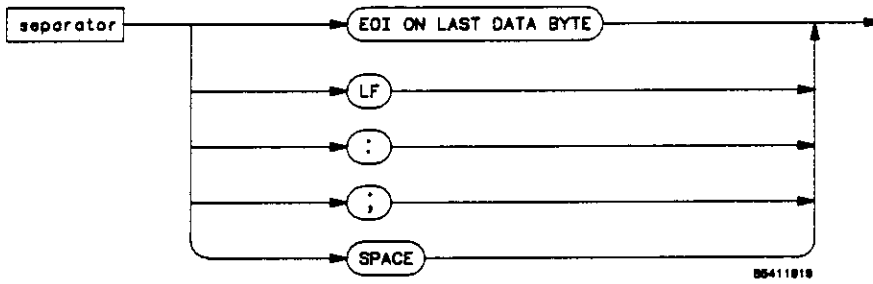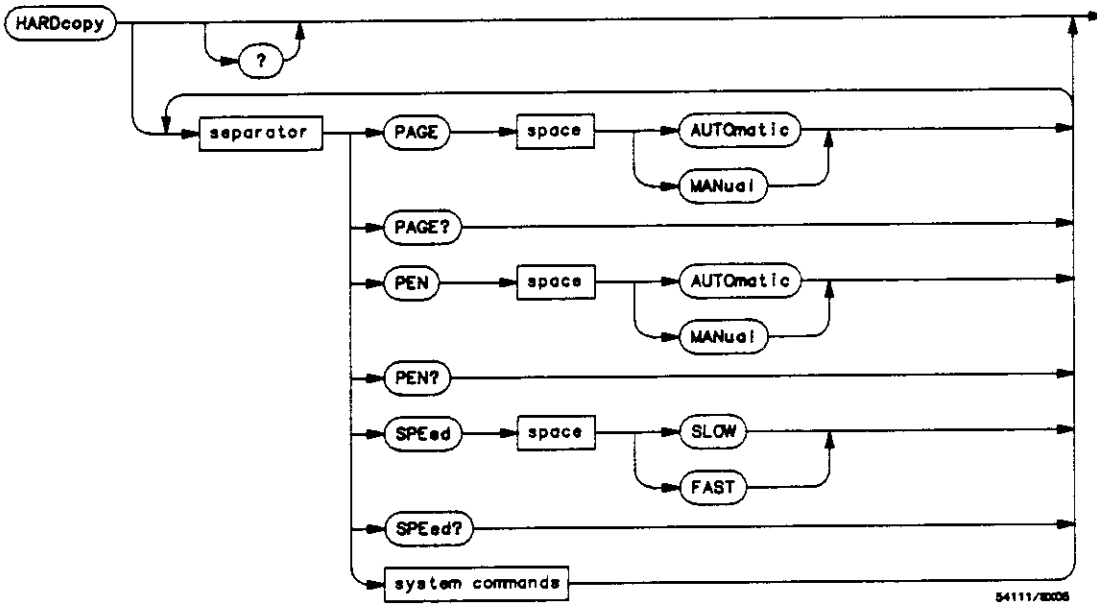
Figure 7-1. Hardcopy Subsystem Syntax Diagrams.

HARDcopy

command/query

The HARDCOPY command selects the HARDCOPY subsystem as the destination for the HARDCOPY subsystem commands.

The HARDCOPY query returns the current parameters for the HARDCOPY subsystem.

Command Syntax: HARDcopy

Example: OUTPUT 707;"HARDCOPY"

Query Syntax: HARDcopy?

Returned Format:
[HARDcopy]<CR><LF>
[PAGE]<argument><CR><LF>
[PEN]<argument><CR><LF>
[SPEed]<argument><CR><LF>

Example:
DIM Hard$[100]
OUTPUT 707;"HEADER ON EOI ON"
OUTPUT 707;"HARDCOPY?"
ENTER 707 USING "-K";Hard$
PRINT USING "K";Hard$"

PAGE   command/query

The PAGE command sets up the HP 54111D to send a form feed
after a hardcopy output to a printer. During a hardcopy output the
HP 54111D ignores page boundaries.

The query returns the current state of the page command.

Command Syntax: PAGE {MANual | AUTOmatic}

Example: OUTPUT 707;"PAGE AUTO"

Query Syntax: PAGE?

Returned Format: PAGE|<state><CR><LF>

                    <state> ::= {MANual | AUTOmatic}

Example: DIM Page$[30]
OUTPUT 707;"PAGE?"
ENTER 707;Page$
PRINT Page$

The PEN command sets the oscilloscope's pen control function. When this command is set to AUTOMATIC, the oscilloscope will instruct the plotter to get a different pen for different parts of the drawing. Up to six pens can be used for different portions of the plot. When this command is set to AUTOMATIC, the HP 54111D assigns the following pen numbers to these functions:

Pen #	Usage
1	Graticule, timebase factors, and Function 1 and associated factors
2	Channel 1 and associated factors
3	Waveform memories and associated factors, and pixel memories 9-10
4	Channel 2 and associated factors
5	Markers and delta measurement results
6	Function 2 and associated factors

When the command is set to the MANUAL mode the plotter will not be instructed to select a pen when a plot is requested. At the completion of the plot an instruction will be sent to cause the plotter to put away the pen.

The PEN query returns the state of the pen control command.

Command Syntax: PEN {MANual | AUTOmatic}

Example: OUTPUT 707;"PEN AUTOMATIC"

Query Syntax: PEN?

Returned Format: [PEN]<state><CR><LF>
        <state> ::= {MANual | AUTOmatic}

Example: DIM Pen$[30]
        OUTPUT 707;"PEN"$
        ENTER 707;PEN$
        PRINT Pen$

command/query

The PRINTER command selects the black and white or the color printer.

Except for four exceptions, the PaintJet color assignments correspond to those on the display and the color menu of the oscilloscope. These exceptions are necessary for better viewability. This means that changing color settings in the color menu, will in some cases change different elements of the output on the PaintJet Printer. The assignments are:

DISPLAY	COLOR No.	PAINTJET
Stored traces (white)	7	Halfbright
Background (black)	0	White
Overlap (magenta)	8,11-15	Black (Color 0)
Channel 1 (yellow)	4	Magenta (Color 8)

EXAMPLE: If the background of the color menu is changed to blue, the PaintJet background will remain white, while the overlap will become blue.

The RESET command does not alter the state of the printer parameter, however, a keydown power up will set the printer selection to MONO.

The printer selection is not saved into the HP 54111D Learn String.

Command Syntax: PRINTER {MONO | COLOR}

Example: OUTPUT 707;"PRINTER MONO"

Query Syntax: PRINTER?

Returned Format: [PRINTER] {MONO | COLOR} <CR> <LF>

Note

There is no shortform HP-IB command for printer selection.

SPEed

<div align="right">

command/query

</div>

The SPEED command specifies the pen speed to be used during plotting. FAST is intended for use on normal paper and SLOW should be used when plotting transparencies.

The query returns the current pen speed.

Command Syntax: SPEed {SLOW | FAST}

Example: OUTPUT 707;"SPEED FAST"

Query Syntax: SPEed?

Returned Format: [SPEed] < state > < CR > < LF >

< state > :: = {SLOW | FAST}

Example:
```
DIM Speed$[30]
OUTPUT 707;"SPEED"
ENTER 707;Speed$
PRINT Speed$
```

8

Measure Subsystem

Introduction

The commands in the MEASURE subsystem are used to make parametric measurements on displayed waveforms, and to report the settings of the voltage and time markers. Some of the commands in this subsystem can be used to set the voltage and time markers to specified voltages, times, or defined events. An example of a defined event is the time at which a waveform crosses a specified voltage level (TVOLt? query). In this case, the time at the voltage crossing is determined and reported to the controller.

Measurement Setup

Time measurements typically should be made at the fastest possible sweep speed, in order to obtain the most measurement accuracy possible. For any measurement to be made, the portion of the waveform required for that measurement must be displayed on the oscilloscope. That is:
- if a period or frequency measurement is to be made, at least one complete cycle must be displayed.
- if a pulse width measurement is to be made, that entire pulse must be displayed.
- if a risetime measurement is to be made, the leading (positive-going) edge of the waveform must be displayed.
- if a falltime measurement is to be made, the trailing (negative-going) edge of the waveform must be displayed.

Measurement Error

If a measurement cannot be made, typically because the proper portion of the waveform is not displayed, the value returned for that parameter is 1.00000E38. This is an error value that is output when a measurement cannot be made or the HP-IB output buffer is empty.

Making Measurements

If more than one waveform, edge, or pulse is displayed, time measurements are made on the first (leftmost) portion of the displayed waveform that can be used.

When any of the defined measurements are requested, the oscilloscope first determines the top (100%) and base (0%) voltages of the waveform. From this information, it can determine the other important voltage values (10% voltage, 90% voltage, and 50% voltage) required to make the measurements. The 10% and 90% voltage values are used in the rise time and fall time measurements. The 50% voltage value is used for measuring frequency, period, pulse width, and duty cycle.

All voltage values are returned in volts. Voltage values returned are measured with zero volts as the reference. The value returned for the VDELta? query is the voltage difference between the VMarker 1 and the VMarker 2.

All time values are returned in seconds. Time values returned are measured with the trigger point (time 0) as the reference. The value returned for TDELta is the time difference between the stop and start markers.

Measurement Queries

In the MEASURE subsystem, measurement results (i.e., automatic parametric measurements) are stored and can be output together. This means that several measurement queries can be sent to the oscilloscope prior to reading (ENTER command) the measurement results. An example of how this is done is shown below.

```
10 OUTPUT 707;"MEASURE SOURCE CHANNEL 1"
20 OUTPUT 707;"VPP? VAV? FREQ? PWID?"
30 ENTER 707;Vpp,Vav,Freq,Pwid
40 PRINT Vpp,Vav,Freq,Pwid
50 END
```

In this example, line 10 selects the measure subsystem and selects the measurement source as channel 1. Line 20 selects the measurements to be made. The measurements are made and the results are stored until read by the controller. Line 30 reads the measurement results, one at a time, into the controller and places them in the numeric variables. Line 40 prints the contents of the numeric variables to the controller display on a single line using the zone print format. It should be noted that the queries in line 20 could have been sent in line 10, after the source selection, with the same result.

Measurement Source Measurements are made on the displayed waveform(s) specified by the SOURCE command. The SOURCE command allows two sources to be specified. When two sources are specified, VMarker 1 and the start marker are assigned to the first specified source and VMarker 2 and the stop marker are assigned to the second specified source. The measurements that use two sources are: ESTART, ESTOP, TSTART, TSTOP, TDELTA, VSTART, VSTOP, VDELTA, PBASE, PTOP, VRELATIVE, VMARKERSET, and VFIFTY.

Some measurements can only be made on a single source. If one of these measurements is made with two sources specified, the measurement is made on the first source and the source specified is changed to a single source.

Subsystem Selection The MEASURE subsystem must be selected prior to sending any MEASURE subsystem commands or queries. This can be done by sending a single line command "MEASURE" or by sending the subsystem selector prior to another command or query. As an example, "MEASURE ALL?" will select the MEASURE subsystem and ask for the instrument to measure and output the results of all parameters of the displayed waveform.

System Command A system command can be sent at any time while in this subsystem. After a system command has been sent, the instrument will return to the subsystem that was selected (in this case MEASURE) prior to the system command.

Refer to figure 8-1 for the syntax diagrams of the MEASURE subsystem.
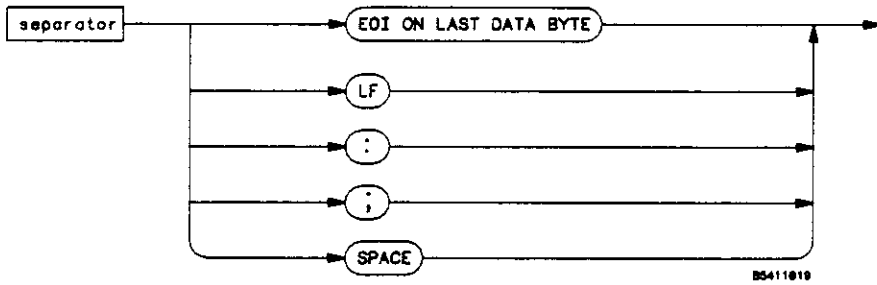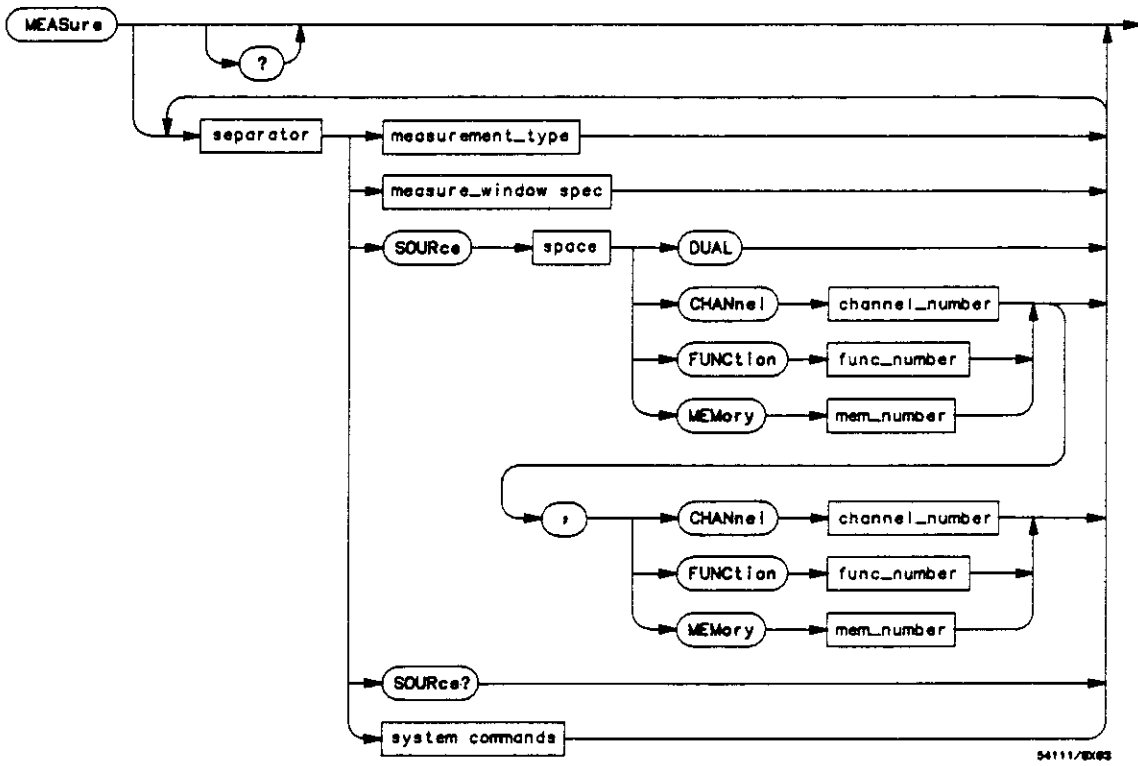
Figure 8-1. Measure Subsystem Syntax Diagrams.

Measure Subsystem
8-4
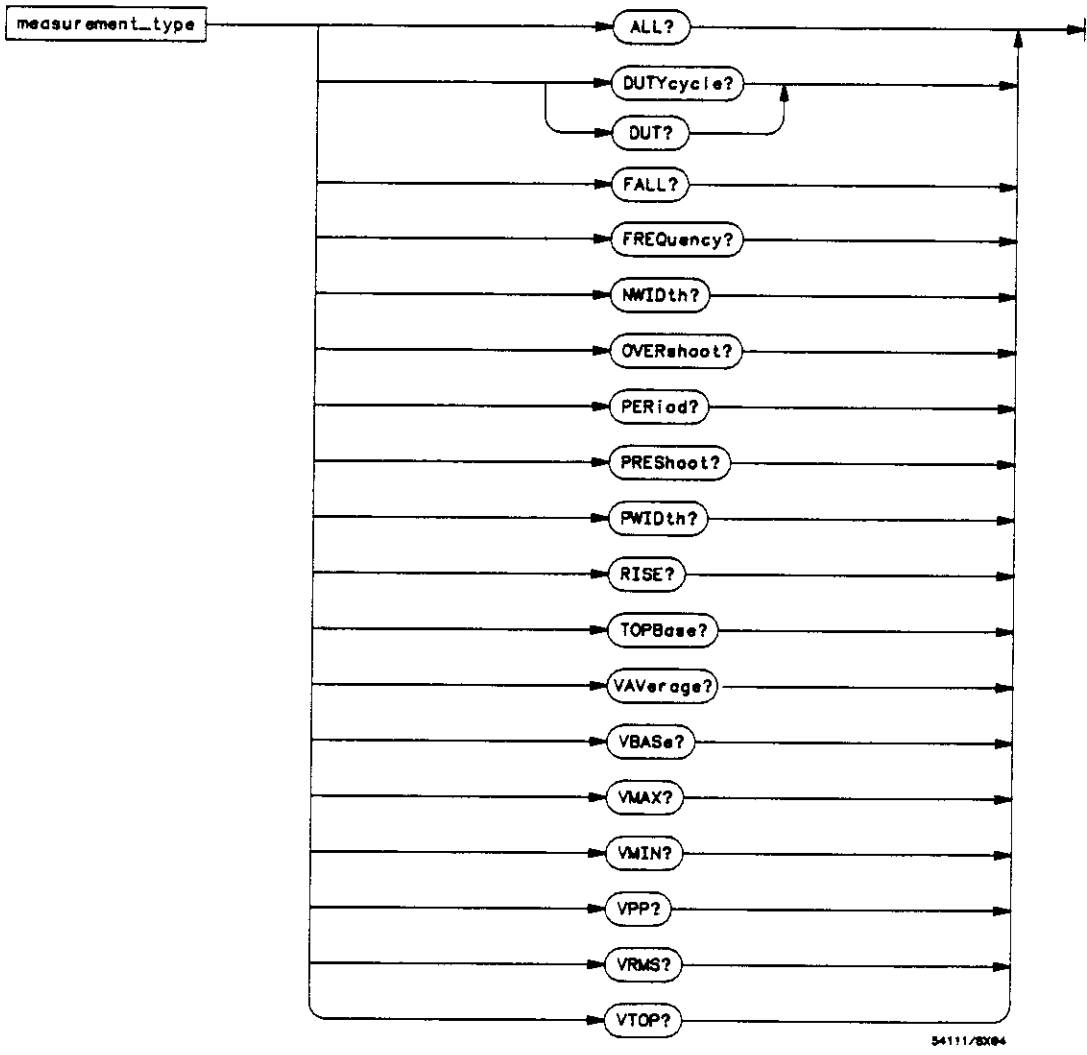
Figure 8-1. Measure Subsystem Syntax Diagrams (continued).

Figure 8-1. Measure Subsystem Syntax Diagrams (continued).
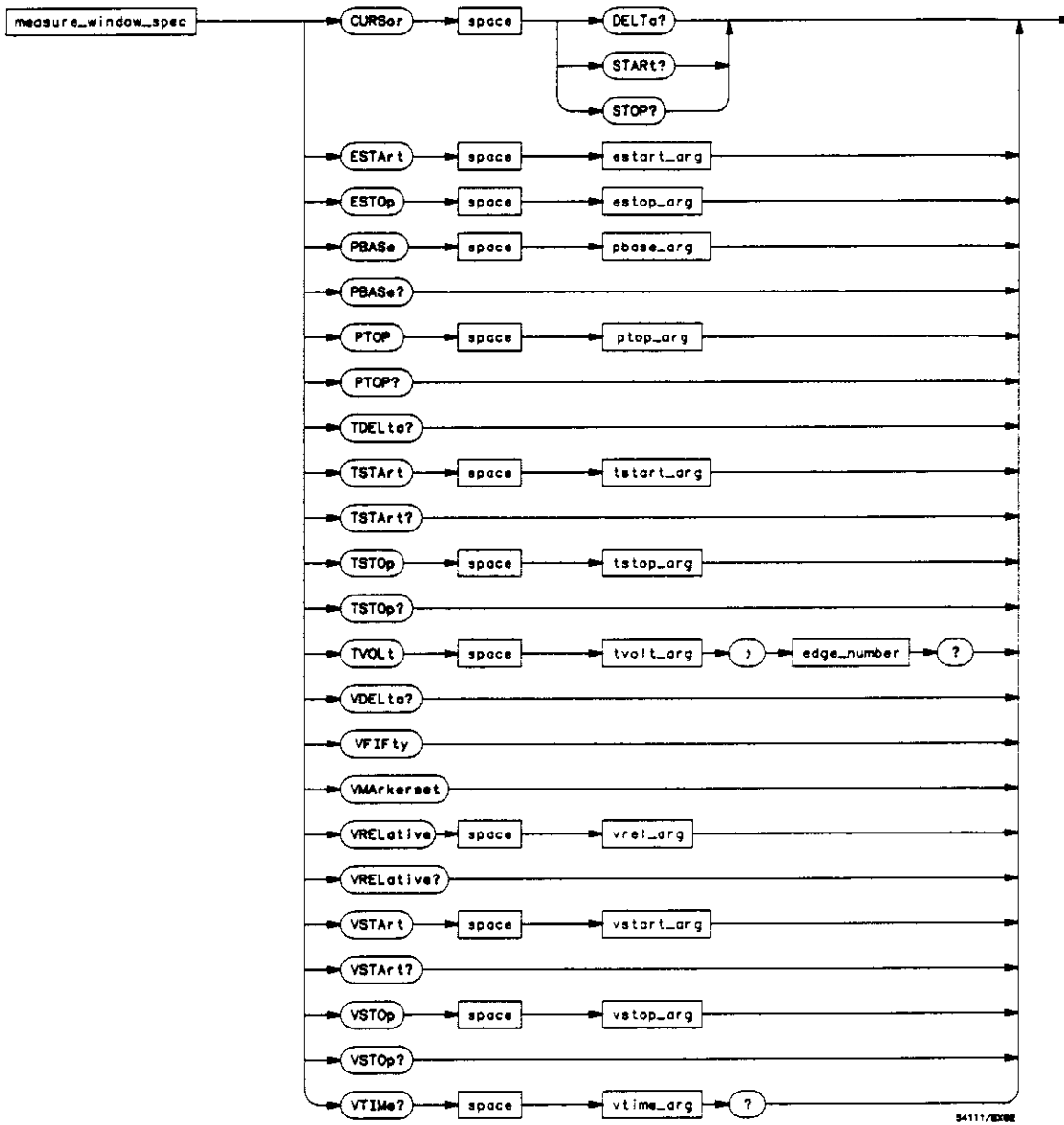
Measure Subsystem
8-6

CHANNEL__NUMBER = An integer, 1 or 2.

EDGE__NUMBER = An integer, 1 through 2000 and −1 through −2000.

ESTART__ARG = An integer, 1 through 2000 and −1 through −2000.

ESTOP__ARG = An integer, 1 through 2000 and −1 through −2000.

FUNC__NUMBER = An integer, 1 or 2.

MEM__NUMBER = An integer, 1 through 8.

PBASE__ARG = An integer, -25 through 125.

PTOP__ARG = An integer, -25 through 125.

TSTART__ARG = A real number. The range of this argument depends on the sweep speed.

TSTOP__ARG = A real number. The range of this argument depends on the sweep speed.

TVOLT__ARG = A real number that is the time at a specified voltage crossing.

VREL__ARG = An integer 0, 10, 20, 50, 80, 90, or 100.

VSTART__ARG = A real number ≤2 X voltage range.

VSTOP__ARG = A real number ≤2 X voltage range.

VTIME__ARG = A real number that is within the horizontal display window.

Figure 8-1. Measure Subsystem Syntax Diagrams (continued).

MEASure   **command/query**

The MEASURE command selects the MEASURE subsystem as the destination for the MEASURE subsystem commands. When this command is sent, no measurements are made.

The MEASURE query responds with selected measurement parameters. Again, no measurements are made when this query is sent. The values returned are the current values (from the previous measurement) of the parameters.

Command Syntax: MEASure

Example: OUTPUT 707;"MEASURE"

Query Syntax: MEASure?

Returned Format: [MEASure]<CR><LF>
[SOURce][CHANnel]<NR1>|[FUNCtion]<NR1>|[MEMory]<NR1><CR><LF>
[PRECision]<argument><CR><LF>
[VDELta]<NR3><CR><LF>
[VSTArt]<NR3><CR><LF>
[VSTOp]<NR3><CR><LF>
[TDELta]<NR3><CR><LF>
[TSTArt]<NR3><CR><LF>
[TSTOp]<NR3><CR><LF>
[PBASe]<NR3><CR><LF>
[PTOP]<NR3><CR><LF>

Example: DIM Mea$[200]
OUTPUT 707;"EOI ON HEADER ON"
OUTPUT 707;"MEASURE?"
ENTER 707 USING "-K";Mea$
PRINT USING "K";Mea$

query

The ALL query makes a set of measurements on the displayed signal and buffers the measurement results for output over the HP-IB. The measurement results are displayed on the CRT if the Measure menu is selected.

For a measurement to be made, the portion of the waveform required for the measurement must be displayed. Measurements are made on the first (left most) displayed waveform. As a general rule, the sweep speed should be set as fast as possible with one complete cycle on the display. This is to obtain the best possible accuracy for the measurement.

If a parameter cannot be measured, the instrument responds with 1.00000E + 38 for that measurement result. All parameters that can be measured will have the correct value as the measurement result.

Refer to the individual commands for information on how the measurements are made and the returned format of the measurement results.

ALL?

Query Syntax: ALL?

Returned Format: [FREQuency]<NR3><CR><LF>
[PERiod]<NR3><CR><LF>
[PWIDth]<NR3><CR><LF>
[NWIDth]<NR3><CR><LF>
[RISE]<NR3><CR><LF>
[FALL]<NR3><CR><LF>
[TOPBase]<NR3><CR><LF>
[VPP]<NR3><CR><LF>
[PREShoot]<NR3><CR><LF>
[OVERshoot]<NR3><CR><LF>
[DUTYcycle]<NR3><CR><LF>
[VRMS]<NR3><CR><LF>
[VMAX]<NR3><CR><LF>
[VMIN]<NR3><CR><LF>
[VTOP]<NR3><CR><LF>
[VBASe]<NR3><CR><LF>
[VAVerage]<NR3><CR><LF>

Example: DIM All$[500]
OUTPUT 707;"EOI ON"
OUTPUT 707;"ALL?"
ENTER 707 USING "-K";"All$"
PRINT USING "K";All$

Note

*These values could be returned to numeric variables
instead of the string variable as shown. If a numeric
variable is used, the headers are ignored by the controller.*

CURSor?   query

The CURSOR query returns the time and voltage values of the specified marker as an ordered pair of time/voltage values.

If delta is specified:
- the instrument returns the value of delta V and delta T.
  If start is specified:
- the positions of VMarker 1 and the start marker are returned.
  If stop is specified:
- the positions of VMarker 2 and the stop marker are returned.

Query Syntax: CURSor {DELTa | STARt | STOP}?

Returned Format: [CURSor]<time>,<voltage><CR><LF>

<time> ::= delta time or start time or stop time

<voltage> ::= delta voltage or V(1) voltage or V(2) voltage

Example: OUTPUT 707;"MEAS SOURCE CH1"
OUTPUT 707;"CURSOR START?"
ENTER 707;Tme,Vlt
PRINT Tme,Vlt

DUTYcycle? | DUT?   **query**

The DUTYCYCLE query measures and outputs the duty cycle of the displayed signal. The value returned for duty cycle is a percentage. The positive pulse width and the period of the displayed signal are measured, then the duty cycle is calculated. The duty cycle is calculated using the following formula:

$$\textbf{duty cycle} = \textbf{(+pulse width/period) X 100}$$

Query Syntax: {DUTYcycle | DUT}?

Returned Format: [DUTYcycle]<value><CR><LF>

<value> ::= percentage of +pulse width to period
(exponential - NR3 format)

Example: OUTPUT 707;"DUTYCYCLE?"
ENTER 707;Dut
PRINT Dut

The ESTART command causes the instrument to position the start marker on the specified edge and slope of the displayed waveform. The start marker is positioned where VMarker 1 intersects the waveform. The desired edge is specified by sending an integer value after the command name. If a positive integer is sent, the oscilloscope will place the start marker on a positive-going waveform edge. If a negative integer is sent, the start marker will be placed on a negative-going waveform edge.

If the VMarker 1 does not intersect the waveform as specified, the error message "Edges required for measurement not found" is displayed.

The ESTART query responds with the currently specified edge.

Command Syntax: ESTArt<edge>

                          <edge> ::=sign and number

(if a positive value is sent, the + sign may be omitted or a space may be used)

Example: OUTPUT 707;"ESTART 2"

This example places the start marker at the second positive-going intersection of the waveform and VMarker 1.

Query Syntax: ESTArt?

Returned Format: [ESTART]<value><CR><LF>

                          <value> ::= {sp | -}specified edge number
                                       (integer - NR1 format)

Example: OUTPUT 707;"ESTART?"
ENTER 707;Estart
PRINT Estart

The ESTOP command causes the instrument to position the stop
marker on the specified edge and slope of the displayed waveform.
The stop marker is positioned where VMarker 2 intersects the
waveform. The desired edge is specified by sending an integer value
after the command name. If a positive integer is sent, the
oscilloscope places the stop marker on a positive-going waveform
edge. If a negative integer is sent, the stop marker is placed on a
negative-going waveform edge.

If VMarker 2 does not intersect the waveform as specified, the error
message "Edges required for measurement not found" is displayed.

The ESTOP query responds with the currently specified edge.

Command Syntax: ESTOP<edge>
                          <edge> ::= sign and number

(if a positive value is sent, the + sign may be omitted or a space
may be used)

Example: OUTPUT 707;"ESTOP-2"

This example places the stop marker at the second negative going
intersection of the waveform and VMarker 2.

Query Syntax: ESTOP?

Returned Format: [ESTOP]<value><CR><LF>
                          <value> ::= {sp | -}specified edge number
                                          (integer - NR1 format)

Example: OUTPUT 707;ESTOP?
                   ENTER 707;Estop
                   PRINT Estop

FALL?

query

The FALL query causes the instrument to measure and output the falltime of the first displayed falling edge. Falltime is measured between the 10% and 90% points of the falling (negative-going) edge. To obtain the best possible measurement accuracy, set the sweep speed as fast as possible while leaving the falling edge of the waveform on the display. The falltime is calculated using the following formula:

falltime = time at 10% point - time at 90% point.

Query Syntax: FALL?

Returned Format: [FALL]<value><CR><LF>

<value> ::= time in seconds between 10% and 90% voltage points (exponential - NR3 format)

Example: OUTPUT 707;"FALL?"
ENTER 707;Fall
PRINT Fall

FREQuency?   **query**

The FREQUENCY query measures and outputs the frequency of the first complete cycle on screen using the 50% levels. The algorithm used is:

> **if first edge on screen is rising**
> > **then**
> > > **frequency = 1/(time at second rising edge**
> > > **- time at first rising edge)**
> > **else**
> > > **frequency = 1/(time at second falling edge**
> > > **- time at first falling edge)**

Query Syntax: FREQuency?

Returned Format: [FREQuency]<value><CR><LF>

> <value> ::= frequency in Hertz
> (exponential - NR3 format)

Example: OUTPUT 707;"FREQuency?"
ENTER 707;Freq
PRINT Freq

query

The NWIDTH query measures and outputs the width of the first negative pulse on screen using the 50% levels. The algorithm used is:

> **if the first edge on screen is rising**
> > **then**
> > > **width** = (time at second rising edge
> > >   - time at first falling edge)
> > **else**
> > > **width** = (time at first rising edge
> > >   - time at first falling edge)

Query Syntax: NWIDth?

Returned Format: [NWIDth]<value><CR><LF>

> <value> ::= negative pulse width in seconds
> (exponential - NR3 format)

Example: OUTPUT 707;"NWIDTH?"
ENTER 707;Nwid
PRINT Nwid

OVERshoot?   query

The OVERSHOOT query measures and outputs the overshoot of a selected signal. Overshoot measures the first edge on screen using the following algorithm:

if the first edge on screen is rising
   then
      overshoot = Vmax - Vtop
   else
      overshoot = Vbase - Vmin

Query Syntax: OVERshoot?

Returned format: [OVERshoot]<value><CR><LF>

<value> ::= overshoot in volts
(exponential - NR3 format)

Example: OUTPUT 707;"OVERSHOOT?"
ENTER 707;Over
PRINT Over

The PBASE command sets VMarker 1 to a percentage of the last established position of the voltage markers. For example, after a TOPBASE operation, VMarker1 would be located at the base (0%) of the signal and VMarker2 at the top (100%) of the signal. If a PBASE 30 command is executed, VMarker1 is moved to the 30% level of the signal. The range of the argument for this command is an integer from −25 to 125. This command also sets the Preset Levels control on the front panel to Variable.

The PBASE query returns the current value of the PBASE setting.

Command Syntax: PBASe<value>

<value> ::= -25% to 125%

Example: OUTPUT 707;"PBASE10"

Query Syntax: PBASE?

Returned Format: [PBASE]<value><CR><LF>

<value> ::= V(1) setting in percent of top-base
(integer - NR1 format)

Example: OUTPUT 707;PBASE?
ENTER 707;Pbase
PRINT Pbase

PERiod? query

The PERIOD command measures and outputs the period of the first complete cycle on screen. The period is measured at the 50% voltage level of the waveform. The algorithm for this measurement is:

> **if the first edge on screen is rising**
>> **then**
>>> **period = (time at second rising edge**
>>>> **- time at first rising edge)**
>>
>> **else**
>>> **period = (time at second falling edge**
>>>> **- time at first falling edge)**

Query Syntax: PERiod?

Returned format: [PERiod]<value><CR><LF>

<value> ::= waveform period in seconds
(exponential - NR3 format)

Example: OUTPUT 707;"PERIOD?"
ENTER 707;Period
PRINT Period

PREShoot?   query

The PRESHOOT query measures and outputs the preshoot of the selected signal. Preshoot measures the first edge on screen using the following algorithm:

if the first edge on screen is rising
  then
    preshoot = Vbase - Vmin
  else
    preshoot = Vmax - Vtop

Query Syntax: PREShoot?

Returned Format: [PREShoot]<value><CR><LF>

                <value> ::= preshoot in volts
                                 (exponential - NR3 format)

Example: OUTPUT 707;"PRESHOOT?"
             ENTER 707;Pres
             PRINT Pres

The PTOP command sets VMarker2 to a percentage of the last established position of the voltage markers. For example, after a TOPBASE operation, VMarker1 would be located at the base (0%) of the signal and VMarker2 at the top (100%) of the signal. If a PTOP 70 command is executed, VMarker2 is moved to the 70% level of the signal. The range of the argument for this command is an integer from −25 to 125. This command also sets the Preset Levels control on the front panel to Variable.

The PTOP query returns the current value of the PTOP setting.

Command Syntax: PTOP<value>

<value> ::= -25% to 125%

Example: OUTPUT 707;"PTOP90"

Query Syntax: PTOP?

Returned Format: [PTOP]<value><CR><LF>
<value> ::= V(2) setting in percent of top-base
(integer - NR1 format)

Example: OUTPUT 707;PTOP?
ENTER 707;Ptop
PRINT Ptop

PWIDth?

query

The PWIDTH query measures and outputs the width of the first displayed positive pulse. Pulse width is measured at the 50% voltage level. The algorithm for this measurement is:

if the first edge on screen is falling
    then
        width = (time at second falling edge
                - time at first rising edge)
    else
        width = (time at first falling edge
                - time at first rising edge)

Query Syntax: PWIDth?

Returned Format: [PWIDth]<value><CR><LF>

           <value> ::= width of positive pulse in seconds
                    (exponential - NR3 format)

Example: OUTPUT 707;"PWIDTH?"
         ENTER 707;Pwid
         PRINT Pwid

RISE?   **query**

The RISE query measures and outputs the risetime of the first displayed rising (positive-going) edge. To obtain the best possible measurement accuracy, set the sweep speed as fast as possible while leaving the leading edge of the waveform on the display. The risetime is determined by measuring the time at the 10% and 90% voltage points on the rising edge, and then the rise time is calculated using the formula:

risetime = (time at 90% point - time at 10% point)

Query Syntax: RISE?

Returned Format: [RISE]<value><CR><LF>

<value> ::= rise time in seconds
(exponential - NR3 format)

Example: OUTPUT 707;"RISE?"
ENTER 707;Rise
PRINT Rise

The SOURCE command selects the source(s) to be used for the measurements. If the source is specified as channel 1 or channel 2, that channel will be used as the source for the MEASURE subsystem commands.

For two source measurements, two parameters are specified after the source command. When two sources are specified, VMarker 1 and the start marker are assigned to the first specified source, and VMarker 2 and the stop marker are assigned to the second specified source. If the keyword DUAL is used as the measurement source, VMarker 1 and the start marker are assigned to channel 1 and VMarker 2 and the stop marker are assigned to channel 2.

The MEASURE commands that can be used when two sources are specified are: ESTART, ESTOP, TSTART, TSTOP, TDELTA, VSTART, VSTOP, VDELTA, PBASE, PTOP, VMARKERSET, VRELATIVE, and VFIFTY.

If two sources, or dual, have been specified, and a measurement is made using a command that can only use one source, then the measurement is made on the first specified source and the source selection is changed.

Generally, in the repetitive mode, HP-IB measurements should not be made on live data. The waveforms should first be acquired into the memories using the DIGITIZE command, and then measurements should be made on the memories.

The reason for not making measurements on live data is that often the waveforms have been cleared when a control such as time/div or vertical offset has been changed and the controller may be so fast that new data has not been acquired to the completion criteria or number of averages desired when the measurement is made.

The SOURCE query returns the current source selection.

Command Syntax: {SOURce | SRC}{[DUAL | <source1> | <source1>, <source2>]

                                      <source1> and <source2> :: =

                                        {CHANnel1 | CHANnel2 | FUNCtion1 | FUNCtion2
                                                                MEMory1 | ... | MEMory8}

Example: OUTPUT 707;"SOURCE CHANNEL1, MEMORY1"

Query Syntax: {SOURce | SRC}?

Returned Format: [SOURce|SRC]{DUAL |<source1> |<source1>, <source2>}<CR><LF>

                                        <source1> and <source2> :: =

                                        {CHANnel1 | CHANnel2 | FUNCtion1 | FUNCtion2 |
                                                                MEMory1 | ... | MEMory8}

Example: DIM Src$[50]
                OUTPUT 707;"SRC?"
                ENTER 707;Src$
                PRINT Src$

                                                  query

The TDELTA query outputs the delta time value. The delta time value is the time difference between the start and stop markers. No measurement is made when this query is received by the oscilloscope. The delta time value is the current value (from previous settings) of the delta time.

Query Syntax: TDELta?

Returned Format: [TDELta]<value><CR><LF>

                      <value> ::= time difference between start and stop marker.
                                      (exponential - NR3 format)

Example: OUTPUT 707;"TDELTA?"
              ENTER 707;Tdel
              PRINT Tdel

TOPBase?   **query**

The TOPBASE query returns the top-base amplitude of the displayed signal. The top-base value is calculated using the formula:

$$amplitude = Vtop \cdot Vbase$$

For more information on how measurements are made refer to Appendix A.

Query Syntax: TOPBase?

Returned Format: [TOPBase]<value><CR><LF>

                                                        <value> ::= difference between top voltage and
                                                                base voltage - delta V value
                                                                (exponential - NR3 format)

Example: OUTPUT 707;"TOPBASE?"
ENTER 707;Topb
PRINT Topb

TSTArt   **command/query**

The TSTART command moves the start marker to the specified time with respect to the trigger time. The specified time can be a positive or negative number. To set the value to a positive value, either a + or <sp> can be used. To set the value to a negative value, a - sign must be sent with the number. The range for this value is -100,000 seconds to 100,000 seconds.

The TSTART query returns the start marker position.

Note

The short form of this command does not follow the defined convention. The short form "TST" is the system command for TEST, therefore be careful not to send this form for the TSTART command.

Command Syntax: TSTArt<start marker time>
                                         <start marker time> ::= time at start marker in seconds

Example: OUTPUT 707;"TSTArt -.001"

Query Syntax: TSTArt?

Returned Format: [TSTArt]<value><CR><LF>
                                         <value> ::= time at start marker in seconds
                                                            (exponential - NR3 format)

Example: OUTPUT 707;"TSTART?"
                        ENTER 707;Tsta
                        PRINT Tsta

<div align="right">**command/query**</div>

The TSTOP command moves the stop marker to the specified time with respect to the trigger time. The specified time can be a positive or negative number. To set the value to a positive value, either a + or <sp> can be used. To set the value to a negative value, a - sign must be sent with the number. The range for this value is -100,000 seconds to 100,000 seconds.

The TSTOP query returns the Stop marker position.

Note

The short form of this command does not follow the defined convention. The short form "TST" is the system command for TEST, therefore be careful not to send this form for the TSTOP command.

Command Syntax: TSTOp<stop marker time>
<stop marker time> ::= time at stop marker in seconds

Example: OUTPUT 707;"TSTOP -1.0E-6"

Query Syntax: TSTOp?

Returned Format: [TSTOp]<value><CR><LF>
<value> ::= time at stop marker in seconds
(exponential - NR3 format)

Example: OUTPUT 707;"TSTOP?"
ENTER 707;Tsto
PRINT Tsto

When the TVOLt query is sent, the displayed signal is searched for the defined voltage level and transition. The time interval between the trigger event and this defined occurrence is returned as the response to this query.

The <voltage> can be specified as a negative or positive voltage. To specify a negative voltage, use a − sign. The sign of <slope> selects a rising(+) or falling(−) edge.

The magnitude of <occurrence> defines the occurrence to be reported. For example, if <slope> is positive and <occurrence> is set to three, the third time the waveform crosses the specified voltage level in the positive direction will be found. Once this voltage crossing is found, the oscilloscope will output the time at that crossing in seconds, with the trigger point (time zero) as the reference.

If the specified crossing cannot be found, the oscilloscope outputs 1.00000E38. This would happen if the waveform does not cross the specified voltage, or if the waveform does not cross the specified voltage the specified number of times in the specified direction on the screen.

Query Syntax: TVOLt<voltage>,<slope><occurrence>?

           <voltage> ::= voltage level the waveform must cross; this can be a
                      positive or negative voltage
           <slope> ::= direction of waveform when <voltage> is crossed;
                      rising (sp or +) or falling(−)
           <occurrence> ::= number of crossing to be reported; (if one, first crossing
                      is reported, if two, second crossing is reported, etc.)

Returned Format: [TVOLt]<time><CR><LF>

           <time> ::= time in seconds of specified voltage crossing
                    (exponential · NR3 format)

Example: OUTPUT 707;"TVOLT − .250, + 3?"
           ENTER 707;Tvolt
           PRINT TVolt

VAVerage? query

The VAVERAGE query finds and outputs the average voltage of the first cycle of the displayed signal. If a complete cycle is not present, the instrument will average the data points on screen.

Note

The measurement is made with the points available on the display. If a complete cycle is not displayed, the value returned will NOT be a true average voltage for the waveform. However, it will be the average of the displayed points.

Query Syntax: VAVerage?

Returned Format: [VAVerage]<value><CR><LF>

<value> ::= average value of displayed data points in volts
(exponential - NR3 format)

Example: OUTPUT 707;"VAVERAGE?"
ENTER 707;Vave
PRINT Vave

VBASe?   **query**

The VBASE query measures and outputs the voltage value at the base of the waveform.

Query Syntax: VBASe?

Returned format: [VBASe]<value><CR><LF>

                      <value> ::= voltage at base of selected waveform.
                                    (exponential - NR3 format)

Example: OUTPUT 707;"VBASE?"
                 ENTER 707;Base
                 PRINT Base

VDELta?   **query**

The VDELTA query outputs the delta voltage between VMarker 1 and VMarker 2. No measurement is made when the VDELTA query is received by the oscilloscope. The delta voltage value that is output is the current value. This is the same value as the front panel delta V.

VDELta = voltage at VMarker 2 - voltage at VMarker 1

Query Syntax: VDELta?

Returned Format: [VDELta]<value><CR><LF>

                        <value> ::= delta V value in volts
                                   (exponential - NR3 format)

Example: OUTPUT 707;"VDELta?"
                 ENTER 707;Vdel
                 PRINT Vdel

command

The VFIFTY command instructs the oscilloscope to find the top and base values of the specified waveforms, then place the voltage markers at the 50% voltage point on the specified source(s).

If only one source has been specified with the source command, the VFIFty command sets both voltage markers (VMarker 1 and VMarker 2) to the 50% voltage level on that source.

If two sources are specified with the source command, Vmarker 1 is set to the 50% level of the first specified source and Vmarker 2 is set to the 50% level of the second specified source.

Command Syntax: VFIFty

Example: OUTPUT 707;"VFIFTY"

Measure Subsystem
8-35

VMArkerset   **command**

The VMARKERSET command corresponds to the Auto Level Set key in the Delta V menu. This command moves both of the voltage markers (VMarker 1 and 2) to the values determined by the VRELATIVE command or PTOP and PBASE commands.

If a PBASE or PTOP command was sent, then VMarker 1 is set to value determined by the PBASE command and VMarker 2 is set to the value determined by the PTOP command.

If the VRELATIVE command was sent, then VMARKER1 and VMARKER2 are set to the values determined by the VRELATIVE percent levels.

Command Syntax: VMArkerset

Example: OUTPUT 707;"SOURCE CHAN 1"
OUTPUT 707;"VMARKERSET"

VMAX?

query

The VMAX query measures and outputs the absolute maximum voltage present on the selected waveform.

Query Syntax: VMAX?

Returned Format: [VMAX]<value><CR><LF>

<value> ::= maximum voltage of selected waveform.
(exponential - NR3 format)

Example: OUTPUT 707;"VMAX?"
ENTER 707;Vmax
PRINT Vmax

VMIN? **query**

The VMIN query measures and outputs the absolute minimum voltage present on the selected waveform.

Query Syntax: VMIN?

Returned Format: [VMIN]<value><CR><LF>

<value> ::= minimum voltage value of the selected waveform. (exponential - NR3 format)

Example: OUTPUT 707;"SOURCE CHAN 1"
OUTPUT 707;"VMIN?"
ENTER 707;Vmin
PRINT Vmin

VPP?

query

The VPP query measures the maximum and minimum voltages for the selected source, then calculates the peak to peak voltage and outputs that value. The peak to peak voltage (Vpp) is calculated using the formula:

$$Vpp = Vmax - Vmin$$

where Vmax and Vmin are the maximum and minimum voltages present on the selected source.

Query Syntax: VPP?

Returned Format: [VPP]<value><CR><LF>

<value> ::= peak to peak voltage of the selected waveform. (exponential - NR3 format)

Example: OUTPUT 707;"SOURCE CHAN 2"
OUTPUT 707;"VPP?"
ENTER 707;Vpp
PRINT Vpp

command/query

The VRELATIVE command moves the voltage markers to defined
percentage points of their last established positions. For example,
after a TOPBASE operation, VMarker 1 would be located at the base
(0%) of the signal and VMarker 2 at the top (100%) of the signal. If
VRELATIVE 10 (or 90) command was executed, VMmarker 1 is
moved to the 10% level and VMarker 2 is moved to the 90% level
the signal. VREL 100 would move the markers back to their original
locations. VREL 50 would move both markers to the 50% point of
their original positions. The values that can be used with this
command are:

> 0 moves VMarker 1 to 0% and VMarker 2 to 100%
> 10 moves Vmarker 1 to 10% and VMarker 2 to 90%
> 20 moves Vmarker 1 to 20% and VMarker 2 to 80%
> 50 moves both markers to 50%
> 80 moves VMarker 1 to 20% and VMarker 2 to 80%
> 90 moves Vmarker 1 to 10% and VMarker 2 to 90%
> 100 moves VMarker 1 to 0% and VMarker 2 to 100%

The VRELATIVE command corresponds to the Preset Levels key in
the Delta V menu. The variable mode is set by executing the PTOP
or PBASE commands.

Note

*The markers can only be set to four defined positions with
this command. Any value will be accepted, however the
markers are set to the nearest defined value. As an
example, sending "VREL 15" sets VMarker 2 to 20% and
VMarker 1 to 80%.*

The VRELATIVE query returns the current relative position of the
markers.

Command Syntax: VRELative<percent>

<percent> ::= {0 | 10 | 20 | 50 | 80 | 90 | 100}

Example: OUTPUT 707;"VRELative 20"

Query Syntax: VRELative?

Returned Format: [VRELative]<value>

<value> ::= marker position in percent{0 | 10 | 20 | 50}
(integer - NR1 format)

Example: OUTPUT 707;"SOURCE CHAN 2"
OUTPUT 707;"VREL?"
ENTER 707;Vrel
PRINT Vrel

Note

*If the query return is 40 then the oscilloscope is in the
variable mode.*

VRMS? query

The VRMS query measures and outputs the RMS voltage of the selected waveform. The RMS voltage is computed over one complete period using the following formula:

$$V_{rms} = \left[\frac{1}{n} \sum_{j=1}^{j=n} V_j{}^2 \right]^{1/2}$$

Where there are n time buckets in one period and V_j is the voltage at bucket j in the time period. Since it is rare for a period to fall precisely within an integral number of time buckets, the algorithm rounds to the nearest time bucket at the beginning and end and uses these as limits.

Query Syntax: VRMS?

Returned Format: [VRMS]<value><CR><LF>

<value> ::= rms voltage of displayed waveform
(exponential - NR3 format)

Example: OUTPUT 707;"VRMS?"
ENTER 707;Vrms
PRINT Vrms

command/query

The VSTART command moves VMarker 1 to the specified voltage.
The range of values is -1000 V to +1000 V with a probe attenuation
factor of x1..

The VSTART query returns the current voltage level of voltage
marker 1.

Command Syntax: VSTArt<voltage>

<voltage> ::= between -1000 V and +1000 V

Example: OUTPUT 707;"VSTART -.01"

Query Syntax: VSTArt?

Returned Format: [VSTArt]<value><CR><LF>
<value> ::= voltage a Vmarker 1
(exponential - NR3 format)

Example: OUTPUT 707;"VSTART?"
ENTER 707;Vsta
PRINT Vsta

**Measure Subsystem
8-43**

The VSTOP command moves VMarker 2 to the specified voltage. The range values for the marker setting is -1000 V to +1000 V with a probe attenuation factor of x1..

The VSTOP query returns the current voltage level of voltage marker 2.

Command Syntax: VSTOp<voltage>

<voltage> ::= between +1000 V and -1000 V.

Example: OUTPUT 707;"VSTOP -.1"

Query Syntax: VSTOp?

Returned Format: [VSTOp]<value><CR><LF>

<value> ::= voltage at Vmarker 2
           (exponential - NR3 format)

Example: OUTPUT 707;"VSTOP?"
ENTER 707;Vstop
PRINT Vstop

VTIMe?   **query**

The VTIME query returns the voltage at a specified time. The time is referenced to the trigger event and must be on screen. The specified time may be - or + (before or after the trigger event). This command functions on single valued waveform records only. If the time with respect to the trigger event is off screen, 1.00000E38 will be returned. If the time point specified does not contain any voltage values, due to the completion criteria being less than 100%, the voltage value at that time will be interpolated using linear interpolation between the closest points before and after the specified time point.

Query Syntax: VTIMe<time>,
   <time> :: = displayed time from trigger in seconds.

Returned Format: [VTIMe]<value><CR><LF>
   <value> ::= voltage at specified time
   (exponential - NR3 format)

Example: OUTPUT 707;"VTIME -.001?"
   ENTER 707;Vtim
   PRINT Vtim

VTOP?   query

The VTOP query returns the voltage at the top of a waveform.

Query Syntax: VTOP?

Returned format: [VTOP]<value><CR><LF>

<value> ::= voltage at top of waveform
(exponential - NR3 format)

Example: OUTPUT 707;"VTOP?"
ENTER 707;Vtop
PRINT Vtop

9

Timebase Subsystem

Introduction The TIMEBASE subsystem commands control the horizontal axis, "X axis," oscilloscope functions.

Subsystem Selection The TIMEBASE subsystem must be selected prior to sending any TIMEBASE subsystem commands or queries. This can be done by sending a single line command "TIMEBASE" or by sending the subsystem selector prior to another command or query. As an example "TIMEBASE DELAY?" will select the TIMEBASE subsystem and ask for the instrument to respond with the current delay value.

System Command A system command can be sent at any time while in this subsystem. After a system command has been sent, the instrument will return to the subsystem that was selected (in this case TIMEBASE) prior to the system command.

Sending Queries In this subsystem if a query is sent followed by another query without a controller read (ENTER) between the queries, only the last response will be received. Although this operation is syntactically correct, the HP 54111D does not queue query responses in this subsystem. Therefore, the response to the first query will be overwritten in the HP-IB output buffer.

See Figure 9-1 for a syntax diagram of the TIMEBASE subsystem commands.

Figure 9-1. Timebase Subsystem Syntax Diagram.

54111/BX21
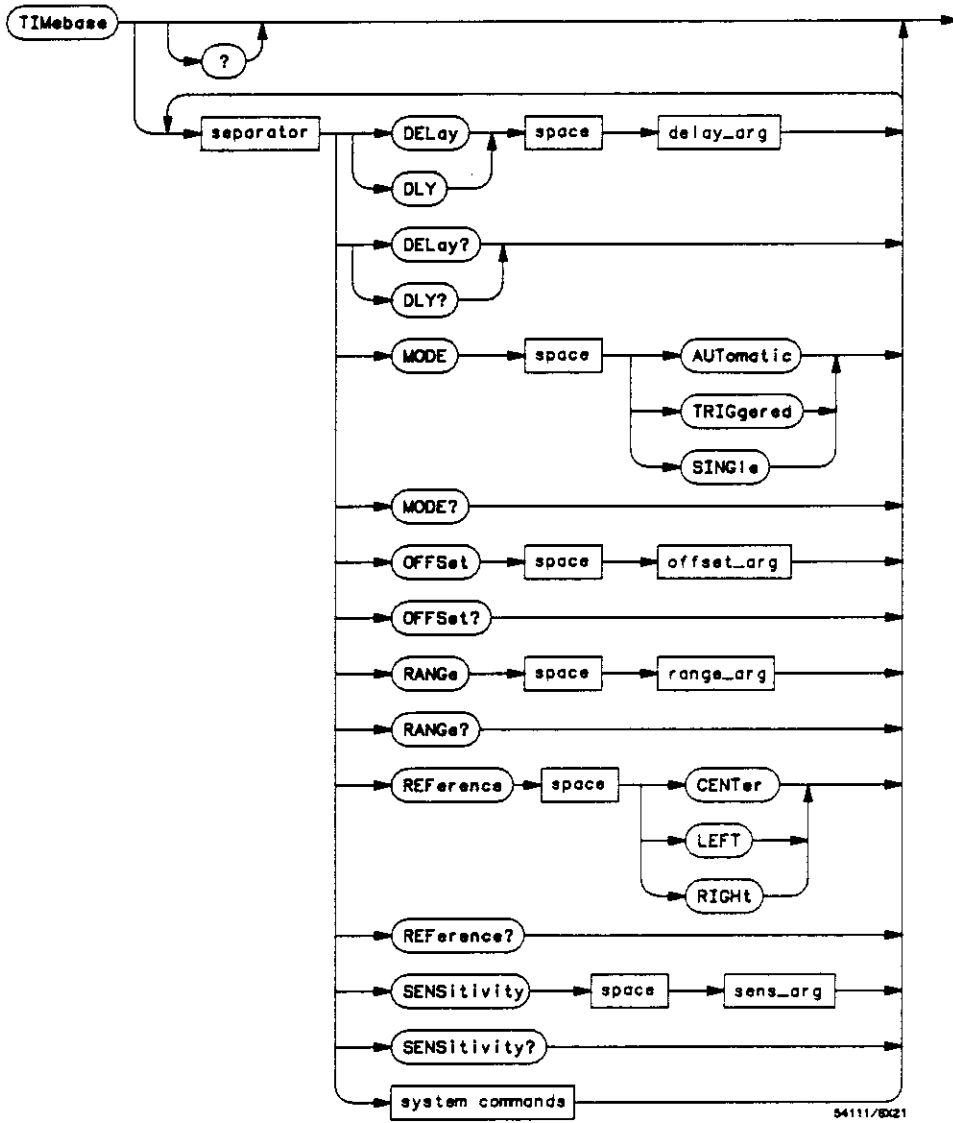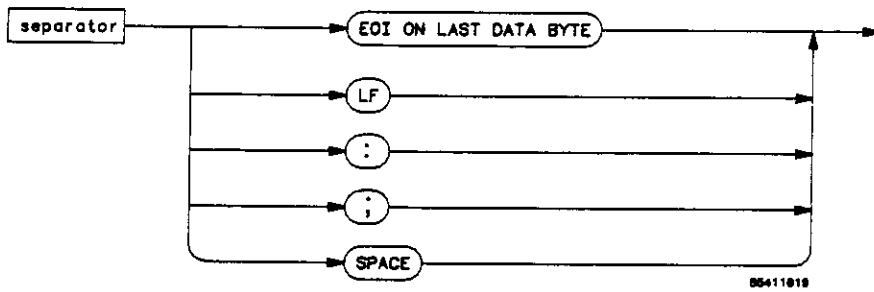
RANGE_ARG = A real number, 1 ns through 10 seconds.

DELAY_ARG = A real number, maximum depends on sweep range.

OFFSET_ARG = Same as DELAY_ARG.

SENS_ARG = A real number, 100 ps through 1 second.

Figure 9-1. Timebase Subsystem Syntax Diagram (continued).

TIMebase   **command/query**

The TIMEBASE command selects the timebase as the destination for the TIMEBASE subsystem commands.

The TIMEBASE query responds with the settings of the TIMEBASE subsystem.

Command Syntax: TIMebase

Example: OUTPUT 707;"TIMEBASE"

Query Syntax: TIMebase?

Returned Format: [TIMebase]<CR><LF>
[MODE]<argument><CR><LF>
[RANGe]<NR3><CR><LF>
[DELay]<NR3><CR><LF>
[REFerence]<NR3><CR><LF>

Example: DIM Time$[100]
OUTPUT 707;"EOI ON"
OUTPUT 707;"TIMEBASE?"
ENTER 707 USING "-K";Time$
PRINT USING "K";Time$

command/query

The DELAY command sets the timebase delay. Delay is the time interval between the trigger event and the displayed delay reference. The displayed reference can be the right edge of the display, left edge of the display, or center of the display.

The DELAY query returns the current delay value. The returned short form of this command is DLY.

Command Syntax: {DELay | DLY }<delay><delay> ::= time in seconds from trigger to display reference. Maximum value depends on TIME/DIV setting.

Example: OUTPUT 707;"DELAY 2E-3"

Query Syntax: {DELay | DLY}?

Returned Format: [DELAY | DLY]<value><CR><LF>
<value> ::= time from trigger to display reference in seconds. Display reference is Right/Left/Center (exponential - NR3 format)

Example: OUTPUT 707;"DELAY?"
ENTER 707;Del
PRINT Del

MODE   **command/query**

The MODE command selects the timebase mode. If the AUTOMATIC mode is selected, the HP 54111D will provide a baseline on the display in the absence of a signal. If a signal is present but is not triggered, the display will be unsynchronized but will not be a baseline. If the TRIGGERED mode is selected and no trigger is present, the HP 54111D will not sweep, and the data acquired on the previous trigger will remain on-screen. The SINGLE mode causes the HP 54111D to make a single acquisition when the next trigger event occurs.

The MODE query returns the current mode.

Command Syntax: MODE {AUTOmatic | TRIGgered | SINGle}

Example: OUTPUT 707;"MODE SINGLE"

Query Syntax: MODE?

Returned Format: [MODE]<argument><CR><LF>
                     <argument> ::= {AUTOmatic | TRIGgered | SINGle}

Example: DIM Mode$[30]
OUTPUT 707;"MODE?"
ENTER 707;Mode$
PRINT Mode$

OFFSet   command/query

The OFFSET command sets the timebase delay. Delay is the time interval between the trigger event and the displayed delay reference. The displayed reference can be the right edge of the display, left edge of the display, or center of the display.

This command performs exactly the same function as the DELay command.

The OFFSET query returns the current delay value with an DELAY or DLY header.

Command Syntax: OFFSet<delay>

               <delay> ::= time in seconds from trigger to display
                             reference. Maximum value depends on TIME/DIV setting.

Example: OUTPUT 707:"OFFSET 1E-4"

Query Syntax: OFFSet?

Returned Format: [DELAY | DLY]<value><CR><LF>

                 <value> ::= time from trigger to display
                         reference in seconds. Display
                         reference is Right/Left/Center
                         (exponential - NR3 format)

Example: OUTPUT 707;"OFFSET?"
            ENTER 707;Offset
            PRINT Offset

RANGe   **command/query**

The RANGE command sets the full scale horizontal time in seconds.
RANGE = 10 X SENSITIVITY. Sensitivity is the TIME/DIV value.

The RANGE query returns the current range value.

Command Syntax: RANGe\<time>

              \<time> :: = 5.0 E-9 through 10 seconds

Example: OUTPUT 707;"RANGE 1"

Query Syntax: RANGe?

Returned Format: [RANGe]\<time>\<CR>\<LF>

              \<time> :: = full scale time (10 X TIME/DIV) in seconds
                      (exponential - NR3 format)

Example: OUTPUT 707;"RANGE?"
           ENTER 707;Range
           PRINT Range

REFerence   **command/query**

The REFERENCE command sets the delay reference to the left, center, or the right side of the screen. This command affects the next acquisition and not the current display.

The query returns the current delay reference.

Command Syntax: REFerence {LEFT | CENTer | RIGHt}

Example: OUTPUT 707;"REFERENCE LEFT"

Query Syntax: REFerence?

Returned Format: [REFerence]<argument><CR><LF>

<argument> ::= {LEFT | RIGHt | CENTer}

Example: DIM Ref$[30]
OUTPUT 707;"REFERENCE?"
ENTER 707;Ref$
PRINT Ref$

Note

In the real time mode, the delay reference also controls the position of the screen display relative to the 8192-point acquisition:
- *When delay reference is left, most of the acquired waveform follows the screen display.*
- *When delay is center, the screen display is centered in the acquired waveform.*
- *When delay reference is right, most of the acquired waveform preceeds the screen display.*

SENSitivity   command/query

The SENSITIVITY command sets TIME/DIV. Range of the
SENSITIVITY command is 500 ps/division to 1 second/division.

The query returns the current time/division.

Command Syntax: SENSitivity<time/division>

<time/division> ::= 5.0E-10 through 1.0 seconds

Example: OUTPUT 707;"SENSITIVITY 1E-7"

Query Syntax: SENSitivity?

Returned Format: [SENSitivity]<value><CR><LF>

<value> ::= time/division in seconds
(exponential · NR3 format)

Example: OUTPUT 707;"SENSITIVITY?"
ENTER 707;Sens
PRINT Sens

10 Trigger Subsystem

Introduction

The commands in the TRIGGER subsystem are used to define the conditions for a trigger. This instrument provides five trigger modes: EDGE mode, PATTERN mode, EVENT DELAY mode, TIME DELAY mode, and STATE mode.

Trigger Levels

In edge mode each trigger source has an associated LEVEL, SLOPE, and PROBE attenuation factor which are used when it is selected as a trigger source. These levels and probe attenuation factors are applicable to other modes, however the slope will depend on the particular mode used.

Source Selection

The SOURCE, ENABLE, and PATH commands are related in that they select the source for commands like LOGIC or LEVEL, however each is used in a slightly different way. The SOURCE command is used to specify the trigger source for the EDGE, STATE, TDLY (time delay), and EDLY (event delay) modes. This is the source that generates the trigger. The ENABLE command is used in the STATE, PATTERN, TDLY (time delay) and EDLY (event delay) modes to specify the source that is used to qualify the trigger. The PATH command is used in the PATTERN and STATE modes to select a pattern element for setup, and in the TDLY and EDLY modes to specify the trigger enable.

Each individual trigger mode keeps track of the last referenced source and it is this source that is addressed by any SLOPE, LOGIC, etc. commands when that mode is re-entered.

Subsystem Selection

The TRIGGER subsystem must be selected prior to sending any trigger subsystem commands or queries. This can be done by sending a single line command "TRIGGER" or by sending the subsystem selector prior to the other command or query. As an example "TRIGGER SOURCE EXTERNAL 3 COUPLING?" will select the TRIGGER subsystem and request the HP 54111D to output the type of coupling for external trigger 3.

System Commands   A system command can be sent at any time while in this subsystem. After a system command has been sent, the instrument will return to the subsystem that was selected (in this case TRIGGER) prior to the system command.

Sending Queries   In this subsystem, if a query is sent followed by another query without a controller read (ENTER) between the queries, only the last response will be received. Although this operation is syntactically correct, the HP 54111D does not queue query responses in this subsystem. Therefore, the response to the first query will be overwritten in the HP-IB output buffer.

See Figure 10-1 for trigger subsystem syntax diagram.
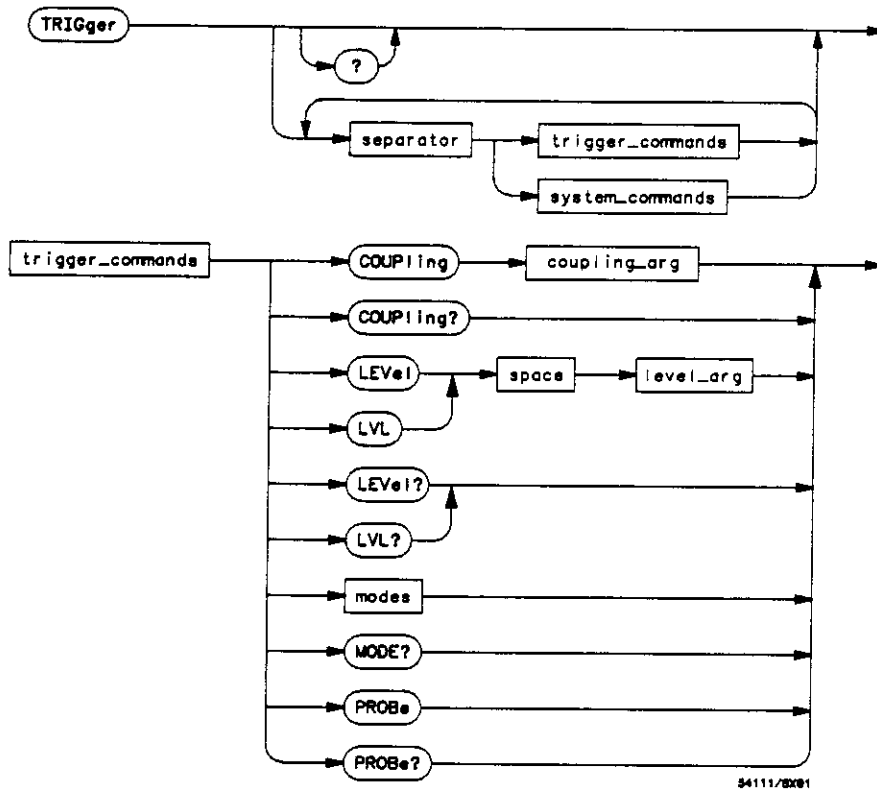
Figure 10-1. Trigger Subsystem Syntax Diagrams.
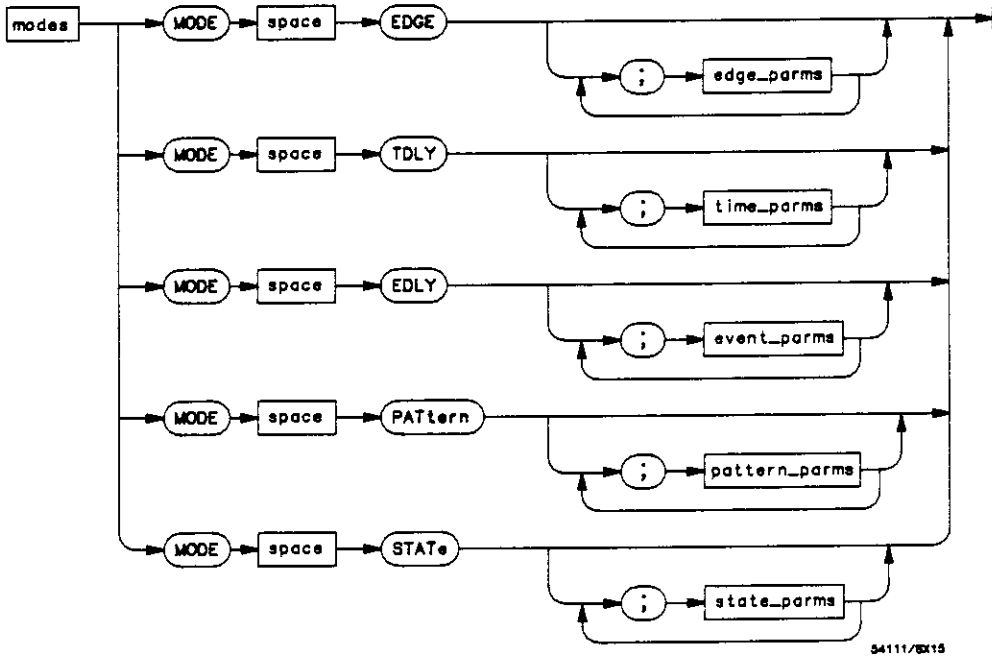
Figure 10-1. Trigger Subsystem Syntax Diagrams (continued).
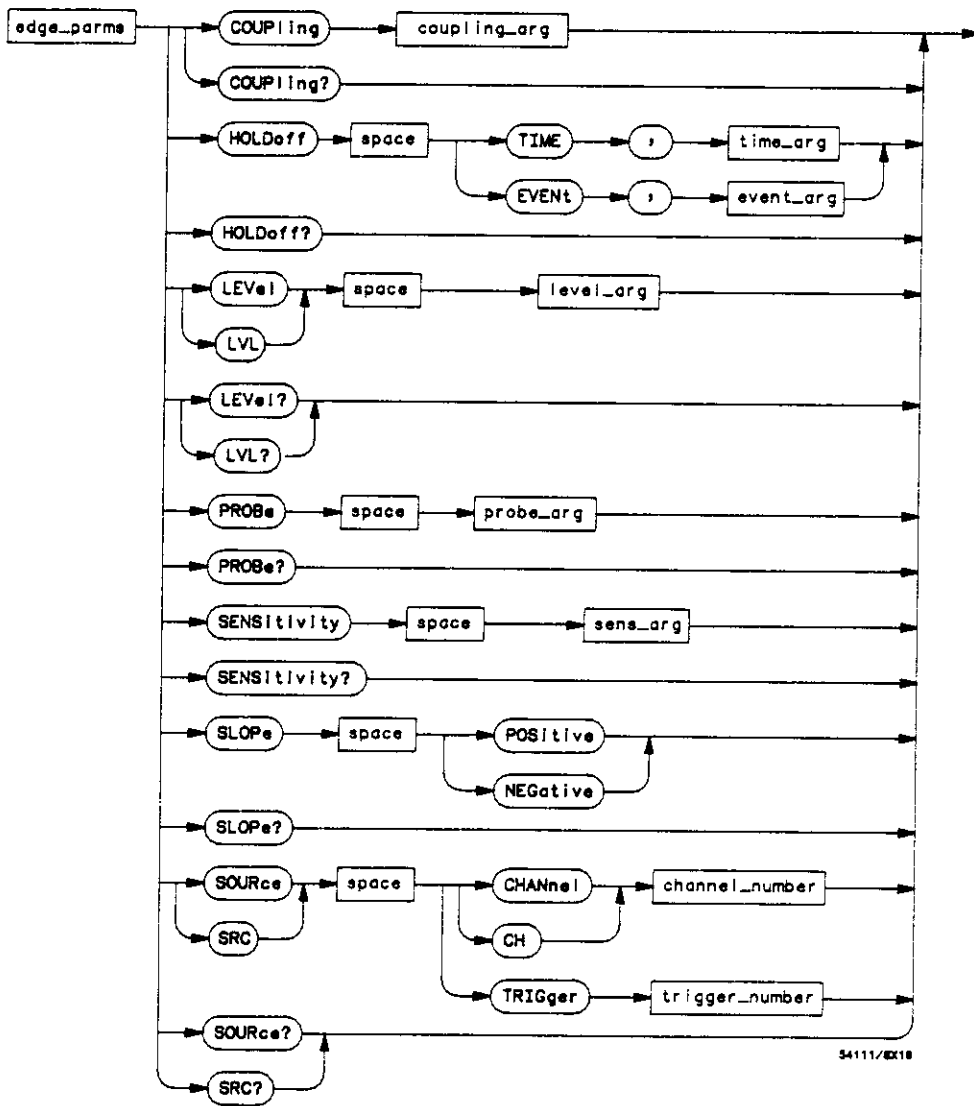
Figure 10-1. Trigger Subsystem Syntax Diagrams (continued).
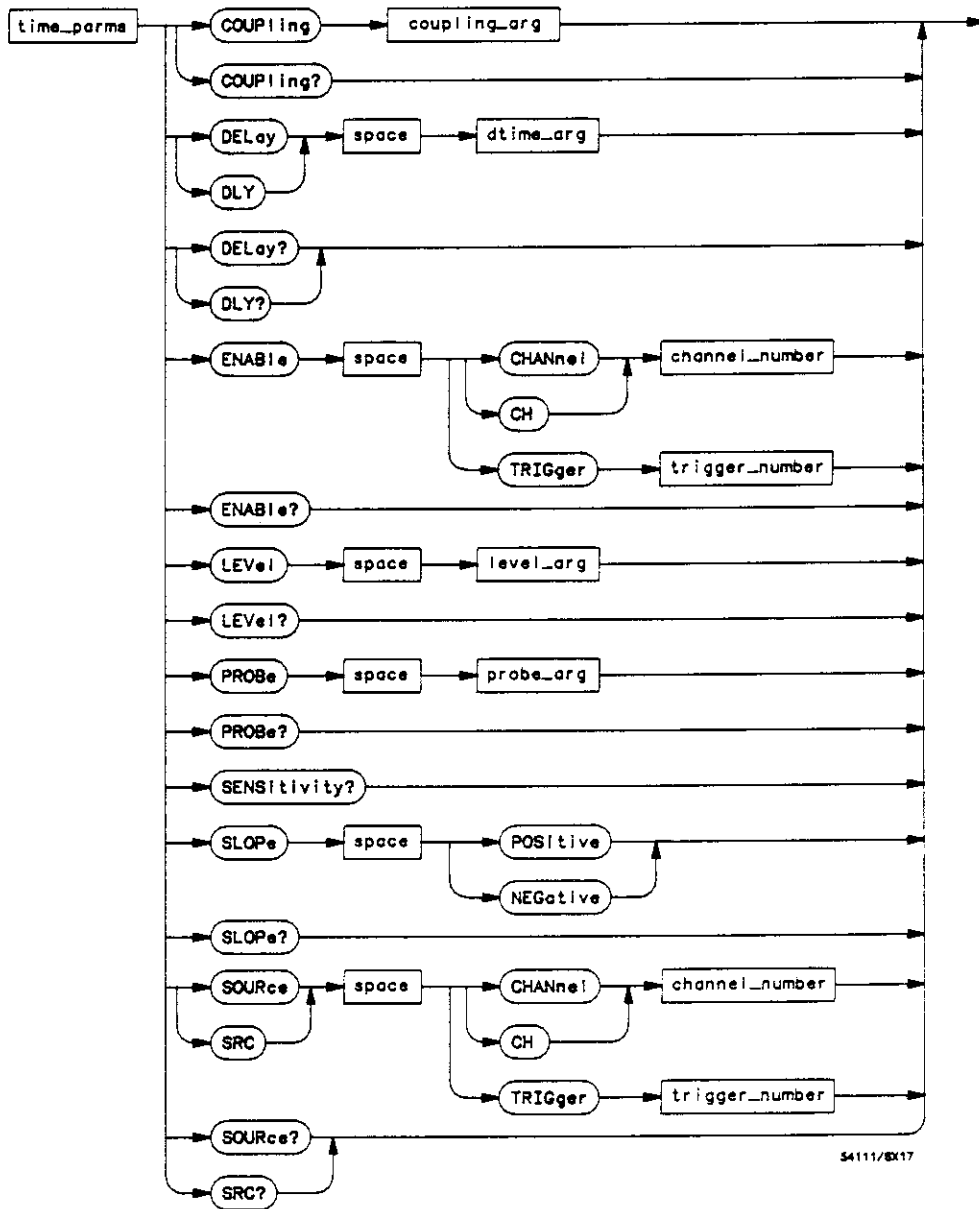
Figure 10-1. Trigger Subsystem Syntax Diagrams (continued).
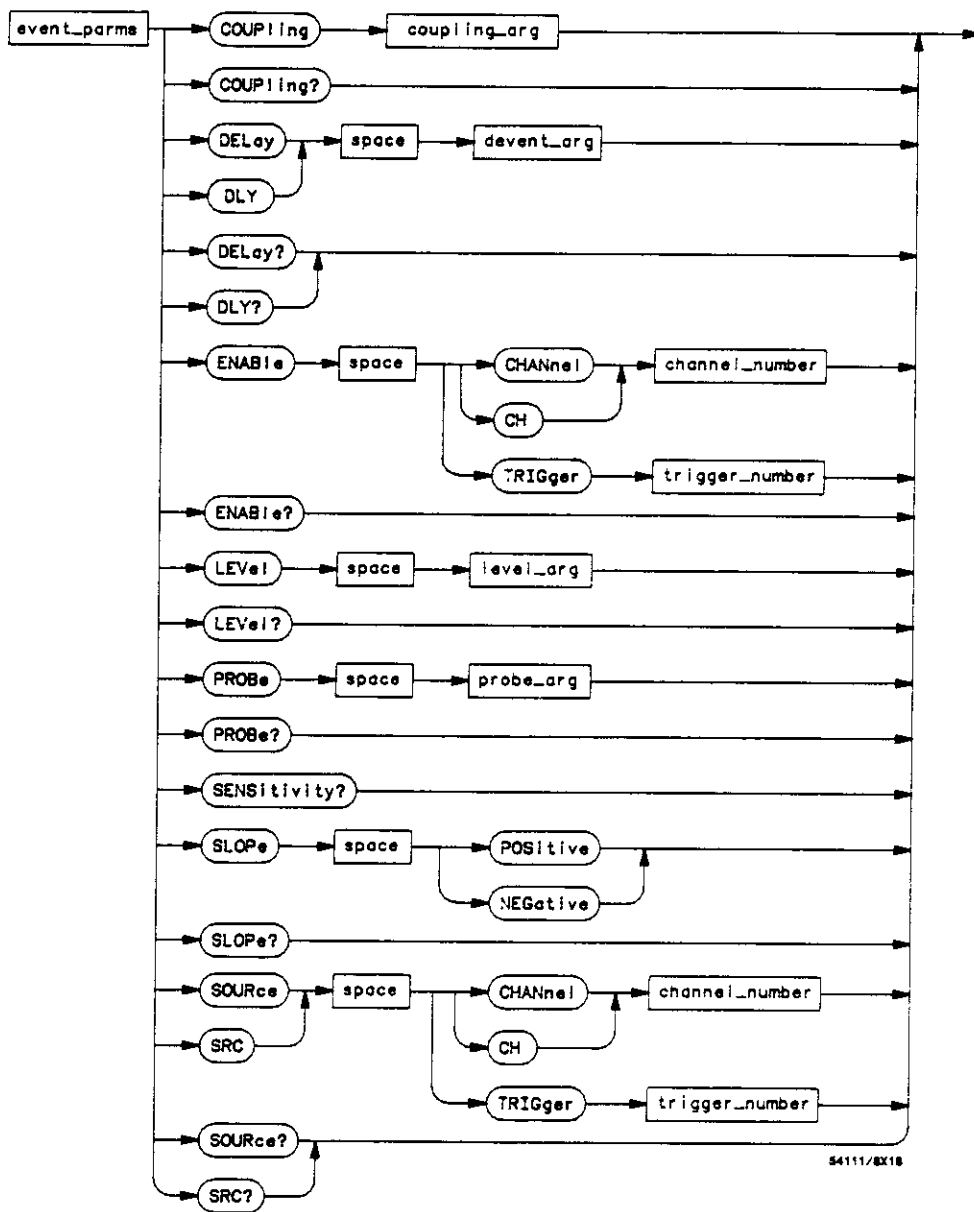
Figure 10-1. Trigger Subsystem Syntax Diagrams (continued).
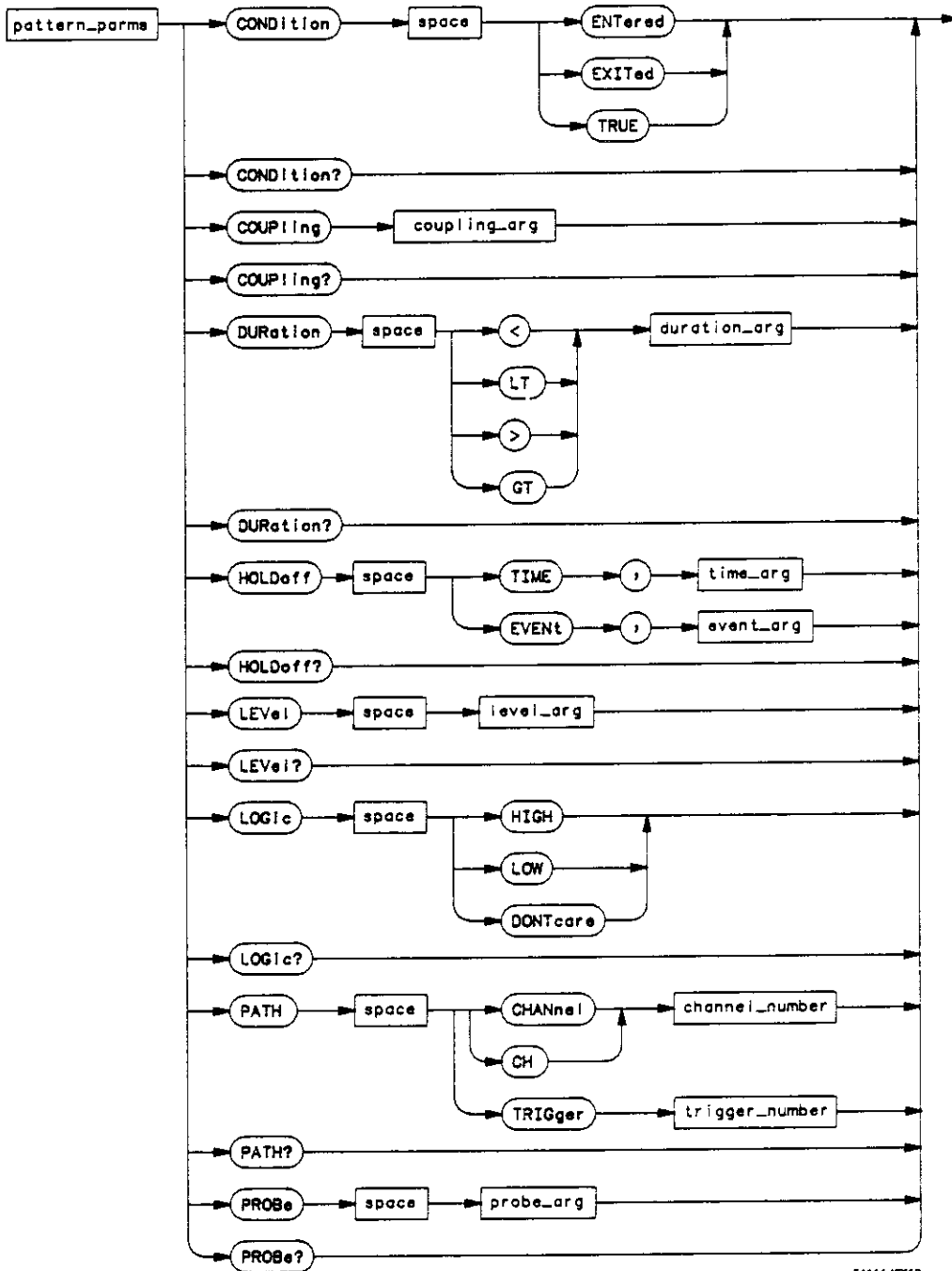
Figure 10-1. Trigger Subsystem Syntax Diagrams (continued).
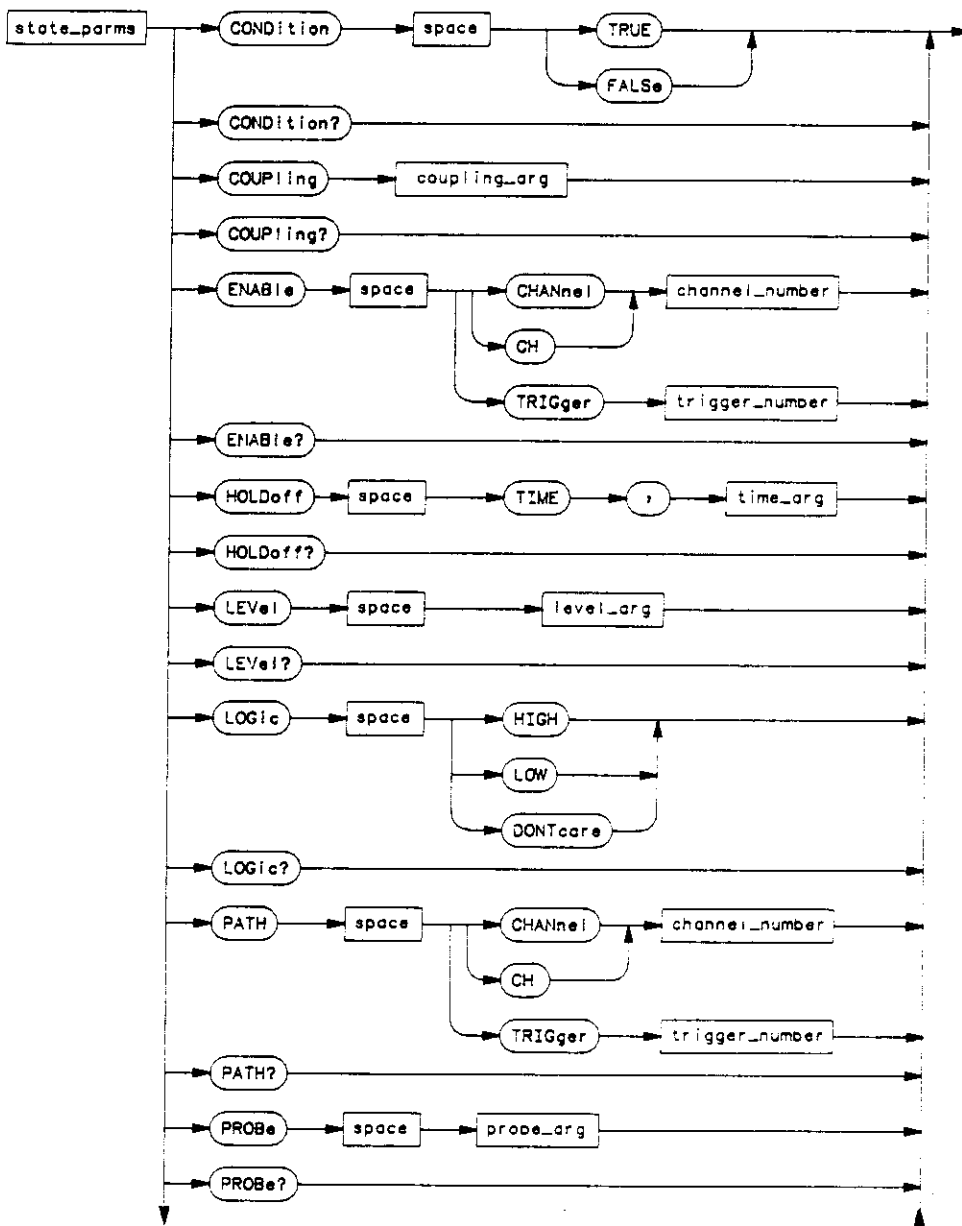
Figure 10-1. Trigger Subsystem Syntax Diagrams (continued).
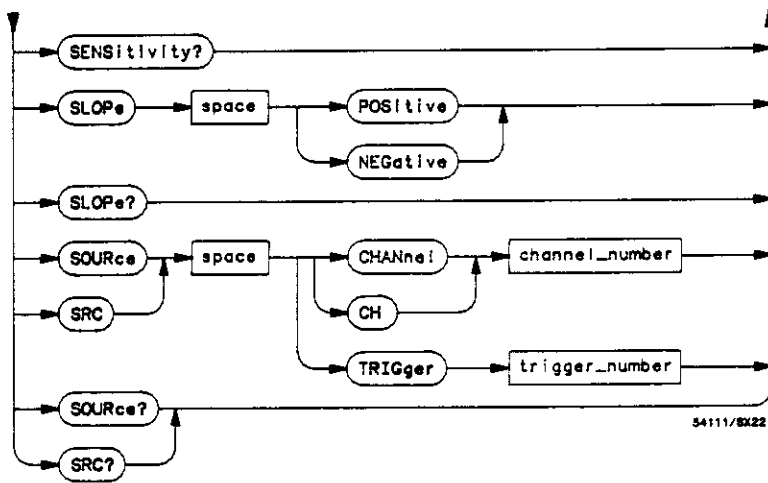
CHANNEL_NUMBER = A real number, 1 or 2.

COUPLING_ARG = AC, DC, or DCFifity.

DEVENT_ARG = A real number, specifying the number of trigger events to delay.

DTIME_ARG = A real number, specifying the delay time in seconds.

DURATION_ARG = A real number, specifing the time a pattern must be present to generate a trigger.

TRIGGER_NUMBER = A real number, 3 or 4 specifying an external trigger source.

EVENT_ARG = A real number, specifing the number of events for the holdoff.

LEVEL_ARG = A real number, specifying the trigger level setting in volts. The range is determined by the sensitivity value of that source.

PROBE_ARG = A real number, specifing the probe attenuation for the current trigger source.

SENS_ARG = An integer (1 or 10) −1 is high, and 10 is low.

TIME_ARG = A real number, specifying the time in seconds for the holdoff.

Figure 10-1. Trigger Subsystem Syntax Diagrams (continued).

The TRIGGER command selects the trigger subsystem as the
destination for the trigger subsystem commands that follow.

The TRIGGER query responds with the subsystem parameters for
the currently selected trigger mode.

Command Syntax: TRIGger

Example: OUTPUT 707;"TRIGGER"

Query Syntax: TRIGger?

Returned Format: TRIGger
MODE EDGE<CR><LF>
SOURce<path name><CR><LF>
PROBe<NR1><CR><LF>
LEVel<NR3><CR><LF>
SLOPe<argument><CR><LF>
HOLDoff TIME<NR3><CR><LF>
COUPling<argument><CR><LF>

TRIGger
MODE PATtern<CR><LF>
CONDition<argument><CR><LF>
DURation<argument><CR><LF>
PATH<path name 1><CR><LF>
PROBe<NR3><CR><LF>
LEVel<NR3><CR><LF>
LOGic<argument><CR><LF>
COUPling<argument><CR><LF>
PATH<path name 2><CR><LF>
PROBe<NR3><CR><LF>
LEVel<NR3><CR><LF>
LOGic<argument><CR><LF>
COUPling<argument><CR><LF>
PATH<path name 3><CR><LF>
PROBe<NR3><CR><LF>
LEVel<NR3><CR><LF>
LOGic<argument><CR><LF>
COUPling<argument><CR><LF>

Returned Format: SENSitivity<NR1><CR><LF>
(continued)   PATH<path name 4><CR><LF>
   PROBe<NR3><CR><LF>
   LEVel<NR3><CR><LF>
   LOGic<argument><CR><LF>
   COUPling<argument><CR><LF>
   SENSitivity<NR1><CR><LF>

   TRIGger
   MODE STATe<CR><LF>
   CONDition<argument><CR><LF>
   PATH<path name 1><CR><LF>
   PROBe<NR3><CR><LF>
   LEVel<NR3><CR><LF>
   LOGic<argument><CR><LF>
   COUPling<argument><CR><LF>
   SENSitivity<NR1><CR><LF>
   PATH<path name 2><CR><LF>
   PROBe<NR3><CR><LF>
   LEVel<NR3><CR><LF>
   LOGic<argument><CR><LF>
   COUPling<argument><CR><LF>
   SENSitivity<NR1><CR><LF>
   PATH<path name 3><CR><LF>
   PROBe<NR3><CR><LF>
   LEVel<NR3><CR><LF>
   LOGic<argument><CR><LF>
   COUPling<argument><CR><LF>
   SENSitivity<NR1><CR><LF>
   SOURce<path name 4><CR><LF>
   PROBe<NR3><CR><LF>
   LEVel<NR3><CR><LF>
   SLOPe<argument><CR><LF>
   HOLDoff TIME<NR3><CR><LF>

   TRIGger
   MODE TDLY<CR><LF>
   ENABle<path name 1><CR><LF>
   PROBe<NR3><CR><LF>
   LEVel<NR3><CR><LF>
   SLOPe<argument><CR><LF>
   DELay<NR3><CR><LF>
   SOURce<path name 2><CR><LF>

Returned Format: PROBe<NR3><CR><LF>
(continued)   LEVel<NR3><CR><LF>
              SLOPe<argument><CR><LF>

              TRIGger
              MODE EDLY<CR><LF>
              ENABle<path name 1><CR><LF>
              PROBe<NR3><CR><LF>
              LEVel<NR3><CR><LF>
              SLOPe<argument><CR><LF>
              DELay<NR1><CR><LF>
              SOURce<path name 2><CR><LF>
              PROBe<NR3><CR><LF>
              LEVel<NR3><CR><LF>
              SLOPe<argument><CR><LF>

Example: DIM Trig$[2000]
              OUTPUT 707;"EOI ON"
              OUTPUT 707;"TRIGGER?"
              ENTER 707 USING "-K";Trig$
              PRINT USING "K";TRIG$ CONDition

CONDition   command/query

The CONDITION command/query is valid only when the trigger mode is PATTERN or STATE. In the PATTERN mode, it specifies whether the trigger is generated on entry to the specified logic pattern or when exiting the specified logic pattern. If TRUE is selected, duration is used as the qualifier for the PATTERN.

In the STATE mode, it specifies whether the trigger is generated when the pattern is present (true) or not present (false).

The CONDITION query returns the currently selected condition.

Command Syntax: CONDition<argument>

                    <argument> ::= {ENTER | EXIT | TRUE}
                              (PATTERN mode)
                    <argument> ::= {TRUE | FALSE}
                              (STATE mode)

Example: OUTPUT 707;"CONDITION EXIT"

Query Syntax: CONDition?

Returned Format: [CONDition]{ENTER | EXIT | TRUE} (PATTERN mode)
                    [CONDition]{FALSE | TRUE} (STATE mode)

Example: DIM Cond$[30]
                OUTPUT 707;"CONDITION?"
                ENTER 707;Cond$
                PRINT Cond$

COUPling   **command/query**

The COUPLING command sets the input coupling of the previously selected trigger SOURCE, ENABLE, or PATH. If you select the AC or DC arguments, 1 MΩ input impedance is provided. If you select the DCFIFTY argument, the selected input will be dc-coupled and the input impedance will be set to 50 Ω.

The COUPLING query returns the current input coupling of the designated trigger source.

Command Syntax: COUPling {AC | DC | DCFifty}

Example: OUTPUT 707;"COUPLING AC"

Query Syntax: COUPling?

Returned Format: [COUPling] {AC | DC | DCFifty}

Example: DIM Coup$[30]
OUTPUT 707;"COUPLING?"
ENTER 707;Coup$
Print Coup$

DELay | DLY   **command/query**

The DELAY command/query is valid only in the events delay (EDLY) or time delay (TDLY) modes. In the time delay mode, this command specifies the delay in seconds. In the events delay mode, this command specifies the number of trigger events.

The DELAY query returns the delay for the current mode.

Command Syntax: {DELay | DLY}{<events> | <time>}

                                        <events> ::= number of trigger events to delay

                                        <time> ::= time of delay in seconds

Example: OUTPUT 707;"DELAY 10"

Query Syntax: {DELay | DLY}?

Returned Format: [DELay]{<events> | <time>}<CR><LF>

                                        <events> ::= number of events the trigger is delayed
                                               (integer - NR1 format)

                                        <time> ::= amount of time the trigger is delayed in seconds
                                             (exponential - NR3 format)

Example: DIM Delay$[30]
OUTPUT 707;"DELAY?"
ENTER 707;Delay$
PRINT Delay$

DURation

<div align="right">

command/query

</div>

The DURATION command/query is valid only in the pattern mode. It specifies the time limit (minimum time for ">" (greater than), or maximum time for "<" (less than)) a pattern must be present to generate a trigger. Pattern duration trigger is implicitly an "EXITED" condition, that is, the trigger coincides with the first event that makes the pattern false.

The DURATION query returns the current selections for duration type and time.

Command Syntax: DURation { LT | < | GT | > }

Example: OUTPUT 707;"DURATION LT 1E-3"

Query Syntax: DURation?

Returned Format: [DURation]{LT | GT}<CR><LF>

Example:
DIM Dur$[30]
OUTPUT 707;"DURATION?"
ENTER 707;Dur$
PRINT Dur$

ENABle   **command/query**

The ENABLE command/query is valid in the PATTERN, STATE, TDLY, or EDLY modes. It is used to specify the source that is to be used as the trigger enable. Once this is specified it is also the source for subsequent SLOPE and PROBE commands.

The ENABLE query returns the current trigger enable source of the present mode.

Command Syntax: ENABle{CHANnel1 | CHANnel2 | TRIGger 3 | TRIGger 4}

Example: OUTPUT 707;"TRIG MODE STATE ENABLE CHANNEL 2"

Query Syntax: ENABle?

Returned Format: [ENABle]<source><CR><LF>

<source> ::= {CHANnel1 | CHANnel2 | TRIGger3 | TRIGger4}

Example: DIM Enable$[30]
OUTPUT 707;"ENABLE?"
ENTER 707;Enable$
PRINT Enable$

HOLDoff   **command/query**

The HOLDOFF command specifies the holdoff time in the EDGE,
PATTERN, or STATE modes, or the holdoff number of events in the
EDGE or PATTERN modes. Each mode has its own holdoff
parameters and "remembers" whether it was using holdoff by time
or holdoff by events.

The HOLDOFF query is valid in the EDGE, PATTERN, or STATE
modes and returns the current holdoff setting for the selected mode.
Time holdoff ranges from 70 ns to 670 ms. Events holdoff ranges
from 2 to 67,000,000 events.

Command Syntax: HOLDoff {[EVENT<events>] | [TIME<time>]}

                                           <events> ::= number of events of holdoff

                                           <time> ::= time in seconds of holdoff

Example: OUTPUT 707;"MODE EDGE HOLDOFF EVENT 100"

Query Syntax: HOLDoff?

Returned Format: [HOLDoff]<mode>, <value><CR><LF>

                                           <mode> ::= current holdoff mode {EVENt | TIME}

                                           <value> ::= holdoff value in this mode

                                                              (mode EVENT - integer - NR1 format;

                                                              mode TIME - exponential - NR3 format)

Example: DIM Hold$[40]
OUTPUT 707;"HOLDOFF?"
ENTER 707;Hold$
PRINT Hold$

LEVel | LVL   **command/query**

The LEVEL command sets the trigger level voltage of the selected SOURCE or PATH. Levels may be set only in edge mode.

The LEVEL query returns the trigger level of the selected SOURCE or PATH.

Command Syntax: {LEVel | LVL} <value>

                                    <value> ::= trigger level setting in volts

Example: OUTPUT 707;"MODE EDGE SOUR CH 2 LEVEL .1"

Query Syntax: {LEVel | LVL}?

Returned Format: [LEVel]<trig lvl><CR><LF>

                                  <trig lvl> ::= trigger level in volts
                                              (exponential - NR3 format)

Example: DIM Level$[30]
OUTPUT 707;"LEVEL?"
ENTER 707;Level$
PRINT Level$

LOGic

command/query

The LOGIC command/query is valid in the STATE and PATTERN modes. The LOGIC command is used to specify the relation between the signal and the defined voltage level that must exist before that part of the pattern is considered valid. If the signal on a selected source or path is greater than the trigger level, that signal is considered HIGH. If it is less than the trigger level, it is considered LOW.

The LOGIC query returns the last specified logic level of the currently enabled source.

Command Syntax: LOGic{LOW | HIGH | DONTcare}

Example: OUTPUT 707;"MODE STATE ENAB CH 2 LOGIC DONTCARE"

Query Syntax: LOGic?

Returned Format: [LOGic]<state><CR><LF>
        <state> ::= {HIGH | LOW | DONTCARE}

Example: DIM Log$[40]
OUTPUT 707;"LOGIC?"
ENTER 707;Log$
PRINT Log$

MODE   **command/query**

The MODE command selects the trigger mode.

The MODE query returns the current trigger mode.

Command Syntax: MODE {EDGE | PATtern | STATe | TDLY | EDLY}

Example: OUTPUT 707;"MODE EDGE"

Query Syntax: MODE?

Returned Format: [MODE]<mode><CR><LF>

<mode>::= {EDGE | PATtern | STATe | TDLY | EDLY}

Example: DIM Mode$[30]
OUTPUT 707;"MODE?"
ENTER YOU;Mode$
PRINT Mode$

PATH

The PATH command/query is valid in the PATTERN, STATE, TDLY, and EDLY modes. This command selects a pattern bit as the source for future probe and logic commands.

The PATH query returns the current trigger source of the present mode.

Command Syntax: PATH {CHANnel1 | CHANnel2 | TRIGger3 | TRIGger4}

Example: OUTPUT 707;"MODE STATE PATH TRIGGER3"

Query Syntax: PATH?

Returned Format: [PATH]<source><CR><LF>

<source>::= {CHANnel 1 | CHANnel 2 | TRIGger 3 | TRIGger 4}

Example:
DIM Path$[30]
OUTPUT 707;"TRIGGER MODE PATTERN PATH?"
ENTER 707;Path$
PRINT Path$

The PROBE command specifies the attenuation factor for the last specified SOURCE or PATH of the current trigger mode. See the PROBE command in chapter 4 of this manual.

The PROBE query returns the current source's probe attenuation factor.

Command Syntax: PROBe<probe atten>

                       <probe atten>::= 1 to 1000

           Example: OUTPUT 707;"PROBE 10"

     Query Syntax: PROBe?

Returned Format: [PROBe]<atten><CR><LF>

                       <atten>::= 1 to 1000

                                (exponential - NR3 format)

          Example: OUTPUT 707;"PROBE?"

                        ENTER 707;Probe$

                        PRINT Probe$

SENSitivity   command/query

The SENSITIVITY command sets the sensitivity for trigger 3 and 4. The argument "1" sets it high (x1), and "10" sets it low (x10). This command can be set only in the edge mode.

The SENSITIVITY query returns the current sensitivity for the selected source (trigger 3 or 4).

Command Syntax: SENSitivity (1 | 10)

Example: OUTPUT 707;"SENSITIVITY 1"

Query Syntax: SENSITIVITY?

Returned Format: [SENSitivity]<sens><CR><LF>

  <sens>:: = 1 or 10

Example: OUTPUT 707;"SENSITIVITY"
ENTER 707;Sens$
PRINT Sens$

SLOPe   command/query

The SLOPE command specifies the trigger slope for the previously specified source.

The SLOPE query returns the current slope for the last selected source of the current mode.

Command Syntax: SLOPe {NEGative | POSitive}

Example: OUTPUT 707;"SLOPE POSITIVE"

Query Syntax: OUTPUT 707;"SLOPE?"

Returned Format: [SLOPe]<slope><CR><LF>
              <slope>::= {POSitive | NEGative}

Example: DIM Slope$[30]
OUTPUT 707;"SLOPE?"
ENTER 707;Slope$
PRINT Slope$

SOURce | SRC   command/query

The SOURCE command/query is valid in the EDGE, STATE, TDLY or EDLY modes. It is used to specify the trigger source. This command also identifies the source for any subsequent SLOPE and PROBE commands.

The SOURCE query returns the current trigger source of the present mode.

Command Syntax: {[SOURce | SRC]}{CHANnel1 | CHANnel2 | TRIGger3 | TRIGger4}

Example: OUTPUT 707;"SOURCE CHANNEL1"

Query Syntax: {SOURce | SRC}?

Returned Format: [SOURce]<source><CR><LF>

<source>::= {CHANnel1 | CHANnel2 | TRIGger3 | TRIGger4}

Example: DIM Src$[30]
OUTPUT 707;"SOURCE?"
ENTER 707;Src$
PRINT Src$

11 Waveform Subsystem

Introduction

The WAVEFORM subsystem is used to transfer waveform data between a controller and the HP 54111D's waveform memories. The waveform record is actually contained in two elements, the waveform data and the preamble. The waveform data is the actual data acquired using the system command DIGITIZE. The preamble contains the information required to properly interpret the waveform data. This includes the number of points acquired, completion, count, format of acquired data, type of acquired data, and input coupling. The preamble also contains the X and Y increments, origins, and references for the acquired data.

The values set in the preamble are determined when the DIGITIZE command is executed and are based on the settings of variables in the ACQUIRE subsystem. Although the preamble values can be changed using a controller, they will not change the way the data was acquired nor will they change the type of data that was actually acquired, the number of points actually acquired, etc. Therefore, extreme caution must be used when changing any waveform preamble values to ensure the data will still be useful. For example, setting POINTS in the preamble to a value different from the actual number of points in the waveform could result in inaccurate data.

The waveform data and preamble must be read (by the controller) or sent (to the HP 54111D) using two separate commands, DATA? and PREAMBLE?.

Refer to figure 11-1, page 11-8 for the syntax diagrams of the WAVEFORM subsystem.

Subsystem Selection

The WAVEFORM subsystem must be selected prior to sending any WAVEFORM subsystem commands or queries. This can be done by sending a single command "WAVEFORM" or by sending the subsystem selector prior to another command or query. As an example "WAVEFORM POINTS?" will return the POINTS value in the currently selected waveform preamble.

System Command   A system command can be sent at any time while in this subsystem. After a system command has been sent, the instrument will return to the subsystem that was selected (in this case WAVEFORM) prior to the system command.

Sending Queries   In this subsystem if a query is sent followed by another query without a controller read (ENTER) between the queries, only the last response will be received. Although this operation is syntactically correct, the HP 54111D does not queue query responses in this subsystem. Therefore, the response to the first query will be overwritten in the HP-IB output buffer.

Data Acquisition Types   There are five types of waveform acquisition types that can be selected using the ACQUIRE subsystem TYPE command. The five types are: NORMAL, FAST, FILTERED, AVERAGE, and VPERSISTENCE. The type of data acquisition selected, and the source(s) specified for the DIGITIZE command will determine where the digitized data is placed.

For example, if ACQUIRE TYPE NORMAL and DIGITIZE 1 are specified, a data record of 8192 points is processed into Memory 1. If DIGITIZE 2 is specified while TYPE NORMAL is selected, a data record of 8192 points is processed into Memory 2.

If ACQuire TYPE FAST and DIGITIZE 1 or DIGITIZE 2 are specified, a data record of 8192 points is formatted and processed in FAST 1 or FAST 2 buffers.

The ACQUIRE TYPES (VPERSISTENCE, AVERAGE, FILTERED) as well as TYPE NORMAL with either DIGITIZE 5 or 6 selected, will digitize a waveform record that is 501 points long and they will be placed in Memories 5 through 8, depending on the digitize source specified.

Refer to table 11-1, page 11-4 for a condensed version of the relationships between the ACQUIRE TYPE, the DIGITIZE command, and where the acquired waveform data is placed.

Normal Data Type - When the NORMAL TYPE is selected using the ACQUIRE subsystem, the two input channels' data and the two functions' (FUNCTION 1 and 2) data can be acquired using the system command DIGITIZE. With the NORMAL type selected and DIGITIZE 1 command sent, the Channel 1 waveform is acquired and placed into Memory 1. Sending a DIGITIZE 2 command acquires the Channel 2 waveform and places the data in Memory 2. If a DIGITIZE 5 is sent, the HP 54111D will acquire the Function 1 data into Memory 5. If a DIGITIZE 6 is sent, the Function 2 data is acquired into Memory 6.

When a DIGITIZE 1 or 2 is used with ACQUIRE TYPE NORMAL, the acquired waveform record is 8192 data points. In this mode an ACQUIRE RESOLUTION value of OFF, 6, 7, or 8 can be selected to change the vertical resolution of the acquired data. When a DIGITIZE 5 or 6 is used with ACQUIRE TYPE NORMAL, the acquired waveform record is 501 data points.

When a DIGITIZE 1 or 2 is used with ACQUIRE TYPE FAST, the acquired waveform record is 8192 data points. The data is formatted and written into the fast buffers. DIGITIZE 5 and 6 are not allowed in ACQUIRE TYPE NORMAL.

Filtered Data Type - When the FILTERED TYPE is selected in the ACQUIRE subsystem, the data from the two input channels and the data from both functions can be acquired using the system command DIGITIZE. With the FILTERED type selected and a DIGITIZE 1 command sent, the Channel 1 waveform is acquired and placed into Memory 5. Sending a DIGITIZE 2 command acquires the Channel 2 waveform and places the data in Memory 6. If a DIGITIZE 5 is sent, the instrument acquires the Function 1 data and places it in Memory 7. Sending a DIGITIZE 6 acquires the Function 2 data and places it in Memory 8.

When the ACQUIRE TYPE FILTERED is selected, all data acquired with the system command DIGITIZE is 501 data points.

Average Data Type - When the AVERAGE TYPE is selected in the ACQUIRE subsystem, the data from the two input channels and the data from both functions can be acquired using the system command DIGITIZE. With the AVERAGE TYPE selected and a DIGITIZE 1 command sent, the Channel 1 waveform is acquired and placed into Memory 5. Sending a DIGITIZE 2 command acquires the Channel 2 waveform and places the data in Memory 6. If a DIGITIZE 5 is sent, the instrument acquires the Function 1 data and places it in Memory 7. Sending a DIGITIZE 6 acquires the Function 2 data and places it in Memory 8.

When the ACQUIRE TYPE AVERAGE is selected, all data acquired with the system command DIGITIZE is 501 data points. In this mode the number of averages can be selected.

Vpersistence Data Type - When the VPERSISTENCE TYPE is selected in the ACQUIRE subsystem, the data from the two input channels and the data from both functions can be acquired using the system command DIGITIZE. With the VPERSISTENCE TYPE

selected and a DIGITIZE 1 command sent, the Channel 1 waveform is acquired and placed into Memory 5. Sending a DIGITIZE 2 command acquires the Channel 2 waveform and places the data in Memory 6. If a DIGITIZE 5 is sent, the instrument acquires the Function 1 data and places it in Memory 7. Sending a DIGITIZE 6 acquires the Function 2 data and places it in Memory 8.

When the ACQUIRE TYPE VPERSISTENCE is selected, all data acquired with the system command DIGITIZE is 501 data points.

Table 11-1 shows the number of points acquired and where the digitized data is placed for each ACQUIRE TYPE and DIGITIZE command combination. This table also indicates where the data is acquired from for each of the DIGITIZE commands.

Refer to the ACQUIRE subsystem (Chapter 3) for more information on waveform data types.

Table 11-4. Digitized Waveform Locations and Points.

DIGITIZE command used	REAL TIME MODE			REPETITIVE MODE
	FAST 8K record	NORMAL 8K record	FILTERED display data only	AVERAGE VPERSISTENCE
DIGITIZE 1 acquires Channel 1	8192 points acquired stored in internal MEMORY FAST 1	8192 points acquired single shot, stored in MEMORY 1	501 points acquired single shot, stored in MEMORY 2	501 points acquired until complete, stored in MEMORY 5
DIGITIZE 2 acquires Channel 2 internal	8192 points acquired stored in MEMORY FAST 2	8192 points acquired single shot, stored in MEMORY 2	501 points acquired single shot, MEMORY 6	501 points acquired until complete, stored in MEMORY 6
DIGITIZE 5 acquires FUNCTION 1	illegal mode	501 points acquired until complete, stored in MEMORY 7		
DIGITIZE 6 acquires FUNCTION 2	illegal mode	501 points acquired until complete, stored in MEMORY 8		

Data Format within the HP 54111D MEMORIES 1-8

The HP 54111D is an 8-bit maximum resolution digitizing oscilloscope. Internally data is kept in 16-bit word format. Valid data, holes, and clipped data are represented internally as shown in table 11-2. When a DIGITIZE command is excuted to capture data to memories 1-8, the 16 bit word format is always used.

DATA FORMAT within the HP 54111D FAST buffer

Data is captured into the Fast buffers according to the format specified in the FAST 1 or FAST 2 Waveform subsystem. If BYTE format is specified, then the data is packed into an eight bit format. If word is specified, the data is in the 16 bit format with the lower eight bits representing the data and the upper eight bits set to zero. The eight bits of data are transferred directly from the A/D converters. The Fast mode is an 8K mode only. There are no holes in the data. Waveform Source Fast 1 and Waveform Source FAST 2 Format should always be set before excuting the DIGITIZE commands.

Table 11-2. Data Values in the HP 54111D.

	UPPER BYTE	LOWER BYTE
Valid data — not clipped	00 hex	0 through 255 decimal (see Note)
Clipped data representation	00 hex	0..3 or 252..255 decimal (see Note)
A hole is represented by −1	FF hex	255 decimal

Note

It should be noted from table 11-2 that the end point data values, such 0..3 and 252..254, are not distinguished from clipped data. It is recommended that all data values within these ranges be treated as clipped data. In BYTE format, the data value 255 is used as the single data value to indicate a hole. In WORD format, the data value FFFFH is used to indicate a hole. A hole is a time point in which no data has been acquired due to a completion criteria of less than 100%.

Waveform Subsystem
11-5

Data Format for HP-IB Transfer There are three formats that can be used to transfer waveform data over the HP-IB. These formats are: ASCII format, WORD format, and BYTE format. Refer to table 11-3 for an overview of how data is transferred over HP-IB in each of these formats.

Table 11-3. Data Format for HP-IB Transfer

Internal data			Output over HP-IB			
	Upper Byte	Lower Byte	ASCII format	WORD Format		BYTE format
				Upper Byte	Lower Byte	
data	[00 hex]	[0...254]	000...254	[00 hex]	[0...254]	0...254
data	[00 hex]	[255]	255	[00 hex}	[FF hex]	254*
hole	[FF hex]	[FF hex]	−001	[FF hex] [FF hex] (−1)		255**
* A data value of 255 is mapped into 254 ** 255 is reserved to indicate holes						

As can be seen from table 11-3 when the WORD format is used two bytes are sent, while the BYTE format uses only one byte of data per point.

When data is transmitted back into the memories in BYTE mode, a value of 255 will be converted back into the internal representation of [FF hex] [FF hex], the internal representation of −1.

WORD Format WORD formatted waveform records are transmitted using the binary block format (#A format) as defined in IEEE Standard 728-1982. When using this format the ASCII character string "#A" is sent first, followed by a two-byte length value (16-bit binary) specifying

the number of bytes to follow. The length value is sent high byte first. The number of bytes is twice the number of words (data points). The number of words is also the value returned by the WAVEFORM POINTS? query. This is followed by a sequence of bytes representing the data words with the most significant byte of each word being transmitted first. **In the HP 54111D, the upper or most significant byte will normally be zero with the lower byte containing a data value of 0 to 255.** If there is a hole in the data, it will be represented by the 16-bit value of -1. Holes will exist only in the limited case where data was acquired in the repetitive mode with sweep ranges set faster than 50 ns/division and 100% completion not selected. The range of data in the word format is from 0 to 255. WORD format is useful in applications where the information is being read directly into an integer array in a 16-bit controller.

BYTE Format The BYTE formatted waveform records are transmitted over the HP-IB using the binary block format (#A format) specified in IEEE Standard 728-1982. The character string "#A" is sent first, followed by a two-byte length value (16-bit binary) specifying the number of bytes to follow. The length value is sent high byte first. The number of bytes transmitted when the format is BYTE is the same as the value returned by the WAVEFORM POINTS? query. This is followed by a sequence of bytes representing the data words. BYTE formatted transfers run faster than WORD formatted transfers in most applications since the number of data elements transmitted over the HP-IB is cut in half.

Eight bits of resolution are retained in the BYTE format by reserving the data value of 255 to indicate holes in the data. Data holes will exist only in the limited case where data was acquired in the repetitive mode with the sweep speed set above 50 ns/division and 100% completion not selected. Actual data values of 255 will be converted to 254 prior to transfer over the HP-IB.

ASCII Format ASCII formatted waveform records are transmitted one value at a time, separated by <CR><LF>s. The data values transmitted are the same as would be sent in the WORD Format except that they are converted to an integer ASCII format (six characters) before being sent over HP-IB. If there is a hole in the data it will be represented by a negative 1 (-000001). Holes exist only in the limited case where data was acquired in the repetitive mode with the sweep speed set above 50 ns/division and where 100% completion was not achieved.

The range of values for data is from 0 to 255.
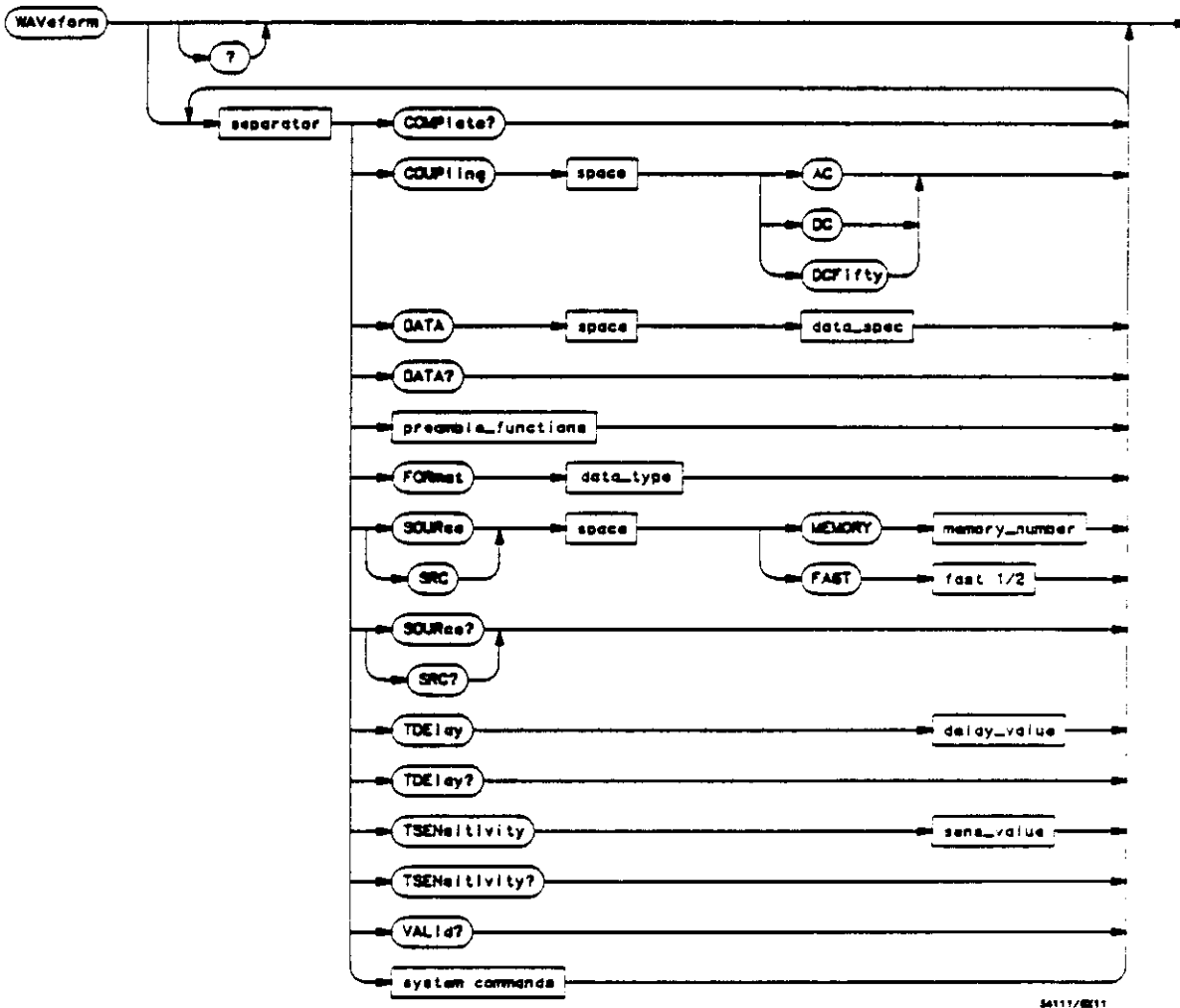
Figure 11-1. Waveform Subsystem Syntax Diagrams.
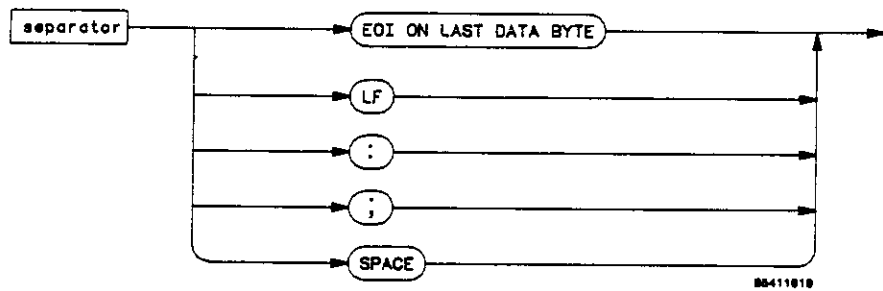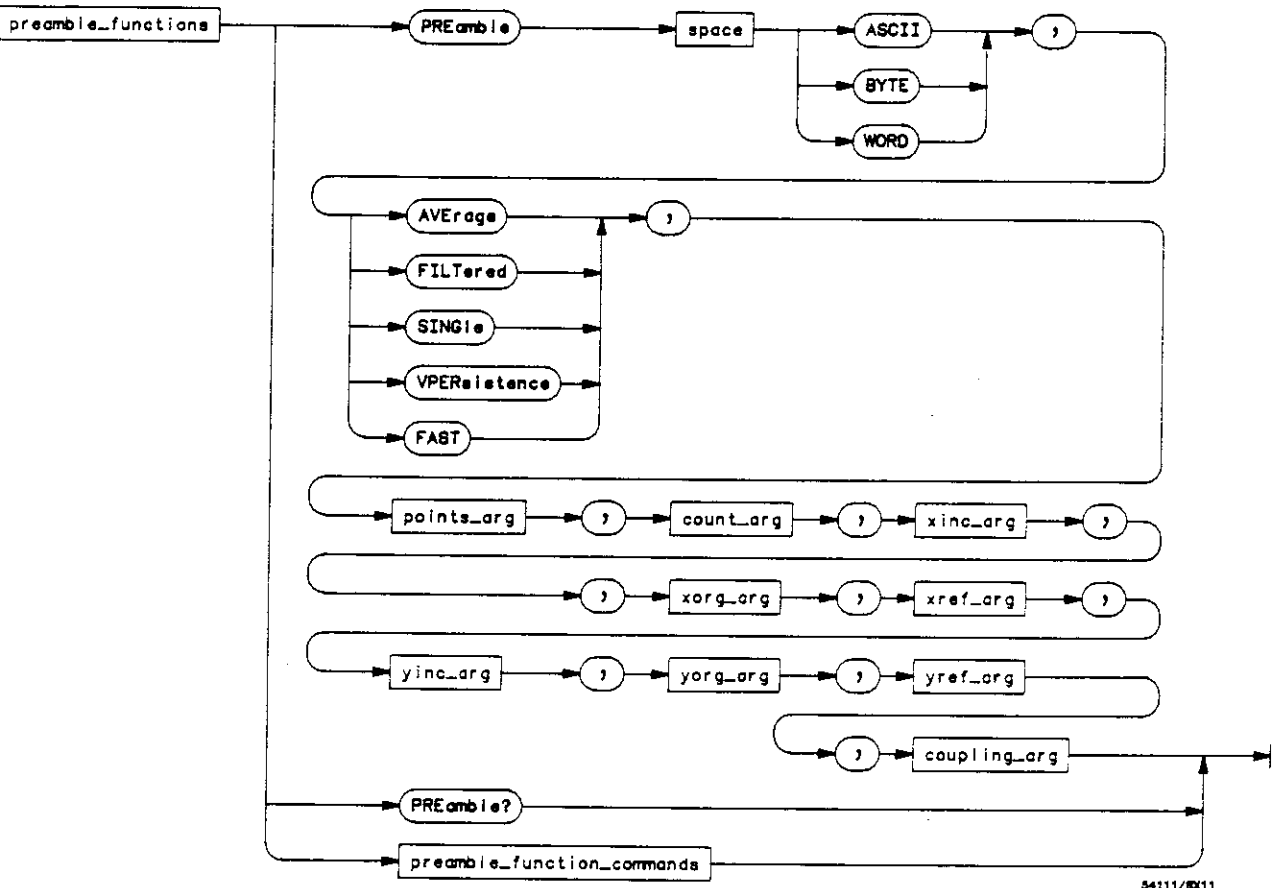
Figure 11-1. *Waveform Subsystem Syntax Diagrams (continued).*

COUNT_ARG = An integer from 1 to 64.

COUPLING_ARG = AC or DC or DCFIFTY or GROUND.

DATA_SPEC = A block of data in #A format as defined in IEEE Std. 728-1982.

DELAY_VALUE = time delay in seconds.

MEMORY_NUMBER = An integer 1 through 8.

FAST_NUMBER = An integer 1 or 2.

POINTS_ARG = An integer, 501 or 8192.

SENS_VALUE = time/div in seconds.

XINC_ARG = A real number from 10 ps to 2 ms.

XORG_ARG = A real number with the following restrictions:
  The maximum value is 60,000 x timebase range or 1.6 s, which ever is greater.
  If the delay reference is left then the minimum value is −200 ms or −(timebase range)
    whichever is smaller.
  Else if the delay reference is center then the minimum value is the lesser of
    −5 x (timebase range) and −200 ms + (10 x timebase range).
  Else if the delay reference is right then the minimum value is the lesser of 0 and
    −200 ms + (10 x timebase range).

XREF_ARG = 0

YINC_ARG = A real number equal to 1/128 x voltage range.

YORG_ARG = A real number with a magnitude less than 1.5 x voltage range.

YREF_ARG = 128.

Figure 11-1. Waveform Subsystem Syntax Diagrams (continued).

Waveform Subsystem
11-10

WAVeform   **command/query**

The WAVEFORM command selects the waveform subsystem as the destination for the WAVEFORM subsystem commands that follow.

The query returns the WAVEFORM subsystem parameters.

Command Syntax: WAVeform

Example: OUTPUT 707;"WAVEFORM"

Query Syntax: WAVeform?

Returned Format: [WAVeform]<CR><LF>
[SOURce]<specified source><CR><LF>
[VALid]<NR1><CR><LF>
[FORMat]<argument><CR><LF>
[TYPE]<argument><CR><LF>
[POINts]<NR1><CR><LF>
[COUNt]<NR1><CR><LF>
[XINCrement]<NR3><CR><LF>
[XORigin]<NR3><CR><LF>
[XREFerence]<NR1><CR><LF>
[YINCrement]<NR3><CR><LF>
[YORigin]<NR3><CR><LF>
[YREFerence]<NR1><CR><LF>
[COUPling]<argument><CR><LF>
[COMPlete]<NR1><CR><LF>

Example: DIM Wav$[300]
OUTPUT 707;"EOI ON"
OUTPUT 707;"WAVEFORM?"
ENTER 707 USING "-K";Wav$
PRINT USING "K";Wav$ COMPlete?

COMPlete? query

This query returns the completion value that is in the currently selected Waveform Subsystem.

Query Syntax: COMPlete?

Example: OUTPUT 707;"COMPLETE?"
ENTER 707;Comp
PRINT Comp

COUNt? | CNT?   query

The response to the COUNT Query varies with the type of Acquisition used.

In the AVERAGE mode the COUNT Query returns the least number of averages which have been achieved by any point in the 501 point average record.

In the VPERSISTENCE, NORMAL and FILTERED mode the COUNT is incremented by one for each data acquisition mode. Where DIGITIZE is excuted, the last count number is written into Waveform Memory Count.

In the FAST mode count is incremented each time a DIGITIZE is excuted.

Count may be cleared with a system ERASE DISPLAY command.

Query Syntax: {COUNt | CNT}?

Returned Format: [COUNt]<value>

                         <value> ::= number of averages in average mode
                                   (integer · NR1 format)

Example: OUTPUT 707;"COUNT?"
                ENTER 707;Count
                PRINT Count

COUPling   **command/query**

The COUPLING command sets the coupling for the input of the selected waveform source.

Command Syntax: COUPling {DC | AC | GROund | GND | DCFifty}

Example: OUTPUT 707;"COUPLING AC"

Query Syntax: COUPling?

Returned Format: Coupling {DC | AC | GROund | DCFifty}

Example: DIM Coup$[30]
OUTPUT 707;"COUPLING?"
ENTER 707;Coup$
PRINT Coup$

DATA   **command/query**

The DATA command causes the instrument to accept a waveform data record over the HP-IB from the controller and store it in the previously specified waveform memory.

Note

The record format must match the format previously specified for the memory by its preamble.

The query returns the waveform record stored in the previously specified waveform memory.

Command Syntax: DATA

Example: OUTPUT 707;"DATA"

Query Syntax: DATA?

Returned format: [DATA]#A<binary block length in bytes><binary block><CR><LF>

The binary block length field is two bytes long. The high byte is passed first.

The following program moves data from the 54111D to the controller and then back to the 54111D using the WAVEFORM DATA command and query. For this example program use the instrument's cal signal and connect it to channel 1.

```
10 CLEAR 707
20 OUTPUT 707;"RESET"
30 OUTPUT 707;"AUTOSCALE"
40 ASSIGN @Fast TO 707;FORMAT OFF
50 OUTPUT 707;"ACQUIRE TYPE NORMAL COMPLETE 90"
60 OUTPUT 707;"DIG1"
70 OUTPUT 707;"HEADER ON LONGFORM OFF EOI ON"
80 OUTPUT 707;"WAVEFORM SOURCE MEMORY1 FORMAT WORD"
90 OUTPUT 707;"DATA?"
100 ENTER 707 USING "#,8A,W";Header$,Byte_len
110 Word_len=Byte_len/2
120 ALLOCATE INTEGER Waveform(1:Word_len)
130 ENTER @Fast;Waveform(*)
140 DIM Preamble$[150]
150 OUTPUT 707;"PREAMBLE?"
160 ENTER 707 USING "-K";Preamble$
170 OUTPUT 707;Preamble$
180 OUTPUT 707 USING "#,8A,W";Header$,Byte_len
190 OUTPUT @Fast;Waveform(*)
200 OUTPUT 707;"VIEW MEMORY1 BLANK CHANNEL1"
210 END
```

FORMat

The FORMAT command sets the data transmission mode for the waveform data. When the ASCII mode is selected the data is ASCII digits with each data value separated by a <CR><LF>. If the BYTE mode is selected the data is sent as eight-bit integers, while a WORD formatted transfer transfers the data as 16-bit integers in two bytes. In the FAST mode only, FORMAT also determines how data is stored in the FAST data buffers.

The query returns the current transfer format for the previously specified memory.

Command Format: FORMat {ASCII | BYTE | WORD}

Example: OUTPUT 707;"FORMAT WORD"

Query Syntax: FORMat?

Returned Format: [FORMat]<mode><CR><LF>

<mode>::= {ASCII | BYTE | WORD}

Example: DIM Format$[30]
OUTPUT 707;"FORMAT?"
ENTER 707;Format$
PRINT Format$

POINts? | PNTS?   **query**

The POINTS query returns the points value in the currently selected
waveform preamble. The POINTS value is the number of points
acquired in the last DIGITIZE command to the selected waveform
memory.

Query Syntax: {POINts | PNTS}?

Returned Format: [POINts]<value><CR><LF>

                            <value>::= number of acquired data points
                                       (integer - NR1 format)

Example: OUTPUT 707;"POINTS?"
                ENTER 707;Points
                PRINT Points

PREamble   **command/query**

This command sends a waveform preamble to the selected waveform memory in the instrument.

The query returns the preamble from the previously specified memory.

Command Syntax: PREamble <preamble block>
                               <preamble block> ::= <format>,
                                                     <type>,
                                                     <points>,
       <count>,
       <xincrement>,
       <xorigin>,
       <xreference>,
       <yincrement>,
       <yorigin>,
       <yreference>,
       <coupling>

Query Syntax: PREamble?

Returned Format: [PREamble]<format parameter>,
       <type parameter>,
       <points NR1>,
       <count NR1>,
       <xincrement NR3>,
       <xorigin NR3>,
       <xreference NR1>,
       <yincrement NR3>,
       <yorigin NR3>,
       <yreference NR1>,
       <coupling parameter>

Example: This example program uses both the command and query form of the PREAMBLE command.

```
10 DIM Pre$[120]
20 OUTPUT 707;"HEADER OFF"
30 OUTPUT 707;"WAVEFORM"
40 OUTPUT 707;"PREAMBLE?"
50 ENTER 707 USING "-K";Pre$
60 PRINT USING "K";Pre$
70 OUTPUT 707 USING "#,K";"PREAMBLE",Pre$
80 END
```

Note

In line 70 of the program example, a space is inserted between the word "PREAMBLE " and the close quote mark. This space must be inside the quotation mark because in this format (#,K) the data is packed together. Failure to put the space will produce a word (PREAMBLEWORD, PREAMBLEBYTE, or PREAMBLEASCII) that is not a proper command word.

SOURce | SRC   command/query

The SOURCE command selects the memory that is to be used as the source for the waveform commands. The command may also be used to set the active display in the Wfm Save menu.

The query returns the currently selected source for the waveform commands.

Command Syntax: {SOURce | SRC} MEMory {1 | ... | 8 | Fast {1 ... 2}

Example: OUTPUT 707;"SOURCE MEMORY3"

Query Syntax: SOURce?

Returned format: [SOURce]<argument><CR><LF>

Example:
DIM Src$[30]
OUTPUT 707;"SOURCE?"
ENTER 707;Src$
PRINT Src$

TDElay   **command/query**

The TDELAY command sets the amount of time delay on the
currently selected Waveform Source.

The query returns the current amount of time delay on a displayed
waveform memory.

Command Syntax: TDElay<value>

                             <value> ::= time delay in seconds

Example: Output 707;"TDELAY .0003"

Query Syntax: TDElay?

Returned Format: [TDElay]<time>

                             <time> ::= time in seconds
                                        (exponential - NR3 format)

Example: OUTPUT 707;"TDElay?"
                ENTER 707;Tde
                PRINT Tde

TSENsitivity   command/query

The TSENsitivity command sets the TIME/DIV for the currently selected Waveform Source. This is similar to the "zoom" function from the front panel (see Chapter 7, *HP 54111D Front-panel Reference Manual*).

The query returns the current TIME/DIV for the displayed waveform.

Command Syntax: TSENSITIVITY<time>

<time> ::= time/div in seconds

Example: OUTPUT 707;"SENSITIVITY 5.0E-6"

Query Syntax: TSENsitivity?

Returned Format: [sensitivity]<time><CR><LF>

<time>::= TIME/DIV setting of currently
selected waveform memory
(exponential - NR3 format)

Example: OUTPUT 707;"TSENSITIVITY?"
ENTER 707;Sens
PRINT Sens

TYPE command/query

The TYPE command sets the data type for the currently selected memory.

The query returns the data type for the previously specified memory.

Command Syntax: TYPE {VPERsistence | FILTered | NORMal | AVErage}

Example: OUTPUT 707;"TYPE AVERAGE"

Query Syntax: TYPE?

Returned Format: [TYPE]<mode><CR><LF>

<mode> ::= {AVErage | FILTered | NORMal | VPERsistence }

Example:
DIM Type$[30]
OUTPUT 707;"TYPE?"
ENTER 707;Type$
PRINT Type$

VALid?   **query**

The VALID query returns 0 if there is no data in the memory. If there is valid data in the previously selected memory the response will be 1.

Query Syntax: VALid?

Returned format: [VALid] {0 | 1}

Example: OUTPUT 707;"VALID?"
ENTER 707;Val
PRINT Val

XINCrement? **query**

This query returns the x-increment value currently in the preamble. This value is the time difference between consecutive data points for NORMAL, AVERAGE, or FILTERED data.

Query Syntax: XINCrement?

Returned Format: [XINCrement]<value><CR><LF>

<value> ::= x-increment in the current preamble
(exponential - NR3 format)

Example: OUTPUT 707;"XINCREMENT?"
ENTER 707;Xin
PRINT Xin

XORigin?   query

The XORIGIN query returns the x-origin value currently in the
preamble. This value is the time of the first data point in the memory
with respect to the trigger point.

Query Syntax: XORigin?

Returned Format: [XORigin]<value><CR><LF>

                        <value> ::= x-origin value currently in preamble
                                   (exponential - NR3 format)

Example: OUTPUT 707;"XORIGIN?"
                 ENTER 707;Xor
                 PRINT Xor

XREFerence?   **query**

The XREFERENCE query returns the current x-reference value in the preamble. This value specifies the data point that is associated with the x-origin data values.

Query Syntax: XREFerence?

Returned Format: [XREFerence]<value><CR><LF>

<value> ::= x-reference value in the current preamble
(exponential - NR3 format)

Example: OUTPUT 707;"XREFERENCE?"
ENTER 707;Xref
PRINT Xref

YINCrement? query

The YINCREMENT query returns the y-increment value currently in the preamble. This value is the voltage difference between consecutive data points.

Query Syntax: YINCrement?

Returned Format: [YINCrement]<value><CR><LF>

<value> ::= y-increment value in the current preamble
(exponential - NR3 format)

Example: OUTPUT 707;"YINCREMENT?"
ENTER 707;Yin
PRINT Yin

YORigin? query

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

Query Syntax: YORigin?

Returned Format: [YORigin]<NR3><CR><LF>

Example: OUTPUT 707;"YORIGIN?"
ENTER 707;Yor
PRINT Yor

YREFerence?   query

The YREFERENCE query returns the current y-reference value in the preamble. This value specifies the data point where the y-origin occurs. The Y-reference is always 128 in the HP 54111D.

Query Syntax: YREFerence?

Returned Format: [Y-REFerence]<value><CR><LF>

> <value> ::= y-reference value in the current preamble
> (exponential - NR3 format)

Example: OUTPUT 707;"YREFERENCE?"
ENTER 707;Yref
PRINT Yref

Note

*Voltage = (point − y-reference) x (y-increment) + (offset).
As an example, if the y-reference is 128, and the y-origin
is −4.40E − 3 V, and the y-increment is 1.5625E − 4 V, then
a data point whose y value is 70 corresponds to 5.45 V.*

12 REMOTE OPERATION

Introduction

This section contains more about the remote operation of the HP 54111D over the Hewlett Packard Interface Bus (HP-IB). With the exception of the line switch, all front panel functions and some instrument features that are remote only operations can be controlled by sending the appropriate command over the HP-IB.

In this manual all HP 54111D program codes are ASCII. Table 12-1 lists ASCII characters and some commonly used equivalent codes.

For additional information concerning HP-IB, refer to IEEE standard 488-1978 or the identical ANSI Standard MC1.1, "IEEE Standard Digital Interface for Programmable Instrumentation."

HP-IB Compatibility

The HP 54111D's HP-IB compatibility as defined in the IEEE standard 488-1978 appears in table 12-2.

Twelve HP-IB meta messages are listed in the left hand column of table 12-2. The most significant of these are the Data messages, which contain the program codes that set the instrument's mode of operation.

The HP 54111D supports the following HP-IB interface functions: SH1, AH1, T5, TE0, L3, LE0, SR1, RL1, PP0, DC1, DT1, C0, E1.

Remote Mode

The HP 54111D communicates over the HP-IB in both the local and remote modes. In the remote mode, all front panel controls except the STANDBY switch and LOCAL key are disabled. When Local Lockout is set, the LOCAL key is also disabled.

The HP 54111D can be addressed to listen or talk while in the remote mode. When addressed to talk, the instrument automatically stops listening and sends either a Data message or the Status Byte.

Table 12-1. Commonly Used Code Conversions.

HP-IB		ASCII	Decimal	Binary	Octal	Hexa-decimal	HB-IB		ASCII	Decimal	Binary	Octal	Hexa-decimal
Addressed Command Group (ACG)	GTL	☺	0	00 000 000	000	00	Talk Address Group (TAG) Note 2	T0	@	64	01 000 000	100	40
		☺	1	00 000 001	001	01		T1	A	65	01 000 001	101	41
		x	2	00 000 010	002	02		T2	B	66	01 000 010	102	42
		—	3	00 000 011	003	03		T3	C	67	01 000 011	103	43
	SDC	■	4	00 000 100	004	04		T4	D	68	01 000 100	104	44
	PPC		5	00 000 101	005	05		T5	E	69	01 000 101	105	45
		L	6	00 000 110	006	06		T6	F	70	01 000 110	106	46
		H	7	00 000 111	007	07		T7	G	71	01 000 111	107	47
	GET	■	8	00 001 000	010	08		T8	H	72	01 001 000	110	48
	TCT	⌐	9	00 001 001	011	09		T9	I	73	01 001 001	111	49
		LF	10	00 001 010	012	0A		T10	J	74	01 001 010	112	4A
		Ω	11	00 001 011	013	0B		T11	K	75	01 001 011	113	4B
		*	12	00 001 100	014	0C		T12	L	76	01 001 100	114	4C
		CR	13	00 001 101	015	0D		T13	M	77	01 001 101	115	4D
		↑	14	00 001 110	016	0E		T14	N	78	01 001 110	116	4E
		↓	15	00 001 111	017	0F		T15	O	79	01 001 111	117	4F
Universal Command Group (UCG)		∴	16	00 010 000	020	10		T16	P	80	01 010 000	120	5D
	LLO	Ä	17	00 010 001	021	11		T17	Q	81	01 010 001	121	51
		Ø	18	00 010 010	022	12		T18	R	82	01 010 010	122	52
		Ü	19	00 010 011	023	13		T19	S	83	01 010 011	123	53
	DCL	å	20	00 010 100	024	14		T20	T	84	01 010 100	124	54
	PPU	ô	21	00 010 101	025	15		T21	U	85	01 010 101	125	55
		ù	22	00 010 110	026	16		T22	V	86	01 010 110	126	56
		ß	23	00 010 111	027	17		T23	W	87	01 010 111	127	57
	SPE	←	24	00 011 000	030	18		T24	X	88	01 011 000	130	58
	SPD	→	25	00 011 001	031	19		T25	Y	89	01 011 001	131	59
		µ	26	00 011 010	032	1A		T26	Z	90	01 011 010	132	5A
		Δ	27	00 011 011	033	1B		T27	[	91	01 011 011	133	5B
		h	28	00 011 100	034	1C		T28	\	92	01 011 100	134	5C
		P	29	00 011 101	035	1D		T29	]	93	01 011 101	135	5D
		⌐	30	00 011 110	036	1E		T30	^	94	01 011 110	136	5E
		⌐	31	00 011 111	037	1F	UNT		_	95	01 011 111	137	5F
Listen Address Group (LAG) Note 1	L0	SP	32	00 100 000	040	20	Secondary Command Group (SCG) Note 3	S0	`	96	01 100 000	140	60
	L1	!	33	00 100 001	041	21		S1	a	97	01 100 001	141	61
	L2	"	34	00 100 010	042	22		S2	b	98	01 100 010	142	62
	L3	#	35	00 100 011	043	23		S3	c	99	01 100 011	143	63
	L4	$	36	00 100 100	044	24		S4	d	100	01 100 100	144	64
	L5	%	37	00 100 101	045	25		S5	e	101	01 100 101	145	65
	L6	&	38	00 100 110	046	26		S6	f	102	01 100 110	146	66
	L7	'	39	00 100 111	047	27		S7	g	103	01 100 111	147	67
	L8	(	40	00 101 000	050	28		S8	h	104	01 101 000	150	68
	L9	)	41	00 101 001	051	29		S9	i	105	01 101 001	151	69
	L10	*	42	00 101 010	052	2A		S10	j	106	01 101 010	152	6A
	L11	+	43	00 101 011	053	2B		S11	k	107	01 101 011	153	6B
	L12	,	44	00 101 100	054	2C		S12	l	108	01 101 100	154	6C
	L13	-	45	00 101 101	055	2D		S13	m	109	01 101 101	155	6D
	L14	.	46	00 101 110	056	2E		S14	n	110	01 101 110	156	6E
	L15	/	47	00 101 111	057	2F		S15	o	111	01 101 111	157	6F
	L16	0	48	00 110 000	060	30		S16	p	112	01 110 000	160	70
	L17	1	49	00 110 001	061	31		S17	q	113	01 110 001	161	71
	L18	2	50	00 110 010	062	32		S18	r	114	01 110 010	162	72
	L19	3	51	00 110 011	063	33		S19	s	115	01 110 011	163	73
	L20	4	52	00 110 100	064	34		S20	t	116	01 110 100	164	74
	L21	5	53	00 110 101	065	35		S21	u	117	01 110 101	165	75
	L22	6	54	00 110 110	066	36		S22	v	118	01 110 110	166	76
	L23	7	55	00 110 111	067	37		S23	w	119	01 110 111	167	77
	L24	8	56	00 111 000	070	38		S24	x	120	01 111 000	170	78
	L25	9	57	00 111 001	071	39		S25	y	121	01 111 001	171	79
	L26	:	58	00 111 010	072	3A		S26	z	122	01 111 010	172	7A
	L27	;	59	00 111 011	073	3B		S27	{	123	01 111 011	173	7B
	L28	<	60	00 111 100	074	3C		S28	|	124	01 111 100	174	7C
	L29	=	61	00 111 101	075	3D		S29	}	125	01 111 101	175	7D
	L30	>	62	00 111 110	076	3E		S30	~	126	01 111 110	176	7E
	UNL	?	63	00 111 111	077	3F		S31	DEL	127	01 111 111	177	7F

NOTES: 1. L<n>=MLA assigned to device number <n>.
2. T<n>=MTA assigned to device number <n>.
3. Meaning defined by Primary Command Group code.

Whether addressed or not, the HP 54111D responds to the Local,
Local Lockout, Clear Lockout/Set Local, Trigger, and Abort
Messages. The instrument may also output a Require Service
Message.

The local to remote mode change is accomplished when a remote
message is sent to the HP 54111D. This message contains two parts:
- Remote Enable (REN) bus control line true.
- Device Listen Address (MLA) received once while REN is true.

All instrument settings remain unchanged with the local-to-remote
transition. The local-to-remote transition disables the front panel
with the exception of the STANDBY power switch and the LOCAL
key.

The HP 54111D must be in the HP-IB Talk/Listen mode before the
local to remote transition can be made.

Table 12-2. HP-IB Message Reference Table.

HP-IB Meta Message	Applicable	Instrument Response	Related Commands and Control Lines		Interface Functions
Data	Yes	All front panel, menu, and remote functions except LINE switch. Also, all instrument settings may be read via the HP-IB.	DAB EOI EOS	MLA UNL MTA UNT OTA	L3 T5
Trigger	Yes	Responds as if the "RUN" System command was issued.	GET MLA		DT1
Clear	Yes	Responds by: • Terminating bus communication • Clearing serial poll bits • Clearing input and output buffers • Clearing error queue and key register	DCL SDC		DC1
Remote	Yes	Enabled to remote mode when the REN bus control line is true. However, it remains in local until it is addressed to listen the first time.	REN MLA		RL1
Local	Yes	Returns from remote to local when it receives the Local message or the LOCAL key is pressed. Settings remain unchanged after the remote-to-local transition.	GTL MLA		RL1
Local Lockout	Yes	When in remote and local lockout is in effect, the front panel is disabled. Only the system controller can return the instrument to local.	LLO		RL1

Table 12.2. HP-IB Message Reference Table (continued)

HP-IB Meta Message	Applicable	Instrument Response	Related Commands and Control Lines	Interface Functions
Clear Lockout Set/Local	Yes	Returns to local and local lockout is clear when the REN bus control line goes false.	REN	RL1
Pass/Take Control	No	The controller subset is not implemented.	TCT	C0
Require Service	Yes	Sets the SRQ line true when one of the service request conditions occur, if it has been enabled to send enabled to send the RQS message for that condition.	SRQ	SR1
Status Byte	Yes	Responds to a Serial Poll Enable (SPE) bus command by sending an 8-bit byte when it is addressed to talk. Bit 6 (RQS bit) is true if the HP 54111D has set the SRQ bus control line true. The byte is cleared after it is read by the HP-IB controller if the RQS bit was set.	SPE SPD STB	T5
Status Bit	No	Does not respond to a parallel poll.	PPE PPC PPD PPU	PP0
Abort	Yes	Is unaddressed to listen or talk.	IFC	T5 L3

Local Mode

When the HP 54111D is in the local mode all front panel controls are operational, and the instrument will not respond to input data over the bus. If the unit is addressed to talk it can send data messages and the status byte. Whether addressed or not, the HP 54111D will respond to the remote, local, local lockout, clear lockout/set local, trigger (GET) and abort messages. The HP 54111D can output a require service message while in the local mode.

This instrument always switches from remote to local whenever it receives the Go-To-Local message (GTL) or the clear lockout/set local message. The clear lockout/set local message sets the remote enable control line (REN) false. If the unit is in the local lockout mode, the LOCAL key on the front panel will be disabled.

The instrument's settings remain unchanged during remote-to-local transition.

Local Lockout

If the HP 54111D is under remote (program) control and the front panel LOCAL key is inadvertently pressed, the instrument returns to local control. Data and/or settings could be changed. To prevent this you may use the Local Lockout message. This command allows return-to-local only under program control.

It should be considered that return-to-local can be accomplished by cycling the power switch, however this technique has several disadvantages. When this method is used to return-to-local the system controller may lose control of the instrument. Several of the HP-IB conditions will be reset to the default states at power up, and the status of any operation that was in progress is unknown by the controller. It is possible that this could "hang-up" the system bus.

Addressing

When the "Talk/Listen" HP-IB mode is selected from the front panel, the instrument may be addressed to talk or listen using the bus commands.

If you address the instrument to listen it will remain configured to listen until it receives an abort message (IFC), its own talk address (MTA), or a universal unlisten command (UNL) from the controller.

If you address the instrument to talk it will remain configured to talk until it receives an abort message (IFC), another instrument's talk address (OTA), its own listen address (MLA), or a universal untalk command (UNT).

The HP 54111D is shipped from the factory in the addressable mode, with the talk and listen addresses set to seven (T=7 and L=7). Refer to table 12-1 for equivalent address codes. The instrument can also be configured in the talk-only or listen-only mode. These modes enable limited bus operation without an HP-IB controller being connected. The HP 54111D's address and addressing mode may be displayed or changed from the front panel.

If the instrument is set to the listen-only mode, it responds to all data messages sent on the HP-IB. However, it cannot output data messages and is inhibited from responding to the remote, local, local lockout, clear lockout/set local, or abort messages. In this mode the unit cannot issue the require service message and cannot respond to

a serial poll. When the instrument is in this mode the front panel is enabled, therefore you can change settings while a program is executing.

If the instrument is set to the talk-only mode it does not respond to any of the bus messages. Select this mode if the HP 54111D is to output data directly to an HP-IB plotter or printer without the aid of an HP-IB controller.

HP-IB Default Conditions

Several HP-IB parameters are reset during power-up, however both the unit's address and addressing mode are saved in non-volatile memory.

HP-IB default conditions are:
- HP-IB local mode
- Local-lockout cleared
- Unaddressed (if in normal addressing mode)
- RQS mask set to decimal 32546 (bits 1,5,8-14 set)
- Status byte register cleared

Data Messages

The HP 54111D communicates on the HP-IB primarily with data messages. The instrument interprets a byte on the eight data lines as a data message when the bus is in the data mode. The bus is in the data mode when the Attention (ATN) control line is false. If the ATN control line is true, the bus is in the command mode.

In the data mode, the HP 54111D can receive and send data messages. Input data messages include the instrument's program commands (device dependent commands) used to program front panel functions and remote functions. Output data messages include instrument status information, the settings of specific functions, measurement results, and the learn and cal strings.

The learn and cal strings are binary data strings that contain a condensed coding of the entire instrument state and calibration factors. Refer to the keywords SETUP, SETUP?, CALIBRATE, and CALIBRATE? in Chapter 2, "System Commands."

Receiving the Data Message

The HP 54111D responds to data messages when it is in the remote mode (REN is true) and the unit is addressed to listen, or when it is in the listen-only mode.

Input data messages contain a string of device dependent commands (program commands) and an end-of-string (EOS) message. The program codes within a data message are executed after the EOS message is received. The following format rules must be observed for all input data messages:

- A linefeed (<LF>) or an EOI is used as the EOS message. Each data message must be terminated by a <LF> or by asserting the EOI (End Or Identify) bus signal line with the last byte in the message.

- The carriage return character (<CR>) is not required before <LF>.

- When more than one command is sent in a data message, a space, semicolon, or colon must be used to separate the program commands.

- The total length of a data message string may not exceed 300 characters.

Syntax errors in a data message are trapped and can be reported over HP-IB. Refer to key words "STATUS?" and "ERR?" in Chapter 2 for details concerning detecting and reporting errors.

Sending Data Messages

The HP 54111D can send data messages in local or remote mode, when addressed to talk, or when in the talk-only mode.

Before the instrument is addressed to talk, the desired output data must be specific with the appropriate input data message. Otherwise, the instrument outputs the over-range value "1E38" by default to complete the bus transaction. If the service request is enabled, a service request will be generated with the "Output Buffer Empty" error.

Receiving the Clear Message

The HP 54111D responds to the Clear message <DCL> and Selected Device Clear <SDC> by:
- Clearing all serial poll status bits.
- Clearing the input and output buffers.
- Clearing the error queue and key register.
- Stopping any measurement or acquisition process except the normal background acquire-display cycle.

Receiving the Trigger Message

The Trigger message (GET, group execute trigger) causes the HP 54111D to make a signal acquisition if the unit was in the STOP/SINGLE mode. If the unit is in the AUTO or TRG'D mode, the trigger message will cause the instrument to enable the trigger repeatedly and display the data it acquires on the CRT. Refer to the RUN command in Chapter 2.

Receiving the Remote Message

The remote message has two parts: the Remote Enable bus control line (REN) is held true, then the controller sends a device listen address <MLA>. Instrument settings are unchanged during the transition from local to remote.

Receiving the Local Message

The local message returns the HP 54111D to front panel control. The Go-To-Local message (GTL bus command) addresses the instrument to listen and then switches it from remote to local. None of the instrument settings are changed during this transition.

Although the local message returns the instrument to front panel control, it does not clear the local lockout if it is set.

Receiving the Local Lockout Message

The local lockout message (LLO bus command) disables the HP 54111D's front panel LOCAL key. Local lockout can be set when the instrument is in either local or remote modes. After the local lockout is set and the unit is in the remote mode, local lockout will be enforced. While the unit is in remote and the local lockout is set, the remote to local transition can only be made using an HP-IB command.

Receiving the Clear Lockout/Set Local Message

The Clear Lockout/Set Local message sets the REN control line false and returns the instrument from the remote mode to the local mode. Instrument settings are not changed by this message. It can be sent when the instrument is either in the remote or local mode. The effect of sending this message when the instrument is in the local mode is to clear the local lockout if it is set.

Sending the Require Service Message

The HP 54111D sends the require service message by setting the Service Request (SRQ) bus control line and bit 6 of the status byte true when a previously programmed condition occurs. The instrument can send the require service message in either local or remote mode. The require service message is cleared when a serial poll is executed by the system controller. During serial poll, the SRQ control line is reset immediately before the instrument places the status byte message on the bus. Refer to Chapter 1 for the conditions that can be selected to cause the require service message. If no conditions are selected, the require service message is disabled.

The HP 54111D will not send a require service message unless it is in the Talk/Listen mode.

Receiving the Abort Message

The abort message (IFC control line true) halts all bus activity. When the HP 54111D receives the abort message, it becomes unaddressed and stops talking or listening. The require service message and the status byte are unaffected by the abort message.

A

HP-IB Data Transfer Tips

Introduction

The HP-IB commands and programs given in this section are based on using an HP Series 200/300 controller. If you are using a different computer, the contents of the literal string will remain the same, but the surrounding commands, including input/output statements, will be different.

Getting Started on the Bus

After connecting the HP-IB cable, put the HP 54111D in the addressable mode and select an address. To do this through the HP 54111D's front panel, press the Utility menu key and then press the HP-IB key. Press the top function key until Talk/Listen is selected. Press the second function and set the HP-IB address. Address 7 is used for each of the examples below.

One of the best ways to see whether the controller and oscilloscope are both set up and talking to each other is to send a simple command to the HP 54111D. A good example is:

Output 707; "AUToscale"

The HP 54111D will display "Measuring" in the upper left corner of the screen, above the waveform display. This indicates that the HP 54111D is performing an Autoscale. For more information on the AUTO-SCALE command, refer to Chapter 2, "System Commands."

Fast Data Transfer

Obtaining the fastest possible waveform capture and transfer requires that the controller and the HP 54111D are set up specifically to accomplish this task. The controller uses buffered I/O (fast handshake) and suspends all processing while the transfer is in process.

The HP 54111D should be set up in the following manner:
* Timebase mode — Single
    — this sets the oscilloscope so that the DIGITIZE command doesn't have to interrupt another task that the HP 54111D is performing.

- blank the display
  - the display is normally updated at the end of a DIGITIZE command so that the user can view the signal that has been acquired. If the display is not blanked, then approximately 200 ms is added to the time required to get the information from the instrument.
- data format set to BYTE
  - in FORMAT BYTE, each data point is passed in a single data byte (8-bit integer).
- data filters should be turned off (RESOLUTION off)
  - the data filters require processing time after the waveform has been digitized. These filters are turned OFF to pass raw data.
- transfer the waveform to memory with a DIGITIZE command
  - the DIGITIZE command arms the trigger circuit and stores a waveform into memory as soon as a trigger occurs.

The following program will digitize a waveform and pass the 8k-point record to an HP Series 200/300 controller in approximately 90 ms.

```
10  ! This program will digitize a waveform present on Channel 1
20  ! and pass the 8k record in approximately 290 ms.
30  !
40  !
50  ! Dimension the arrays and setup the I/O buffer
60  !
70      INTEGER Wave(1:4098) BUFFER,Record(1:8196)
80      ASSIGN @Buff TO BUFFER Wave(*);FORMAT OFF
90      ASSIGN @Scope TO 707
100 !
110 ! Setup the 54111 for single shot acquisition with the
120 ! display blanked
130 !
140     OUTPUT @Scope;"ACQ TYPE NORM RESO OFF"
150     OUTPUT @Scope;"TIM MODE SINGLE REF LEFT"
160     OUTPUT @Scope;"DISP BLANK CHAN1 BLANK CHAN2 BLANK
            MEM1 BLANK MEM2"
170     OUTPUT@Scope;"WAV SRC MEM1 FORM BYTE HEAD OFF"
180 !
190 ! Digitize the waveform and pass to controller
200 !
210      OUTPUT @Scope:"DIG1 DATA?"
220      TRANSFER @Scope TO @Buff;COUNT 8196,WAIT
230 !
240 ! Unpack the I/O buffer array
250 !
260      ENTER @Buff USING "B";Record(*)
270      END
```

The program listed above is beneficial for those who need the absolute maximum transfer rate possible. The HP 54111D's BYTE mode sends half the number of bytes of data as the WORD mode. The FORMAT command in line 80 is the fastest data transfer mode for the HP Series 200/300 controllers and uses a 16-bit integer format. The drawback to using the BYTE transfer method is that once the transfer is complete, the I/O buffer must be unpacked. Unpacking the array requires about six seconds (line 260).

If the objective of a fast transfer is to digitize a waveform, pass it to the controller, and then be able to interpret the data as quickly as possible, then the WORD format must be used. WORD format passes the data points in two 8-bit bytes, which represent the data as 16-bit integers. This format is totally compatible with the format off mode of the HP Series 200/300 controllers. Although the additional data bytes require an extra 30 ms of bus time, there is no need to unpack the array in the controller. This saves six seconds.

The program can be modified for WORD transfer by changing the following lines:

```
70    INTEGER Wave (1:8196) BUFFER, Record(1:8196)
170   OUTPUT @Scope; "WAV SRC MEM1 FORM WORD HEAD OFF"
220   TRANSFER @SCOPE TO @BUFF;COUNT 16388,WAIT
```

Data Format   Data is transmitted from the HP 54111D as quantization-level information (figure A-1). The corresponding time value for a particular quantization level is not transmitted, but it can be calculated because the quantization levels are transmitted in order starting at the beginning of the record. Voltage/time pairs can be calculated as shown in figure A-1.
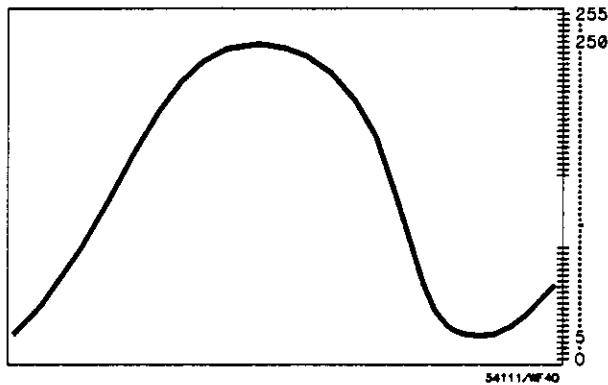
Figure A-1. Data is transferred as quantization levels

Voltage/Time Pairs

To change an array of data into voltage and time pairs, you must request additional information from the HP 54111D. The information includes YREF, YORG, XREF, XINC, XORG. These can be requested using the programming commands described in Chapter 11, "Waveform Subsystem."

Here are the equations to change quantization level data received from the HP 54111D into voltage and time:

$$V = [(YP - YREF) \times YINC] + YORG$$
$$T = [(XP - XREF) \times XINC] + XORG$$

— YP = quantization-level data from the HP 54111D.
— XP = the number of the data point. The data is transmitted in order, starting at the beginning of the record.
— YINC, XINC = scale factors for the vertical and horizontal.
— YREF, XREF = reference values for vertical and horizontal.
— YORG, XORG = offset and delay values for vertical and horizontal.

The values for YINC, YREF, XINC, and XORG can be requested from the HP 54111D using commands described in Chapter 11, "Waveform Subsystem."

B

Example/Demo Programs

Introduction

This chapter contains example programs using the command set for the HP 54111D. In general, they use the long form of the command with alpha (as opposed to numeric) arguments with each command having a separate output statement for clarity. To optimize speed, switch to concatenated short form numerics.

Throughout these examples, the HP 54111D is assumed to be at address 7, the hardcopy devices at address 1, and the system bus at 700. The input signal used is the CAL signal from the front panel connected to channel 1 with a 10:1 probe.

All programs were developed on an HP Series 200 computer using HP BASIC 4.0. Several examples use the BASIC command "ENTER 2." This pauses program execution until the "ENTER" key is depressed on the controller. This is used to separate different blocks in the example to dramatize the features, allow for user interaction, or to wait for the HP 54111D to finish an operation such as hardcopy output or an acquisition.

Vertical Channel Setup Program

```
10  ! This sample program demonstrates some of the commands
20  ! used to set a vertical channel, in this case channel 1.
30  ! This program works well using the cal signal from the
40  ! rear panel of the instrument. The program sets the
50  ! probe attenuation factor for channel 1 to 10:1.
60  !
70  CLEAR 707                            ! Device clear command,
80  ! initializes HP-IB registers.
90  !
100 OUTPUT 707;"AUTOSCALE"              ! Autoscales the unit
110 OUTPUT 707;"ACQ TYPE NORMAL"        ! Real time acquistions
120 OUTPUT 707;"CHANNEL1"               ! Enter Ch1 subsystem
130 GOTO 360
140 OUTPUT 707;"SENS .5"                ! Sensitivity = 500 mV/div
150 OUTPUT 707;"OFFSET 2"               ! Set offset to 2 V
160 OUTPUT 707;"PROBE 10"               ! Sets probe attenuation to 10:1
170 REAL Offset,Range
180 INTEGER J
190 Offset=2                            ! Set offset variable to 2V
200 BEEP
210 DISP "PRESS ENTER - OFFSET positions the waveform on the screen"
220 ENTER 2
230 DISP ""
240 FOR J=1 TO 17
250     OUTPUT 707;"OFFSET";Offset      ! Sets next offset
260     Offset=Offset-.2
270     WAIT .3
280 NEXT J
290 !
300 BEEP
310 DISP "PRESS ENTER - the vertical SENSITIVITY scales the signal"
320 ENTER 2
330 DISP ""
340 !
350 OUTPUT 707;"OFFSET .4"              ! Center screen at 400 mV
360 OUTPUT 707;"range .880"             ! Sets vertical range to 880mV
370 !                                       Can also use "SENSITIVITY"
380 !                                       volts/div
390 Range=.88                           ! Sets range variable to 880 mV
400 !
410 FOR J=1 TO 35
420 OUTPUT 707;"RANGE";Range            ! Sets new range
430 Range=Range+.05
440 WAIT .3
450 NEXT J
460 !
470 !
480 LOCAL 707                           ! Returns the 54111D to local
490 END                                 ! operation.
```

Timebase Program

```
10   ! This is a sample program demonstrating the TIMEBASE
20   ! subsystem. The front panel CAL signal works well
30   ! with this program.
31   !
40   CLEAR 707                          ! Device clear command
50                                      ! initializes the HP-IB registers.
60   !
80   OUTPUT 707;"AUTOSCALE"
90   REAL Sens,Delay
100  INTEGER J
110  OUTPUT 707;"ACQUIRE TYPE FILT"
120  OUTPUT 707;"TIMEBASE"             ! Enter TIMEBASE subsytem.
130  OUTPUT 707;"SENSITIVITY 50E-6"    ! Sets timebase to 200 ns/div.
140                                    !   Can also use "RANGE" for
150                                    ! second full screen.
160  OUTPUT 707;"STOP"
170  !
180  OUTPUT 707;"DELAY 0.0 "           ! Set delay to 0.
190  OUTPUT 707;"REFERENCE CENTER"     ! Puts delay reference at
200                                    ! center of graticule.
210                                    !
220  Delay=0.                          ! Sets delay to 0.
230  FOR J=1 TO 100
240    OUTPUT 707;"OFFSET ";Delay      ! "OFFSET" = "DELAY."
250                                    ! Delay=Delay+4.E-5
260  NEXT J
270  !
280  !
290  BEEP
300  DISP "PRESS ENTER - The horizontal time will be changed"
310  ENTER 2
320  DISP ""
330                                    !
340  Range=.01                         ! Sets full scale to 100 ms
350                                    ! (i.e., 10 ms/div).
360                                    !
370  OUTPUT 707;"TIM DELAY 0"          ! Return delay to 0
380  FOR J=1 TO 15
390    OUTPUT 707;"TIM RANGE";Range    ! Next full scale range.
400    Range=Range/1.7
410  NEXT J
420  !
430  END
```

Measurement Setup Program

```
10  ! This sample program demonstrates some of the commands in the
20  ! Measure Subsystem. It assumes that the front panel CAL signal
30  ! has been connected to CHANNEL 1.
40  !
50  CLEAR 707
60  DIM Measure$[400]
70  OUTPUT 707;"AUTOSCALE"               !
80  OUTPUT 707;"ACQUIRE TYPE             ! Sets display mode to Normal
    NORMAL"
90  OUTPUT 707;"TRIGGER SLOPE POS"  ! Selects positive slope
100 OUTPUT 707;"TIMEBASE RANG 2E-3" ! 200 μs per division
110 OUTPUT 707;"CHAN1 RANG 1.2 OFFS .4"   ! Chan1 set to 1.2 volts full
120                                  ! scale centered at .4 volts
130 !
140 OUTPUT 707;"PROBE 10"               ! Set attenuation to 10:1
150 OUTPUT 707;"DISPLAY"                ! Selects Display Subsystem
160 OUTPUT 707;"FORMAT SINGLE"          ! Selects full screen display
170 OUTPUT 707;"VIEW CHAN1 BLANK        !
    CHAN2"
180 OUTPUT 707;"TMARKER ON"             ! Turn on Tmarkers
190 !
200 !
210 OUTPUT 707;"MEASURE"                ! Selects Measure subsystem
220 OUTPUT 707;"SOURCE CHANNEL 1"  ! Channel 1 is the measurement
                                         source
230 OUTPUT 707;"VSTART .4"              ! Sets voltage markers to
240 OUTPUT 707;"VSTOP .4"               ! .4V. This will be used as a
250 !                                     reference for the edge find
260 !                                     function.
270 BEEP
280 DISP "PRESS ENTER watch the time markers move to signal EDGES"
290 ENTER 2                             ! This statement causes a pause in
300 !                                     the program, press ENTER on the
310 !                                     controller to continue.
320 DISP ""
330 !
340 INTEGER J
350 FOR J=1 TO 4
360     OUTPUT 707;"ESTART +";J         ! Find Jth positive edge
370     WAIT .75
380     OUTPUT 707;"ESTOP -";J          ! Find Jth negative edge
390     WAIT .75
400 NEXT J
410 !
420 BEEP
430 DISP "PRESS ENTER – VOLTAGE and TIME markers increment across
    screen"
```

```
440 ENTER 2
450 DISP ""
460 !
470 !
480 !
490 !
500 REAL Delay,Offset
510 Delay=0
520 Offset=0                                     ! Initialize offset variable
530 FOR J=1 TO 21
540    OUTPUT 707;"TSTART";-1.E-3-Delay    ! Move time start marker
550    OUTPUT 707;"TSTOP";1.E-3+Delay      ! Move time stop marker
560    OUTPUT 707;"VSTART";(-.2-Offset)/10  ! Move voltage start marker
570    OUTPUT 707;"VSTOP";1+Offset         ! Move voltage stop marker
580    Offset=Offset-.06
590    Delay=Delay-1.00E-4
600 NEXT J
610 !
620 BEEP
630 DISP "PRESS ENTER to take AUTOMATIC MEASUREMENTS"
640 ENTER 2
650 DISP ""
660 !
670 !
680 OUTPUT 707;"MEASURE "
690 OUTPUT 707;"ALL?"                       ! Measure all parameters. They will
700 !                                        be displayed on scope, and are
710 !                                        availible over HP-IB.
720 !
730 ENTER 707 USING "-K";Measure$
740 PRINT USING "K";Measure$
750 !
760 !
770 BEEP
780 DISP "PRESS ENTER to accurately measure RISE TIME"
790 ENTER 2
800 DISP ""
810 !
820 !
830 OUTPUT 707;"TIMEBASE RANGE          ! Expand the display
    5E-6"
840 !
850 OUTPUT 707;"MEASURE"
860 OUTPUT 707;"RISE?"                   ! Measure rise time
870 !
880 ENTER 707;Measure$
890 PRINT Measure$
900 LOCAL 707                            ! Puts the 54111D in local
910 END                                  ! operation.
```

Memories/ Single-Shot Measurements Program

```
10  ! This sample program demonstrates using the memories for
20  ! measurements and performing measurements on single shot
30  ! data. Before running the program, connect the CAL signal
40  ! to the Channel 1 input.
50  !
60                                          ! Setup signal.
70  !
80  CLEAR 707                               ! Initialize HP-IB registers.
90  OUTPUT 707;"AUTOSCALE"                  ! Scales the signal.
100 OUTPUT 707;"TIMEBASE RANGE 500-6"       ! Sets to 25 ns/division.
110 !
120 OUTPUT 707;"ACQUIRE TYPE NORM"          ! Set for real time
130 OUTPUT 707;"DIGITIZE CHANNEL 1"         ! Acquire Ch1 data to Memory 1.
140 OUTPUT 707;"BLANK CHANNEL 1"            ! Turn off Ch1.
150 OUTPUT 707;"VIEW MEMORY 1"
160 !
170                                         ! Measure peak to peak voltage
180 !                                         of memory 1.
180 !
190 OUTPUT 707;"MEAS SOURCE MEMORY1"
200 OUTPUT 707;"VPP?"                       ! Set measurement source.
210 DIM Vpp$[20]                            ! Measure peak to peak
                                              voltage.
220 ENTER 707;Vpp$
230 PRINT Vpp$                              ! Print results.
240 END
```

Learn String Program

```
10  !
20  ! This program demonstrates some of the learn string capabilities.
30  CLEAR 707
40  !
50  !
60  DIM Setting$[308]              ! 298 = # of bytes in learn string
70  !                                plus 10 = "SETUP #A**"
80  !                                Where SET = header
90  !                                  #A = indicates binary block
100 !                                  ** = 2 byte integer = length
110 OUTPUT 707;"LONGFORM ON"      ! Use longform of mnemonics.
120 OUTPUT 707;"HEADER ON"        ! Tells the 54111D to precede the
130 !                              learn string with a header.
140 OUTPUT 707;"EOI ON "          ! Tells the 54111D to output an EOI
150 !                              with the last byte.
160 OUTPUT 707;"SETUP?"           ! This asks the scope for the learn string
170 ENTER 707 USING "-K";Setting$ ! Reads in header and string. (-K tells
180 !                              computer to treat CR & LF as data).
190 LOCAL 707                     ! Puts scope in local operation.
200 !
210 !
220 !
230 DISP "Change the instrument SETTINGS, then PRESS ENTER"
240 ENTER 2                       ! This allows you to change the scope's
250 !                              setup.
260 !                              Hit "ENTER" to continue
270 !
280 DISP "The instrument has returned to the PREVIOUS SETTINGS"
290 !
300 OUTPUT 707;Setting$           ! Outputs the learn string and header.
310 !                              Scope is reset to previous setup.
320 !
330 LOCAL 707                     ! Returns scope to local operation.
340 !
350 END
```

Plotter Program

```
 10  ! This sample program demonstrates some of the commands in the
 20  ! Hardcopy subsystem and the PLOT command. It assumes that
 30  ! the scope is at address 7, the plotter is at address 5, and
 40  ! that the system bus is 700.
 50  !
 60  CLEAR 707
 70  OUTPUT 707;"HARDCOPY"        ! Puts the 54111D in the HARDCOPY
 80  !                              subsystem.
 90  OUTPUT 707;"PEN AUTO"        ! Sets the 54111D to the auto
100  !                              pen mode.
110  !
120  OUTPUT 707;"SPEED SLOW"      ! Sets the plotter speed to 10
130  !
140  OUTPUT 707;"PLOT"           ! Outputs data to the plotter
150  SEND 7;UNT UNL              ! Clears bus
160  SEND 7;LISTEN 5             ! Tells printer to listen
170  SEND 7;TALK 7              ! Sets scope to talk mode
180  SEND 7;DATA                ! Sets the HP-IB ATN line true
190  !                            causing data transfer to begin
200  WAIT 180                  ! Wait 3 minutes for transfer to
210  !                           complete.
220  !                           Note: If programming, use the
230  !                           SRQ capabilities of the 54111D
240  !                           to determine when the transfer
250  !                           is complete (see next example
260  !                           program). Attempting to program
270  !                           the 54111D while making
280  END                       ! a hardcopy dump will cause errors.
```

Hardcopy (Service Request) Program

```
10  ! This sample program demonstrates some of the commands in the
20  ! Hardcopy subsystem. The service request is used to detect
30  ! when printing is complete. The program assumes that a
40  ! graphics printer is used and its address is set to 1.
50  !
60  CLEAR 707
70  OUTPUT 707;"REQUEST 4112"      ! Request mask where:
80                                  ! 4096 = Hardcopy complete, bit 12
90                                  !    16 = Ready bit - bit 4,
100                                 !         Set so bit 12 causes SRQ
110 OUTPUT 707;"REQUEST ON"        ! Enables scope's Service Request
120 ON INTR 7 GOTO 260             ! Exit printing routine after SRQ
130 ENABLE INTR 7;2                ! Enables SRQ on bus #7
140 DISABLE INTR 7                 ! Disables all interrupts on bus #7
150 OUTPUT 707;"HARDCOPY"          ! Puts the scope in the HARDCOPY
160                                 ! subsystem.
170 OUTPUT 707;"PAGE AUTO"         ! Sets the scope to automatically
180                                 ! output a formfeed
190 OUTPUT 707;"PRINT"             ! Starts print buffering
200 SEND 7;UNT UNL                 ! Clears bus
210 SEND 7;LISTEN 1                ! Tells printer to listen
220 SEND 7;TALK 7                  ! Puts the scope in talk mode
230 SEND 7;DATA                    ! Lowers the ATN line @ controller
240 ENABLE INTR 7                  ! Enables SRQ interrupt
250 GOTO 250                       ! Loops until printing is complete
260 A=SPOLL(707)                   ! Clear service request
270 END
```

Service Request (Printer) Program

```
 10  ! This sample program demonstrates more uses of the Service
 20  ! Requests (SRQs). This set of instructions uses: Hardcopy
 30  ! Done, Local, Front Panel Service, Ready bit and Ready Masks.
 40  ! The scope will monitor the front panel for SRQ and echo
 50  ! any activity. Any advisories or acquisitions initiated
 60  ! by the front panel will be disclosed. These examples assume
 70  ! the scope to be at address 7 and the printer to be at address
 80  ! 01 on bus #7.
 90  !
100  CLEAR 707
110  DIM K$[80],A$[80]
120  !
130  ON INTR 7 GOSUB Srq__svc        ! Goto "Srq__svc" on Service Request
140  ENABLE INTR 7;2                 ! Enables SRQ on bus #7
150  DISABLE INTR 7                  ! Disables all interrpts on bus #7
160  !
170  PRINT
180  OUTPUT 707;"RESET"              ! Resets 54111D.
190  OUTPUT 707;"AUTOSCALE"
200  OUTPUT 707;"ACQUIRE MODE AVERAGE"
210  WAIT 4                          ! Puts scope into averaged mode.
220  INTEGER Rqsmask                 ! Wait for data accumulation
230  Rqsmask=4096+16+4               ! request mask
240  !                                  4096 = Hardcopy done - bit 12
250  !                                    16 = Ready - bit 4
260  !                                     4 = Front Panel Service - bit 2
270  !                                        Set so bit 12 causes an SRQ
280  OUTPUT 707;"REQUEST";Rqsmask    ! Sends Request Mask
290  OUTPUT 707;"REQUEST ON"         ! Set RQS - bit 6 in the request mask
300  OUTPUT 707;"LONGFORM ON"        ! Sets longform for headers.
310  OUTPUT 707;"HEADER ON"          ! Sets headers on for queries.
320  Stat=SPOLL(707)                 ! Serial Poll scope, should be 0.
330  PRINT "Result of Serial Poll is ";Stat
340  !
350  Dump__flag=0
360  OUTPUT 707;"PAGE AUTO"          ! Sets auto form feed.
370  OUTPUT 707;"PRINT"              ! Starts hardcopy print
380  SEND 7;UNT UNL                  ! Turns off entire bus.
390  SEND 7;LISTEN 1                 ! Sets printer to listen.
400  SEND 7;TALK 7                   ! Sets scope to talk.
410  SEND 7;DATA                     ! Lower ATN line @ controller
420  !
430  ENABLE INTR 7;2                 ! Enables interrupts on bus #7
440  !
450  IF Dump__flag=0 THEN
460      PRINT
```

```
470   PRINT " Waiting for hardcopy transfer to complete."
480   PRINT "Time availible for other tasks."
490   PRINT "!!!! Bus is NOT availible !!!!!"
500   WAIT 2
510   GOTO 450
520 END IF
530 GOTO 530
540 !
550 !
560 !
570 ! Service request interrupt routine
580 !
590 Srq__svc:                          !
600   Stat=SPOLL(707)                  ! Perform serial poll
610   !                                  and clear SRQ.
620   INTEGER J
630   PRINT
640   PRINT "Service Request Status= ";Stat
650   !
660   !
670   IF BIT(Stat,0) THEN              ! 54111D is not a controller.
680     PRINT "RQC should not be set - PROBLEM" ! RQC cannot be set by the
690                                    ! 54111D
700   END IF
710   !
720   !
730   IF BIT(Stat,1) THEN              ! RAM power failure
740     PRINT "PWR status has been set-WHY?"
750   END IF
760   !
770   !
780   IF BIT(Stat,2) THEN              ! Front Panel Service
790     PRINT "FPS status has been set"
800     OUTPUT 707;"KEY?"             ! Asks for key code.
810     ENTER 707;K$                  ! Reads key code.
820     OUTPUT 707;K$                 ! Outputs key code.
830     PRINT " "&K$
840   END IF
850   !
860   !
870   IF BIT(Stat,3) THEN              ! Local operation occurred.
880     PRINT "LCL operation has occured"
890   END IF
900   !
910   !
920   IF BIT(Stat,4) THEN              ! Ready bit - only bit active
930     PRINT "Hardcopy Complete !!"  ! in the Ready byte is Hardcopy
940     IF Dump__flag=0 THEN          ! Complete.
950       SEND 7;UNT UNL
960       Dump__flag=1
```

```
970   END IF
980   PRINT "Now try pressing keys, they will echo from controller"
990 END IF
1000 !
1010 !
1020 IF BIT(Stat,5) THEN              ! Go read the errors.
1030   REPEAT
1040     OUTPUT 707;"ERR?"            ! Asks for next error in queue.
1050     ENTER 707;A$                 ! Reads next error.
1060     PRINT "Error was : ";A$       ! Prints next error.
1070   UNTIL VAL(A$[9,12])=0          ! Until error queue is empty.
1080 END IF
1090 !
1100 !
1110 IF BIT(Stat,6) THEN             ! A SRQ was generated by someone.
1120   OUTPUT 707;"REQUEST?"         ! Asks for mask value.
1130   ENTER 707;A$                  ! Reads mask value.
1140   PRINT A$&" is the mask value"
1150 END IF
1160 !
1170 !
1180 IF BIT (Stat,7) THEN            ! Advisory has been initiated
1190   OUTPUT 707;"DSP?"             ! Asks for advisory
1200   ENTER 707;A$                  ! Reads advisory
1210   PRINT A$&"is the Advisory"     ! Prints advisory
1220 END IF
1230 !
1240 !
1250 ENABLE INTR 7
1260 RETURN
1270 !
1280 END
```

Service Request (Data Acquisition) Program

```
10  ! This sample program demonstrates some of the uses of
20  ! Service Requests (SRQs). This set of instructions uses
30  ! the Acquisition Done, Local, Front Panel Service,
40  ! Ready and Ready masks. An acquisition that will produce
50  ! buffered results will be started. When a SRQ is sent, the
60  ! results will be read and displayed. The scope will then
70  ! monitor the front panel keys using SRQs and echo any
80  ! activity. Acquisitions initiated from the front panel
90  ! will be disclosed.
100 !
110 CLEAR 707
120 !
130 PRINTER IS 1                      ! Display is PRINT destination
140 DIM B$(1:16)[30],K$[80],A$[80]
150 !
160 ON INTR 7 GOSUB Srq__svc          ! Goto "Srq__svc" on Service Request.
170 ENABLE INTR 7;2                   ! Enables SRQ on bus #7.
180 DISABLE INTR 7                    ! Disables all interrupts on bus #7.
190 !
200 PRINT
210 OUTPUT 707;"RESET"               ! Resets 54111D.
220 OUTPUT 707;"AUTOSCALE"
230 INTEGER Rqsmask
240 Rqsmask=1024+16+4                ! Request mask where:
250 !                                   1024 = Acquisition done - bit 10.
260 !                                     16 = Ready - bit 4.
270 !                                      4 = Front Panel Service - bit 2.
280 !                                          Set so bit 10 causes an SRQ
290 OUTPUT 707;"REQUEST";Rqsmask! Sends Request Mask.
300 OUTPUT 707;"REQUEST ON"          ! Sets RQS - bit 6 in Request mask.
310 OUTPUT 707;"LONGFORM ON"         ! Turns on longform for headers.
320 OUTPUT 707;"HEADER ON"           ! Turns headers on for queries.
330 Stat=SPOLL(707)                  ! Serial Poll scope, should be 0
340 PRINT "Result of Serial Poll is ";Stat
350 !
360 Meas__flag=0
370 OUTPUT 707;"ACQUIRE TYPE AVERAGE" ! Selects the average mode
380 OUTPUT 707;"COUNT 64"            ! 64 Averages for completion
390 OUTPUT 707;"MEASURE "            ! Enters MEASURE subsystem.
400 OUTPUT 707;"SOURCE MEM5"         ! Memory 5 is source for measurement
410 OUTPUT 707;"DIGITIZE 1"
420 !
430 ENABLE INTR 7;2                  ! Enables interuppts on bus #7.
440 !
450 BEEP
460 PRINT
470 PRINT " Waiting for averages and measurement"
480 PRINT " to complete. Time available for "
```

```
490 PRINT " other controller tasks. "
500 PRINT
510 PRINT " THE BUS IS AVAILABLE"
520 !
530 OUTPUT 707;"ALL?"                    ! Requests ALL measurements
540 !
550 GOTO 550                             ! Loop until Service Request occurs.
560 !
570 !
580 !
590 ! Service Request Interupt Routine
600 !
610 Srq__svc:!
620 BEEP
630 Stat=SPOLL(707)                      ! Performs a Serial Poll
640                                      ! and clears SRQ.
650 INTEGER J
660 PRINT
670 PRINT "Service Request Status= ";Stat
680 !
690 !
700 IF BIT(Stat,0) THEN                  ! Request Control - 54111D is
710   PRINT "RQC should not be set - PROBLEM" ! not a controller and
720 END IF                                    ! cannot send a RQC.
730 !
740 !
750 IF BIT(Stat,1) THEN                  ! RAM power failure.
760   PRINT "PWR status has been set. WHY?"
770 END IF
780 !
790 !
800 IF BIT(Stat,2) THEN                  ! Front Panel Service.
810   PRINT "FPS status has been set"
820   OUTPUT 707;"Key?"                  ! Asks for key code.
830   ENTER 707;K$                       ! Reads key code.
840   OUTPUT 707;K$                      ! Outputs key code.
850   PRINT " "&K$
860 END IF
870 !
880 !
890 IF BIT(Stat,3) THEN                  ! Local operation occurred
900   PRINT "LCL operation has occurred"
910 END IF
920 !
930 !
940 IF BIT(Stat,4) THEN                  ! Ready bit - only bit active
950   PRINT "Acquisition done !"         ! in the Ready byte is Acq done
960   IF Meas__flag=0 THEN               ! First time thru?
970     FOR J=1 TO 16                    !
980       ENTER 707;B$(J)                ! Reads measurement results
```

```
990      PRINT B$(J)                    ! and prints them.
1000     NEXT J
1010     Meas_flag=1
1020     PRINT
1030     PRINT "Press some keys. The key number will be printed out."
1040  END IF
1050 END IF
1060 !
1070 !
1080 IF BIT(Stat,5) THEN                ! GO read the errors.
1090   REPEAT
1100     OUTPUT 707;"ERR?"              ! Asks for next error in queue.
1110     ENTER 707;A$                   ! Reads error.
1120     PRINT "Error was : ";A$        ! Prints error.
1130   UNTIL VAL(A$[9,12])=0            ! Until error queue is empty.
1140 END IF
1150 !
1160 !
1170 IF BIT (Stat 6) THEN              ! A SRQ has been generated.
1180    OUTPUT 707;"REQUEST?"          ! Ask for mask value.
1190    ENTER 707:A$                   ! Reads mask value.
1200    PRINT A$& "is the mask value"  ! Prints mask value.
1210 END IF
1220 !
1230 !
1240 ! ENABLE INTR 7                    ! SRQ disables interrupts.
1250                                    ! This enables them
1260 RETURN                            ! This re-enables them.
1270 !
1280 END
```

Display (Color) Program

```
10  ! This sample program demonstrates some of the text commands
20  ! in the Display subsystem.
40  !
50  ! Disable the instrument from writing text to the screen and
60  ! set the colors.
70  !
80  CLEAR 707
90  OUTPUT 707;"DISPLAY MASK 0"      ! Disable the instrument from
100 !                                  writing text to the screen
105 OUTPUT 707;"ATTRIBUTE ENABLE" ! Enable embedded attributes
110 OUTPUT 707;"PRIORITY ON"         ! Text overwrites graphics
120 OUTPUT 707;"SETCOLOR DEFAULT" ! Turn on the default colors
130 !
140 ! Write a color block to each position on screen
150 !
160 FOR Row=0 TO 22
170   FOR Column=0 TO 71
180   Attribute=129                    ! Use the inverse video attribute
190   Attribute=Attribute+8*((Row+Column) MOD 16)   ! Set the color
200   OUTPUT 707;"ROW ";Row;"COLUMN ";Column    ! Send row and column
210   ! Output a single character block with its attribute
220   OUTPUT 707 USING "8A,B,2A";"STRING """,Attribute," """
230   NEXT Column
240 NEXT Row
250 !
260 ! To get back to the normal display, turn the instrument off
270 ! and then on again. This resets the display MASK.
290 !
300 END
```

C

Format Statement Descriptions

Introduction

This appendix contains information about the program format statement contained in the program examples in this manual. The information in this appendix has been copied from the *HP BASIC 4.0 Language Reference Manual*, HP manual reorder number 98613-90051, copyright 1985.

The three program commands in this appendix contain most of the format ("USING") statements and parameter definitions that are used in this programming manual. This appendix is included as a convenience to the user who may not have the
HP BASIC 4.0 Language Reference Manual. If you do have the
HP BASIC 4.0 Language Reference Manual available, please refer to it for a more complete description of the programming statements.

ENTER

Option Required	None
Keyboard Executable	Yes
Programmable	Yes
In an IF...THEN...	Yes

This statement is used to input data from a device, file, string, or buffer and assign the values entered to variables.

literal form of image specifier:

Item	Description/Default	Range Restrictions
I/O path name	name assigned to a device, devices, mass storage file, or buffer	any valid name
record number	numeric expression, rounded to an integer	1 through $2^{31}-1$
device selector	numeric expression, rounded to an integer	—
source string name	name of a string variable	any valid name
subscript	numeric expression, rounded to an integer	$-32{,}767$ through $+32{,}767$
image line number	integer constant identifying an IMAGE statement	1 through 32,767
image line label	name identifying an IMAGE statement	any valid name
image specifier	string expression	(see drawing)
numeric name	name of a numeric variable	any valid name
string name	name of a string variable	any valid name
beginning position	numeric expression, rounded to an integer	1 through 32,767
ending position	numeric expression, rounded to an integer	0 through 32,767
substring length	numeric expression, rounded to an integer	0 through 32,767
image specifier list	literal	(see next drawing)
repeat factor	integer constant	1 through 32,767
literal	string constant composed of characters from the keyboard, including those generated using the ANY CHAR key	quote mark not allowed

Format Statement Descriptions
C-4
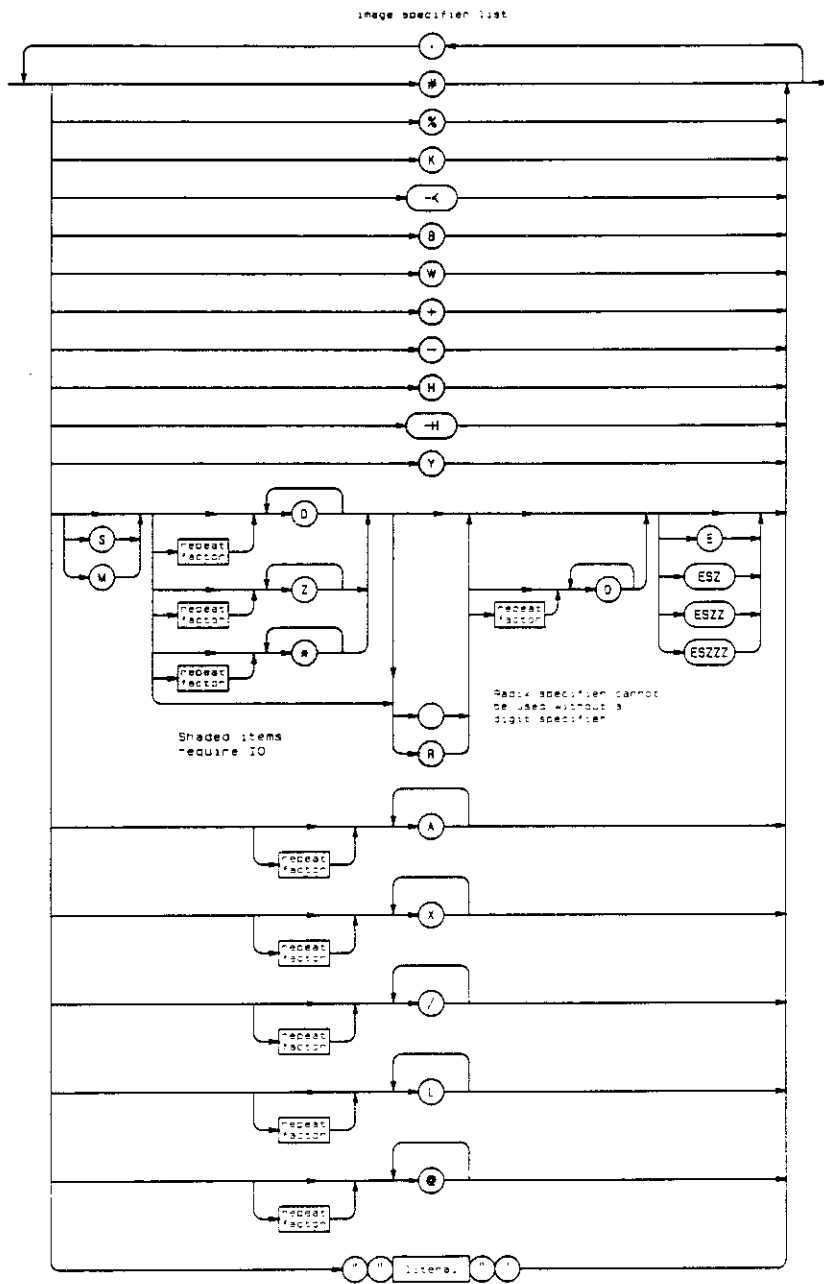
Example	ENTER 705;Number,String$
Statements	ENTER @File;Array(*)
	ENTER @Source USING Fmt5;Item(1),Item(2),Item(3)
	ENTER 12 USING "#,6A";A$[2;6]

Semantics

The Number Builder

If the data being received is ASCII and the associated variable is numeric, a number builder is used to create a numeric quantity from the ASCII representation. The number builder ignores all leading non-numeric characters, ignores all blanks, and terminates on the first non-numeric character, or the first character received with EOI true. Numeric characters are 0 through 9, +, −, decimal point, e, and E, in a meaningful numeric order. If the number cannot be converted to the type of the associated variable, an error is generated. If more digits are received than can be stored in a variable of type REAL, the right-most digits are lost, but any exponent will be built correctly. Overflow occurs only if the exponent overflows.

Arrays

Entire arrays may be entered by using the asterisk specifier. Each element in an array is treated as an item by the ENTER statement, as if the elements were listed separately. The array is filled in row-major order (right-most subscript varies fastest).

Files As Source

If an I/O path has been assigned to a file, the file may be read with ENTER statements. The file must be an ASCII or BDAT file. The attributes specified in the ASSIGN statement are used only if the file is a BDAT file. Data read from an ASCII file is always in ASCII format. Data read from a BDAT file is considered to be in internal format if FORMAT is OFF, and is read as ASCII characters if FORMAT is ON.

Serial access is available for both ASCII and BDAT files. Random access is available for BDAT files. The file pointer is important to both serial and random access. The file pointer is set to the beginning of the file when the file is opened by an ASSIGN. The file pointer always points to the next byte available for ENTER operations.

Random access uses the record number parameter to read items from a specific location in a file. The record specified must be before the end-of-file. The ENTER begins at the beginning of the specified record.

It is recommended that random and serial access to the same file not be mixed. Also, data should be entered into variables of the same type as those used to output it (e.g., string for string, REAL for REAL, etc.).

Devices As Source

An I/O path name or a device selector may be used to ENTER from a device. If a device selector is used, the default system attributes are used. If an I/O path name is used, the ASSIGN statement determines the attributes used. If multiple devices are specified in the ASSIGN, the ENTER sets the first device to be talker, and the rest to be listeners.

If FORMAT ON is the current attribute, the items are read as ASCII. If FORMAT OFF is the current attribute, items are read from the device in the computer's internal format. Two bytes are read for each INTEGER, eight bytes for each REAL. Each string entered consists of a four-byte header containing the length of the string, followed by the actual string characters. The string must contain an even number of characters; if the length is odd, an extra byte is entered to give alignment on the word boundary.

CRT As Source

If the device selector is 1, the ENTER is from the CRT. The ENTER reads characters from the CRT, beginning at the current print position (print position may be modified by using TABXY in a PRINT statement). The print position is updated as the ENTER progresses. After the last non-blank character in each line, a line-feed is sent with a simulated "EOI." After the last line is read, the print position is off the screen. If the print position is off screen when an ENTER is started, the off-screen text is first scrolled into the last line of the display.

Keyboard As Source

ENTER from device selector 2 may be used to read the keyboard. An entry can be terminated by pressing ENTER, EXECUTE, RETURN, CONTINUE, or STEP. Using ENTER, EXECUTE, RETURN, or STEP causes a CR/LF to be appended to the entry. The CONTINUE key adds no characters to the entry and does not terminate the

ENTER statement. If an ENTER is stepped into, it may be stepped out of, even if the CONTINUE key is pressed. An HP-IB EOI may be simulated by pressing CTRL E before the character to be sent, if this feature has been enabled by an appropriate CONTROL statement to the keyboard.

Strings As Source

If a string name is used as the source, the string is treated similarly to a file. However, there is no file pointer, each ENTER begins at the beginning of the string and reads serially within the string.

Buffers As Source (Requires TRANS)

When entering from an I/O path assigned to a buffer, data is removed from the buffer beginning at the location indicated by the buffer's empty pointer. As data is received, the current number-of-bytes register and empty pointer are adjusted accordingly. Encountering the fill pointer (buffer empty) produces an error unless a continuous inbound TRANSFER is filling the buffer. In this case, the ENTER will wait until more data is placed in the buffer.

Since devices are logically bound to buffers, an ENTER statement cannot intercept data while it is travelling between the device and the buffer. If an I/O path is currently being used in an outbound TRANSFER, and an ENTER statement uses it as a source, execution of the ENTER is deferred until the completion of the TRANSFER. An ENTER can be concurrent with an inbound TRANSFER only if the source is the I/O path assigned to the buffer.

An ENTER from a string variable that is also a buffer will not update the buffer's pointers and may return meaningless data.

ENTER With USING

When the computer executes an ENTER USING statement, it reads the image specifier, acting on each field specifier (field specifiers are separated from each other by commas) as it is encountered. If no variable is required for the field specifier, the field specifier is acted upon without referencing the enter items. When the field specifier references a variable, bytes are entered and used to create a value for the next item in the enter list. Each element in an array is considered a separate item.

The processing of image specifiers stops when a specifier is encountered that has no matching enter item. If the image specifiers are exhausted before the enter items, the specifiers are reused, starting at the beginning of the specifier list.

Entry into a string variable always terminates when the
dimensioned length of the string is reached. If more variables
remain in the enter list when this happens, the next character
received is associated with the next item in the list.

When USING is specified, all data is interpreted as ASCII
characters. FORMAT ON is always assumed with USING, regardless
of any attempt to specify FORMAT OFF.

Effects of the image specifiers on the ENTER statement are shown
in the following table.

Image Specifier	Meaning
K	Freefield Entry Numeric: Entered characters are sent to the number builder. Leading non-numeric characters are ignored. All blanks are ignored. Trailing non-numeric characters and characters sent with EOI true are delimiters. Numeric characters include digits, decimal point, +, −, e, and E when their order is meaningful. String: Entered characters are placed in the string. Carriage-return not immediately followed by line-feed is entered into the string. Entry to a string terminates on CR/LF, LF, a character received with EOI true, or when the dimensioned length of the string is reached.
−K	Like K except that LF is entered into a string, and thus CR/LF and LF do not terminate the entry.
H	Like K except that the European number format is used. Thus, a comma is the radix indicator and a period is a terminator for a numeric item (requires IO).
−H	Same as −K for strings; same as H for numbers (requires IO).
S	Same action as D.
M	Same action as D.
D	Demands a character. Non-numerics are accepted to fill the character count. Blanks are ignored; other non-numerics are delimiters.

Image Specifier	Meaning
Z	Same action as D.
*	Same action as D (requires IO).
.	Same action as D.
R	Like D, R demands a character. When R is used in a numeric image, it directs the number builder to use the European number format. Thus, a comma is the radix indicator and a period is a terminator for the numeric item (requires IO).
E	Same action as 4D.
ESZ	Same action as 3D.
ESZZ	Same action as 4D.
ESZZZ	Same action as 5D.
A	Demands a string character. Any character received is placed in the string.
X	Skips a character.
literal	Skips one character for each character in the literal.
B	Demands one byte. The byte becomes a numeric quantity.
W	Demands one 16-bit word, which is interpreted as a 16-bit, two's complement integer. If either an I/O path name with the BYTE attribute or a device selector is used to access an 8-bit interface, two bytes will be entered; the most-significant byte is entered first. If an I/O path name with the BYTE attribute is used to access a 16-bit interface, the BYTE attribute is overridden and one word is entered in a single operation. If an I/O path name with the WORD attribute is used to access a 16-bit interface, one byte is entered and ignored when necessary to achieve alignment on a word boundary. If the source is a file, string variable, or buffer, the WORD attribute is ignored and all data are entered as bytes; however, one byte will be entered and ignored when necessary to achieve alignment on a word boundary.

Image Specifier	Meaning
Y	Like W except that pad bytes are never entered to achieve word alignment. If an I/O path name with the BYTE is used to access a 16-bit interface, the BYTE attribute is not overridden (as with W specifier above) (requires IO).
#	Statement is terminated when the last ENTER item is terminated. EOI and line-feed are item terminators, and early termination is not allowed.
%	Like # except that an END indication (such as EOI or end-of-file) is an immediate statement terminator. Otherwise, no statement terminator is required. Early termination is allowed if the current item is satisfied.
+	Specifies that an END indication is required with the last character of the last item to terminate the ENTER statement. Line-feeds are not statement terminators. Line-feed is an item terminator unless that function is suppressed by −K or −H (requires IO).
−	Specifies that a line-feed terminator is required as the last character of the last item to terminate the statement. EOI is ignored, and other END indications, such as EOF or end-of-data, cause an error if encountered before the line-feed (requires IO).
/	Demands a new field; skips all characters to the next line-feed. EOI is ignored.
L	Ignored for ENTER.
@	Ignored for ENTER.

Enter Statement Termination   A simple ENTER statement (one without USING) expects to give values to all the variables in the enter list and then receive a statement terminator. A statement terminator is an EOI, a line-feed received at the end of the last variable (or within 256 characters after the end of the last variable), an end-of-data indication, or an end-of-file. If a statement terminator is received before all the variables are satisfied, or no terminator is received within 256 bytes after the last variable is satisfied, an error occurs. The terminator requirements can be altered by using images.

An ENTER statement with USING, but without a % or # image specifier, is different from a simple ENTER in one respect. EOI is not treated as a statement terminator unless it occurs on or after the last variable. Thus, EOI is treated like a line-feed and can be used to terminate entry into each variable.

An ENTER statement with USING that specifies a # image requires no statement terminator other than a satisfied enter list. EOI and line-feed end the entry into individual variables. The ENTER statement terminates when the variable list has been satisfied.

An ENTER statement with USING that specifies a % image allows EOI as a statement terminator. Like the # specifier, no special terminator is required. Unlike the # specifier, if an EOI is received, it is treated as an immediate statement terminator. If the EOI occurs at a normal boundary between items, the ENTER statement terminates without error and leaves the value of any remaining variable unchanged.

OUTPUT

Option Required	None
Keyboard Executable	Yes
Programmable	Yes
In an IF...THEN...	Yes

This statement outputs items to the specified destination.

Expanded diagram:

Format Statement Descriptions
C-12

literal form of image specifier:

Item	Description/Default	Range Restrictions
I/O path name	name assigned to a device, devices, mass storage file, or buffer	any valid name
record number	numeric expression, rounded to an integer	1 through $2^{31}-1$
device selector	numeric expression, rounded to an integer	
destination string name	name of a string variable	any valid name
subscript	numeric expression, rounded to an integer	$-32{,}767$ through $+32{,}767$
image line number	integer constant identifying an IMAGE statement	1 through 32,766
image line label	name identifying an IMAGE statement	any valid name
image specifier	string expression	(see drawing)
numeric array name	name of a numeric array	any valid name
string array name	name of a string array	any valid name
image specifier list	literal	(see next drawing)
repeat factor	integer constant	1 through 32,767
literal	string constant composed of characters from the keyboard, including those generated using the ANY CHAR key	quote mark not allowed

image specifier list

Shaded items
require IO

Radix specifier cannot
be used without a
digit specifier

Format Statement Descriptions

Example Statements

```
OUTPUT  701;Number,String$
OUTPUT  @File;Array(*),END
OUTPUT  @Rand, 5 USING Fmt1;Item(5)
OUTPUT  12 USING "#,6A";B$[2;6]
OUTPUT  @Printer;Rank;Id;Names$
```

Semantics

Standard Numeric Format The standard numeric format depends on the value of the number being displayed. If the absolute value of the number is greater than or equal to $1E-4$ and less than $1E+6$, it is rounded to 12 digits and displayed in floating point notation.

If it is not within these limits, it is displayed in scientific notation. The standard numeric format is used unless USING is selected, and may be specified by using K in an image specifier.

Arrays Entire arrays may be output by using the asterisk specifier. Each element in an array is treated as an item by the OUTPUT statement, as if the items were listed separately, separated by the punctuation following the array specifier. If no punctuation follows the array specifier, a comma is assumed. The array is output in row-major order (right-most subscript varies fastest.)

Files As Destination If an I/O path has been assigned to a file, the file may be written to with OUTPUT statements. The file must be an ASCII or BDAT file. The attributes specified in the ASSIGN statement are used if the file is a BDAT file.

Serial access is available for both ASCII and BDAT files. Random access is available for BDAT files. The end-of-file marker (EOF) and the file pointer are important to both serial and random access. The file pointer is set to the beginning of the file when the file is opened by an ASSIGN. The file pointer always points to the next byte to be written by serial OUPUT operations. The EOF pointer is read from the media when the file is opened by an ASSIGN. On a newly-created file, EOF is set to the beginning of the file. After each

OUTPUT operation, the EOF is updated **internally** to the maximum of the file pointer or the previous EOF value. The EOF pointer is updated on the **media** at the following times:
- When the current end-of-file changes.
- When END is specified in an OUTPUT statement directed to the file.
- When a CONTROL statement directed to the I/O path name changes the position of the EOF.

Random access uses the record number parameter to write items to a specific location in a file. The OUPUT begins at the start of the specified record and must fit into one record. The record specified cannot be beyond the record containing the EOF, if EOF is at the first byte of a record. The record specified can be one record beyond the record containing the EOF, if EOF is not at the first byte or record. Random access is always allowed to records preceding the EOF record. If you wish to write randomly to a newly-created file, either use a CONTROL statement to position the EOF in the last record, or write some "dummy" data into every record.

When data is written to an ASCII file, each item is sent as an ASCII representation with a two-byte length header. Data sent to a BDAT file is sent in internal format if FORMAT is OFF, and is sent as ASCII characters if FORMAT is ON.

Devices As Destination An I/O path or a device selector may be used to direct OUTPUT to a device. If a device selector is used, the default system attributes are used. If an I/O path is used, the ASSIGN statement used to associate the I/O path with the device also determines the attributes used. If multiple listeners were specified in the ASSIGN, the OUPUT is directed to all of them. If FORMAT ON is the current attribute, the items are sent in ASCII. Items followed by a semicolon are sent with nothing following them. Numeric items followed by a comma are sent with a comma following them. String items followed by a comma are sent with a CR/LF following them. If the last item in the OUPUT statement has no punctuation following it, the current end-of-line (EOL) sequence is sent after it. Trailing punctuation eliminates the automatic EOL.

If FORMAT OFF is the current attribute, items are sent to the device in the computer's internal format. Punctuation following items has no effect on the OUTPUT. Two bytes are sent for each INTEGER, eight bytes for each REAL. Each string output consists of a four string byte header containing the length of the string,

followed by the actual characters. If the number of characters is odd, an additional byte containing a blank is sent after the last character.

CRT As Destination If the device selector is 1, the OUTPUT is directed to the CRT. OUTPUT 1 and PRINT differ in their treatment of separators and print fields. The OUTPUT format is described under "Devices As Destination." See the PRINT keyword (page C-22) for a discussion of that format. OUTPUT 1 USING and PRINT USING to the CRT produce similar actions.

Keyboard As Destination Outputs to device selector 2 may be used to simulate keystrokes. ASCII characters can be sent directly (i.e., "hello"). Non-ASCII keys (such as EXECUTE) are simulated by a two-byte sequence.

When simulating keystrokes, unwanted characters (such as the EOL sequence) can be avoided with an image specifier (such as "#B" or "#,K"). See "OUTPUT With USING," page C-18.

Strings As Destination If a string is used for the destination, the string is treated similarly to a file. However, if there is no file pointer, each OUTPUT begins at the beginning of the string, and writes serially within the string.

Buffers As Destination (Requires TRANS) When the destination is an I/O path name assigned to a buffer, data is placed in the buffer beginning at the location indicated by the buffer's fill pointer. As data is sent, the current number-of-bytes register and fill pointer are adjusted accordingly. Encountering the empty pointer (buffer full) produces an error unless a continuous outbound TRANSFER is emptying the buffer. In this case, the OUTPUT will wait until there is more room in the buffer for data.

If an I/O path is currently being used in an inbound TRANSFER, and an OUTPUT statement uses it as a destination, execution of the OUTPUT is deferred until the completion of the TRANSFER. An OUTPUT can be concurrent with an outbound TRANSFER only if the destination is the I/O path assigned to the buffer.

An OUTPUT to a string variable that is also a buffer will not update the buffer's pointers and will probably corrupt the data in the buffer.

Using END With Devices The secondary keyword END may be specified following the last item in an OUTPUT statement. The result, when USING is not specified, is to suppress the EOL (End-of-Line) sequence that would

otherwise be output after the last byte of the last item. If a comma is used to separate the last item from the END keyword, the corresponding item terminator is output (CR/LF for string items or comma for numeric items).

With HP-IB interfaces, END specifies an EOI signal to be sent with the last data byte of the last item. However, if no data is sent from the last output item, EOI is not sent. With data communications interfaces, END specifies an end-of-data indication to be sent with the last byte of the last output item.

OUTPUT With USING
When the computer executes an OUTPUT USING statement, it reads the image specifier, acting on each field specifier (field specifiers are separated from each other by commas) as it is encountered. If nothing is required from the output items, the field specifier is acted upon without accessing the output list. When the field specifier requires characters, it accesses the next item considered a separate item.

The processing of image specifiers stops when a specifier is encountered that has no matching output item. If the image specifiers are exhausted before the output items, they are reused, starting at the beginning.

If a numeric item requires more decimal places to the left of the decimal point than are provided by the field specifier, an error is generated. A minus sign takes a digit place if M or S is not used, and can generate unexpected overflows of the image field. If the number contains more digits to the right of the decimal point than specified, it is rounded to fit the specifier.

If a string is longer than the field specifier, it is truncated, and the rightmost characters are lost. If it is shorter than the specifier, trailing blanks are used to fill out the field.

Effects of the image specifiers on the OUTPUT statement are shown in the following table.

Image Specifier	Meaning
K	Compact field. Outputs a number or string in standard form with no leading or trailing blanks.
−K	Same as K.
H	Similar to K except that the number is printed using the European number format (comma radix) (requires IO).
−H	Same as H (requires IO).
S	Outputs the number's sign (+ or −)
M	Outputs the number's sign if negative—a blank if positive.
D	Outputs one-digit character. A leading zero is replaced by a blank. If the number is negative and no sign image is specified, the minus sign will occupy a leading digit position. If a sign is printed, it will "float" to the left of the left-most digit.
Z	Same as D except that leading zeros are output.
*	Like D except that asterisks are printed instead of leading zeros (requires IO).
.	Outputs a decimal-point radix indicator.
R	Outputs a comma radix indicator (European radix) (requires IO).
E	Outputs an E, a sign, and a two-digit exponent.
ESZ	Outputs an E, a sign, and a one-digit exponent.
ESZZ	Same as E.
ESZZZ	Outputs an E, a sign, and a three-digit exponent.
A	Outputs a string character. Trailing blanks are output if the number of characters specified is greater than the number available in the corresponding string. If the image specifier is exhausted before the coresponding string, the remaining characters are ignored.

Format Statement Descriptions
C-19

Image Specifier	Meaning
X	Outputs a blank.
literal	Outputs the characters contained in the literal.
B	Outputs the characters represented by one byte of data. This is similar to the CHR$ function. The number is rounded to an INTEGER and the least-significant byte is sent. If the number is greater than 32,767, then 255 is used. If the number is less than −32,768, then 0 is used.
W	Outputs a 16-bit word as a two's complement integer. The corresponding numeric item is rounded to an INTEGER. If it is greater than 32,767, then 32,767 is sent. If it is less than −32,768, then −32,768 is sent. If either an I/O path name with the BYTE attribute or a device selector is used to access an 8-bit interface, two bytes will be output; the most-significant byte is sent first. If an I/O path name with the BYTE attribute is used to access a 16-bit interface, a null pad byte is output whenever necessary to achieve alignment on a word boundary. If the destination is a BDAT file, string variable, or buffer, the BYTE or WORD attribute is ignored and all data are sent as bytes; however, pad byte(s) will be output when necessary to achieve alignment on a word boundary. The pad character may be changed by using the CONVERT attribute.
Y	Like W except that no pad bytes are output to achieve word alignment. If an I/O path with the BYTE attribute is used to access a 16-bit interface, the BYTE attribute is not overridden (as with the W specifier above) (requires IO).
#	Suppresses the automatic output of the EOL (end-of-line) sequence following the last output item.
%	Ignored in OUTPUT images.
+	Changes the automatic EOL sequence that normally follows the last output item to a single carriage-return (requires IO).
−	Changes the automatic EOL sequence that normally follows the last output item to a single line-feed (requires IO).

Image Specifier	Meaning
/	Outputs a carriage-return and a line-feed.
L	Outputs the current EOL sequence. The default EOL characters are CR and LF. If the destination is an I/O path name with the WORD attribute, a pad byte may be sent after the EOL characters to achieve word alignment.
@	Outputs a form-feed.

END With OUTPUT...USING

Using the optional secondary keyword END in an OUTPUT...USING statement produces results which differ from those in an OUTPUT statement without USING. Instead of always suppressing the EOL sequence, the END keyword only suppresses the EOL sequence when no data is output from the last output item. Thus, the # image specifier generally controls the suppression of the otherwise automatic EOL sequence.

With HP-IB interfaces, END specifies an EOI signal to be sent with the last byte output. However, no EOI is sent if no data is sent from the last output item or the EOL sequence is suppressed. With data communications interfaces, END specifies an end-of-data indication to be sent at the same time an EOI would be sent on HP-IB interfaces.

PRINT

Option Required	None
Keyboard Executable	Yes
Programmable	Yes
In an IF...THEN...	Yes

This statement sends items to the PRINTER IS device.
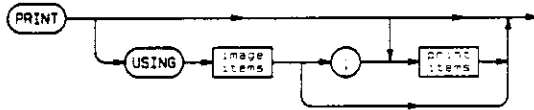
Expanded diagram

literal form of image specifier

image specifier list

S
M

D
Z
#

repeat factor

E
ESZ
ESZZ
ESZZZ

Radix specifier cannot
be used without a
digit specifier

Shaded items
require IO

.
R

A

X

/

L

@

repeat factor
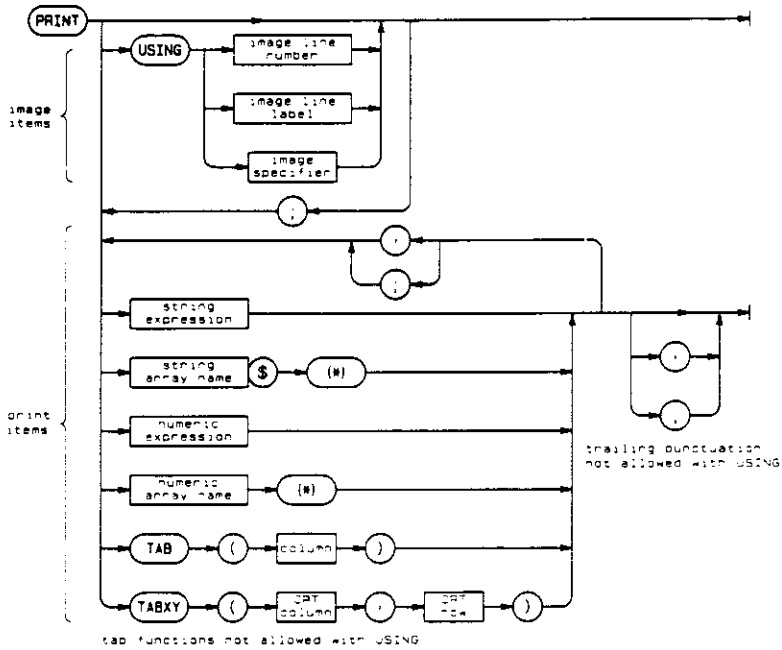
literal
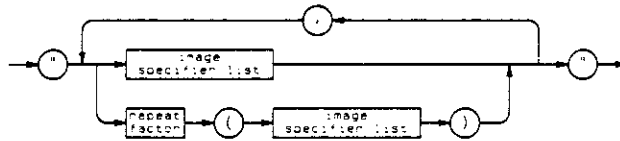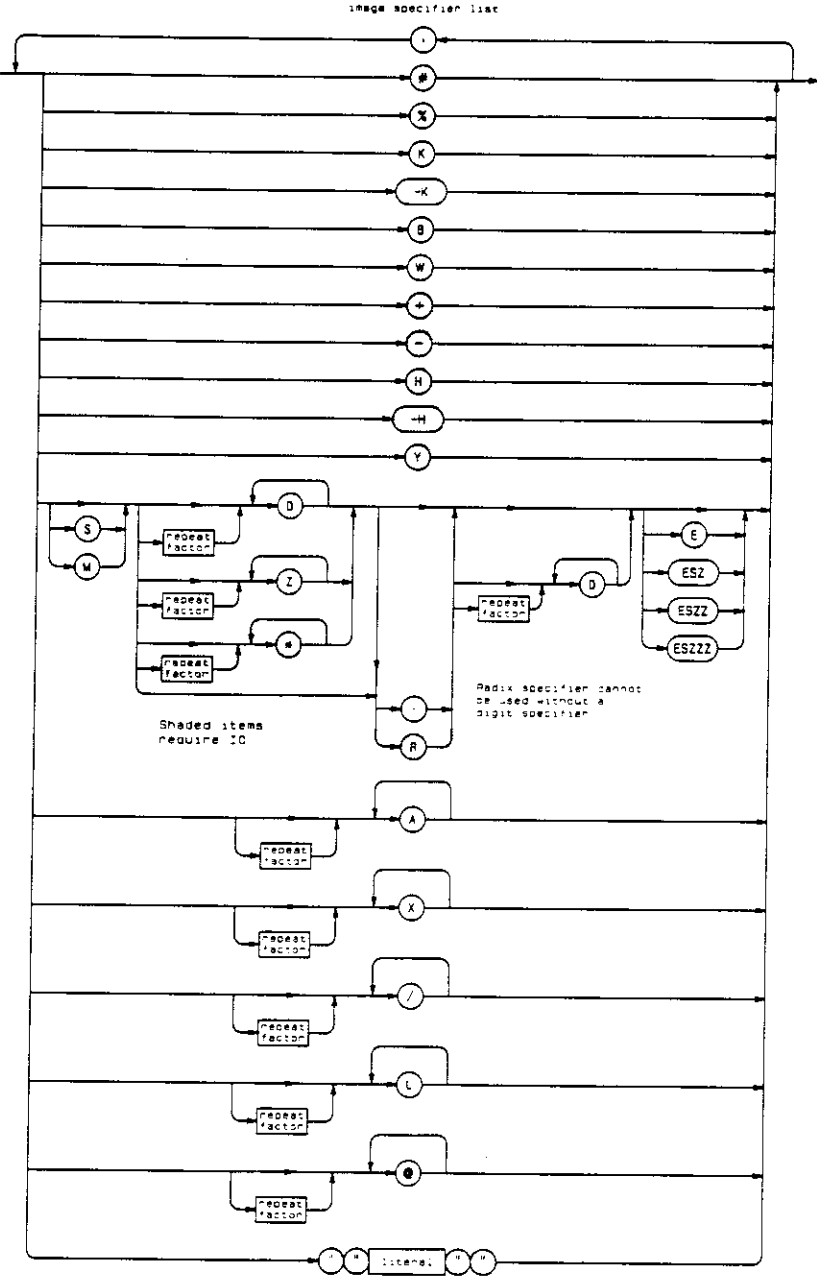
Item	Description/Default	Range Restrictions	Recommended Range
image line number	integer constant identifying an IMAGE statement	1 thru 32,766	—
image line label	name identifying an IMAGE statement	any valid name	—
image specifier	string expression	(see drawing)	—
numeric array name	name of a numeric array	any valid name	—
string array name	name of a string array	any valid name	—
column	numeric expression, rounded to an integer	−32,768 thru +32,767	device dependent
CRT column	numeric expression, rounded to an integer	0 thru 32,767	1 thru screen width
CRT row	numeric expression, rounded to an integer	0 thru 32,767	1 thru 18
image specifier list	literal	(see next drawing)	—
repeat factor	integer constant	1 thru 32,767	—
literal	string constant composed of characters from the keyboard, including those generated using the ANY CHAR key	quote mark not allowed	—

Format Statement Descriptions
C-24

Example	PRINT "LINE";Number
Statements	PRINT Array (*);
	PRINT TABXY(1,1),Header\$,TABXY(Col,3),Message\$
	PRINT USING "5Z,DD";Money
	PRINT USING Fmt3;Id,Item\$,Kilograms/2.2

Semantics

Standard Numeric Format The standard numeric format depends on the value of the number being displayed. If the absolute value of the number is greater than or equal to $1E-4$ and less than $1E+6$, it is rounded to 12 digits and displayed in floating point notation. If it is not within these limits, it is displayed in scientific notation. The standard numeric format is used unless USING is selected, and may be specified by using K in an image specifier.

Automatic End-of-Line Sequence After the print list is exhausted, an End-of-Line (EOL) sequence is sent to the PRINTER IS device, unless it is suppressed by trailing punctuation or a pound-sign (#) image specifier. The printer width for EOL sequences generation is set to the screen width (50, 80, or 128 characters) for CRTs and to 80 for external devices unless the WIDTH attribute of the PRINTER IS statement was specified. WIDTH is off for files. This "printer width exceeded" EOL is not suppressed by trailing punctuation, but can be suppressed by the use of an image specifier.

Control Codes Some ASCII control codes have a special effect in PRINT statements if the PRINTER IS device is the CRT (device selector = 1):

Character	Keystroke	Name	Action
CHR$(7)	CTRL-G	bell	Sounds the beeper.
CHR$(8)	CTRL-H	backspace	Moves the print position back one character.
CHR$(10)	CTRL-J	line-feed	Moves the print position down one line.
CHR$(12)	CTRL-L	form-feed	Prints two line-feeds, then advances the CRT buffer enough lines to place the next item at the top of the CRT.
CHR$(13)	CTRL-M	carriage-return	Moves the print position to column 1.

The effect of ASCII control codes on a printer is device dependent. See your printer manual to find which control codes are recognized by your printer and what effects they will have.

CRT Enhancements There are several character enhancements (such as inverse video and underlining) available on some CRTs. They are accessed through characters with decimal values above 127.

Arrays Entire arrays may be printed using the asterisk specifier. Each element in an array is treated as a separate item, as if the elements were all listed and separated by the punctuation following the array specifier. If no punctuation follows the array specifier, a comma is assumed. The array is printed in row-major order (right-most subscript varies fastest).

PRINT Fields If PRINT is used without USING, the punctuation following an item determines the width of the item's print field; a semicolon selects the compact field, and a comma selects the default print field. Any trailing punctuation will suppress the automatic EOL sequence, in addition to selecting the print field to be used for the print item preceding it.

The compact field is slightly different for numeric and string items. Numeric items are printed with one trailing blank. String items are printed with no leading or trailing blanks.

The default print field prints items with trailing blanks to fill to the beginning of the next ten-character field.

Numeric data is printed with one leading blank if the number is positive, or with a minus sign if the number is negative, whether in compact or default field.

TAB   The TAB function is used to position the next character to be printed on a line. In the TAB function, a column parameter less than one is treated as one. A column parameter greater than zero is subjected to the following formula:
TAB position = ((column − 1)MOD width) + 1; where "width" is 50 for the Model 226 CRT, 128 for Model 237, and 80 for all other devices. If the TAB position evaluates to a column number less than or equal to the number of characters printed since the last EOL sequence, then an EOL sequence is printed, followed by (TAB position − 1) blanks. If the TAB position evaluates to a column number greater than the number of characters printed since the last EOL, sufficient blanks are printed to move to the TAB position.

TABXY   The TABXY function provides X − Y character positioning on the CRT. It is ignored if a device other than the CRT is the PRINTER IS device. TABXY(1,1) specifies the upper left-hand corner of the CRT. If a negative value is provided for CRT row or CRT column, it is an error. Any number greater than the screen width for CRT column is treated as the screen width. Any number greater than 18 for CRT row is treated as 18; on a Model 237 this is extended to 41 rows. If 0 is provided for either parameter, the current value of that parameter remains unchanged.

PRINT
With USING   When the computer executes a PRINT USING statement, it reads the image specifier, acting on each field specifier (field specifiers are separated from each other by commas) as it is encountered. If nothing is required from the print items, the field specifier is acted upon without accessing the print list. When the field specifier requires characters, it accesses the next item in the print list, using the entire item. Each element in an array is considered a separate item.

The processing of image specifiers stops when a specifier is encountered that has no matching print item. If the image specifiers are exhausted before the print items, they are reused, starting at the beginning.

Format Statement Descriptions
C-27

If a numeric item requires more decimal places to the left of the decimal point than are provided by the field specifier, an error is generated. A minus sign takes a digit place if M or S is not used, and can generate unexpected overflows of the image field. If the number contains more digits to the right of the decimal point than specified, it is rounded to fit the specifier.

If a string is longer than the field specifier, it is truncated, and the rightmost characters are lost. If it is shorter than the specifier, trailing blanks are used to fill out the field.

Effects of the image specifiers on the PRINT statement are shown in the following table.

Image Specifier	Meaning
K	Compact field. Prints a number or string in standard form with no leading or trailing blanks.
−K	Same as K.
H	Similar to K except that the number is printed using the European number format (comma radix) (requires IO)
−H	Same as H (requires IO).
S	Prints the number's sign (+ or −)
M	Prints the number's sign if negative—a blank if positive.
D	Prints one-digit character. A leading zero is replaced by a blank. If the number is negative and no sign image is specified, the minus sign will occupy a leading digit position. If a sign is printed, it will "float" to the left of the left-most digit.
Z	Same as D except that leading zeros are printed.
*	Like Z except that asterisks are printed instead of leading zeros (requires IO).
.	Prints a decimal-point radix indicator.

Image Specifier	Meaning
R	Prints a European comma radix indicator (requires IO).
E	Prints an E, a sign, and a two-digit exponent.
ESZ	Prints an E, a sign, and a one-digit exponent.
ESZZ	Same as E.
ESZZZ	Prints an E, a sign, and a three-digit exponent.
A	Prints a string character. Trailing blanks are output if the number of characters specified is greater than the number available in the corresponding string. If the image specifier is exhausted before the corresponding string, the remaining characters are ignored.
X	Prints a blank.
literal	Prints the characters contained in the literal.
B	Prints the characters represented by one byte of data. This is similar to the CHR$ function. The number is rounded to an INTEGER and the least-significant byte is sent. If the number is greater than 32,767, then 255 is used. If the number is less than −32,768, then 0 is used.
W	Prints two characters represented by the two bytes in a 16-bit, two's complement integer word. If it is greater than 32,767, then 32,767 is used. If it is less than −32,768, then −32,768 is used. On an 8-bit interface, the most-significant byte is sent first. On a 16-bit interface, the two bytes are sent as one word in a single operation.
Y	Same as W (requires IO).
#	Suppresses the automatic output of the EOL (end-of-line) sequence following the last print item.
%	Ignored in PRINT image.
+	Changes the automatic EOL sequence that normally follows the last print item to a single carriage-return (requires IO).

Image Specifier	Meaning
−	Changes the automatic EOL sequence that normally follows the last print item to a single line-feed (requires IO).
/	Sends a carriage-return and a line-feed to the PRINTER IS device.
L	Sends the current EOL sequence to the PRINTER IS device. The default EOL characters are CR and LF. If the destination is an I/O path name with the WORD attribute, a pad byte may be sent after the EOL characters to achieve word alignment.
@	Sends a form-feed to the PRINTER IS device.

Index

E

F

G

H

N

O

P

T

U

V

Scans
By
Outsource-Options
(NR1DX)

Outsource-Options
Suite 2M
187 Depot Road,
Hollis, NH 03049

nr1dx@arrl.net or Henderson@outsource-options.com

"High resolution scans of obsolete technical manuals"

If you purchased this manual from us (typically through our Ebay name of NR1DX) thank you very much. If you received this from a well-meaning "friend" for free we would appreciate your treating this much like you would "share ware". By that we mean a donation of at least $5-10 per manual is appreciated in recognition of the time, energy and quality of effort that went into preserving this manual. Donations via PayPal go to: nr1dx@arrl.net or can be mailed to us the address above. This is not a hobby for us but a semi-retirement business used to augment our diminished income in the face of the current degraded world economy.

Typically the scans in our manuals are done as follows;
1) Typed text pages are typically scanned in black and white at 300 dpi.
2) Photo pages are typically scanned in gray scale mode at 600 dpi
3) Schematic diagram pages are typically scanned in black and white at 600 dpi unless the original manual had colored high lighting (as is the case for some 70's vintage Tektronix manuals).

Original pages, which exceed the 8.5 x 11 format, are scanned with enough overlap so that they can be printed and pieced. This is done for primarily for two reasons, the time involved in post processing these to their original size is not trivial and most people do not have the ability to print 11 X 17 pages anyway.

If your looking for a quality scanned technical manual in PDF format please drop us a line at nr1dx@arrl.net. and we will be happy to email you a current list of the manuals we have available. If you don't see the manual you need on the list drop us a line anyway we may still be able to point you to other sources. If you have an existing manual you would like scanned please write for details, This can often be done very reasonably in consideration for adding your manual to our library.

Thanks

Dave Henderson, NR1DX

Dave Henderson
Outsource-Options