Programmer Manual

Tektronix

SCD1000 & 5000
Transient Waveform Recorders

070-7315-02

This document applies for firmware version V1.80.

Please check for change information at the rear of this manual.

First Printing: September 1992
Revised Printing: November 1992

Instrument Serial Numbers

Each instrument manufactured by Tektronix has a serial number on a panel insert or tag, or stamped on the chassis. The first letter in the serial number designates the country of manufacture. The last five digits of the serial number are assigned sequentially and are unique to each instrument. Those manufactured in the United States have six unique digits. The country of manufacture is identified as follows:

B010000	Tektronix, Inc., Beaverton, Oregon, USA
E200000	Tektronix United Kingdom, Ltd., London
J300000	Sony/Tektronix, Japan
H700000	Tektronix Holland, NV, Heerenveen, The Netherlands

Instruments manufactured for Tektronix by external vendors outside the United States are assigned a two digit alpha code to identify the country of manufacture (e.g., JP for Japan, HK for Hong Kong, IL for Israel, etc.).

Tektronix, Inc., P.O. Box 500, Beaverton, OR 97077

Printed in U.S.A.

WARRANTY

Tektronix warrants that this product will be free from defects in materials and workmanship for a period of one (1) year from the date of shipment. If any such product proves defective during this warranty period, Tektronix, at its option, either will repair the defective product without charge for parts and labor, or will provide a replacement in exchange for the defective product.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period and make suitable arrangements for the performance of service. Customer shall be responsible for packaging and shipping the defective product to the service center designated by Tektronix, with shipping charges prepaid. Tektronix shall pay for the return of the product to Customer if the shipment is to a location within the country in which the Tektronix service center is located. Customer shall be responsible for paying all shipping charges, duties, taxes, and any other charges for products returned to any other locations.

This warranty shall not apply to any defect, failure or damage caused by improper use or improper or inadequate maintenance and care. Tektronix shall not be obligated to furnish service under this warranty a) to repair damage resulting from attempts by personnel other than Tektronix representatives to install, repair or service the product; b) to repair damage resulting from improper use or connection to incompatible equipment; or c) to service a product that has been modified or integrated with other products when the effect of such modification or integration increases the time or difficulty of servicing the product.

THIS WARRANTY IS GIVEN BY TEKTRONIX WITH RESPECT TO THIS PRODUCT IN LIEU OF ANY OTHER WARRANTIES, EXPRESSED OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPAIR OR REPLACE DEFECTIVE PRODUCTS IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Contents

Introduction

GPIB Introduction

GPIB Command Set

Status and Events

Program Examples

Alternate Interfaces

ASCII & GPIB Code Chart

Change Information

List of Figures

List of Tables

Introduction

Overview

The SCD1000 and SCD5000 are high-speed, transient digitizers based on CRT scan conversion.

Both SCD digitizers have an input impedance of 50 Ω with programmable input settings. Table 1-1 lists the features of each system.

Table 1-1: SCD1000/SCD5000 Characteristics

Characteristics	SCD1000	SCD5000
Impedance	50 Ω	50 Ω
Input BW	DC to 1 GHz	DC to 4.5 GHz DC to 3.0 GHz with Option 01
# Input Channels	2	1
Input Range	10 mV to 10 V	5 V 10 V with Option 01
Input Offset	±2.5 × full-scale range	±0.8 × full-scale range
Input Coupling	AC,DC,OFF	DC Only
Trigger Source	Any input channel external input	External or internal calibrator signal (Internal trigger pickoff with Option 01)
Trigger Coupling	AC or DC (Internal only)	AC Only DC Only with Option 01

Instrument Control

Instrument functions of the SCD are controlled through either the IEEE–488.1 interface or through the front panel Display Unit (DU). When the interface is used, an IEEE–488 controller programs the SCD and receives information through a set of SCD commands. When the display unit is used, front panel controls provide:

- instrument set up
- display of waveforms
- waveform measurements
- output to a hard copy unit

Input Channels

The SCD1000 includes two input channels. The SCD5000 has only one input channel. In the SCD1000, different vertical modes allow acquisition from either of the channels (CHA or CHB) or allows the algebraic addition of the channels (ADD). The number of channels used for acquisition affects other parameters such as the data statement.

Other programmable input parameters include full-scale range (SCD1000 only), offset, coupling, and signal inversion (SCD1000 only).

In the SCD1000, the input range for each channel can be set from 100 mV to 10 V full scale. Programmable offset values range from 250 mV to 25 V. SCD5000 input range is fixed at 5 volts (10 volts with Option 01).

In the SCD1000, signals can be AC or DC coupled or disconnected from the input. In the SCD5000, the input signal is DC coupled. Input impedance is 50 Ω in both instruments.

Acquisition Sequence and Acquisition Process

An acquisition sequence starts when the digitizer recognizes a trigger event defined by the trigger parameters or when auto-triggering is initiated. After recognizing the trigger event, the SCD writes the event onto the CRT target. The SCD reads the target and digitizes the data on the target, storing the data in a data record. An acquisition sequence is finished when the record is filled.

An acquisition process is the filling of all required records. If more than one record must be filled to complete an acquisition process, records are consecutively filled from the specified start record through the specified number of records set by the acquisition system commands, or to the maximum available records.

Acquisition System

The acquisition system controls the sweep mode, time window, acquisition mode, record length, and acquisition state.

Time Window — The time window programs the acquisition duration. Time window settings are from 5 ns to 100 µs.

Acquisition Modes — The acquisition mode programs the number of records acquired during an acquisition process. NORMAL acquisition mode always fills only the programmed start record per acquisition process. Depending on trigger parameters, subsequent acquisitions may occur, but for each acquisition process, only the programmed start record is filled. In ADVANCE mode, the digitizer fills a specified number of records to complete an acquisition process. Acquisition processes may be repeated due to trigger settings, but each acquisition process fills only the number of specified records. In ADVANCE mode, each record is stamped with a time identifying when the acquisition began. AVERAGE acquisition mode allows 1 to

1024 averages to be performed to increase the signal-to-noise ratio.The maximum number of available records for acquisition is 16. Records 1, 2, 3, and 4 use non-volatile memory for storage and will retain data across power downs.

Record Length — Record length can be programmed to 256, 512, or 1024 data points.

Acquisition State — The acquisition state controls the starting and stopping of the acquisition process. RUN and STOP immediately affect the acquisition process. RUN continuously acquires data. HOLDNXT completes one *acquisition process* before stopping the digitizer. The completion of the current acquisition process depends on the recognition of enough trigger events and may include the filling of one or more records depending on the acquisition mode (NORMAL or ADVANCE).

Triggering

The trigger system defines the parameters of the trigger event. The trigger event is defined by its source, level, and slope. Level can be defined in either volts or percent.The position of the time window relative to the trigger event can be set using a trigger delay setting.

The SCD can also be triggered from the **Manual Trig** button on the Display Unit and from the IEEE–488 interface using the MTRIG command.

Mode — AUTO trigger mode automatically triggers the digitizer approximately 360 ms after the start of an acquisition sequence, if a trigger event has not already occurred. NORMAL trigger mode allows the digitizer to trigger only with the recognition of a trigger event as defined by the trigger parameters.

Source — In the SCD1000, the trigger source can be from one of the channels, the sum of the channels, or from the external trigger input on the front panel of the SCD. In the SCD5000, the source can be either the external input connector or the internal time calibration source. In the SCD5000 with Option 01 installed, the trigger source can be either the external input connector, or the internal trigger pickoff signal.

Level — With an internal trigger source, the trigger level can be set in the range of ±Vertical Range (CHA, CHB, or Add (AC) SCD1000), ±(Vertical Range / 2) (CHA, CHB, or Add (DC) SCD1000), or ±5 V (SCD5000 with Option 01). Level for an internal source can be specified in either percent of the full-scale range or voltage.

Although the trigger level can be specified in volts, the trigger level is internally expressed as a percentage of the full-scale range plus offset (Level=Trig% × Range + Offset). This is done so that once the level is set, changing range and offset does not affect the relative trigger level.

Specifying the level over the maximum allowable value causes the SCD to coerce the level to the maximum allowable level. With an external trigger source, the trigger must be specified as a voltage level in the range of ±1.0 V (SCD1000) or ±0.5 V (SCD5000).

Slope — Trigger slope can be positive (PLUS) or negative (MINUS).

Coupling — In the SCD1000, trigger coupling can be AC or DC. DC coupling passes all components of the signal to the trigger circuits. AC coupling blocks all DC components and attenuates frequencies below 2 kHz. In the SCD5000, the trigger signal is AC coupled only.

Delay — The time window can be delayed relative to the trigger event up to five times the length of the time window. See Figure 1-1. Trigger delay can be expressed in percentage of the record or seconds.

Arm — External arming allows an externally applied signal to enable trigger recognition when the acquisition state is Hold Next or Running. The arming signal is applied to the rear panel Arm In connector. To enable, select External Arm from the Trigger menu or issue the appropriate GPIB command. A trigger will not be recognized until the arming signal (ground or TTL low) is received.

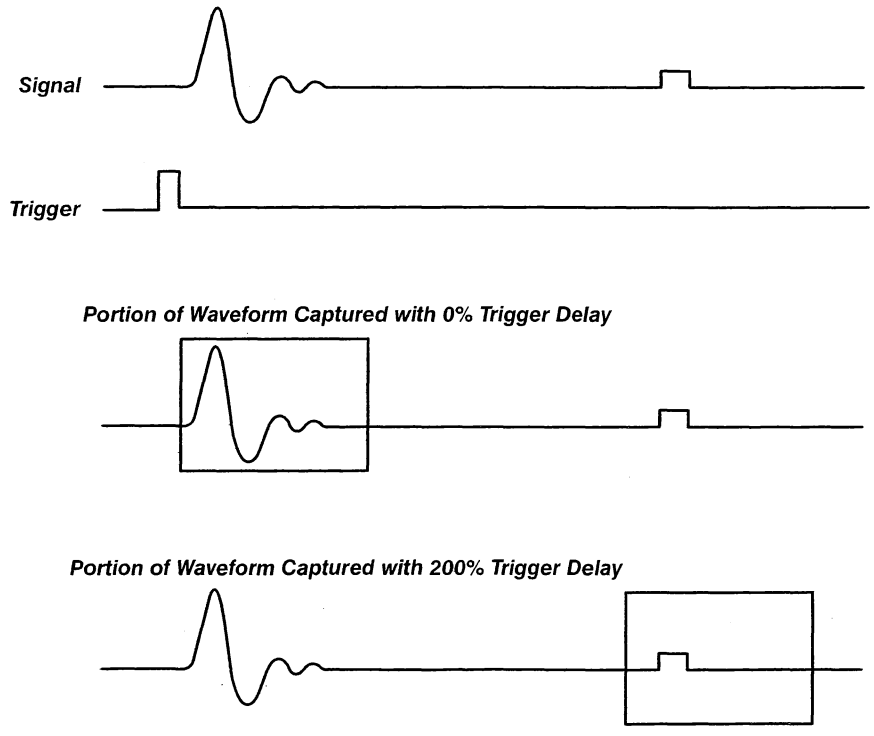


Figure 1-1: Trigger Delay

Internal Calibration & Diagnostics

An internal calibration function provides:

- vertical channel amplitude and offset adjustments
- input impedance bias current adjustment (SCD1000)
- trigger level and delay adjustments
- window size adjustment
- CRT adjustments
- CRT geometry correction

Internal calibration, which is only executed at user request, is initiated from the Display Unit or over the GPIB using the CALIBRATE commands.

Self-tests can be run for the acquisition and processor subsystems, as well as for the front panel. TEST commands allow entire subsystems or portions of a subsystem to be tested one time or several times (LOOP). The test results can be displayed as PASS/FAIL or PASS/FAIL and include a descriptive string. See the Service Manual for a more detailed description of calibration and diagnostics.

Factory Settings

An initialization function returns the digitizer's settings to *factory settings* stored in ROM. These factory settings cannot be changed, but are useful to place the instrument in a known state. The settings (see Table 3-8) are a good starting point to begin instrument set-up.

As the SCD is used, all instrument settings are saved in non-volatile RAM at power-down so that the digitizer powers up with the same settings that were selected when it was turned off.

Initialization to factory settings can be limited to just GPIB-related functions or just instrument functions, or both the GPIB and instrument functions can be reset to their factory settings. See the INIT command in Table 3-8.

Display Unit

The Display Unit (DU) is a control and display device. In addition to a high resolution, 640 $\times$ 400 pixel Liquid Crystal Display (LCD), the DU contains dedicated control keys, programmable soft keys, and a variable knob.

Depending on the display mode (waveform or menu), the LCD panel displays either waveforms and status information or waveforms and SCD menus. Programmable soft keys change functions according to menu labels to allow control of instrument settings, display modes, cursor positioning, and other functions. The variable knob allows easy adjustment of numeric values of functions.

The Display Unit can operate simultaneously with the GPIB interface functions.

Menu System

Figure 1-2 shows the DU displaying menus. The DU displays three types of menus: mode menus, function menus, and an auxiliary menu. Mode menus are displayed along the bottom of the screen and allow selection of function menus for different systems of the digitizer, such as Trigger, Vertical, Cursors, etc. Function menus appear along the left side of the display and allow changing values for each of the parameters associated with a system, such as trigger system Level, Position, Coupling, etc. The auxiliary menu appears at the right side of the display and labels the associated soft keys for various functions. User-defined key labels and the variable knob's last setting are part of the auxiliary menu.

Button presses can be emulated or queried over the GPIB using the ABSTOUCH command. See Table 3-8 for more information.

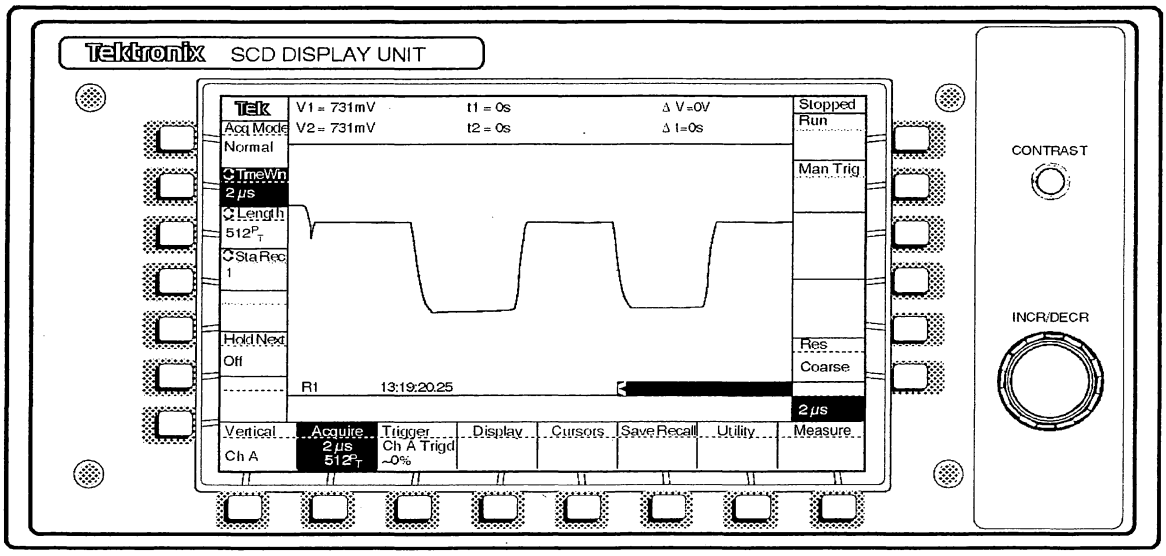


Figure 1-2: Display Unit Showing Menus

Window System

Through soft key selections, the LCD panel can be divided into one, two, or four separate display windows. (This is not affected by, nor does it affect, the vertical mode setting.) Any record from any selected channel can be displayed in any window. Figure 1-3 is an example of a four-window display with waveforms.

When a waveform is displayed, each window contains the following information:

- the digitized waveform
- the record number
- cursors (if selected)
- a ground potential indicator (if in range)
- a time stamp indicating the time the acquisition began
- a reference bar, indicating the relative portion of the record that can be seen in the display

The status information displayed next to each window (when no menu is selected) includes:

- channel number from which the waveform was acquired
- vertical full-scale range setting
- vertical offset setting
- vertical expansion (zoom) setting

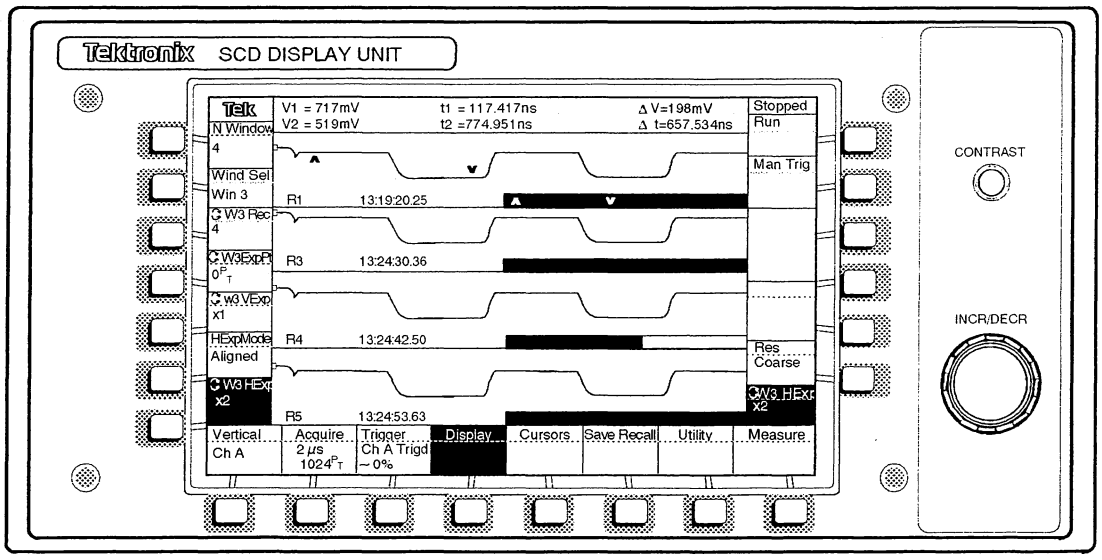


Figure 1-3: Four Window Display

Vertical & Horizontal Expansion (Zoom)

Vertical and horizontal expansion functions display a selectable portion of the waveform in the entire window. Expansion allows individual data points of a record to be seen and increases vertical visual resolution to see discrete digitized levels.

Windows can be individually expanded or aligned with a selected window.

A 1024 point record can be horizontally expanded by a factor of ×2. Records with 256 or 512 data points are shown at ×1 only. The display window's horizontal axis is 512 data points. At ×1, the entire record is displayed in the window.

Vertical expansion factors are ×1, ×2, ×4, ×8, and ×16. At ×1, the entire vertical range of the acquired waveform is displayed. The acquired vertical resolution of a waveform is 2048 levels (11 bits). However, the displayed vertical resolution of a waveform depends on the number of windows in the display: 256 points for one window, 128 points/window for two windows, 64 points/window for four windows.

Cursor Measurements

The SCD provides two cursors that can be placed in any one or two windows. Using the cursors, absolute and differential measurements of voltage and time can be made (frequency measurements are obtained by inverting the time measurements). Absolute measurements are referenced to ground for voltage and the left edge of the window for time. Differential measurements (Δt, Δv, and Δf) are made between the two cursors.

NOTE

Other messages can be displayed in the cursor readout zone. These messages will overwrite cursor information. The cursor information is automatically updated when the user adjusts the cursor position with the variable knob.

Cursor response is improved when Acquire state is set to stopped.

Debug Mode

A debug mode displays IEEE–488.1 bus traffic on the Display Unit. Only bus traffic for the digitizer is displayed; other device traffic is not displayed. Besides display of bus traffic, debug features include:

- display of error codes where they occur in the bus transactions
- scrolling of the status area
- display of control and other special characters

NOTE

Turn the cursors off when using Debug mode. If the cursors are turned on, the cursor results will overwrite the debug information. This information can be retrieved, however, by using the Recall Stat utility function.

SCD Setup Switches

This section describes the parameters to consider when setting up the SCD. The following information does not include pinouts and other IEEE–488.1 bus data. For more information on IEEE–488.1, see Section 2.

Setup switches are located on the SCD's rear panel (Figure 1-4) select the digitizer's IEEE–488.1 (GPIB) bus and instrument settings.

GPIB Address

GPIB switches 4 through 8 set the bus address from 0 to 30. Each switch is a binary value: 1, 2, 4, 8, and 16.

NOTE

Each instrument on the bus must have a unique address.

GPIB Message Terminator

GPIB switch 3 selects the message terminator as either EOI or EOI/LF. Section 2 describes the message terminator.

The factory setting is EOI.

NOTE

The SCD will also accept LF only as a message terminator when the switch is set to EOI/LF; this function is explained in more detail in Section 2. If the controller accepts EOI as the terminator, the switch should be set to EOI.

Option

GPIB switch 2 is used to set the instrument to either talk/listen or talk-only GPIB mode. The talk-only mode is to be used in conjunction with an HPGL plotter. It is also used in conjunction with MPU board switches 2 and 3 in order to select from optional modes which are available (see Table 1-2). Note that switch 2 is labelled Option on the rear panel of the instrument.

Dump Continuous Mode (Option 2F) — The instrument can be placed in Dump Continuous Mode by setting instrument switch 2 to ON, MPU switch 2 to ON, and MPU switch 3 to OFF. Setting instrument switch 2 to OFF will remove the instrument from this mode.

Auto Record Mode — Sets the instrument to "repeat mode" at power up (waveforms are acquired and transmitted immediately). The instrument may be placed in Auto Record Mode by setting instrument switch 2 to ON, MPU switch 2 to OFF, and MPU switch 3 to ON. Setting instrument switch 2 to OFF will remove the instrument from this mode. Other combinations of MPU switches 2 and 3 are available, but are reserved for future use.

NOTE

Refer MPU board switch changes to qualified service personnel.

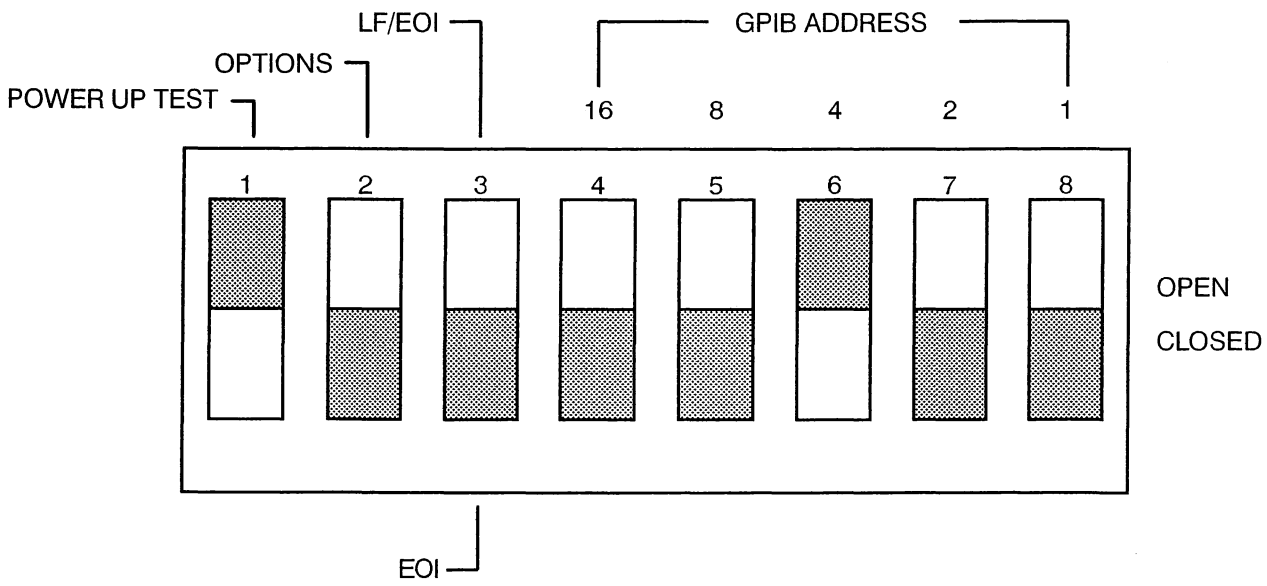


Figure 1-4: SCD1000/SCD5000 GPIB and Instrument Switches

Table 1-2: Optional Mode Switching

	Switch and Location		
Function	GPIB switch 2 on rear panel	Switch 2 on MPU board	Switch 3 on MPU board
Dump Continuous mode (Option 2F)	Open	Open	Closed
Auto Record mode	Open	Closed	Open
GPIB talk-only mode	*Open	*Closed	*Closed

*** Default for standard instrument**

Power Up Test Bypass

Instrument switch 1 determines whether or not the digitizer performs a self-test upon power-up. When OFF, the digitizer bypasses the self-test (the SAFEGUARD PUPTST command is ignored). When ON, the SAFEGUARD PUPTST command determines whether or not the digitizer performs a self-test upon power up. See Table 3-11.

Factory setting is ON.

GPIB Introduction

This section introduces IEEE–488.1 programming concepts including syntax, command processing conventions, interface messages, and SCD programming examples. This section describes both device-dependent functions (SCD functions) and interface functions (low-level IEEE–488.1 functions).

The IEEE–488.1 interface (GPIB) is based on the IEEE Std 488.1–1987 *Digital Interface for Programmable Instrumentation*. This specification defines mechanical, electrical, and functional interface elements that enable data transfer between compatible devices. The SCD digitizer adheres to this standard.

The IEEE–488.1 uses a bit-parallel, byte-serial binary data format with a maximum transmission rate of 500 Kbytes/s. The interface allows connection of as many as 15 devices (including the controller) in a linear, star, or combined configuration. The total GPIB transmission cabling should not exceed 2 meters per device.

IEEE–488.1 devices include instruments that communicate bidirectionally (talk and listen) and unidirectionally (listen only or talk only). Each device on the bus has its own unique address and must be addressed and placed in a talk or listen mode before the controller can communicate with it.

Command Processing Conventions

Several command processing conventions affect the way programs are written. SCD command processing conventions are described in this section.

Upper and Lower Case

The digitizer ignores the case of alphabetical text that is input. Thus, `rqs ON` and `RQS ON` are identical. The digitizer always returns upper case only to the IEEE–488.1 port. Therefore, query responses always return as upper case characters.

Abbreviations and Minimum

Any command word in a command line can be abbreviated to a minimum ambiguity and be properly interpreted by the digitizer. For example, `TRIGGER`, `TRIGGE`, `TRIGG`, `TRIG`, and `TRI` are identical commands because each of the abbreviated forms includes the command's minimum ambiguity (`TRI`).

The minimum ambiguity for each command word is defined in the command set tables (Section 3).

Quotes

Double quotes can be used inside a quoted string argument by entering the quotes twice. For example, `"Press ""User1"" Key"` causes `Press "User1" Key` to be sent to the digitizer.

White Space

White space is any CR, space, LF, or TAB character. The digitizer always ignores white space. If the terminator switch is set to EOI/LF (line-feed termination), line-feeds will not be interpreted as white space if properly placed in the message. The line-feed (LF) character is recognized as such anywhere but inside of a quoted string. There, line-feed is ignored.

Message Terminators

A message terminator indicates the end of an IEEE–488.1 message. Using the switch on the rear panel, the SCD can be set to recognize messages terminated by either EOI or EOI/LF (line feed).

When EOI is the terminator, any data byte on the bus when EOI is asserted is recognized as the end of a message. EOI only is made for controllers that want to avoid the overhead of the 2 extra bytes (CR + LF).

When EOI/LF is the terminator, either the LF (line-feed) character properly placed in a message, or any data byte on the bus simultaneously with EOI asserted, is recognized as the end of an input (to the SCD) message. The digitizer will not terminate on only a LF character embedded in binary data transfers or in quoted strings.

When EOI/LF is the terminator, the SCD transmits a Carriage Return character followed by Line Feed (LF) and simultaneously asserts EOI to terminate an output message. If the controller supports EOI, the terminator switch should be set to EOI (not EOI/LF). This eliminates any unwanted terminations if the binary waveform data sent to the controller contains line-feed characters.

Depending on placement, the LF character may be interpreted as white space as described above.

Longforms and Shortforms (LONGFORM command)

The `LONGFORM` command controls the number of characters the digitizer returns to the controller as a result of a query command.

When `LONGFORM` is `OFF`:

- the digitizer returns the abbreviated form of command elements (for example, `TRI COUP:AC` or `CURS OFF`)
- responses to `ALLEV?` and `EVENT?` queries are limited to the abbreviated form of the message and the event code (for example, `EVE 121`).

When LONGFORM is ON:

- the digitizer returns the complete spelling of the command element (for example, TRIGGER COUPLING:AC or CURSORS OFF)
- responses to ALLEV? and EVENT? queries include the complete spelling of the command, the event code, and a quoted string describing the event code (for example, EVENT 155,"Invalid string input")

The PATH command (described below) also affects digitizer responses. The LONGFORM command is described in Table 3-8.

Removing Command Echoes in Responses (PATH command)

The PATH command controls whether or not the digitizer includes in its response the query command it received from the controller. When PATH is ON, the command is included with the response. For example, CHA RANGE:.5 is returned when PATH is ON. When PATH is OFF, the query command is not included. The example just given is shortened to .5 when PATH is OFF.

Command Syntax

There are two types of SCD commands: set commands and query commands. These types are described below, but their syntactical forms differ slightly as described here.

IEEE–488.1 set command syntax consists of headers, links, arguments, and delimiters. Set commands have the following syntactical form:

```
<header><space delimiter ( )><optional link><colon
delimiter (:)><optional argument>
```

For example: TRIGGER SOURCE:CHA

Some SCD set commands do not include a link. A few set commands have neither link nor argument.

Query commands have a similar form except that the header includes a query indicator a question mark (?) and does not include an argument or the colon delimiter. Some SCD query commands do not include a link.

Query commands have the following syntactical form:

```
<header><?><space delimiter><optional link>
```

For example: TRIGGER? SOURCE

Set and Query Commands

SCD commands can either be set commands or query commands. Syntax for set and query commands differ slightly as described above.

Set Commands

Set commands instruct the instrument to do something, such as set up a parameter, start a process, etc. Set commands can be of three types:

`CALIBRATE TRIGGER` (set without link)

`TRIGGER COUPLING:AC` (set with link)

`MTRIG` (set with neither link nor argument)

Query Commands

Query commands instruct the instrument to prepare to transfer instrument or other settings or waveform data to the controller. Once a query command has been sent, the device is talk addressed to allow transfer of data from the digitizer's output buffer to the IEEE–488.1 bus.

Most query commands are derived from set commands; they allow checking the current setting of a parameter set by a set command. Query commands are similar in appearance to set commands except for a question mark added to the header (`ACQUIRE?`). Queries may or may not have a link; these commands never have an argument, therefore the colon delimiter separating link and argument is not legal. Here are some examples of query commands:

```
EVENT?

TRIGGER? COUPLING
```

Queries can be general or specific. A general query requests settings or data for many links. In the query

```
CHA?
```

the digitizer returns all of the settings of channel A (SCD1000 only).

A specific query requests settings or data for one link. In the query

```
REPSET? NREPEAT
```

only the setting of `NREPEAT` will be returned to the digitizer if it is talk addressed before another command is sent.

Out of Phase Query

Several queries can be concatenated into one command line as explained later in this section. If a query (single or multiple queries) is sent to the digitizer and the instrument is not talk addressed before another query or command is sent to the same device, the first query is disregarded, and the response to it is cleared. The data requested by a second query is sent as usual, if the digitizer is subsequently talk addressed. In order to get the information from the digitizer, it must be talk addressed after the query is sent.

Oversized Query Response

Some general query commands may produce a response that is too big for the digitizer's output buffers. If all of this data is not talked out of the instrument, the front panel of the digitizer will stop functioning, which makes the instrument appear to lock-up. The front panel will resume normal operation when the remaining data is transmitted.

Set-Only and Query-Only Commands

Some commands are query-only or set-only. For example, the `EVENT?` query has no corresponding set command. The `ERASE <NRx>` command has no corresponding query.

There may be times when a query-only command will be sent to the digitizer in a set command. This is most likely to occur when the results of a query are stored external to the digitizer and are later returned to the digitizer in a corresponding set command. An example of this is the `ID?` query. In response to a `ID?` query, the digitizer will return a string which includes the `ID?` query response. When this string is returned to the digitizer as a set command, a command error is generated. Some query-only commands cause the instrument to generate an error when returned as a set command; others don't. The command set in Section 3 of this manual indicates which query-only commands will generate an error when sent to the instrument as a set command.

Headers

A header identifies a set of commands that affect a category of functions of the instrument, such as `TRIGGER` settings, `ACQUIRE` settings, or `TEST` functions. In the commands:

```
CHA RANGE:.5
TRIGGER SOURCE:CHA
```

`CHA` and `TRIGGER` are headers.

The simplest SCD command consists of just a header, for example:

```
MTRIG
```

Links

A link further specifies a particular parameter of a category of functions that are identified by a header. In the command:

```
CHA OFFSET:.5
```

`OFFSET` is an SCD link specifying a particular parameter of channel 1. Links are separated from the header by a space delimiter (ASCII 32) or a tab character (ASCII 11).

Many commands have links; however, some do not have links, such as:

`PATH OFF` (command without link)

Arguments

An argument sets the state or value of a parameter specified by a link or header. Most commands require arguments. However, some commands do not have arguments. The argument is separated from the link or header by a colon delimiter (:). An argument can be:

- a symbol to set a parameter's state
- a numeric to set a parameter's value
- a quoted string to specify a string of characters

Character String Arguments — In the command

```
DISPLAY ON
```

the argument ON is a symbol that turns on the optional Display Unit. Notice that this command has no link.

Numeric Arguments — In the command

```
CHA OFFSET:.5
```

the numeric value .5 sets the offset of channel A. The SCD accepts the following numeric arguments:

- signed integer
- unsigned integer (unsigned numbers are always interpreted as positive)
- floating point value with no exponent
- floating point value with an exponent

Although the digitizer can receive any of these numeric expressions as a numeric argument, numeric responses from the digitizer follow certain numeric conventions. The convention used depends on the command. Some responses are unsigned integers; some are floating point values with an exponent. The command set tables in Section 3 identify the numeric convention used for each appropriate command.

Quoted Strings Arguments — In the command

```
USER1 "Grp Exec","Trig"
```

the quoted strings `Grp Exec` and `Trig` label one of the user-definable keys on the Display Unit.

Quoted strings can be delimited by double quotes (").

Quotes can be used inside the string by entering the quotes twice. For example, `"Press ""User1"" Key"` causes `Press "User1" Key` to be displayed.

Delimiters

Colon (:) — Separates a link from its following argument.

Comma (,) — Separates an argument from the next link in a command line, or it separates multiple links for some commands. The comma is used to include more than one link in a command line in order to set several parameters of a single header. For example, to change several trigger parameters in one command string:

```
TRIGGER COUPLING:AC,MODE:AUTO,SLOPE:PLUS
```

To separate multiple arguments in a single command:

```
ABSTOUCH 0, 8
```

Semicolon (;) — Separates a group of links and arguments of one header from the next header in a command line. The semicolon is used to include more than one header in a command line. For example: `VMODE ADD;CHA RANGE:100E-3,OFFSET:.5, TYPEOFFSET:VOLTS;CHB RANGE:1,OFFSET:.5,TYPEOFFSET:VOLTS.`

Concatenation of Commands

Multiple set and query commands can be sent in the same command line if properly delimited. (See Delimiters above.) For example, the following command lines

```
VMODE ADD <EOI>

CHA RANGE:100E-3,TYPEOFFSET: VOLTS, OFFSET:-1.25 <EOI>

CHB RANGE: 200E-3,TYPEOFFSET: VOLTS, OFFSET:-1.50
<EOI>

ACQUIRE STATE:HLDNXT <EOI>
```

could be concatenated into one command line such as

```
VMODE ADD;CHA RANGE:1.00E-3,TYPEOFFSET:VOLTS, OFF-
SET:-1.25;  CHB RANGE:100E-3,TYPEOFFSET:VOLTS, OFF-
SET:-1.50;   ACQUIRE STATE:HLDNXT<EOI>
```

Talking With Nothing to Say (TWNTS)

If a response is requested of the digitizer without it first having been queried, it responds with a TWNTS message while asserting the EOI line. The message is one byte long with the value FF <EOI> in EOI mode and FF CR LF <EOI> in LF/EOI mode.

TWNTS will not occur if the digitizer is currently acquiring.

Interface Messages

Interface messages are low-level commands generated by the GPIB interface software in the controller or composed according to the IEEE–488.1 standard. Unlike instrument commands, interface messages cannot be sent as character strings.

The following descriptions are provided as an overview of how these GPIB messages relate to the SCD. All of these messages appear on the bus with the attention line (ATN) asserted. For complete descriptions of the interface messages and resultant interface states, see ANSI/IEEE Std 488.1–1987.

The SCD supports the IEEE–488.1 interface functions as follows:

- Acceptor Handshake (AH1)
- Controller (C0)
- Device Clear (DC1)
- Device Trigger (DT1)
- Tri-state Bus (E2)
- Listener (L4)
- Parallel Poll (PP0)
- Remote/Local (RL0 however, the SCD generally follows the state transitions of remote and local instrument control; see Local Lockout below)
- Service Request (SR1)
- Source Handshake (SH1)
- Talker (T5)

Listen Address (LA) and Talk

Listen Address (LA) messages condition the SCD to receive commands. Talk Address (TA) messages condition the SCD to respond to queries and serial polls. The SCD receives its Listen Address when the data on the bus equals decimal 32 plus the address set on the SCD's rear panel address switches. The SCD receives its Talk Address when the data on the bus equals decimal 64 plus the address set on the SCD's rear panel address switches. For example, if the SCD is set to address 20 on the dip switch, then the listen address is 32+20=52 and the talk address is 64+20=84.

Local Lockout (LLO)

Remote With Lockout State (RWLS) inhibits front panel operation, which prevents the front panel controls from affecting the SCD. While in this state, the front panel LOCK (red) and GPIB (yellow) LEDs are on. These LEDs will only show if there is no front panel attached to the instrument.

The SCD powers on in the local state (LOCS). RWLS can be achieved by asserting REN, listen addressing the box, and sending the LLO (Local Lockout) message. The front panel controls can also be turned off by sending the FPANEL OFF command (see Table 3-13).

Unlisten (UNL) and Untalk (UNT)

The Unlisten (UNL) message is equivalent to talk address decimal 31, so the address sent is 32+31=63. The UNL message cancels the LA message. The Untalk (UNT) message is equivalent to listen address decimal 31, so the address sent is 64+31=95. The UNT message cancels the TA message. The Untalk and Unlisten commands are universal commands. All instruments on the bus stop talking and listening when the controller sends UNT and UNL messages.

Device Clear (DCL)

The Device Clear (DCL) message initializes communication between the SCD and the controller. In response to DCL, the digitizer clears input and output messages as well as unexecuted control settings. Errors and events waiting to be reported are cleared, except for the power-on event. The SRQ message is cleared, unless SRQ is true from a power-on condition.

Interface Clear (IFC)

Interface Clear (IFC) is a signal line of the IEEE−488.1 cable. When IFC is asserted, both the Talk and Listen functions are placed in an idle state. This produces the same effects as receiving both the Untalk and Unlisten commands. IFC resets the interface only and does not affect any instrument functions. The input and output buffers are not cleared.

Selected Device Clear (SDC)

Selected Device Clear (SDC) performs the same function as DCL, but requires the instrument to have been listen-addressed (more than one instrument can be simultaneously addressed and thus affected by SDC). This function allows the controller to perform a device clear on selected instruments. When the SCD receives an SDC, it executes a Device Clear (explained above).

Serial Poll Enable (SPE) and Serial Poll Disable (SPD)

The Serial Poll Enable (SPE) message causes the SCD to transmit its serial-poll status byte when it is talk-addressed. The Serial Poll Disable (SPD) message returns the digitizer to normal operation. If SRQ was asserted, it is cleared when the digitizer is polled.

Parallel Poll

The SCD does not support parallel polling commands.

Group Execute Trigger (GET)

The SCD supports the Group Execute Trigger (GET) function. In the SCD, the DT command (Table 3-8) enables the SCD to recognize the GET command and enter one of the acquisition states (STOP, RUN, or HOLDNXT). Get requires the device to be a listener. When GET is received after DT is received, the digitizer enters the set acquisition state. This capability allows many instruments to be synchronized by having them wait for the GET command before executing their instructions. A multiple digitizer system can use the GET command to acquire many channels at the same time.

Device Trigger (DT)

Device Trigger programs the digitizer's response to the GET message. The DT command allows the user to program the digitizer to enter one of the three acquisition states upon receiving GET.

Device-Dependent Programming

High-level programming languages, such as BASIC, C, and Pascal, are used to create programs that send GPIB messages to devices and receive data and responses from the devices. Statements in these languages usually contain three parts:

- input/output keyword (such as `PRINT` or `READ`)
- IEEE–488.1 logical unit designator, which may be an address or a name (such as `20` or `DIG`)
- instrument command or response formed by a character string or string-variable designator (such as `CHA RANGE:2`)

Generic Programming Language

Because the SCD can be controlled by several different computer types, a "generic" computer language is used in the following examples to replace language constructs from other languages that provide input, output, and other statements. Table 2-1 lists the generic language constructs used in the examples.

Table 2-1: Generic Language Constructs for Examples

Generic Construct	Description
Sendstring @address	Send ASCII string to device at @address
Readstring @address	Read ASCII string from device at @address
Readintarray @address	Read binary-encoded integer array values, most significant bit first from device at @address.
Serial poll (address,statusb)	Perform a serial poll to read status byte from device requesting service. Device address is obtained during poll.
Dim	Allocate space for arrays or strings. For example: Dim Integer Intwfm(1024)
While/Wend	While command for looping requirements.
Writedisk	Save data to disk.
@Screen	Replaces @address in command. Use to output to the computer monitor.
@Variablename	Replaces @address in command. Used as a variable to identify an address.

In the following example, the SCD is set to address 20.

All examples assume that proper configuration and declaration to the GPIB port and device have been done prior to these statements. The examples show proper command syntax.

Output Statement Examples

Output statements send commands and other data to the digitizer. The following examples show several commands used to set up the vertical inputs of the SCD, set up the trigger system, and begin an acquisition. Any SCD commands may replace the ones following the generic output statement. (The following commands are written for an SCD1000 and could be concatenated as explained earlier.)

```
Sendstring @20: "VMODE ADD;CHA RANGE:100E-3"
Sendstring @20: "CHB RANGE:200E-3"
Sendstring @20: "TRIGGER MODE:NORMAL,SOURCE:CHA"
Sendstring @20: "ACQUIRE STATE:HLDNXT"
```

Input Statement Examples

Input statements allow the controller to receive waveform data and other information from the digitizer into arrays or variables. In the following examples, variables and arrays have been dimensioned large enough to hold the expected data

```
Readstring @20: SETTINGS$
Readintarray @20: Intwfm(i)
```

Query Command and Response Examples

Query and input operations may be specified by separate statements, or, if the controller permits, a prompting input statement can perform both functions. The following example queries for and then acquires the channel 1 settings of the device at address 20 (SET$ has been dimensioned as a string variable large enough to accommodate all data coming from the device).

```
Sendstring @20:"CHA?"
Readstring @20:SET$
```

In this operation, the controller addresses the device as a listener and sends the query command, "CHA?", over the bus. The controller then reassigns the instrument to be a talker and receives the characters into the target variable SET$. The variable then contains the channel 1 information, which can be displayed on the console:

```
Sendstring @Screen: SET$
CHA RANGE:2.0E+0,OFFSET:0,TYPEOFFSET:PERCENT,COU-
PLING:DC
```

Instrument Settings Transfer

Setup parameters can be copied to the controller using the SET? query command. The settings can be saved in a pre-defined string variable and then written to a disk file. Once saved in the controller, the SCD settings can be modified at a later time and returned to the digitizer. The entire setup is described in ASCII characters, as defined in the command tables of Section 3.

The following example shows how to dimension a string variable to receive the current digitizer settings, modify them, and then send them back to the digitizer.

```
DIM SET$ (600)
Sendstring @20:"SET?"
Readstring @20:SET$
.
commands that may modify the settings in SET$
.
Sendstring @20: SET$
```

Handling Service Request (SRQ) and Event Codes

The most recent RQS command (see Table 3-7) determines whether the digitizer asserts the SRQ control line of the bus when either an error or a change in status occurs. The RQS command is always set to ON at power-up.

If the controller is configured and programmed appropriately, an asserted SRQ line interrupts its normal program flow. To service the interrupt, the controller polls each device on the bus. In response to being polled, the interrupting device returns a status byte, which reveals the type of event that occurred. The interrupting device then clears the SRQ line.

If another SRQ is pending, either from the same or another instrument on the GPIB, the SRQ line will be re-asserted. The SRQ line is re-asserted each time an SRQ needs to be handled. If the controller does not respond to the SRQ, the instrument continues to operate and communicate normally, even though the condition that caused the SRQ may invalidate an operation such as a measurement, setting, or acquisition.

After reading a status byte, the program may request more information about the event by sending the EVENT? query command. The device returns a number (and a descriptive string if the digitizer is programmed to do so) that identifies the specific event. Section 4 defines the various status bytes, event codes, and errors.

The following example shows how to read the status byte and obtain the event code after SRQ has been asserted. (The device address is obtained during the poll routine and stored in the variable, DEV.) The status byte and associated event code are then displayed on the controller's screen. The variables for device address (DEV) and status byte (STATUSB) are integer. However, because the LONGFORM command is set to ON, a string is returned with every event query, which is stored in the string variable, EVENT$. (See descriptions of these commands in Section 1.)

```
Serialpoll: (DEV,STATUS)
Sendstring @DEV: "LONGFORM ON;PATH ON"
Sendstring @DEV:"EVENT?"
Readstring @DEV:EVENT$
Sendstring @Screen: "SRQ from ";DEV;", status= ";STA-
TUS;
EVENT$
```

Sending an `SRQMASK USRx:ON` command (where x=1 or 2) to the SCD allows an SRQ to be generated when the appropriate user button on the Display Unit is pressed. (Each of the user buttons can be labeled. See Section 3 for the appropriate commands.) The SCD sends unique status byte and event code values to signify when a user button is pressed. These values can be used to control program flow by waiting for the operator to press a user button before performing other actions, such as acquiring a waveform or making a cursor measurement.

In addition, `SRQMASK ABSTOUCH:ON` allows any key on the Display Unit to generate an SRQ. See Table 3-7 for information on the SRQMASK command.

Device Dependent Command Set Listing

Table 2-2 alphabetically lists all the SCD commands described in Section 3. In the table, spelling of headers, links, and arguments is done with both uppercase and lowercase characters. Uppercase characters indicate the minimum ambiguity of each command. The entire spelling (longform) is in uppercase and lowercase. Other conventions follow those of the tables in Section 3.

See Section 3 for descriptions of each of these commands.

Command Set Table Format Conventions

The following format conventions are used in the command set tables:

Items included in brackets ([...]) are optional items.

<x> represents an alphacharacter

<NR1> represents a signed integer.

<NR2> represents a floating point number with no exponent

<NR3> represents a floating point number with an exponent

<NRx> represents an <NR1>, <NR2>, or <NR3>

<ui> represents an unsigned integer with no leading space

<qstring> represents a quoted string ("xxxxx" or 'xxxxx')

<bblock> represents a Tek Codes & Formats Binary Block

Spelling for headers, links, and arguments is done with uppercase and lowercase characters; however, the command's minimum ambiguity appears in uppercase characters (for example, `TRI`) while the longform includes both uppercase and lowercase characters (for example, `TRIgger`).

Each table's Description column includes a brief description, numeric limits (where appropriate), factory settings (where appropriate), an example of the query or command, and, for queries, an example of the SCD response.

Factory settings are the values programmed in the SCD when first shipped from the factory. Subsequent programming of the SCD causes values to be changed. These values are saved in memory when the unit is turned off. The digitizer powers on with the saved settings; the SCD does not return to the factory settings each time it is turned on. See Factory Settings in Section 1 and the INIT command in Table 3-8 for more information.

All example responses are representative of the results when the `PATH` and `LONGFORM` commands are `ON`.

Commands are sorted in order of typical importance with queries following commands. For example, in Table 3-7 all `SRQMASK` commands are listed followed by all `SRQMASK?` queries.

Table 2-2: Alphabetical Command Set Listing

Header	Link	Argument	Reference Section 3
ABStouch		<NRx>,<NRx>	Table 3-8
ABStouch		CLEar	Table 3-8
ABStouch?			Table 3-8
ACQUire	AVErage:	<NRx>	Table 3-3
ACQUire	GEOmetry	ON OFF RUN	Table 3-3
ACQUire	HLDNxt	ON OFF	Table 3-3
ACQUire	LENgth:	<NRx>	Table 3-3
ACQUire	MODe:	NORmal ADVance AVErage	Table 3-3
ACQUire	NRECord:	<NRx>	Table 3-3
ACQUire	STARt	<NRx>	Table 3-3
ACQUire	STATe:	STOP RUN HLDNxt	Table 3-3
ACQUire	TIME:	<NRx>	Table 3-3
ACQUire?			Table 3-3
ACQUire?	AVErage		Table 3-3
ACQUire?	GEOmetry		Table 3-3
ACQUire?	HLDNxt		Table 3-3
ACQUire?	LASt		Table 3-3
ACQUire?	LENgth		Table 3-3
ACQUire?	MODe		Table 3-3
ACQUire?	NRECord		Table 3-3
ACQUire?	STARt		Table 3-3
ACQUire?	STATe		Table 3-3
ACQUire?	TIME		Table 3-3
ALLEv?			Table 3-7
AREA?			Table 3-18

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
AREAZone		DISTal MEAszone MESial PROXimal	Table 3-18
AREAZone?			Table 3-18
ARM		INTERnal EXTernal	Table 3-4
ARM?			Table 3-4
BASE?			Table 3-18
BASEAber?			Table 3-18
BASEMode	METhod:	ABSOlute HISTOGram HISTOMean MINImum	Table 3-18
BASEMode	LEVEl:	<NRx>	Table 3-18
BASEMode?			Table 3-18
BASEMode?	METhod		Table 3-18
BASEMode?	LEVEl		Table 3-18
BASETop?			Table 3-18
BATdate?			Table 3-14
BELl	BUTton:	ON OFF	Table 3-13
BELl	KNOb:	ON OFF	Table 3-13
BELl	RINg		Table 3-13
BELl?			Table 3-13
BELl?	BUTton		Table 3-13
BELl?	KNOb		Table 3-13
CALIBRATE		[ALL] CRT GEOmetry INPut HORizontal TRIgger VERTical	Table 3-14
CALIBRATE?			Table 3-14

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
CALOut	CH<x>	AMPL OFF TIME	Table 3-14
CALOut	EXTernal	AMPL TIME	Table 3-14
CALOut?			Table 3-14
CALOut?	CH<x>		Table 3-14
CALOut?	EXTernal		Table 3-14
CALIBRATOr	AMPLItude:	<NRx>	Table 3-14
CALIBRATOr	TIME:	<NRx>	Table 3-14
CALIBRATOr?			Table 3-14
CALIBRATOr?	AMPLItude		Table 3-14
CALIBRATOr?	TIME		Table 3-14
CCOnstant?	<ui>		Table 3-14
CDAte?			Table 3-14
CH<x>	COUPling:	AC (SCD1000) DC OFF (SCD1000)	Table 3-2
CH<x>	INVert:	OFF (SCD1000) ON (SCD1000)	Table 3-2
CH<x>	OFFSet:	<NRx>	Table 3-2
CH<x>	RANge: (SCD1000)	<NRx>	Table 3-2
CH<x>	TYPEOffset:	PERCent VOLts	Table 3-2
CH<x>?			Table 3-2
CH<x>?	COUPling		Table 3-2
CH<x>?	INVert (SCD1000)		Table 3-2
CH<x>?	OFFSet		Table 3-2
CH<x>?	RANge (SCD1000)		Table 3-2
CH<x>?	TYPEOffset		Table 3-2
CH<x>?	PROBe (Option 1E)		Table 3-2
CH?			Table 3-2

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
CLOck	DATE:	<qstring>	Table 3-15
CLOck	TIME:	<qstring>	Table 3-15
CLOck?			Table 3-15
CLOck?	DATE		Table 3-15
CLOck?	TIME		Table 3-15
CROSS?	DFAll DRIse MAXLoc MFAll MINLoc MPEriod MRIse PFAll PRIse		Table 3-18
CROSS?			Table 3-18
CRS?			Table 3-9
CRS1	LOCTn:	WIN<ui>	Table 3-9
CRS2	LOCTn:	WIN<ui>	Table 3-9
CRS1	XPOint:	<NRx>	Table 3-9
CRS2	XPOint:	<NRx>	Table 3-9
CRS1?			Table 3-9
CRS2?			Table 3-9
CRS1?	LOCTn		Table 3-9
CRS2?	LOCTn		Table 3-9
CRS1?	XTIMe		Table 3-9
CRS2?	XTIMe		Table 3-9
CRS1?	XPOint		Table 3-9
CRS2?	XPOint		Table 3-9
CRS1?	YCOord		Table 3-9
CRS2?	YCOord		Table 3-9
CRSD	TYPETime:	HZ SECond	Table 3-9
CRSD?			Table 3-9

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
CRSD?	T		Table 3-9
CRSD?	TYPETime		Table 3-9
CRSD?	Y		Table 3-9
CRTBkgnd		<NRx>	Table 3-15
CRTBkgnd?			Table 3-15
CURSors		ON OFF	Table 3-9
CURSors?			Table 3-9
CURVe?			Table 3-6
DATA	BYTEOrder:	MSB LSB	Table 3-6
DATA	CNTrecord:	<NRx>	Table 3-6
DATA	COUNt:	<NRx>	Table 3-6
DATA	FLAGbit:	ON OFF	Table 3-6
DATA	STARt:	<NRx>	Table 3-6
DATA	STREcord:	<NRx>	Table 3-6
DATA?			Table 3-6
DATA?	BYTEOrder	MSB LSB	Table 3-6
DATA?	CNTrecord		Table 3-6
DATA?	COUNt		Table 3-6
DATA?	FLAGbit	ON OFF	Table 3-6
DATA?	STARt		Table 3-6
DATA?	STREcord		Table 3-6
DEBug	GPIb:	ON OFF	Table 3-8
DEBug?	[GPIb]		Table 3-8
DIAg?			Table 3-14
DISplay		ON OFF	Table 3-12

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
DISplay?			Table 3-12
DISTAl?			Table 3-18
DISTLevel	PERCent:	<NRx>	Table 3-18
DISTLevel	LEVEl:	<NRx>	Table 3-18
DISTLevel?			Table 3-18
DISTLevel?	PERCent		Table 3-18
DISTLevel?	LEVEl		Table 3-18
DT		RUN STOP HLDNxt OFF	Table 3-8
DT?			Table 3-8
ERAse		<NRx>	Table 3-10
EVEnt?			Table 3-7
EVQty?			Table 3-7
FALL?			Table 3-18
FALLSlew?			Table 3-18
FOCus		<NRx>	Table 3-15
FOCus?			Table 3-15
FPANel		ON OFF	Table 3-13
FPANel?			Table 3-13
FPStat?		<NRx>	Table 3-7
FREquency?			Table 3-18
GEOMArray?			Table 3-6
GRATicule		ON OFF	Table 3-9
GRATicule?			Table 3-9
HELp?			Table 3-8
HEXPMd		ALIgned INDep	Table 3-12
HEXPMd?			Table 3-12

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
ID?			Table 3-7
INIt		PANel GPIb [ALL]	Table 3-8
INTEnsity		<NRx>	Table 3-15
INTEnsity?			Table 3-15
INTERPolate?		<NRx>	Table 3-6
LEVMode		PERCent ABSOlute	Table 3-18
LEVMode?			Table 3-18
LINArray?			Table 3-6
LLSet?		<bblock>	Table 3-10
LONgform		ON OFF	Table 3-8
LONgform?			Table 3-8
MAXImum?			Table 3-18
MEAN?			Table 3-18
MEASUre	FUNction:	ON OFF	Table 3-18
MEASUre	MANmeas		Table 3-18
MEASUre	MEASZone:	CURSors FULl WIN1 WIN2 WIN3 WIN4	Table 3-18
MEASUre	WAVfrm:	<NRx>	Table 3-18
MEASUre	WINDow:	WIN1 WIN2 WIN3 WIN4	Table 3-18
MEASUre?			Table 3-18
MEASUre?	FUNction MEASZone WAVfrm WINDow		Table 3-18

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
MESIal?			Table 3-18
MESLevel	PERCent:	<NRx>	Table 3-18
MESLevel	LEVEl:	<NRx>	Table 3-18
MESLevel?			Table 3-18
MESLevel?	PERCent LEVEl		Table 3-18
MINImum?			Table 3-18
MSList	AREA:	ON OFF	Table 3-18
MSList	BASE:	ON OFF	Table 3-18
MSList	BASEAber:	ON OFF	Table 3-18
MSList	BASETop:	ON OFF	Table 3-18
MSList	DISTAL:	ON OFF	Table 3-18
MSList	FALL:	ON OFF	Table 3-18
MSList	FALLSlew:	ON OFF	Table 3-18
MSList	FREquency:	ON OFF	Table 3-18
MSList	MAXImum:	ON OFF	Table 3-18
MSList	MEAN:	ON OFF	Table 3-18
MSList	MESIal:	ON OFF	Table 3-18
MSList	MINImum:	ON OFF	Table 3-18
MSList	PERIod:	ON OFF	Table 3-18
MSList	PK_pk:	ON OFF	Table 3-18
MSList	PROXImal:	ON OFF	Table 3-18

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
MSList	RISE:	ON OFF	Table 3-18
MSList	RISESlew:	ON OFF	Table 3-18
MSList	RMS:	ON OFF	Table 3-18
MSList	TOP:	ON OFF	Table 3-18
MSList	TOPAber:	ON OFF	Table 3-18
MSList	WIDth:	ON OFF	Table 3-18
MSList	CLEar		Table 3-18
MSList?			Table 3-18
MSList?	AREA		Table 3-18
MSList?	BASE		Table 3-18
MSList?	BASEAber		Table 3-18
MSList?	BASETop		Table 3-18
MSList?	DISTAL		Table 3-18
MSList?	FALL		Table 3-18
MSList?	FALLSlew		Table 3-18
MSList?	FREquency		Table 3-18
MSList?	MAXimum		Table 3-18
MSList?	MEAN		Table 3-18
MSList?	MESIal		Table 3-18
MSList?	MINimum		Table 3-18
MSList?	PERiod		Table 3-18
MSList?	PK_pk		Table 3-18
MSList?	PROXImal		Table 3-18
MSList?	RISE		Table 3-18
MSList?	RISESlew		Table 3-18
MSList?	RMS		Table 3-18

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
MSList?	TOP		Table 3-18
MSList?	TOPAber		Table 3-18
MSList?	WIDth		Table 3-18
MTRig			Table 3-4
NWIn		1 2 4	Table 3-12
NWIn?			Table 3-12
OPTion?			Table 3-7
PATh		ON OFF	Table 3-8
PATh?			Table 3-8
PDAte		ON OFF	Table 3-17
PDAte?			Table 3-17
PERIod?			Table 3-18
PLOT?			Table 3-17
PTItle		<qstring>	Table 3-17
PTItle?			Table 3-17
PK_pk?			Table 3-18
PROXImal?			Table 3-18
PROXLevel	PERCent:	<NRx>	Table 3-18
PROXLevel	LEVEl:	<NRx>	Table 3-18
PROXLevel?			Table 3-18
PROXLevel?	PERCent LEVEl		Table 3-18
RAW		LINArray REFArray	Table 3-15
RAW?			Table 3-15
RECAll		<NRx>	Table 3-10
REFArray?			Table 3-6
REFList?			Table 3-16

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
REPEat?			Table 3-6
REPSet	NREPEat:	<NRx>	Table 3-6
REPSet?			Table 3-6
REPSet?	NREPEat		Table 3-6
RESUlts?			Table 3-18
RESUlts?	AREA		Table 3-18
RESUlts?	BASE		Table 3-18
RESUlts?	BASEAber		Table 3-18
RESUlts?	BASETop		Table 3-18
RESUlts?	DISTAL		Table 3-18
RESUlts?	FALL		Table 3-18
RESUlts?	FALLSlew		Table 3-18
RESUlts?	FREquency		Table 3-18
RESUlts?	MAXImum		Table 3-18
RESUlts?	MEAN		Table 3-18
RESUlts?	MESIal		Table 3-18
RESUlts?	MINImum		Table 3-18
RESUlts?	PERIod		Table 3-18
RESUlts?	PK_pk		Table 3-18
RESUlts?	PROXImal		Table 3-18
RESUlts?	RISE		Table 3-18
RESUlts?	RISESlew		Table 3-18
RESUlts?	RMS		Table 3-18
RESUlts?	TOP		Table 3-18
RESUlts?	TOPAber		Table 3-18
RESUlts?	WIDth		Table 3-18
RISE?			Table 3-18
RISESlew?			Table 3-18
RMS?			Table 3-18

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
RQS		ON OFF	Table 3-7
RQS?			Table 3-7
SAFEguard	ACQProtect	ON OFF	Table 3-11
SAFEguard	LINConvert	<NRX>	Table 3-11
SAFEguard	PROTect:	ON OFF	Table 3-11
SAFEguard	PUPtst:	ON OFF	Table 3-11
SAFEguard	SECURE		Table 3-11
SAFEguard	STOPAcq		Table 3-11
SAFEguard?			Table 3-11
SAFEguard?	ACQProtect		Table 3-11
SAFEguard?	PROTect		Table 3-11
SAFEguard?	PUPtst		Table 3-11
SAVE		<NRx>	Table 3-10
SET?			Table 3-10
SETRef		ON OFF RUN	Table 3-16
SETRef?			Table 3-16
SRQmask	ABSTouch:	ON OFF	Table 3-7
SRQmask	CMDerr:	ON OFF	Table 3-7
SRQmask	EXERr:	ON OFF	Table 3-7
SRQmask	EXWarn:	ON OFF	Table 3-7
SRQmask	INErr:	ON OFF	Table 3-7
SRQmask	INWarn:	ON OFF	Table 3-7

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
SRQmask	OPCmpl:	ON OFF	Table 3-7
SRQmask	USR1:	ON OFF	Table 3-7
SRQmask	USR2:	ON OFF	Table 3-7
SRQmask?			Table 3-7
SRQmask?	ABSTouch		Table 3-7
SRQmask?	CMDerr		Table 3-7
SRQmask?	EXERr		Table 3-7
SRQmask?	EXWarn		Table 3-7
SRQmask?	INErr		Table 3-7
SRQmask?	INWarn		Table 3-7
SRQmask?	OPCmpl		Table 3-7
SRQmask?	USR1		Table 3-7
SRQmask?	USR2		Table 3-7
TESt	LOOp:	ON OFF	Table 3-14
TESt	NUM:	<NRx>	Table 3-14
TESt	SYS:	ALL DIG FP MPU OPTion	Table 3-14
TESt	VERBose:	ON OFF	Table 3-14
TESt?			Table 3-14
TESt?	LOOp		Table 3-14
TESt?	NUM		Table 3-14
TESt?	SYS		Table 3-14
TESt?	VERBose		Table 3-14
TEXt	CHAR:	<NRx>	Table 3-12
TEXt	CLEar:	<NRx>	Table 3-12

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
TEXt	LINE:	<NRx>	Table 3-12
TEXt	STRIng:	<qstring>	Table 3-12
TIMESt?			Table 3-6
TOP?			Table 3-18
TOPAber?			Table 3-18
TOPMode	METhod:	ABSOlute HISTOGram HISTOMean MAXimum	Table 3-18
TOPMode	LEVEl:	<NRx>	Table 3-18
TOPMode?			Table 3-18
TOPMode?	METhod		Table 3-18
TOPMode?	LEVEl		Table 3-18
TRIgger	COUPling:	AC DC	Table 3-4
TRIgger	DELay:	<NRx>	Table 3-4
TRIgger	LEVEl:	<NRx>	Table 3-4
TRIgger	MODe:	AUTo NORmal	Table 3-4
TRIgger	SLOpe:	PLUs MINUs	Table 3-4
TRIgger	SOUrce:	CHA (SCD1000) CHB (SCD1000) ADD (SCD1000) EXTernal INTernal (SCD5000 Opt 01) CALIBRATOR (SCD5000)	Table 3-4
TRIgger	TYPEDelay:	PERCent SECond	Table 3-4
TRIgger	TYPELevel:	PERCent VOLts	Table 3-4
TRIgger?			Table 3-4
TRIgger?	COUPling		Table 3-4
TRIgger?	DELay		Table 3-4
TRIgger?	LEVEl		Table 3-4

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
TRIgger?	MODe		Table 3-4
TRIgger?	SLOpe		Table 3-4
TRIgger?	SOUrce		Table 3-4
TRIgger?	TYPEDelay		Table 3-4
TRIgger?	TYPELevel		Table 3-4
UID		<qstring>	Table 3-7
UID?			Table 3-7
UNIts?			Table 3-18
UNIts?	AREA		Table 3-18
UNIts?	BASE		Table 3-18
UNIts?	BASEAber		Table 3-18
UNIts?	BASETop		Table 3-18
UNIts?	DISTAL		Table 3-18
UNIts?	FALL		Table 3-18
UNIts?	FALLSlew		Table 3-18
UNIts?	FREquency		Table 3-18
UNIts?	MAXImum		Table 3-18
UNIts?	MEAN		Table 3-18
UNIts?	MESIal		Table 3-18
UNIts?	MINImum		Table 3-18
UNIts?	PERIod		Table 3-18
UNIts?	PK_pk		Table 3-18
UNIts?	PROXImal		Table 3-18
UNIts?	RISE		Table 3-18
UNIts?	RISESlew		Table 3-18
UNIts?	RMS		Table 3-18
UNIts?	TOP		Table 3-18
UNIts?	TOPAber		Table 3-18
UNIts?	WIDth		Table 3-18

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
USER1		<qstring1><qstring2>	Table 3-8
USER2		<qstring1><qstring2>	Table 3-8
USER1?			Table 3-8
USER2?			Table 3-8
VERSion?			Table 3-7
VMOde		CHA (SCD1000) CHB (SCD1000) ADD (SCD1000)	Table 3-1
VMOde?			Table 3-1
VIDeo		ON OFF	Table 3-14
VIDeo?			Table 3-14
WAVfrm?			Table 3-6
WFMpre?			Table 3-5
WFMpre?	BIT/nr		Table 3-5
WFMpre?	BN.fmt		Table 3-5
WFMpre?	BYT/nr		Table 3-5
WFMpre?	BYT/or		Table 3-5
WFMpre?	CRVchk		Table 3-5
WFMpre?	ENCdg		Table 3-5
WFMpre?	NR.pt		Table 3-5
WFMpre?	PT.Fmt		Table 3-5
WFMpre?	PT.Off		Table 3-5
WFMpre?	WFId		Table 3-5
WFMpre?	XINcr		Table 3-5
WFMpre?	XUNit		Table 3-5
WFMpre?	XZEro		Table 3-5
WFMpre?	YMUlt		Table 3-5
WFMpre?	YOFf		Table 3-5
WFMpre?	YUNit		Table 3-5

Table 2-2: Alphabetical Command Set Listing (Cont.)

Header	Link	Argument	Reference Section 3
WFMpre?	YZEro		Table 3-5
WFTx		DL IL TCF	Table 3-5
WFTx?			Table 3-5
WIDth?			Table 3-18
WIN<ui>	EXPnt:	<NRx>	Table 3-12
WIN<ui>	HEXPNd:	<NRx>	Table 3-12
WIN<ui>	RECOrd:	<NRx>	Table 3-12
WIN<ui>	VEXpnd:	<NRx>	Table 3-12
WIN<ui>?			Table 3-12
WIN<ui>?	EXPnt		Table 3-12
WIN<ui>?	HEXPNd		Table 3-12
WIN<ui>?	RECOrd		Table 3-12
WIN<ui>?	VEXpnd		Table 3-12
WIN?			Table 3-12

GPIB Command Set

This section lists the GPIB commands for the SCD1000/SCD5000. Command syntax and other general information about the IEEE–488.1 interface are provided in Section 2. This section includes commands for the vertical system, arm and trigger systems, and acquisition parameters. Commands that affect data and waveforms, status and events, waveform preamble, and diagnostics and calibration are also listed, as well as GPIB-related commands, and other instrument commands. Other tables list initialization values and value limits.

Command Table Summary

The following tables are in this section.

Vertical Commands

In the Vertical commands, <x> can be A, or B (for the SCD1000 only); <x> is NULL for SCD5000.

Table 3-1: Vertical Mode Commands

Header	Link	Argument	Description
VMOde		ADD CHA CHB	SCD1000 Only. Selects input source from either channel or the algebraic sum of both. Factory setting: CHA Example: `VMODE ADD`
VMOde?			SCD1000 only. Queries for input channel selection. Example: `VMODE?` Response: `VMODE CHA`

Table 3-2: Vertical Channel Commands

Header	Link	Argument	Description
CH<x>	COUPling:	AC (SCD1000) DC OFF(SCD1000)	SCD1000: Sets the specified channel coupling to AC, DC,or OFF (input disconnected from signal). SCD5000: DC coupling only. Factory setting: AC Example: `CHA COUPLING:AC`
CH<x>	INVert:	ON OFF	SCD1000 only. Inverts the signal from the specified channel (x). Factory setting: OFF Example: `CHB INVERT:ON`
CH<x>	OFFSet:	<NRx>	Sets the specified channel input offset to <NRx>Limits: SCD1000:±250 mV to ±25 V (1X probe attenuation); SCD5000: ±4 V Factory setting: 0 Example: `CHA OFFSET:1.25` (SCD1000) Example: `CH OFFSET:1.25` (SCD5000)
CH<x>	RANge:	<NRx>	SCD1000 only. Sets specified channel full scale range. The valid settings are 100 mV, 200 mV, 500 mV, 1 V, 2 V, 5 V,10 V (1X probe attenuation). Limits: 100 mV to 10 V (1X probe attenuation) Factory setting: 1 V Example: `CHA RANGE:200E-3`
CH<x>	TYPEOffset:	PERCent VOLts	Sets the specified channel input offset unit to PERCent of full-scale range or VOLts. Factory setting: VOLTS Example: `CHA TYPEOFFSET:PERCENT`

Table 3-2: Vertical Channel Commands (Cont.)

Header	Link	Argument	Description
CH<x>?			Queries for all settings for the specified channel: COUPling, RANge, TYPEOffset, OFFSet, INVert, and PROBe Example: `CHB?` Response: `CHB COUPLING:AC,RANGE:200.E-3, TYPEOFFSET:PERCENT,OFFSET:10.E-1, INVERT:OFF,PROBE:"NOT INSTALLED."`
CH<x>?	COUPling		Queries for the channel's input coupling setting. SCD1000 responds with AC, DC, or OFF. SCD5000 responds with DC only. Example: `CHA? COUPLING` Response: `CHA COUPLING:AC`
CH<x>?	INVert		SCD1000 only. Queries for the channels signal invert setting. Example: `CHB? INVERT` Response: `CHB INVERT:OFF`
CH<x>?	OFFSet		Queries for the channel's offset. The response is a floating point number with an exponent. Example: `CHB? OFFSET` Response: `CHB OFFSET:100.E-3`
CH<x>?	RANge		Queries for the channel's full scale range setting. SCD1000 responds with it's range setting; SCD5000 responds with a range of 5 Volts. Example: `CHA? RANGE` Response: `CHA RANGE:200.E-3` (SCD1000) Response: `CH RANGE:5.0` (SCD5000)
CH<x>?	TYPEOffset		Queries for the channel's input offset units. Example: `CHA? TYPEOFFSET` Response: `CHA TYPEOFFSET:VOLTS`
CH<x>?	PROBe (Option 1E)		Queries for the channel's probe value. Returns the Level (I or II) of probe attached to the channel and a string of information generated by the probe. If no probe is attached, the return value is: "NOT INSTALLED." Example: `CHA? PROB` Response: `CHA PROBE:"NOT INSTALLED,"`
CH?			Queries for all settings of all channels: COUPling, RANge, TYPEOffset, OFFSet, and INVert. Example: `CH?` Response: `CHA COUPLING:AC,RANGE:200.E-3, YPEOFFSET:PERCENT,OFFSET:10.E-1, INVERT:ON,PROBE:"NOT INSTALLED";CHB COUPLING:DC,RANGE:500E-3,TYPEOFFSET:VOLTS, OFFSET:1.E0,INVERT:ON,PROBE:"NOT INSTALLED"`

Acquire Commands

Table 3-3: Acquire Commands

Header	Link	Argument	Description
ACQuire	AVErage:	<NRx>	Sets the number of averages to perform in AVErage mode. Limits: 1 to 1024 Factory setting: 16
			Number of Averages / **Resolution of Curve Data** 1–3 / 11 bits 4–15 / 12 bits 16–98 / 13 bits 99–1024 / 14 bits
ACQuire	GEOmetry:	ON OFF RUN	Enables internal vertical geometry correction. Factory setting SCD1000 OFF, SCD5000 ON Performs internal calibration to create the correction table to improve vertical linearity using the instrument's current acquisition settings.
ACQuire	HLDNxt:	ON OFF	Turns on or off holdnext acquisition mode. ACQuire STATe must be set to HLDNxt. Factory setting: OFF Example: `ACQUIRE HLDNXT:ON`
ACQuire	LENgth:	256 512 1024	Selects the record length in sample points of all records. See Acquisition System in Section 1. Limits: 256, 512, 1024. Factory setting: 512 Example: `ACQUIRE LENGTH:256`
ACQuire	MODe:	NORmal ADVance AVErage	Sets the acquisition mode. NORmal fills only the record specified by the STArt command. ADVance fills a specified number of consecutive records set by NRECord starting with record specified by the STArt command. AVErage averages a number of acquisitions set by ACQuire:AVErage. The result is placed in the record specified by the STArt command. See also Acquisition Sequence and Acquisition Process in Section 1. Factory setting: NORmal Example: `ACQUIRE MODE:ADVANCE`
ACQuire	NRECord:	<NRx>	Sets the number of consecutive records to fill in ADVance mode. Limits: 1 to 16. Factory setting: 1 Example: `ACQUIRE NRECORD:4`

Table 3-3: Acquire Commands (Cont.)

Header	Link	Argument	Description
ACQuire	STARt:	<NRx>	Selects the record where the next acquisition starts. Records 1, 2, 3 and 4 are stored in nonvolatile memory. Limits: 1 to 16. Factory setting: 1 Example: `ACQUIRE START:2`
ACQuire	STATe:	STOp RUN HLDNxt	Controls the acquisition state. STOp immediately stops the acquisition sequence. RUN starts the acquisition sequence and causes the digitizer to perform as many acquisitions as possible. HLDNxt completes one acquisition process and then stops. See Acquisition Sequence and Acquisition Process in Section 1 for more information. At the end of each acquisition process an operation complete (OPC) SRQ is generated.
ACQuire	TIME:	<NRx>	Sets the time window duration. Limits: 5 ns to 100 µs Factory setting: 1E−6 (1 µs) Example: `ACQUIRE TIME:10E-9`
ACQuire?			Queries for all the acquisition settings: STATe, HLDNxt, TIMe, LENgth, MODe, NRECord, STArt, and LASt Example: `ACQUIRE?` Response: `ACQUIRE STATE:STOP,HLDNXT:OFF,TIME:5.E-9LENGTH:512, MODE:ADVANCE,NRECORD:8,AVERAGEL16,START:2, LAST:2,GEOMETRY:OFF`
ACQuire?	AVErage		Queries for the number of acquisitions to average when in the Average mode. Example: `ACQUIRE? AVERAGE` Response: `ACQUIRE AVERAGE:16`
ACQuire?	GEOmetry		Queries for the state of internal geometry correction. Example: `ACQUIRE? GEOMETRY` Response: `ACQUIRE GEOMETRY ON/OFF`
ACQuire?	HLDNxt		Queries for the state (ON or OFF) of the holdnext acquisition mode. Example: `ACQUIRE? HLDNXT` Response: `ACQUIRE HLDNXT:OFF`
ACQuire?	LASt		Query only. Queries for the number of the last valid record. Only completed records are valid. Example: `ACQUIRE? LAST` Response: `ACQUIRE LAST: 6`

Table 3-3: Acquire Commands (Cont.)

Header	Link	Argument	Description
ACQuire?	LENgth		Queries for the record length in sample points. Example: `ACQUIRE? LENGTH` Response: `ACQUIRE LENGTH:1024`
ACQuire?	MODe		Queries for the acquisition mode (NORmal, ADVance or AVErage). Example: `ACQUIRE? MODE` Response: `ACQUIRE MODE:ADVANCE`
ACQuire?	NRECord		Queries for the number of records to acquire when in ADVance mode. Example: `ACQUIRE? NRECORD` Response: `ACQUIRE NRECORD:8`
ACQuire?	STARt		Queries for the first record to be filled. Example: `ACQUIRE? START` Response: `ACQUIRE START:2`
ACQuire?	STATe		Queries for the state of the acquisition process (STOp, RUN,or HLDNxt). Example: `ACQUIRE? STATE` Response: `ACQUIRE STATE:HLDNXT`
ACQuire?	TIME		Queries for the time window setting. Example: `ACQUIRE? TIME` Response: `ACQUIRE TIME:5.E-9`

Trigger Commands

Table 3-4: Trigger Commands

Header	Link	Argument	Description
ARM		INTERnal EXTernal	Selects the source of the trigger arming signal: INTERnal automatically arms at the start of each sequence; EXTernal arms at a closure to ground of the rear panel ARM INPUT connector. This function operates with later version Control boards; noncompatible versions elicit an error message. Example: `ARM INTERnal`
ARM?			Queries for the setting of the arming mode (Internal or External). Example: `ARM?`
MTRig			Set only. Immediately triggers the digitizer. Example: `MTRIG` (This command has no argument.)
TRIgger	COUPling:	AC DC (SCD1000 and SCD5000 with Option 01)	SCD1000: Sets trigger signal coupling to AC,or DC (AC attenuates signal components <1 kHz). SCD5000: AC coupling only. Factory setting: AC Example: `TRIGGER COUPLING:DC`
TRIgger	DELay:	<NRx>	Positions the time window relative to the trigger event. Trigger delay can be specified in terms of percent of the record length or seconds. See Triggering in Section 1. Limits: 0 to 9 times the record length. Factory setting: 0 Example:`TRIGGER POSITION:50` (Delay is 50%.)
TRIgger	LEVEl:	<NRx>	Sets trigger level to the value specified by <NRx>. Units specified by TYPELevel command (% or volts). See Section 1 for more information on triggering. Limits: AC Coupling: ±100% of vertical range; DC Coupling: ±50% of vertical range Factory setting: 0.0 volts Example: `TRIGGER LEVEL:25` (Trigger level is set to 25% of full-scale range if the typelevel is set to percent.)
TRIgger	MODe:	AUTo NORmal	Selects trigger mode. In AUTo mode, triggering occurs when the trigger event is detected or 360 ms after the start of acquisition sequence, whichever comes first. In Normal mode, only a proper trigger event can trigger the digitizer. Factory setting: AUTo Example: `TRIGGER MODE:NORMAL`

Table 3-4: Trigger Commands (Cont.)

Header	Link	Argument	Description
TRIgger	SLOpe:	PLUs MINUs	Sets trigger slope to positive (PLUs) or negative (MINUs) edge triggering Factory setting: PLUs Example: `TRIGGER SLOPE:MINUS`
TRIgger	SOUrce:	SCD1000: CHA CHB ADD EXTernal SCD5000 INTernal CALIBRATOR SCD5000 with Option 01 INTernal EXTernal	Sets trigger source to channel A, B, the algebraic sum of A and B,or the external input. SCD5000: Sets the trigger source to the external input or the internal time calibrator signal. SCD5000 Option 01 sets the trigger source to the external input or the internal trigger pickoff. Factory setting: SCD1000: CHA; SCD5000: EXTernal Example: `TRIGGER SOURCE:CHB`
TRIgger	TYPEDelay	PERCent SECond	Sets the unit of trigger delay to percent or seconds. Factory setting: SECond Example: `TRIGGER TYPEDELAY:PERCENT`
TRIgger	TYPELevel:	PERCent VOLts	Sets the units of trigger level to percent of full scale range or to volts. PERCent allowed only for internal trigger sources. External source forces TYPELevel to VOLts. See also TRIgger LEVEl command and Triggering in Section 1. Factory setting: VOLTs Example: `TRIGGER TYPELEVEL:PERCENT`
TRIgger?			Queries for all trigger settings: SOUrce, TYPELevel, LEVEl, TYPEDelay, DELlay, SLOpe, COUPling, and MODe. Example: `TRIGGER?` Response: `TRIGGER SOURCE:CHB,TYPELEVEL:PERCENT,LEVEL:40, TYPEDELAY:PERCENT,DELAY:50,SLOPE:PLUS, COUPLING:AC,MODE:AUTO`
TRIgger?	COUPling		Queries for the setting of the trigger coupling. Example: `TRIGGER? COUPLING` Response: `TRIGGER COUPLING:AC`
TRIgger?	DELay		Queries for the setting of the trigger delay. Example: `TRIGGER? DELAY` Response: `TRIGGER DELAY:150` (Delay is 150 percent.)
TRIgger?	LEVEl		Queries for the setting of the trigger level. Example: `TRIGGER? LEVEL` Response: `TRIGGER LEVEL:125.E-3`

Table 3-4: Trigger Commands (Cont.)

Header	Link	Argument	Description
TRIgger?	MODe		Queries for the setting of the trigger mode (AUTo or NORmal). Example: `TRIGGER? MODE`
TRIgger?	SLOpe		Queries for the setting of the trigger slope (PLUs or MINUs). Example: `TRIGGER? SLOPE` Response: `TRIGGER SLOPE:MINUS`
TRIgger?	SOUrce		Queries for the setting of trigger source: SCD1000: CHA, CHB, ADD, or EXTernal: SCD5000: EXTernal, INTernal Option 01), or CALIBRATOR. Example: `TRIGGER? SOURCE` Response: `TRIGGER SOURCE:EXTERNAL`
TRIgger?	TYPEDelay		Queries for unit of the trigger delay (PERCent or SECond). Example: `TRIGGER? TYPEDELAY` Response: `TRIGGER TYPEDELAY:SECOND`
TRIgger?	TYPELevel		Queries for unit of the trigger level (PERCent or VOLts). Example: `TRIGGER? TYPELEVEL` Response: `TRIGGER TYPELEVEL:PERCENT`

Waveform Preamble Commands

The waveform preamble contains scaling, encoding and other information that the controller can use to reconstruct the waveform from the data. Some of the data in the query are values set by the DATA command parameters and other commands. See Table 3-6 for DATA commands.

Table 3-5: Waveform Preamble Commands

Header	Link	Argument	Description
WFMpre?			Query only. Queries for all WFMpre data. Example: `WFMPRE?` Response: `WFMPRE WFID:"CH1 7 89-09-22 07:24:33 25",ENCDG:BINARY,NR.PT:512,PT.FMT:Y,XINCR: 5.E-9,PTOFF:64,XZERO:0,XUNIT:SECONDS, YMULT:3.91.E-3,YZERO:100.E-3,YOFF:127, YUNIT:VOLTS,BYT/NR:1,BN.FMT:RI,BIT/NR:8, CRVCHK:NULL`
WFMpre?	BIT/nr		Query only. Queries for the number of bits per binary waveform data point (the sample point). The range is 11 to 14 bits. Example: `WFMPRE? BIT/NR` Response: `WFMPRE BIT/NR:11`
WFMpre?	BN.fmt		Query only. Queries for the Tek Codes & Formats binary number format. This value is always RI (right justified). Example: `WFMPRE? BN.FMT` Response: `WFMPRE BN.FMT:RI`
WFMpre?	BYT/nr		Query only. Queries for the number of bytes per binary waveform data point (sample point). This value is always 2 bytes per sample point. Example: `WFMPRE? BYT/NR` Response: `WFMPRE BYT/NR:2`
WFMpre?	BYT.or		Query only. Queries the order of curve data byte order. The first byte transmitted of the curve data word is identified using this query. Example: `WFMPRE? BYT.OR` Response: `WFMPRE BYT.OR:MSB`
WFMpre?	CRVchk		Query only. Queries for the checksum (NONE, NULL, or CHKSM0) that is appended to the binary waveform data stream. NONE is returned when WFTx is IL. NULL is returned when WFTx is DL. CHKSM0 is returned when WFTx is TCF. See WFTx command on page 3-12. Example: `WFMPRE? CRVCHK` Response: `WFMPRE CRVCHK:NULL`

Table 3-5: Waveform Preamble Commands (Cont.)

Header	Link	Argument	Description
WFMpre?	ENCdg		Query only. Queries for the encoding of the binary waveform data stream sent from the digitizer. This value is always BINARY. Example: `WFMPRE? ENCDG` Response: `WFMPRE ENCDG:BINARY`
WFMpre?	NR.pt		Query only. Queries for the number of points in the waveform to be transmitted. This value is set by the DATA COUNt and DATA CNTrecord commands (see page 3-13). Response is a signed integer. Example: `WFMPRE? NR.PT` Response: `WFMPRE NR.PT:512`
WFMpre?	PT.Fmt		Query only. Queries for the point format of the binary waveform data. This value is always "Y" meaning that the byte defines the amplitude of the waveform at each sample interval. Example: `WFMPRE? PT.FMT` Response: `WFMPRE PT.FMT:Y`
WFMpre?	PT.Off		Query only. Queries for the number of sample points between the trigger point and the first point being transmitted. This value is affected by the DATA STARt and TRIGGER POSITION commands. Example: `WFMPRE? PT.OFF` Response: `WFMPRE PT.OFF:127`
WFMpre?	WFId		Query only. Queries for the waveform identification string. The response is a quoted string indicating the channel number, record number, date and time of acquisition, and the number of missing data points on the centroided waveform ("Ch# REC# date time xx"). If more than one record is being sent,only the beginning record number is indicated. Example: `WFMPRE? WFID` Response: `WFMPRE WFID:"CHA 4 89-12-15 23:14:22.62 54"`
WFMpre?	XINcr		Query only. Queries for the sample interval of the waveform. The response is a floating point number with an exponent. This value is set by the ACQUire TIME command. Example: `WFMPRE? XINCR` Response: `WFMPRE XINCR:50.E-9`
WFMpre?	XUNit		Query only. Queries for the horizontal unit of measure for the waveform. This value is always SEConds. Example: `WFMPRE? XUNIT` Response: `WFMPRE XUNIT:"SECONDS"`

Table 3-5: Waveform Preamble Commands (Cont.)

Header	Link	Argument	Description
WFMpre?	XZEro		Query only. Queries for the horizontal offset of the waveform data. The response is <NR3> and is always zero. Example: `WFMPRE? XZERO` Response: `WFMPRE XZERO:0E-3`
WFMpre?	YMUl		Query only. Queries for the vertical scale factor (multiplier in volts) of the waveform data. This number is any of the full-scale vertical range settings divided by 512 or 2048 (the current vertical resolution). The response is a floating point number with an exponent. Example: `WFMPRE? YMULT` Response: `WFMPRE YMULT:3.91.E-3`
WFMpre?	YOFf		Query only. Queries for the center value of the waveform data. Example: `WFMPRE? YOFF` Response: `WFMPRE YOFF:1024`
WFMpre?	YUNit		Query only. Queries for the vertical unit of measure for the waveform. Example: `WFMPRE? YUNIT` Response: `WFMPRE YUNIT:"VOLTS"`
WFMpre?	YZEro		Query only. Queries for the vertical offset of the waveform. This value is set by the CH<x> OFFSET command. The response is a floating point number with an exponent. Example: `WFMPRE? YZERO` Response: `WFMPRE YZERO:100.E-3`
WFTx		DL IL TCF	Sets the waveform transfer format. DL = Definite Length Binary Block. IL = Indefinite Length Binary Block. TCF = Tek Codes and Formats. See Section 5 for more information on transfer formats. Factory setting: DL Example: `WFTX IL`
WFTx?			Queries for the waveform transfer format (DL, IL, or TCF). Example: `WFTX?` Response: `WFTX TCF`

Data and Waveform Commands

The CURVE? query causes the SCD to transmit waveform data to the controller. The amount of data that is sent is defined by the DATA command parameters (DATA statement). See Table 3-5 earlier in this section.

Table 3-6: Data and Waveform Commands

Header	Link	Argument	Description
CURVe?			Query only. Sends the SCD's binary waveform data to the GPIB port. The data sent is specified by the DATA command parameters. Example: `CURVE?` The response to CURVe? depends on the transmission format (WFTx commands) and the DATA statement. See page 3-12 for the WFTx command. ***NOTE:*** *Data which was interpolated rather than recorded is flagged by setting the 15th bit in the data word (Data would be of the pattern: 4xxxH).*

DATA	BYTEOrder:	MSB
		LSB
			MSB selects the most significant byte of the data portion to be transmitted first.
			LSB selects the least significant byte of the data portion to be transmitted first
			Example:
			`DATA BYTEORDER: MSB`
			`DATA? BYTEORDER`

DATA	CNTrecord:	<NRx>	Sets the number of records to be transferred. The first record is set by the DATA STREcord command. Limits: 1 to 16 Factory setting: 1 Example: `DATA CNTRECORD:4`
DATA	COUNt:	<NRx>	Sets the number of points in the curve to be transferred (the starting point is included in the transfer). If COUNt:0 is specified,the entire record is transmitted. Limits: 0 to record length Factory setting: 0 Example: `DATA COUNT:128`

DATA	FLAGbit:	ON
		OFF
			ON selects the interpolation flag bit (bit 15) to be enabled on all curve data that was generated by interpolation.
			OFF selects masking of the interpolation flag bit (bit 15) on all curve data output.
			Example:
			`DATA FLAGBIT:ON`
			`DATA? FLAGBIT`

DATA	STARt:	<NRx>	Sets the starting point in the selected record where transferred waveform data starts. Limits: 1 to record length. Factory setting: 1 Example: `DATA START:64`

Table 3-6: Data and Waveform Commands (Cont.)

Header	Link	Argument	Description
DATA	STREcord:	<NRx>	Selects the first record to be transferred. Limits: 1 to 16 Factory setting: 1 Example: `DATA STRECORD:10`
DATA?			Queries for all settings of the data command for all channels: CNTrecord, COUNt, STARt and STREcord. Example: `DATA?` Response: `DATA CNTRECORD:0,COUNT:1024,START:1,STRECORD:1`
DATA?	BYTEOrder		Queries for the data transmission byte order. Example: `DATA? BYTEORDER` Response: `DATA BYTEORDER:MSB`
DATA?	CNTrecord		Queries for the number of records to include in a waveform transfer. Example: `DATA? CNTRECORD` Response: `DATA CNTRECORD:16`
DATA?	COUNt		Queries for the number of points to include in a waveform data transfer. Example: `DATA? COUNT` Response: `DATA COUNT:1024`
DATA?	FLAGbit		Queries the status of the interpolation flag bit (bit 15) for curve data output. Example: `DATA? FLAGBIT` Response: `DATA? FLAGBIT:OFF`
DATA?	STARt		Queries for the data transfer starting point in the selected record. Example: `DATA? START` Response: `DATA START:64`
DATA?	STREcord		Queries for first record to include in a waveform transfer. Example: `DATA? STRECORD` Response: `DATA STRECORD:1`
GEOMArray?			Query only. Sends the vertical geometry correction array data to the GPIB port only. Only DL format is allowed and the byte order is always MSB first. The WFTx setting is ignored (see WFTx command on page 3-12). See section **?** for more information. Example: `GEOMARRAY?` Response: `GEOMARRAY #3581 <data>... {null checksum}`

Table 3-6: Data and Waveform Commands (Cont.)

Header	Link	Argument	Description
INTERPolate?		<NRx>	Query only. Queries for the number of interpolated data points of the specified record. Limits: 1 to 16 Example: `INTERPOLATE? 1:12`
INTERPolate?			Query only. Queries for the number of interpolated data points of all records. Example: `INTERPOLATE? 1:12, 2:512, 3:1024 {etc... to 16 recs}`
LINArray?			Query only. Sends the raw, uncentroided waveform data to the GPIB port. Only DL format is allowed. The WFTx setting is ignored (see WFTx command on page 3-12). Refer to Figure 5-1 for data format. Example: `LINARRAY?` Response: `LINARRAY #517584 <data>...{null checksum}`
REFArray?			Query only. Sends the blanking pixel map of the target to the GPIB. This array is used in the digitizer to remove target irregularities. Only DL format is allowed. The WFTx setting is ignored (see WFTx command on page 3-12). Refer to Figure 5-1 for data format. Example: `REFARRAY?` Response: `REFARRAY #6524289 <data>...{null checksum}`
REPEat?			Query Only. Starts repeat cycle set up by the REPSet command. The SCD repeatedly (according to values set by REPSet command) acquires and then transmits binary waveform data to the controller. (Any command from the bus or front panel aborts the REPEat process.) The data sent is specified by the DATA command parameters. Example: `REPEAT?` The response to this command is the waveform data acquired and transmitted to the controller. Transmission format is specified by the WFTX command (see page 3-12).
REPSet	NREPEat:	<NRx>	Sets the SCD to execute repeat mode acquisition and transfer mode. In this mode, the SCD is set to capture the number of records defined using NRECORD command and automatically transfer the waveform data to a waiting GPIB controller. The controller does not need to query for the waveform data transfer using the CURVe? query. If <NRx> = 0, the process is repeated indefinitely until the SCD is listen addressed with any command. At that time, the process is terminated. Limits: 0 to $(2^{32}-1)$ Factory setting: 1 Example: `REPSET NREPEAT:10`

Table 3-6: Data and Waveform Commands (Cont.)

Header	Link	Argument	Description
REPSet?			Queries for the number of times to execute the repeat cycles. Example: `REPSET?` Response: `REPSET NREPEAT:10`
REPSet?	NREPEat		Queries for the number of times to execute the repeat cycle. Example: `REPSET? NREPEAT` Response: `REPSET NREPEAT:12`
TIMESt?			Query only. Queries for the time stamps of all acquired records. The value of the time stamp is the date and time of each trigger event, in yy-mm-dd hh:mm:ss.ss format. Example: `TIMEST?` Response: `TIMEST? 1:"89-12-15 12:42:53.42:,2:"89-12-15 13:22:54.22", 3:"89-12-15 16:22:33.23", 4:"89-12-15 18:22:34.76", 5:"...."... to 16 records.`
TIMESt?	<ui>		Query only. Queries for the date/time stamp of the specified record. The value of the time stamp is the date and time of the trigger event in yy-mm-dd hh:mm:ss.ss format. Example: `TIMEST? 12` Response: `TIMEST 12:"89-12-15 12:42:33.61"`
WAVfrm?			Query only. Sends the waveform preamble and waveform data to the GPIB port. The transmitted data depends on the DATA statement and the WFTx command. This query is equivalent to sending a WFMpre? query followed by a CURVe? query. See Table 3-5 for the WFMpre? query and WFTx commands. Example: `WAVFRM?` Response: See WFMpre? query in Table 3-5 and CURVe? query at the beginning of this table.

Status and Event Commands

The following commands are used to control and report the details of operating status to the controller. Details on status bytes and events codes can be found in Section 4.

Table 3-7: Status and Event Commands

Header	Link	Argument	Description
ALLEv?			Query only. Queries the SCD for all event codes. If the LONgform command is ON, the event code is followed by a quoted string describing the event code. If OFF, only the event code is sent. To return all event codes, either the RQS OFF command must first be sent or all events must have been serially polled. If RQS is ON, only the most recent event code will be returned followed by an error. See RQS command below. See also Section 4 for more information on the event codes. Example: `ALLEV?` Response: `ALLEV 156, "Unknown symbol", 157, "Syntax error"` (RQS is OFF.)
EVEnt?			Query only. Queries the SCD for the most recent event code. If the LONgform command is set to ON, the event code is followed by a quoted string describing the event code. If LONgform is OFF, only the event code is sent. If RQS is ON, an event code will be returned only if a serial poll has been performed before requesting the event code. If RQS is OFF, a serial poll is not necessary prior to requesting the event code. However, corresponding status bytes are lost when consecutive event codes are requested. See the RQS command later in this table. See also Section 4 for information on status bytes and event codes. Example: `EVENT?` Response: `EVENT 455,"Self test completed successfully"`
EVQty			Query only. Queries the SCD for how many events are waiting to be queried (up to a maximum of 20). The response is a signed integer. Example: `EVQTY?` Response: `EVQTY 3`

FPStat?			Query only. Queries for all 20 Front Panel status messages.
			Example: `FPSTAT?`
			Response: `FPSTAT 1: "Power on initialization complete",1:"",`
			`2:"Self test completed successfully",2:"", ...`
			`{continue for all 20 status msgs}`

Table 3-7: Status and Event Commands (Cont.)

Header	Link	Argument	Description
FPStat?		<NRx>	Query only. Queries for one of the 20 Front Panel status messages that are buffered within the system. Status messages may occupy one or two lines on the display. Each of the status lines are output as separate strings using this query. Limits: 1 to 20 Example: `FPSTAT? 1` Response: `FPSTAT 1: "Power on initialization complete?,1:""`
ID?			Query only. Queries the SCD for information about the digitizer. The response includes the firmware version numbers for the digitizer and Display Unit. Example: `ID?` Response: `ID "TEK/SCD1000,V81.1,DIG/1.0, DSY/3.00`
OPTion?			Query only. Queries for the options installed in the digitizer. Possibilities are OPT 1E: TEK type II probe OPT 2E: SMA input connectors OPT 1P: Fast wfm capture OPT 2F: HSDO & battery backed-up linear array OPT 01: Delay line (SCD5000 only) Example: `OPTION?` Response: `OPTION "01 Delay line"`
RQS		ON OFF	Enables or disables the digitizer's ability to assert the SRQ line when an event occurs or a condition changes. ON enables; OFF disables. When OFF, the digitizer does not request service from the controller. The controller must poll the digitizer to determine if it needs to be serviced. See also SRQMask command in this table and Section 4 for more details. Factory setting: ON Example: `RQS OFF`
RQS?			Queries for the RQS status. Example: `RQS` Response: `RQS ON`
SRQmask	ABSTouch:	ON OFF	Enables (ON) or disables (OFF) SRQ assertion as the result of a front panel change, either by the ABSTouch command or button pushes. See also the ABSTouch command in Table 3-8. Factory setting: OFF Example: `SRQMASK ABSTOUCH:OFF`

Table 3-7: Status and Event Commands (Cont.)

Header	Link	Argument	Description
SRQmask	CMDerr: EXERr: EXWarn: INErr: INWarn: OPCmpl: USR1: USR2:	ON OFF	Enables or disables the ability of the SCD to assert SRQ on each of these status conditions. See Section 4 for more information on errors, warnings, user, and operation complete conditions. Factory settings: CMDerr: ON,EXErr: ON,EXWarn: ON, INERr:ON,INWarn: ON,OPCmpl: ON,USR1: OFF,USR2: OFF Example: `SRQMASK CMDERR:ON,EXERR:ON,EXWARN:ON, INERR:ON,INWARN:ON,OPCMPL:ON,USR1:OFF,USR2:OFF`
SRQmask?			Queries for the states (ON or OFF) of all SRQ masks. See Section 4 for more information on errors, warnings, user, and operation complete conditions. Example: `SRQMASK?` Response: `SRQMASK ABSTOUCH:OFF,CMDERR:ON, EXERR:ON,XWARN:ON,INERR:ON,INWARN:ON, OPCMPL:ON,USR1:OFF,USR2:OFF`
SRQmask?	ABSTouch CMDerr EXERr EXWarn INErr INWarn OPCmpl USR1 USR2		Queries for the states (ON or OFF) of the individual SRQMask settings. See Section 4 for more information on errors, warnings, user, and operation complete conditions. Example: `SRQMASK? ABSTOUCH` Response: `SRQMASK ABSTOUCH:ON.`
UID		<qstring>	Assigns an identifying name to the device. The name must be a quoted string no longer than ten characters (for example, "MSTRSCD"). Set at factory to instrument serial number. Example: `UID "DIG_2"`
UID?			Queries for the instrument's ID string. The response is a ten character quoted string. Example: `UID?` Response: `UID "B010101"`
VERSion?			Query only. SCD sends firmware version number. Response is <digitizer firmware version #>,<display firmware version #>. Example: `VERSION?` Response: `VERSION "DIG FW#1.0 DSY FW#3.00"`

GPIB Related Commands

Table 3-8: GPIB Related Commands

Header	Link	Argument	Description
ABSTouch		<NRx>,<NRx>	Emulates a touch to a front-panel button or a "click" of the variable knob designated by the coordinates <NRx> and <NRx>. Figure 3-1 identifies the coordinates possible with the SCD Display Unit. These coordinates are stored in a buffer (ABSTouch buffer). The last 20 ABSTouch commands and front panel button presses are stored. ABSTouch coordinates from the front panel are not stored if RWLS is active. Factory setting: None Example: `ABSTOUCH 0,8`
		CLEar	Clears the coordinates in the ABSTouch buffer. The ABSTouch buffer is always cleared at power-up. Example: `ABSTOUCH CLEAR`
ABSTouch?			Queries for the coordinates in the ABSTouch buffer. The response is two signed integers indicating the touch area. See Figure 3-1 for touch area coordinates. If the ABSTouch buffer is empty, the response is −1,−1. Example: `ABSTOUCH?` Response: `ABSTOUCH 2,8`

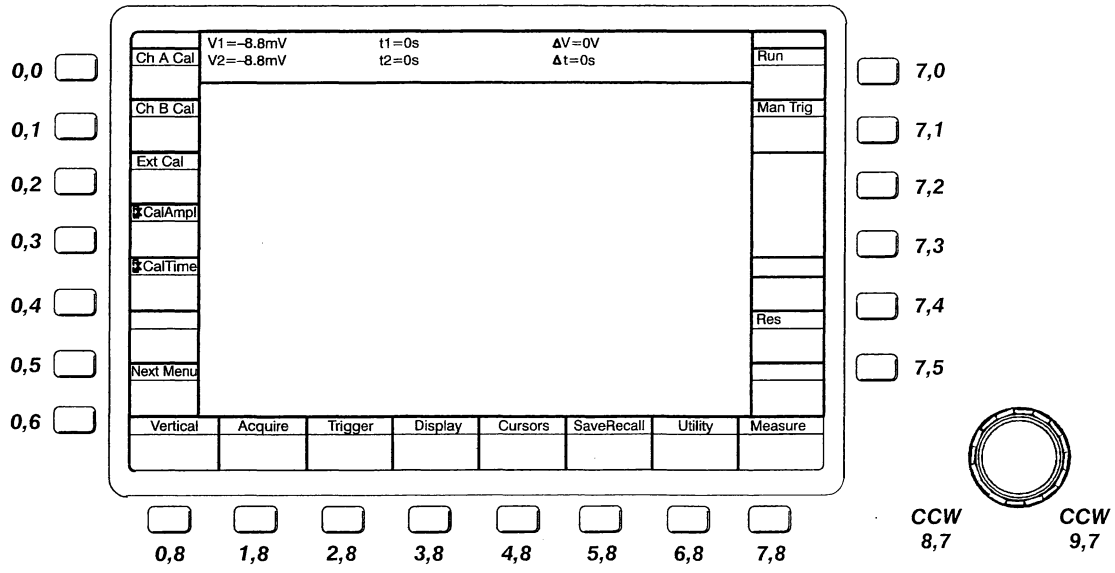


Figure 3-1: Front Panel Coordinates for ABSTouch

Table 3-8: GPIB Related Commands (Cont.)

Header	Link	Argument	Description
DEBug	GPIb:	ON OFF	Sets the state of GPIB debugging. When ON, the IEEE−488.1 bus traffic to the SCD is displayed on the Display Unit. See Debug Mode in Section 1. Factory setting: OFF Example: `DEBUG GPIB:ON`
DEBug?	GPIb		Queries for the state (OFF or ON) of GPIB debugging. Example: `DEBUG? GPIB` Response: `DEBUG ON`
DT		RUN STOp HLDNxt OFF	Sets the acquisition state the SCD enters when the next GET is received. See Table 3-3 (ACQUire STATe command) for more information on these settings Factory setting: OFF Example: `DT RUN`
DT?			Queries for the group execute trigger acquisition state. Example: DT? Response: `DT RUN`
HELp?			Queries for list of all SCD command headers. Example: `HELP?`
INIt		PANel GPIb [ALL]	Set only. Resets settings to factory settings for instrument (PANel), IEEE−488.1 (GPIB), or both (ALL). See Tables 3-20 and 3-21 for factory settings of PANel and GPIb. An OPC SRQ is generated when PANel or GPIb is sent. Example: `INIT ALL`
LONgform		ON OFF	Controls number of characters reported to controller. When ON, complete spelling of headers, links, and arguments are reported. (Affected by PATH command.) Responses to EVENT? and ALLEV? queries include event numbers and quoted string description of event. number(s). When OFF, headers, links, and arguments are abbreviated to minimum ambiguity. Responses to EVENT? and ALLEV? queries are limited to event number(s). Factory setting: ON Example: `LONGFORM ON`
LONgform?			Queries for the setting (OFF or ON) of the LONgform command. Example: `LONGFORM?` Response: `LONGFORM ON`

Table 3-8: GPIB Related Commands (Cont.)

Header	Link	Argument	Description
PATh		ON OFF	Sets the type of response to queries. When ON, the header and link are returned with the argument (for example, CHA OFFSET:10). When OFF, only the arguments are returned to the query (for example,10). Factory setting: ON Example: `PATH OFF`
PATh?			Queries for the state (ON or OFF) of the path setting. Example: `PATH?` Response: `PATH ON`
USER1 USER2		<qstring1>, <qstring2>	Labels the Display Unit's front panel USER buttons with <qstring1> on line 1 and <qstring2> on line 2. The strings can be up to eight characters each. Factory setting: `"","` Example: `USER1"Group","Trig"`
USER1? USER2?			Queries the specified front panel USER button for its labels. The response is a set of two quoted strings (<qstring>) containing the labels for each line. Pressing either user button causes a USER SRQ to be generated if the USER SRQ is unmasked. Example: `USER1?` Response: `USER1 "Send","Wavefrm"`

Cursor Commands

Table 3-9: Cursor Commands

Header	Link	Argument	Description
CRS?			Queries for all settings of both cursors. Example: `CRS?` Response: `CRS1LOCTN:WIN2,XPOINT:123,XTIME:65.E-9, YCOORD:2.57.E-6;CRS2LOCTN:WIN2,XPOINT:158, XTIME:83.E-9,YCOORD:1.07.E-6`
CRS1 CRS2	LOCTn:	WIN<ui>	Places the specified cursor in the window specified by <ui>. Limits: <ui> can be from 1 to 4. Factory setting: Cursor 1: Window 1; Cursor 2: Window 1 Example: `CRS1 LOCTN:WIN1`
CRS1 CRS2	XPOint:	<NRx>	Places the specified cursor at sample point location <NRx>. Limits: <NRx> can be from 0 to (Record Length − 1) Factory setting: Cursor 1: point 0; Cursor 2: point 0 Example: `CRS1 XPOINT:127`
CRS1? CRS2?			Queries for the settings of each of the above cursor commands: LOCTn, XPOint, XTIMe, and YCOord. Example: `CRS1?` Response: `CRS1 LOCTN:WIN2,XPOINT:123, XTIME:65.E-9,YCOORD:2.57.E-6`
CRS1? CRS2?	LOCTn		Queries for the window number in which the specified cursor is located. Example: `CRS2? LOCTN` Response: `CRS2 LOCTN:WIN4`
CRS1? CRS2?	XTIMe		Query only. Queries for the time in seconds of the selected cursor. The response is a floating point number with an exponent. Example: `CRS1? XTIME` Response: `CRS1 XTIME:112.E-3`
CRS1? CRS2?	XPOint		Queries for the sample point of the specified cursor. The response is an integer. Example: `CRS2? XPOINT` Response: `CRS2 XPOINT:365`
CRS1? CRS2?	YCOord		Query only. Queries for the amplitude (in volts) of the specified cursor. The response is a floating point number with an exponent. Example: `CRS1? YCOORD` Response: `CRS1 YCOORD:65.E-2`

Table 3-9: Cursor Commands (Cont.)

Header	Link	Argument	Description
CRSD	TYPETime:	HZ SECond	Sets the unit of Δt measurements to hertz or seconds. Measurements between cursor 1 and cursor 2 are displayed in the selected unit. Factory setting: SECond Example: `CRSD TYPETIME:SECOND`
CRSD?			Queries for the typetime value and the Δt and Δy measurement values. Example: `CRSD` Response: `CRSD TYPETIME:SECOND,T:14.3.E-9, Y:1.54.E-6`
CRSD?	T		Query only. Queries for the Δt measurement value in seconds. Δt is the time difference between the locations of cursor 2 and cursor 1. The response is a floating point number with an exponent. Example: `CRSD? T` Response: `CRSD T:122.E-9`
CRSD?	TYPETime		Queries for the units of Δt measurements. Example: `CRSD? TYPETIME` Response: `CRSD TYPETIME:HZ`
CRSD?	Y		Query only. Queries for the Δy measurement value in volts. Δy is the voltage difference between the locations of cursor 2 and cursor 1. The response is a floating point number with an exponent. Example: `CRSD? Y` Response: `CRSD Y:12.5.E-3`
CURSors		ON OFF	Turns ON or OFF the display of the cursors. Factory setting: ON Example: `CURSORS ON`
CURSors?			Query for the state (ON or OFF) of the cursor display. Example: `CURSORS?` Response: `CURSORS ON`
GRATicule		ON OFF	Turns graticule waveform display mode on or off. This command applies to all waveform display windows. Factory setting: OFF Example: `GRATICULE ON`
GRATicule?			Queries for the state of the graticule display mode. Example: `GRATIcule.` Response: `GRATICULE ON`

Save/Recall Settings Commands

Table 3-10: Save/Recall Commands

Header	Link	Argument	Description
ERAse		<NRx>	Clears the saved settings located in <NRx>. Factory setting: None Example: `ERASE 2`
LLSet?			Queries for the contents of the LLSet binary block. The format is <&><data><&><data><&><data>...<EOI>. The LLSet binary block contains all of the digitizer settings. The user may store the results of an LLSet? query and return them to the digitizer at a later time to restore the digitizer to the state it was in when the LLSet? query was executed.
		<bblock>	This command contains no header. It's argument is a binary block which was generated by an LLSet? query and stored external to the instrument. This binary block specifies the settings for the instrument. LLSet sets up the digitizer quicker than using an ASCII string generated by querying all settings and resending it. However, the settings should also be saved as an ASCII string because the <bblock> may become obsolete if the SCD firmware version is changed.
RECAll		<NRx>	Recalls the instrument settings stored in a nonvolatile RAM area specified by <NRx>. (The settings are stored in nonvolatile RAM using the SAVE command as described below.) An OPC SRQ is generated upon completion. Limits: <NRx> ranges from 1 to 10. Factory setting: None Example: `RECALL 1`
SAVE		<NRx>	Stores the current instrument settings in the location in nonvolatile memory specified by <NRx>. Settings can be recalled using the RECAll command. An OPC SRQ is generated upon completion. Limits: <NRx> ranges from 1 to 10. Factory setting: None Example: `SAVE 1`
SET?			Queries for all current instrument settings. Example: `SET?` Response: `H1 RANGE:1,OFFSET:10.....`

Instrument/Data Protection Commands

Table 3-11: Instrument/Data Protection Commands

Header	Link	Argument	Description
SAFEguard	ACQProtect:	ON OFF	When ON, the digitizer will not initiate an acquisition or a function that performs acquisitions such as calibration, setref, and diagnostics from front panel input. GPIB command input is uneffected by this control. When OFF, the front panel has it's normal operational control. Factory setting: OFF Example: SAFEGUARD ACQPROTECT: OFF
SAFEguard	LINConvert:	<NRx>	Set only. Centroid the current linear array (HSDO array if installed). The current instrument setup is used to determine the record to place the centroided results and the generation of the waveform preamble. Record scaling, record placement and geometry correction may all be controlled by adjusting the instrument setting prior to running this command. Since no trigger event is associated with the centroided record the timestamp is marked "TIME UNKNOWN". This command when used with SAFEguard STOPAcq will allow you to generate a centroided and corrected curve from the battery backed up Option 2F Linear Array. If the instrument power was inadvertently removed too soon after a trigger event for the instrument to completely process the waveform. Limits: 1 to 16 Example: SAFEGUARD LINCONVERT: 2
SAFEguard	PROTect: (SCD1000 only)	ON OFF	When ON, the digitizer automatically disconnects the inputs when it detects an overvoltage condition. When OFF, the inputs are not protected. However, a service request can be issued if an overvoltage condition is detected. ***NOTE*** *Operation of the instrument with Protect = OFF will damage the instrument and void the warranty if an input overvoltage condition occurs.* Factory setting: ON Example: SAFEGUARD PROTECT:OFF
SAFEguard	PUPtst:	ON OFF	When ON and instrument switch #8 is OFF, the digitizers executes a self-test at power-up. When OFF and instrument switch #8 is OFF, the digitizer does not execute a self-test. Factory setting: ON. Example: SAFEGUARD PUPTST:OFF

Table 3-11: Instrument/Data Protection Commands (Cont.)

Header	Link	Argument	Description
SAFEguard	SECURE		Set only. Erases all memory data and resets all parameters to their factory settings.Diagnostics are run (if enabled). POWER ON and SETTINGS LOST SRQs are generated. Example: `SAFEGUARD SECURE`
SAFEguard	STOPAcq		Set only. Causes the Acquisition machine of the instrument to immediately move to the acquire stop state from the acquire holdnext or acquire run states. If no trigger is received prior to this command the curve data and timestamp of the previous acquisition will be preserved. The waveform preamble data for the curve that remains will be lost. Example: `SAFEGUARD STOPACQ`
SAFEguard?			Queries for state of all SAFEguard settings. Example: `SAFEGUARD?` Response: `SAFEGUARD PROTECT:ON,PUPTST:OFF`
SAFEguard?	ACQProtect		Queries for the acquire protect setting (ON or OFF). Example: `SAFEGUARD? ACQPROTECT` Response: `SAFEGUARD ACQPROTECT:OFF`
SAFEguard?	PROTect (SCD1000 only)		Queries for the input protection setting (OFF or ON). Example: `SAFEGUARD? PROTECT` Response: `SAFEGUARD PROTECT:ON`
SAFEguard?	PUPtst		Queries for the power-up self-test setting (OFF or ON). Example: `SAFEGUARD? PUPTST` Response: `SAFEGUARD PUPTST`

Display Commands

Table 3-12: Display Commands

Header	Link	Argument	Description
DISplay		ON OFF	Turns ON or OFF the Display Unit. (The SCD executes faster when the display is OFF.) Factory setting: ON Example: `DISPLAY OFF`
DISplay?			Queries for the state of the Display Unit. Example: `DISPLAY?` Response: `DISPLAY OFF`
HEXPMd		ALIgned INDep	Selects whether expansion occurs for all windows at the same time by the same expansion setting (ALIgned) or for only the selected window (INDep). Factory setting: ALIGNED Example: `HEXPMD INDEP`
HEXPMd?			Queries for the horizontal expansion mode (ALIGNED or INDEP). Example: `HEXPMD?` Response: `HEXPMD INDEP`
NWIn		1 2 4	Selects the number of windows for displaying waveforms. 1, 2,or 4 windows can be displayed at the same time. Factory setting: 1 Example: `NWIN 4`
NWIn?			Queries for the number of displayed windows (1, 2, or 4). Example: `NWIN?` Response: `NWIN 4`
TEXt	CHAR:	<NRx>	Set only. Specifies the starting column for the TEXT STRIng command. Limits: 1 to 64 Factory setting: 1 Example: `TEXT CHAR:10`
TEXt	CLEar:	<NRx>	Set only. Clears the text on line <NRx> of the display. If <NRx> is 0, all lines are cleared. Factory setting: None Example: `TEXT CLEAR:2`

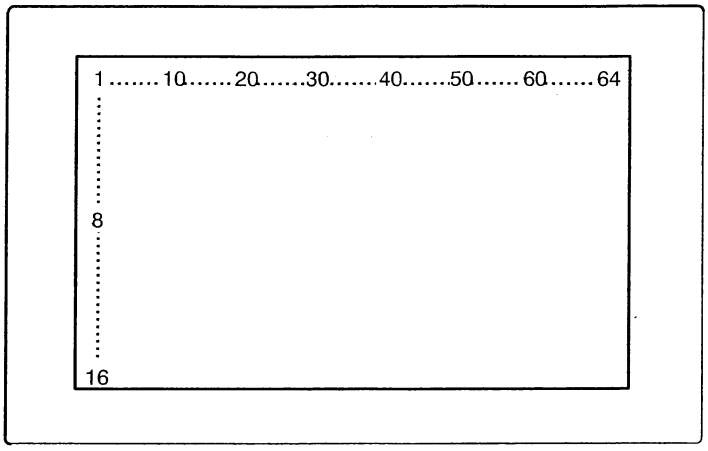
Table 3-12: Display Commands (Cont.)

Header	Link	Argument	Description
TEXt	LINE:	<NRx>	Set only. Specifies the starting line number for the TEXT STRIng command. Record bar lines cannot be used for displaying text. Limits: 1 to 16. Rows are divided evenly among displayed windows. (See Figure 3-2.) Factory setting: 1 Example: `TEXT LINE:6`
TEXt	STRIng:	<qstring>	Set only. Writes the text string (<qstring>) at the location specified by TEXT CHAR and TEXT LINE commands. Text is only allowed in waveform areas. Table 3-21 lists the TEXT command character set. Record 0 may be used to display text only. Limits: 16 rows × 64 cols. Rows are divided evenly among the number of displayed windows. See Figure 3-2. Factory setting: " " Example: `TEXT STRING:"Trig position -25%."`
WIN<ui>	EXPnt:	<NRx>	Selects point of waveform around which expansion takes place. Limits: 0 to (Record Length − 1) Factory setting: 0 Example: `WIN1 EXPNT:38`
WIN<ui>	HEXPNd:	<NRx>	The horizontal expansion factor can be set to 1, 2, 4, or 8 to expand 512 or 1024 data point records. Otherwise HEXPNd equals 1, 2, or 4. Limits: 1, 2, 4, 8 Factory setting: 1 Example: `WIN2 HEXPND:2`
WIN<ui>	RECOrd:	<NRx>	Displays waveform data from record <NRx> in window <ui>. Factory setting: Win1: 1; Win2: 1; Win3:1; Win4: 1 Example: `WIN4 RECORD:24`
WIN<ui>	VEXpnd:	<NRx>	Vertically expands window <ui> by <NRx>. <NRx> can be 1, 2, or 4. If <NRx> is 1, the entire vertical range is displayed. Factory setting: 1 Example: `WIN4 VEXPND:2`

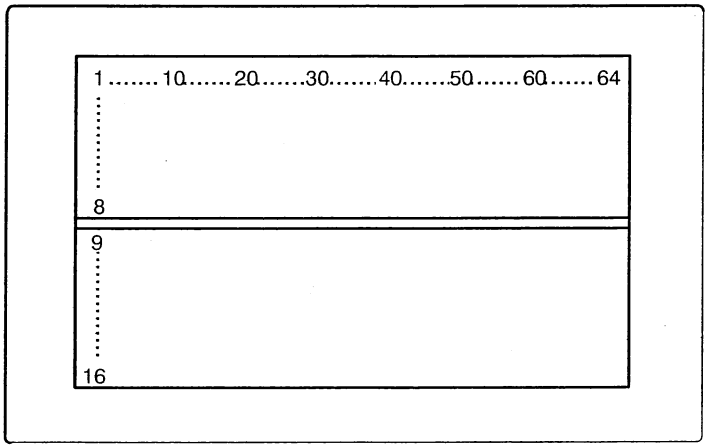
Table 3-12: Display Commands (Cont.)

Header	Link	Argument	Description

One Window



Two Windows



Four Windows

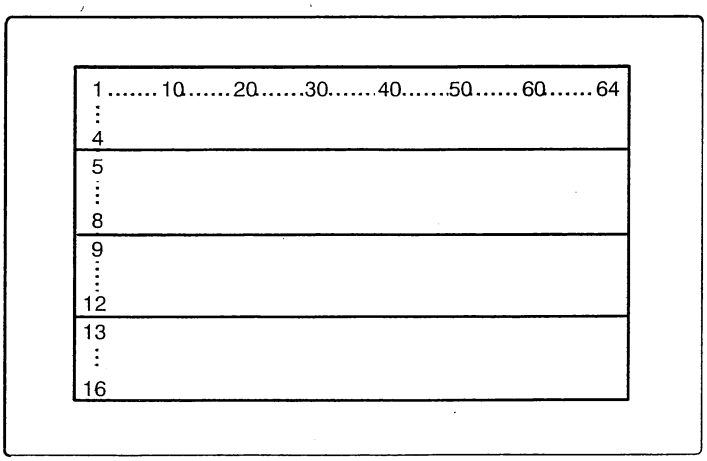


Figure 3-2: Text Areas

Table 3-12: Display Commands (Cont.)

Header	Link	Argument	Description
WIN<ui>?			Queries for all the settings (EXPnt, HEXPNd, RECOrd, VEXpnd) of window <ui>. Example: `WIN1?` Response: `WIN1EXPNT:129,HEXPND:8,RECORD:2, VEXPND:2`
WIN<x>?	EXPnt		Queries for window <ui>'s expansion point. The response is a signed integer. Example: `WIN3? EXPNT` Response: `WIN3 EXPNT:116`
WIN<x>?	HEXPNd		Queries for horizontal expansion value of window <ui>. The response is a signed integer: 1, 2, 4, or 8. Example: `WIN4? HEXPND` Response: `WIN4 HEXPND:4`
WIN<x>?	RECOrd		Queries for the record from which window <ui> displays its data. The response is a signed integer. Example: `WIN2? RECORD` Response: `WIN2 RECORD:12`
WIN<x>?	VEXpnd		Queries for the vertical expansion value of window <ui>. The response is a signed integer. Example: `WIN4? VEXPND` Response: `WIN4 VEXPND:2`
WIN?			Queries for all the settings (EXPnt, HEXPNd, RECOrd,V EXpnd) of all the windows (1, 2, 3, 4). Example: `WIN?` Response: `WIN1 EXPNT:129,HEXPND:1,RECORD:2, VEXPND:2;WIN2 EXPNT:332,HEXPND:2,RECORD:7, VEXPND:4;WIN3 EXPNT:129,HEXPND:1,RECORD:2, VEXPND:2;WIN4 EXPNT:332,HEXPND:1,RECORD:7, VEXPND:4`

Front Panel Commands

Table 3-13: Front Panel Commands

Header	Link	Argument	Description
BELl	BUTton:	ON OFF	Turns sound for button clicks ON or OFF. Factory setting: ON Example: `BELL BUTTON: OFF`
BELl	KNOb:	ON OFF	Turns sound for knob clicks ON or OFF. Factory setting: ON Example: `BELL KNOB: ON`
BELl	RINg		Sounds a beep. Example: `BELL RING`
BELl?			Queries for the settings of both the knob and button sounds. Example: `BELL?` Response: `BELL BUTTON:ON,KNOB:ON`
BELl?	BUTton		Queries for the setting (ON or OFF) of the button sound. Example: `BELL? BUTTON` Response: `BELL BUTTON:OFF`
BELl?	KNOb		Queries for the setting (ON or OFF) of the knob sound. Example: `BELL? KNOB` Response: `BELL KNOB:OFF`
FPAnel		ON OFF	When ON, front panel changes from the knob and buttons are allowed. When OFF, the front panel is locked out as in RWLS.
FPAnel?			Queries for the state (OFF or ON) of the FPANEL command. Example: `FPANEL?` Response: `FPANEL OFF`

Diagnostics and Calibration Commands

Diagnostics allow checking of several SCD subsystems. Calibration provides internal calibration of digitizer circuits. Calibration should be run whenever the values from the SCD are suspect, or whenever the operating environment of the SCD changes. Calibration should not be run within the first 20 minutes after power-up.

Table 3-14: Diagnostic and Calibration Commands

Header	Link	Argument	Description
BATdate?			Queries for the date when the NVRAM ICs were first powered. (NVRam battery life is 8–10 years). The format is yy-mm-dd. Example: `BAT?` Response: `BAT "89-09-24"`
CALIBRATE		[ALL] CRT GEOmetry HORizontal INPut VERTical TRIgger	Performs internal calibration of input circuitry (SCD1000 only). Performs internal calibration of horizontal circuitry. Performs internal calibration of vertical circuitry. Performs internal calibration of trigger circuitry. Performs internal calibration to align CRT tilt and intensity. Performs internal calibration to create the correction table to improve vertical linearity. Performs internal calibration of horizontal, vertical, and trigger circuitry. Each calibrate function generates an operation complete (OPC) SRQ when done. Example: `CAL TRIGGER`
CALIBRATE?			Queries for self-calibration status. The response is a quoted string that identifies the sections that failed or passed. Example: `CALIBRATE?` Response: `CALIBRATE "PASSED"`
CALOut	EXTernal:	AMPL TIME	Connects the time calibrator signal to the external cal connector. In the SCD1000 both channels A & B are forced to OFF. Connects the amplitude calibrator signal to the external cal connector. In the SCD1000 both channels A & B are forced to OFF. Factory setting: TIME (SCD1000), AMPL (SCD5000) Example: `CALOUT EXTERNAL:AMPL`

Table 3-14: Diagnostic and Calibration Commands (Cont.)

Header	Link	Argument	Description
CALOut	CH[A|B]:		SCD1000 only
		AMPL	Connects the 50 Ω amplitude calibrator signal to the channel [A|B] input. Channel [A|B] vertical coupling is forced to OFF. Calout External is forced to Time and the alternate calout channel is forced to OFF.
		AMPL4[50]	Connects the 50 Ω amplitude calibrator signal to the channel [A|B] input. Channel [A|B] vertical coupling is forced to OFF. Calout External is forced to Time and the alternate calout channel is forced to OFF.
		OFF	Disconnects calibrator signals from the channel [A|B] input. Factory setting: OFF Example: `CALO CHA:TIME`
		TIME	Connects the time calibrator signal to the channel [A|B] input. Channel [A|B] vertical coupling is forced to OFF. Calout External is forced to Amplitude and the alternate calout channel is forced to OFF.

CALOut?			Queries the settings of the EXTernal CHA and CHB calibrator settings.
			Example: `CALOUT?`
			Response:
			`CALOUT EXTERNAL: AMPL;`
			`CHA: OFF,`
			`CHB: OFF`

CALOut?	CH[A|B]		Queries for the selection of the calibrator signal supplied to CH[A|B]. Example: `CALOUT? CHB` Response: `CALOUT CHB:OFF`
CALOut?	EXTernal		Queries for the selection of the signal connected to the calibrator output connection. Example: `CALOUT? EXTERNAL` Response: `CALOUT EXTERNAL:TIME`
CALIBRATOr	AMPLItude:	<NRx>	Sets the calibrator signal amplitude. Limits: SCD1000: 0 VDC, ±40 mVDC, ±80 mVDC, ±0.2 VDC, ±0.4 VDC, ±0.8 VDC, ±2 VDC, and ±2.5 VDC; SCD5000: 0 VDC, ±0.5 VDC, ±1.0 VDC, ±2.0 VDC, ±3.0 VDC, and ±4.0 VDC Factory setting: 2.5 (SCD1000), 2.0 (SCD5000) Example: `CALIBRATOR AMPLITUDE:2`

Table 3-14: Diagnostic and Calibration Commands (Cont.)

Header	Link	Argument	Description
CALIBRATOr	TIME:	<NRx>	Sets the calibration signal time period. Limits: 4E−9 to 80E−6 Factory setting: 800E−9 Example: `CALIBRATOR TIME:4E-9`
CALIBRATOr?			Queries for all settings of the calibrator signal: amplitude and time period. Example: `CALIBRATOR?` Response: `CALIBRATOR AMPLITUDE:1.0,TIME:8.E-6`
CALIBRATOr?	AMPLItude		Queries for the calibrator signal amplitude. Example: `CALIBRATOR? AMPLITUDE` Response: `CALIBRATOR AMPLITUDE:2.5`
CALIBRATOr?	TIME		Queries for the calibrator signal time period. Example: `CALIBRATOR? TIME` Response: `CALIBRATOR TIME:800.E-9`
CCOnstant?			Queries for all calibration constants contained in NVRAM. Example: `CCONSTANT?` Response: `CCONSTANT 1:2048...351:0.00`
CCOnstant?	<ui>		Queries for the calibration constant specified by <ui>. Example: `CCONSTANT? <33>` Response: `CCONSTANT 33:1234`
CDAte?			Queries for the date when the last calibration was performed. The response is a quoted string. The format is yy-mm-dd. Example: `CDATE?` Response: `CDATE "89-09-24"`
DIAg?			Query only. Queries for results of the last diagnostics executed, which can be either power-up self-test or tests set up by TESt command. Response is PASSED, FAILED, or BYPASSED. If TESt VERBose is ON (see below), an ASCII string is also reported describing the test result. Example: `DIAG?` Response: `DIAG PASSED`
TESt	LOOp:	ON OFF	Sets test looping. When ON, the selected tests are repeated until any GPIB command is issued. When OFF, the tests are done only once. (Must be enabled by an internal jumper.) Factory setting: OFF Example: `TEST LOOP:ON`

Table 3-14: Diagnostic and Calibration Commands (Cont.)

Header	Link	Argument	Description
TESt	NUM:	<NRx>	Runs only the test specified by (<NRx>) as referenced in Table 3-19. Factory setting: 1 Example: `TEST NUM:3`
TESt	SYS:	ALL	Runs all tests, and generates an operation complete (OPC) SRQ when done. Factory setting: ALL Example: `TEST SYS:FPASYS`
		DIG	Runs tests associated with the processor board only. See Table 3-19 for more information.
		FP	Runs tests associated with the front panel only. See Table 3-19 for more information.
		MPU	Runs tests associated with the acquisition system only. See Table 3-19 for more information.
		OPTion	Runs test associated with the HSDO and Fast Waveform Capture options. See Table 3-19.
TESt	VERBose:	ON OFF	Sets DIAG? response format. If ON, an ASCII string of up to 130 characters describes the results of the first test that failed,or the last test executed if no failures were detected. If OFF, the response to queries on test results is abbreviated to PASSED or FAILED. Factory setting: OFF Example: `TEST VERBOSE:OFF`
TESt?			Queries for all test settings: LOOp, NUM, SYS,and VERBose. Example: `TEST?` Response: `TEST LOOP:OFF,NUM:1,SYS:FP,VERBOSE:OFF`
TESt?	LOOp		Queries for the state of test looping. See TESt LOOp earlier. Example: `TEST? LOOP` Response: `TEST LOOP:OFF`
TESt?	NUM		Queries for the number of the test to be run (see Table 3-19). The response is an unsigned integer. See TESt NUM command earlier. Example: `TEST? NUM` Response: `TEST NUM:3`
TESt?	SYS		Queries for the group of tests to be run. The response indicates which subsystem is checked: acquisition (MPU), front panel (FP), processor board (DIG),or ALL. See TESt SYS earlier. Example: `TEST? SYS` Response: `TEST SYS:FP`

Table 3-14: Diagnostic and Calibration Commands (Cont.)

Header	Link	Argument	Description
TESt?	VERBose		Queries for state (OFF or ON) of test verbosity. See TESt VERBose earlier. Example: `TEST? VERBOSE` Response: `TEST VERBOSE:OFF`
VIDeo		ON OFF	When ON, target video can be viewed in real time using an external monitor (factory service switch and internal jumper must also be set). Set to OFF at power-up. (Intended for calibration use only.) Factory setting: OFF Example: `VIDEO ON`
VIDeo?			Queries for the setting of the video command. Example: `VIDEO?` Response: `VIDEO OFF`

Utility Commands

Table 3-15: Utility Commands

Header	Link	Argument	Description
CLOck	DATE:	<qstring>	Sets the date of the internal clock in yy-mm-dd format. Factory setting: Current date Example: `CLOCK DATE:"89-12-23"`
CLOck	TIME:	<qstring>	Sets the time of the internal clock in hh:mm:ss format. The time should be set in 24-hour format. Factory setting: Current time Example: `CLOCK TIME:"14:25:12"`
CLOck?			Queries for the time and date. Example: `CLOCK?` Response: `CLOCK DATE:"89-12-22",TIME:"22:12:34"`
CLOck?	DATE		Queries for the date. Response is a quoted string in yy-mm-dd format. Example: `CLOCK? DATE` Response: `CLOCK DATE:"89-12-21"`
CLOck?	TIME		Queries for the time. Response is a quoted string in hh:mm:ss format. Example: `CLOCK TIME` Response: `CLOCK TIME:"15:23:33"`
CRTBkgnd		<NRx>	Sets CRT background sensitivity. Over enhancement produces a noisy target image background and elicits the error message: "linear array overflow". Limits: 0 to 100% Example: `CRTBgnd 30`
CRTBkgnd?			Queries for the CRT background sensitivity. Example: `CRTBkgnd?` Response: `CRTBkgnd 30`
FOCus		<NRx>	Adjusts the write gun's focus. Affects only the current time window. Limits: 0 to 100% Factory setting: 50 Example: `FOCUS 25`
FOCus?			Queries for the current focus setting (0 to 100%). Example: `FOCUS?` Response: `FOCUS 32`

Table 3-15: Utility Commands (Cont.)

Header	Link	Argument	Description
INTEnsity	<NRx>		Adjusts the write gun's beam intensity. Affects only the current time window. Limits: 0 to 100% Factory setting: Depends on the time window Example: `INTENSITY 34`
INTEnsity?			Queries for the intensity setting (0 to 100%). Example: `INTENSITY?` Response: `INTENSITY 34`
RAW			The RAW command selects what will be seen on the display when the target image is selected. To set the display to view the target image over GPIB the following procedure should be used: 1. Save the current settings: SAVE 9. 2. INIT the instrument: INI. 3. Recall the saved settings: REC 9. 4. Use the ABSTouch command to select the Utility Mode menu: ABS 6,8. (The target image is seen on the LCD display when Utility Mode menu containing the CRT settings is selected.)
		LINArray	Selects the waveform stored in the linear array when viewing the target image.
		REFArray	Selects the reference array defect map when viewing the target image.
RAW?			Queries for the selection of what will be seen when the display is set to view the target image is selected. Example: `RAW?` Response: `RAW LINEARRAY`

Reference Array Correction Commands

Table 3-16: Reference Array Correction Commands

Header	Link	Argument	Description
REFList?			Query only. Returns the reference array defects coordinate list. This is a list of the locations on the target where defects were detected during the reference array correction cycles. The format of the defects coordinate list is: REFList min_x, min_y, max_x,max_y,....
SETRef		OFF ON RUN	Turns off Reference Array correction. Turns on Reference Array correction. Runs Reference Array correction cycles. Reference array correction cycles are run to record accumulated target defect data. A more detailed description of reference array correction is given in the SCD1000/SCD5000 Operator's Manual. Any further input on the front panel or the GPIB will abort the correction cycle. Factory setting: ON
SETRef?			Query for the setting of target defect correction. If SETRef is running, reference array creation will be stopped and the reference array data lost. SETRef must be run again, without interruption to restore the correction array. Example: `SETRef?` Response: `SETREF ON`

Plot Commands

Table 3-17: Plot Commands

Header	Link	Argument	Description
PDAte		ON OFF	If on the current date setting in the SCD is added to the plot. Factory setting: ON Example: `PDAte OFF`
PDAte?			Queries whether PDAte is on or off for plots. Example: `PDate?` Response: `PDAte ON`
PLOT?			When the SCD is next talked addressed the unit will send a HPGL compatible representation of the information currently on the display unit. Example: `PLOT?`
PTItle		<qstring>	Up to 50 character string to use for plot title. This must be sent as a quoted string. To eliminate any title from the plot send the PTItle command with a single space quoted string ("Ê"). To restore the SCD ID? as the title send the PTItle command as a null string ("""). Factory setting: Instrument ID? response Example: `PTItle "Event #7"` Example: `PTItle "Ê"` Example: `PTItle ""`
PTItle?			Queries for the current plot title string Example: `PTItle?` Response: `PTItle "Test #11"`

Measurement Commands

Table 3-18: Measurement Commands

Header	Link	Argument	Description
AREA?			Query for AREA measurement of waveform data bounded by AREAZone. See RESUlts? AREA – Units of measure "Vs" (vertical units seconds).
AREAZone		DISTAl MEASZone MESial PROXimal	Used to determine boundaries for an area measurement. If MEASZone is selected then the area measurement is bound by the current selection for MEASZone, either full waveform or between cursors. Factory setting: MEASZone Example: `AREAZone PROXimal`
AREAZone?			Returns the current means being used to bound area measurements. Example: `AREAZone?` Response: `AREAZone MEASZone`
BASE?			Query for BASE line value as determined by BASEMode METhod or LEVEl of waveform data bounded by MEASUre MEASZone. See RESUlts? BASE – Units of measure "V" (vertical units).
BASEAber?			Query for BASEAber value (Minimum value – Base value) as determined by BASEMode METhod or LEVEl of waveform data bounded by MEASUre MEASZone. See RESUlts? BASEAber – Units of measure "V" (vertical units)
BASEMode	METhod:	ABSOlute HISTOGram HISTOMean MINImum	Select method to calculate baseline of the measurement zone. Baseline can be based from a histogram of the lower half of the vertical range of the data, as either maximum occurrence or average count.
BASEMode	LEVEl:	<NRx>	Further baseline can be based on the minimum value within the measurement zone or based on a user selected value (selection of ABSolute.enables LEVEl). Factory setting: METhod: HISTOGram LEVEl:0 Example: `BASEMode METhod:MINImum`
BASEMode?			Queries for current method and level of baseline. Example: `BASEMode?` Response: `BASEMode METhod:HISTOMean,LEVEl:240E-3`
BASEMode?	METhod		Queries for current means of establishing the baseline. Example: `BASEMode? METhod` Response: `BASEMode METhod:HISTOGram`

Table 3-18: Measurement Commands (Cont.)

Header	Link	Argument	Description
BASEMode?	LEVEl		Queries for the current LEVEl to use for baseline. Example:`BASEMode? LEVEl` Response: `BASEMode LEVEl: 2.6`
BASETop?			Query for BASETop value (Top value − Base value) as determined by BASEMode and TOPMode METhod or LEVEl of waveform data bounded by MEASUre MEASZone. See RESUlts? BASETop − Units of measure are "V" (vertical units).
CROSS?	DFAll DRIse MAXLoc MFAll MINLoc MPEriod MRIse PFAll PRIse		Queries for the interpolated waveform array index of an individual crossing location within the measurement zone. Returns −1 if the crossing was not found. Example: `CROSS? PRISE` Response `CROSS PRISE:47.83`
CROSS?			A query only that returns an array of 9 interpolate array indexes of the waveform data representing the points used for timing measurements plus the first location of minimum and maximum values found in the measurement zone. Returns −1 if the crossing was not found. Example: `CROSS?` Response: `CROSS PRISE:101.01,PFALL:86.99, DRISE:134.74,DFALL:59.45,MRISE:119.20, MFALL:71.56,MPERIOD:175.27,MAXLOC:45.00, MINLOC:511.00`
DISTAl?			Query for value of DISTAl used in all timing measurements. See RESUlts? DISTAl − Units of measure "V" (vertical units).
DISTLevel	PERCent: LEVEl:	<NRx> <NRx>	Used to set the value to be used for the distal level crossing (crossing closest to the top line). Whether PERCent or LEVEl is used is determined by LEVMode. Factory setting: DISTLevel PERCent: 90 DISTLevel LEVEl: 0 Example: `DISTLevel PERCent: 80`
DISTLevel?			Query for current level of distal level. Example: `DISTLevel?` Response: `DISTLevel PERCent: 90, LEVEl: 2.7`
DISTLevel?	PERCent LEVEl		Query for current value of distal level for either PERCent or LEVEl. Example: `DISTLevel? PERCent` Response: `DISTLevel PERCent: 90`

Table 3-18: Measurement Commands (Cont.)

Header	Link	Argument	Description
FALL?			Query for value (Proximal crossing – Distal crossing on a qualified edge) as determined by DISTLevel, PROXLevel and LEVMode of the first full qualified falling edge of the waveform data bounded by MEASUre MEASZone. See RESUlts? FALL – Units of measure "s" (seconds)
FALLSlew?			Query for fall time slew rate value ((Distal amplitude – Proximal amplitude) / (Proximal crossing – Distal crossing on a qualified edge)) as determined by DISTLevel, PROXLevel and LEVMode of the first full qualified falling edge of the waveform data bounded by MEASUre MEASZone. See RESUlts? FALLSlew – Units of measure "V/ns"
FREquency?			The reciprocal of the period determined by three consecutive edge qualified mesial level crossing of waveform data bounded by MEASUre MEASZone. See RESUlts? FREquency – Units of measure "Hz"
LEVMode		PERCent ABSOlute	Use either the percent or absolute value for the proximal, mesial, and distal levels. Factory setting: LEVMode PERCent Example: `LEVMode ABSOlute`
LEVMode?			Query for style of level being used to determine proximal, mesial, and distal crossings. Example: `LEVMode?` Response: `LEVMode PERCent`
MAXImum?			Query for maximum value of the waveform data bounded by MEASUre MEASZone. See RESUlts? MAXImum – Units of measure "V" (vertical units).
MEAN?			Query for mean value of the waveform data bounded by MEASUre MEASZone. See RESUlts? MEAN – Units of measure "V" (vertical units).
MEASUre	FUNction:	ON OFF	Turns execution of measurements on or off. Factory setting: MEASUre FUNction: OFF Example: MEASUre FUNction: ON
MEASUre	MANmeas		Executes the measurement routine on the currently acquired waveform data in the selected record. It will also display the measurements that have been selected.

Table 3-18: Measurement Commands (Cont.)

Header	Link	Argument	Description
MEASUre	MEASZone:	CURSors FULl WIN1 WIN2 WIN3 WIN4	Select the waveform region over which measurements will be made. Measurements can be made over all of the waveform data, data between cursors (inclusive) or data between the end points of a display window. ***NOTE:*** *The Cursors and Measurement Zone Window are not tied to the Measured Record. The Measurement record and the record displayed in the measurement zone window do not have to be the same. The measurement zone window simply defines end points that are then applied to the measurement record.* Factory setting: MEASUre MEASZone: FULl Example: `MEASUre MEASZone: CURSors`
MEASUre	WAVfrm:	<NRx>	Assigns record to make the waveform measurements on. Range is record 1 to record 16. Factory setting: MEASUre WAVfrm: 1 Example: `MEASUre WAVfrm: 12`
MEASUre	WINDow:	WIN1 WIN2 WIN3 WIN4	Tells which waveform window to display results of measurements (maximum of 8 at any one time) selected for display. Factory setting: MEASUre WINDow: WIN1 Example: `MEASUre WINDow: WIN2`
MEASUre?			Query to determine current status of measurement function, the measurement zone, and display area. Example: `MEASUre?` Response: `MEASUre FUNction: ON, WAVfrm: 1, MEASZone:CURSors, WINDow: WIN1`
MEASUre?	FUNction: MEASZone: WAVfrm: WINDow:		Query for current state of measurement execution status, measurement zone, which waveform record to measure, or waveform window of display selected results. Example: `MEASUre? FUNction` Response: `MEASUre FUNction: OFF`
MESIal?			Query for value of MESIal used in all timing measurements. See RESUlts? MESIal – Units of measure "V" (vertical units)
MESLevel	PERCent: LEVEl:	<NRx> <NRx>	Used to set the value to be used for the mesial level crossing (crossing between the proximal and distal crossings). WhetherPERCent or LEVEl is used is determined by LEVMode. Factory setting: MESLevel PERCent: 50 MESLevel LEVEl: 0 Example: `MESLevel PERCent: 55`
MESLevel?			Query for current level of mesial level Example: `MESLevel?` Response: `MESLevel PERCent: 50, LEVEl: 2.7`

Table 3-18: Measurement Commands (Cont.)

Header	Link	Argument	Description
MESLevel?	PERCent LEVEl		Query for current value of mesial level for either percent or level. Example: `MESLevel? PERCent` Response: `MESLevel PERCent: 46`
MINImum?			Query for minimum value of the waveform data bounded by MEASUre MEASZone. See RESUlts? MINImum – Units of measure "V" (vertical units)
MSList	AREA: BASE: BASEAber: BASETop: DISTAl: FALL: FALLSlew: FREquency: MAXImum: MEAN: MESIal: MINImum: PERIod: PK_pk: PROXImal: RISE: RISESlew: RMS: TOP: TOPAber: WIDth:	ON OFF	Determines which measurements will be displayed on the display unit (maximum of 8 at any one time). Position on screen determined by MEASUre WINDow. Factory setting: ALL MSList selections are OFF. Example: `MSList MINImum ON, MAXImum ON, PERIod: ON,PROXima: OFF`
MSList	CLEar		Turns OFF all currently selected displayed measurements.
MSList?			Query for the display state of all of the measurements. Example: `MSList?` Response: `MSList MAXImum: OFF, TOP: OFF, DISTAl: OFF, MESIal: OFF, PROXImal: OFF, BASE: OFF, MINImum: OFF, PK_pk: OFF, BASETop: OFF, TOPAber: OFF, BASEAber: OFF, RISE: OFF, RISESlew: OFF, FALL: OFF, FALLSlew: OFF, WIDth: OFF, PERIod: OFF, FREquency: OFF, AREA: OFF, MEAN: OFF, RMS: OFF`

Table 3-18: Measurement Commands (Cont.)

Header	Link	Argument	Description
MSList?	AREA BASE BASEAber BASETop DISTAl FALL FALLSlew FREquency MAXImum MEAN MESIal MINImum PERIod PK_pk PROXImal RISE RISESlew RMS TOP TOPAber WIDth		Queries for the display state of each parameter. Example: `MSList? AREA` Response: `MSList AREA: OFF`
PERIod?			The period determined by three consecutive edge qualified mesial level crossing of waveform data bounded by MEASUre MEASZone. See RESUlts? PERIod – Units of measure "s".
PK_pk?			Query for the peak to peak value (MAXImum value – MINImum value) of waveform data bounded by MEASUre MEASZone. See RESUlts? PK_pk – Units of measure are "V" (vertical units).
PROXImal?			Query for value of PROXImal used in all timing measurements. See RESUlts? PROXImal – Units of measure "V" (vertical units).
PROXLevel	PERCent: LEVEl:	<NRx> <NRx>	Used to set the value to be used for the proximal level crossing (crossing nearest the baseline). Whether PERCent or LEVEl is used is determined by LEVMode. Factory setting: PROXLevel PERCent: 10 PROXLevel LEVEl: 0 Example: `PROXLevel PERCent:20`
PROXLevel?			Query for current level of proximal level. Example: `PROXLevel?` Response: `PROXLevel PERCent: 20, LEVEl: -1.7`
PROXLevel?	PERCent LEVEl		Query for current value of proximal level for either percent or level. Example: `PROXLevel? PERCent` Response: `PROXLevel PERCent:12`

Table 3-18: Measurement Commands (Cont.)

Header	Link	Argument	Description
RESUlts?			Returns all of the measurements made on the selected waveform over the specified measurement zone. If a measurement is not found NAN (2.0E+308) is returned. Example: `RESUlts?` Response: `MAXImum: 212.4E-3, TOP: 167.0E-3, DISTAl: 118.1E-3, MESIal: -77.4E-3, PROXImal: -272.9E-3, BASE: -321.8E-3, MINImum: -345.2E-3, PK_pk: 557.6E-3, BASEAber: 23.4E-3, TOPAber: 45.4E-3, BASETop: 488.8E-3, RISE: 330.E-9, RISESlew: 1.184E-3, FALL: 270.E-9, FALLSlew: 1.451E-3, WIDth: 466.E-9, PERIod:1.015E-6, FREQuency: 985.221E3, AREA:-298.7E-9, MEAN: -60.2E-3, RMS: 186.3E-3`
RESUlts?	AREA BASE BASEAber BASETop DISTAl FALL FALLSlew FREquency MAXImum MEAN MESIal MINImum PERIod PK_pk PROXImal RISE RISESlew RMS TOP TOPAber WIDth		Queries for individual measurements made on the selected waveform over the selected measurement zone. If a measurement is not found NAN (2.0E+308) is returned. Example: `RESUlts? BASE` Response: `RESUlts BASE: -3211.8E-3`
RISE?			Query for rise time value (Distal crossing – Proximal crossing on a qualified edge) as determined by DISTLevel, PROXLevel and LEVMode of the first full qualified falling edge of the waveform data bounded by MEASUre MEASZone. See RESUlts? RISE – Units of measure "s" (seconds).
RISESlew?			Query for rise time slew rate value (Distal amplitude – Proximal amplitude) / (Distal crossing – Proximal crossing) on a qualified edge) as determined by DISTLevel, PROXLevel and LEVMode of the first full qualified falling edge of the waveform data bounded by MEASUre MEASZone. See RESUlts? RISESlew – Units of measure "V/ns".

Table 3-18: Measurement Commands (Cont.)

Header	Link	Argument	Description
RMS?			Query for RMS value (square root of the mean of the sum of the squares – AC+DC measurement) of the waveform data bounded by MEASUre MEASZone. See RESUlts? RMS – Units of measure "V" (vertical units).
TOP?			Query for TOP line value as determined by TOPMode METhod or LEVEl of waveform data bounded by MEASUre MEASZone. See RESUlts? TOP – Units of measure "V" (vertical units)
TOPAber?			Query for TOPAber value (Maximum value – Top value) as determined by TOPMode METhod or LEVEl of waveform data bounded by MEASUre MEASZone. See RESUlts? TOPAber – Units of measure "V" (vertical units).
TOPMode	METhod: LEVEl:	ABSOlute HISTOGram HISTOMean MAXImum <NRx>	Select method to calculate the top line of the measurement zone. Top line can be based from a histogram of the upper half of the vertical range of the data, as either maximum occurrence or average count. Further, Top Line can be based on the maximum value within the measurement zone or based on a user selected value. Selection of ABSolute.enables LEVEl. Factory setting: METhod: HISTOGram LEVEl:0 Example: `TOPMode METhod: MAXImum`
TOPMode?			Queries for current method and level of the top line. Example: `TOPMode?` Response: `TOPMode METhod: HISTOMean, LEVEl: 240E-3`
TOPMode?	METhod		Queries for current means of establishing the top line. Example: `TOPMode? METhod` Response: `TOPMode METhod: HISTOGram`
TOPMode?	LEVEl		Queries for the current level to use for the top line. Example: `TOPMode? LEVEl` Response: `TOPMode LEVEl: 2.6`

Table 3-18: Measurement Commands (Cont.)

Header	Link	Argument	Description
UNIts?			Queries for units of measure of all measurements made on the selected waveform over the selected measurement zone. Note "V" represents volts which is the native vertical mode of the SCD. If a probe that encodes other units is used these will be used (e.g., Tek Type II I/F optical probe would use "µW". Example: `UNIts?` Response: `MAXImum: "V", TOP: "V", DISTAl: "V",MESIal: "V", PROXImal: "V", BASE: "V", MINImum: "V", PK_pk: "V", BASEAber: "V", TOPAber: "V", BASETop: "V", RISE: "s", RISESlew: "V/ns", FALL: "s", FALLSlew: "V/ns", WIDth: "s",PERIod: "s", FREQency: "Hz", AREA: "Vs", MEAN: "V", RMS: "V"`
UNIts?	AREA BASE BASEAber BASETop DISTAl FALL FALLSlew FREquency MAXImum MEAN MESIal MINImum PERIod PK_pk PROXImal RISE RISESlew RMS TOP TOPAber WIDth		Queries for units of measure of individual measurements made on the selected waveform over the selected measurement zone. Note "V" represents volts which is the native vertical mode of the SCD. If a probe that encodes other units is used these will be used (e.g., Tek Type II I/F optical probe would use "µW". Example: `UNIts? FALL` Response: `UNIts FALL: "s"`
WIDth?			The width determined by the first two consecutive edge qualified mesial level crossing of waveform data bounded by MEASUre MEASZone. See RESUlts? WIDth – Units of measure "s".

Table 3-19: Test List For The Test Command

Number	Name	Subsystem	Description
1	Real Time Clock	MPU	Verifies that tick interrupts occur at the correct rate.
2	GPIB	MPU	Verifies the GPIB interface.
3	Bus Error	MPU	Verifies the bus error detect logic.
4	Timer	MPU	Verifies that the timer interrupts at the correct rate.
5	ROM0 Part Number	MPU	Retrieves the part number from the ROM header.
6	ROM1 Part Number	MPU	Retrieves the part number from the ROM header.
7	ROM2 Part Number	MPU	Retrieves the part number from the ROM header.
8	ROM3 Part Number	MPU	Retrieves the part number from the ROM header.
9	Display Unit ROM Part Number	MPU	Retrieves the part number from the display unit ROM.
10	NVRAM	MPU	Verifies that NVRAM works correctly.
11	Video	FP	Verifies video RAM/LCD display.
12	Button	FP	Exercises knobs and buttons.
13	Front Panel Communication	FP	Verifies communication between the front panel 68705 CPU and the 68010 MPU.
14	Digital Acquisition With Memory Test	DIG	Verifies digital acquisition hardware and memory (reference & linear array).
15	Digital Acquisition Without Memory Test	DIG	Same test as 14 without the memory test.
16	Serial Bus	DIG	Verifies communications over the serial bus to the analog, read, and write boards.
17	Option Test	OPT	Verifies HSDO (Option 2F) and Fast Waveform Capture (Option 1P) options are correctly readable and writeable by the 68010 MPU.
18	Option 01 Communication Test	OPT	Checks Option 01 host port communication.
19	Option 2F Battery Back-up Test	OPT	Checks that NVRAM on the Option 2F board will retain waveform data across a power-down.

Table 3-20: Instrument Factory Settings and Limits

Header	Link	Factory Settings	Limits (N/A = Not Applicable)
VMOde		CHA	SCD1000 only
CH<x>	RANge	1	100E−3 to 10 (probe attenuation = x1)
CH<x>	OFFSet	0	±1 to ±100
CH<x>	TYPEOffset	VOLTS	N/A
CH<x>	COUPling	AC	DC, AC & OFF allowed only on SCD1000
CH<x>	INVert	OFF	ON, OFF (SCD1000) only
TRIgger	COUPling	AC	Depends on Source. See Triggering in Section 1.
		DC (SCD1000 and SCD5000 Opt 01)	
TRIgger	MODE	AUTO	N/A
TRIgger	LEVEl	0.0	Internal source: ±0.625*full-scale voltage range
			External source: ±6.25 volts
TRIgger	TYPELevel	VOLTS	N/A
TRIgger	DELay	0	0 to 9 times record length
TRIgger	TYPEDelay	SECOND	N/A
TRIgger	SLOpe	PLUS	N/A
TRIgger	SOUrce	INTERNAL (SCD5000 with Opt 01)	N/A
		EXTERNAL (SCD5000) CHA (SCD1000)	
ARM		INTERnal	N/A
ACQuire	MODe	NORMAL	N/A
ACQuire	STATe	STOP	N/A
ACQuire	TIME	1E−6 (1 µs)	5E−9 to 100E−6 (5 ns to 100 µs)
ACQuire	LENgth	512	256, 512, or 1024.
ACQuire	NRECord	1	1 to 16
ACQuire	STARt	1	1 to 16
ACQuire	HLDnxt	OFF	N/A
ACQuire	DBLsweep	OFF	N/A
ACQuire	GEOmetry	OFF(SCD1000)	
		ON (SCD5000)	
ACQuire	AVErage	16	1 to 1024
ACQuire	LASt	1	1 to 16
CRS1	LOCTn	WIN1	1 ≤LOCTn ≤4
CRS1	XPOint	0	0 ≤XPOint ≤(RL−1)
CRS2	LOCTn	WIN1	1 ≤LOCTn ≤4
CRS2	XPOint	0	0 ≤XPOint ≤(RL−1)
CRSD	TYPETime	SECOND	N/A

NVRAM Calibration Constants 06/24/2012

CCONSTANT 0:2048,
1:2048,
2:117,
3:37,
4:128,
5:114,
6:127,
7:127,
8:127,
9:127,
10:150,
11:127,
12:120,
13:129,
14:2048,
15:2048,
16:127,
17:110,
18:127,
19:24,
20:3280,
21:2052,
22:127,
23:250,
24:127,
25:127,
26:127,
27:127,

SCD1000
Response to CCO? command
for S/N B040272

Table 3-20: Instrument Factory Settings and Limits (Cont.)

Header	Link	Factory Settings	Limits (N/A = Not Applicable)
CURSors		ON	N/A
HEXPMd		ALIGNED	N/A
NWIn		1	1, 2, or 4
GRAticule		OFF	N/A
CRTBkgrd		0	0 to 100
INTEnsity		Set by CRT self-cal	0 to 100
FOCus		Set by CRT self-cal	0 to 100
MEASUre	FUNction	OFF	
MEASUre	MEASZone	FULl	
MEASUre	WAVfrm	1	
MEASUre	WINDow	WIN1	
WIN<ui>	EXPnt	0	0 to (Record Length − 1)
WIN<ui>	HEXPNd	1	1, 2, 4, or 8
WIN<ui>	RECOrd	All windows: 1	1 ≤ RECOrd ≤ nNRECord
WIN<ui>	VEXpnd	1	1, 2, or 4 (SCD5000)
SAVe		1	1 to 10
RECAll		1	1 to 10
VIEW		OFF	Accessible from Display Unit only

Table 3-21: IEEE−488.1 Factory Settings and Argument Limits

Header	Link	Factory Settings	Limits (N/A = Not Applicable)
BELl	BUTton	ON	N/A
BELl	KNOb	ON	N/A
DATA	BYTEOrder	MSB	N/A
DATA	CNTrecord	1	1 ≤ CNTrecord ≤ (Number of Records)
DATA	COUnt	0	0 ≤ COUnt ≤ (Record Length-STArt)
DATA	FLAGbit	ON	N/A
DATA	STArt	1	1 ≤ STArt ≤ Record Length
DATA	STRecord	1	1 ≤ STRecord ≤ NRECord
DEBug	GPIb	OFF	N/A
DISplay		ON (only if Display Unit is present)	N/A
DT		OFF	N/A
FPAnel		ON (only if Display Unit is present)	N/A
LONgform		ON	N/A
PATh		ON	N/A
REPSet	NREPEat	1	0 ≤ NREPEat ≤ ($2^{32}-1$) (0 = infinite repeat)
RQS		ON	N/A
SAFEguard	ACQProtect	OFF	N/A
SAFEguard	LINConvert	N/A	1 ≤ RECOrd ≤ 16
SAFEguard	PROTect	ON	SCD1000 only
SAFEguard	PUPtst	ON	Affected by rear panel switch #8
SRQmask	ABSTouch	OFF	N/A
SRQmask	CMDerr	ON	N/A
SRQmask	EXErr	ON	N/A
SRQmask	EXWarn	ON	N/A
SRQmask	INErr	ON	N/A
SRQmask	INWarn	ON	N/A
SRQmask	OPCmpl	ON	N/A
SRQmask	USR1	OFF	N/A
SRQmask	USR2	OFF	N/A
TESt	LOOp	OFF	N/A
TESt	NUM	1	1 ≤ NUM ≤ 42
TESt	SYS	ALL	N/A
TESt	VERBose	OFF	N/A

Table 3-21: IEEE−488.1 Factory Settings and Argument Limits (Cont.)

Header	Link	Factory Settings	Limits (N/A = Not Applicable)
TEXt	STRIng	"" (null string)	N/A
TEXt	CHAR	1	1 ≤CHAR ≤64
TEXt	CLEar	N/A	0 ≤ <NRx> ≤16
TEXt	LINE	1	1 ≤LINe ≤16
USER1		"","" null string,null string	≤8 characters for each string
USER2		"","" null string,null string	≤8 characters for each string
WFTx		DL	N/A

Table 3-22: Text Command Character Set

Decimal	Binary	Character	Decimal	Binary	Character	Decimal	Binary	Character
0	00000000	NU	29	00011101	GS	58	00111010	:
1	00000001	SH	30	00011110	RS	59	00111011	;
2	00000010	SX	31	00011111	US	60	00111100	<
3	00000011	EX	32	00100000	<space>	61	00111101	=
4	00000100	ET	33	00100001	!	62	00111110	>
5	00000101	EQ	34	00100010	"	63	00111111	?
6	00000110	AK	35	00100011	#	64	01000000	@
7	00000111	BL	36	00100100	$	65	01000001	A
8	00001000	BS	37	00100101	%	66	01000010	B
9	00001001	HT	38	00100110	&	67	01000011	C
10	00001010	LF	39	00100111	'	68	01000100	D
11	00001011	VT	40	00101000	(	69	01000101	E
12	00001100	FF	41	00101001	)	70	01000110	F
13	00001101	CR	42	00101010	*	71	01000111	G
14	00001110	SO	43	00101011	+	72	01001000	H
15	00001111	SI	44	00101100	,	73	01001001	I
16	00010000	DL	45	00101101	-	74	01001010	J
17	00010001	D1	46	00101110	.	75	01001011	K
18	00010010	D2	47	00101111	/	76	01001100	L
19	00010011	D3	48	00110000	0	77	01001101	M
20	00010100	D4	49	00110001	1	78	01001110	N
21	00010101	NK	50	00110010	2	79	01001111	O
22	00010110	SY	51	00110011	3	80	01010000	P
23	00010111	EB	52	00110100	4	81	01010001	Q
24	00011000	CN	53	00110101	5	82	01010010	R
25	00011001	EM	54	00110110	6	83	01010011	S
26	00011010	SB	55	00110111	7	84	01010100	T
27	00011011	EC	56	00111000	8	85	01010101	U
28	00011100	FS	57	00111001	9	86	01010110	V

Table 3-22: Text Command Character Set (Cont.)

Decimal	Binary	Character	Decimal	Binary	Character	Decimal	Binary	Character
87	01010111	W	113	01110001	q	139	10001100	å
88	01011000	X	114	01110010	r	140	10001100	Æ
89	01011001	Y	115	01110011	s	141	10001101	æ
90	01011010	Z	116	01110100	t	142	10001110	ç
91	01011011	[	117	01110101	u	143	10001111	ß
92	01011100	\	118	01110110	v	144	10010000	Ñ
93	01011101	]	119	01110111	w	145	10010001	ñ
94	01011110	^	120	01111000	x	146	10010010	¿
95	01011111	‘	121	01111001	y	147	10010011	i
96	01100000	_	122	01111010	z	148	10010100	A<tilde>
97	01100001	a	123	01111011	{	149	10010101	a<tilde>
98	01100010	b	124	01111100	|	150	10010110	A<‘>
99	01100011	c	125	01111101	}	151	10010111	O<tilde>
100	01100100	d	126	01111110	~	152	10011000	o<tilde>
101	01100101	e	127	01111111	<shaded box>	153	10011001	E<‘>
102	01100110	f	128	10000000	Ä	154	10011010	Ø
103	01100111	g	129	10000001	ä	155	10011011	ø
104	01101000	h	130	10000010	Ö	156	10011100	Œ
105	01101001	i	131	10000011	Ö	157	10011101	œ
106	01101010	j	132	10000100	Ü	158	10011110	Ç
107	01101011	k	133	10000101	ü	159	10011111	∞
108	01101100	l	134	10000110	à	160	10100000	<smiling face>
109	01101101	m	135	10000111	é	161	10100001	°
110	01101110	n	136	10001000	á	162	10100010	<ground symbols>
111	01101111	o	137	10001001	è	163	10100011	<smiling face>
112	01110000	p	138	10001010	Å	164	10100100	<db symbol>

Table 3-22: Text Command Character Set (Cont.)

Decimal	Binary	Character	Decimal	Binary	Character	Decimal	Binary	Character
165	10100101	BWL	188	10111100	<smiling face>	211	11010011	Σ
166	10100110	HZ	189	10111101	<falling edge symbol>	212	11010100	τ
167	10100111	RAD	190	10111110	<smiling face>	213	11010101	υ
168	10101000	DEG	191	10111111	<smiling face>	214	11010110	ν
169	10101001	<smiling face>	192	11000000	Π	215	11010111	ω
170	10101001	<smiling face>	193	11000001	α	216	11011000	χ
171	10101011	<arrows>	194	11000010	γ	217	11011001	ξ
172	10101100	<smiling face>	195	11000011	δ	218	11011010	ζ
173	10101101	<rising edge symbol>	196	11000100	Δ	219	11011011	φ
174	10101110	PT	197	11000101	ε	220	11011100	Λ
175	10101111	NU	198	11000110	φ	221	11011101	Ψ
176	10110000	1	199	11000111	Γ	222	11011110	σ
177	10110001	2	200	11001000	Θ	223	11011111	Ξ
178	10110010	3	201	11001001		224	11100000	⊘
179	10110011	4	202	11001010	ψ	225	11100001	≠
180	10110100	5	203	11001011	ϰ	226	11100010	Æ
181	10110101	6	204	11001100	λ	227	11100011	¨
182	10110110	7	205	11001101	μ	228	11100100	Ú
183	10110111	8	206	11001110	η	229	11100101	Π
184	10111000	9	207	11001111	Ω	230	11100110	∞
185	10111001	0	208	11010000	π	231	11100111	÷
186	10111010	<smiling face>	209	11010001	θ	232	11101000	¬
187	10111011	<smiling face>	210	11010010	ϱ	233	11101001	±

Table 3-22: Text Command Character Set (Cont.)

Decimal	Binary	Character	Decimal	Binary	Character	Decimal	Binary	Character
234	11101010	≠	242	11110010	<right half of T/left half of e in Tek>	250	11111010	
235	11101011	≤	243	11110011	<right half of e/left third of k in Tek>	251	11111011	
236	11101100	≥	244	11110100	<middle third of k in Tek>	252	11111100	
237	11101101	©	245	11110101	<right third of k in Tek>	253	11111101	
238	11101110	®	246	11110110		254	11111110	
239	11101111	≈	247	11110111		255	11111111	
240	11110000	—	248	11111000				
241	11110001	<left half of T in Tek>	249	11111001				

Status and Events

This section describes the status and event system of the SCD. A general description of the system and concepts is provided along with tables of status bytes and event codes.

If the RQS ON command has been sent, the SCD can request service from the controller by asserting the SRQ control line. In response, the controller performs a serial poll, reading a *status byte* from each device on the IEEE–488.1 bus. The status byte indicates the current condition of each device. If the device requested service, a bit of the status byte is set.

The status byte also generally indicates the reason for requesting service. More information on the request is indicated by an event code, which the controller can query using the EVENT? query command (all event codes can be queried by sending the ALLEV? query command). The information obtained from the event codes can be used during program execution and is also helpful during program development and troubleshooting.

Through this process, involving serial polls, status bytes, and event codes, the SCD can report operating status to the controller. For more information on the status and event query commands, see Table 3-7.

Status Bytes

The SCD status byte bit assignments are shown in Figure 4-1.

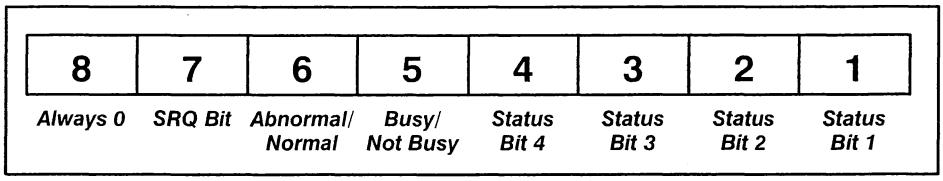


Figure 4-1: Status Byte Bit Assignments

Bit 7 (SRQ Bit) is set to 1 when the digitizer requests service (referred to by "S" in Table 4-1). The controller reads this bit during a serial poll to determine the requesting device. When RQS is ON, the digitizer sets this bit and asserts the SRQ line. When RQS is OFF, the digitizer clears this bit and does not assert the SRQ line. The controller must then poll the SCD and read the four status bits to determine if the SCD needs to be serviced. The RQS bit also affects how event codes can be read from the digitizer. See Reading the Event Codes later in this section.

Bit 6 (Abnormal/Normal) is set to 1 when the status is abnormal. The bit is set to 0 when the status is normal. Abnormal states include errors and warnings generated during processing. Normal states include power-on, operation complete, and user requests. See Table 4-1.

Bit 5 (Busy/Not Busy) is set to 1 when the digitizer is busy. It is set to 0 when the digitizer is not busy. The digitizer is busy when internal processing makes it unavailable to the controller, or when an acquisition is in progress.

Bits 4 through 1 (System Status) generally indicates a reason for requesting service.

Table 4-1 lists the status byte values and corresponding system states generated by the 8 bits of the status byte. Further information about the condition can be obtained from the event code.

Table 4-1: SCD1000/SCD5000 Status Bytes

	Binary			Decimal			
				SRQ Asserted		SRQ Unasserted	
Title	**Bit —**	**8765**	**4321**	**Not Busy**	**Busy**	**Not Busy**	**Busy**
System Status (Normal):							
No Status To Report		0000	0000	0	16	0	16
Power On		0S0B	0001	65	81	1	17
Operation Complete		0S0B	0010	66	82	2	18
User Request		0S0B	0011	67	83	3	19
System Status (Abnormal):							
Command Error		0S1B	0001	97	113	33	49
Execution Error		0S1B	0010	98	114	34	50
Internal error		0S1B	0011	99	115	35	51
Execution Warning		0S1B	0101	101	117	37	53
Internal Warning		0S10	0110	70	102	6	38

Table 4-2: Normal Status Conditions

Status	Conditions
No Status To Report	No event or device dependent status to report.
Power On	Indicates to the controller that the digitizer has finished its power-up sequence and is ready.
Operation Complete	Identifies when a task has been completed, such as a repeat sequence.
User Request	Identifies that a user-programmable button has been pressed.

Table 4-3: Abnormal Status Conditions

Status	Conditions
Command Error	Indicates that the digitizer could not understand the command it received.
Execution Error	Indicates that the digitizer recognized the command it received but could not successfully execute it.
Internal Error	Identifies a functional problem with the digitizer, such as a system failure, etc. The data should be considered suspect and the problem investigated.
Execution Warning	Indicates that the digitizer was able to understand and execute the command but there was some problem with it, such as an argument that was outside the commands limits.
Internal Warning	Identifies when the digitizer detects a problem, such as an over-voltage condition, over-temperature condition, calibration failure, etc., but is able to continue operation. The data should be considered suspect and the condition should be investigated.

All of these status conditions generate one or more event codes to further indicate the abnormal status.

Event Codes

Event codes are returned with a number from 0 to 999 and a descriptive string (if LONGFORM is ON). The event codes are categorized as shown in Table 4-4.

Table 4-4: Event Code Groups

Event Class	Event Code Range
Command Errors	100–199
Execution Errors	200–299
Internal Errors	300–399
System Events	400–499
Execution Warnings	500–599
Internal Warnings	600–699
Device-Dependent Events	700–799
Not Currently Used	800–900

Reading the Event Registers

Status bytes and event codes are generated at the same time; an event code always accompanies a status byte through the event system. However, the event code is not simultaneously read with the status byte. Figure 4-2 illustrates how status bytes and event codes are placed on the bus. A *serial poll* places the status byte on the IEEE–488.1 bus and moves the corresponding event code into a polled event code register. An *event query command* places the event code on the bus.

The controller must poll the digitizer and read the status byte to determine if the SCD needs to be serviced. The controller may also query for an event code or for all event codes. However, the RQS command setting affects how the digitizer responds to serial polls and event queries.

With RQS ON, an event code can only be transmitted if its corresponding status byte has first been polled. Thus, consecutive event queries without prior serial polls, or a query for all events, will return a special event code identifying that a serial poll must first be done. Depending on the situation, one normal event code may also be returned, but with RQS ON, no more than one event code plus the *special event code* will be returned.

With RQS OFF, event codes can be consecutively transmitted, or all event codes can be transmitted, without first polling the device. However, since status bytes and event codes travel through the event system together, corresponding status bytes are lost with consecutive event queries or a query for all events. Similarly, consecutive serial polls will cause status bytes to be transmitted, but corresponding event codes will be lost.

Table 4-4 lists the results of serial poll/event query combinations for RQS ON and RQS OFF. The Slash / character separates the commands as they occur.

Table 4-5: Effects of RQS on Status Bytes and Event Codes

RSQ State	Assert SRQ	Serial Poll/ Event?[1]	Serial Poll Allev?[1]	No Serial Poll/Event?[1]	No Serial Pll/Allev?[1]	Serial Poll/ Serial Poll[1]
OFF	NO	Most recent sb followed by corresponding ec	Most recent sb, corresponding ec; all other ec's (corresponding sb's are lost)	Most recent sb lost; corresponding ec is sent	All sb's lost; ec's sent	Most recent sb sent followed by next sb; first ec lost
ON	YES	Most recent sb followed by corresponding ec	Most recent sb, corresponding ec followed by sec	Not allowed; sec returned	Not allowed; sec sent	Most recent sb sent followed by next sb; first ec lost

1 sb=status byte; ec=event code; sec=the special event code described above.

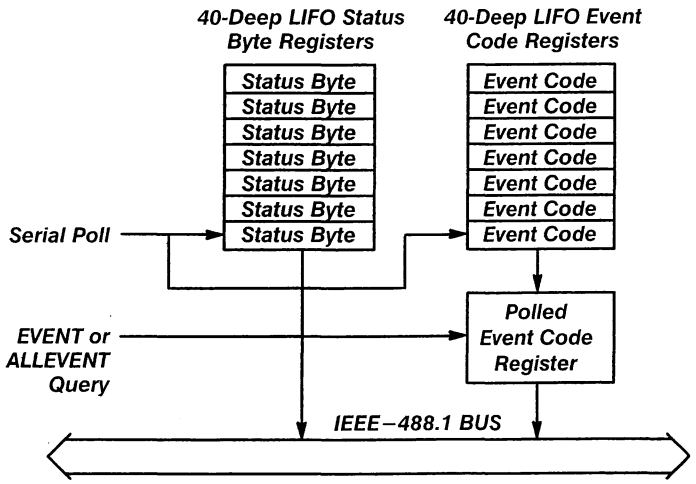


Figure 4-2: Status Byte and Event Registers Model

Event Code Tables

Tables 4-6 through 4-11 list the SCD event codes and event code description strings for all event groups.

Table 4-6: Command Error Events (SRQ 97, 113)

Code	Message
108	Checksum error in binary block transfer
151	Symbol or number too long
152	Invalid or out of range input character
154	Invalid number input (syntax incorrect)
155	Invalid string input
157	Syntax error
160	Too many binary points
162	Numerical overflow
163	Numerical underflow

Table 4-7: Execution Error Events (SRQ 98, 114)

Code	Message
203	I/O buffers full. Untalk query in output buffer prevents instrument from processing new commands
252	Illegal date/time argument
253	Saved setting buffer #<NR1> is empty
255	Window #<NR1> not displayed
256	No data available for selected channel/record in window <NR1>
257	Command disabled by internal switch (see GPIB command set)
260	Too many points for TC&F transfer
261	Calibration constant #<NR1> is not valid
268	Acquire status must be running before manually triggering
269	Reference Array Data lost
270	Reference array update aborted, data lost
271	Vertical Geometry constants invalid, reset to zero
272	Setting is illegal with current option configuration

Table 4-8: Internal Error Events (SRQ 99, 115)

Code	Message
350	Self test failure
352	Fast waveform comm. fault; Option has been disabled

Table 4-9: System Error Events (SRQ 99, 115)

Code	Message
400	No status to report
401	Power on initialization complete
403	User requested SRQ #1
451	User requested SRQ #2
452	Front panel input generates SRQ
454	SRQ pending
455	Self test completed successfully
456	<string> calibration passed
458	Selected front panel setting #<NR1> recalled
459	Save in buffer #<NR1> complete
461	RAM has been erased
462	Front-panel initialization is complete
463	GPIB initialization is complete
465	Acquisition done
466	Reference array update completed, <number> defects found

Table 4-10: Execution Warning Events (SRQ 101, 117)

Code	Message
551	<string> argument is out of range
553	Expansion factor on window #<NR1> forced to a power of 2
554	Expansion point aligned on a <NR1> point boundary
555	Input number too large; forced to maximum value
556	Input number too small; forced to minimum value
560	Invalid point count specified for curve transfer
561	curve data not from same acquire cycle
565	Repeat mode aborted on input
566	Reference array update overflow, <number> defects processed
567	Reference array update in progress, DO NOT power down

Table 4-11: Internal Warning Events (SRQ 70, 102)

Code	Message
655	Input channel <string> overvoltage
656	Target protect, check Z-axis
657	Linear array overflow in record <NR1>
658	Missing data in record <NR1>
660	Calibration failure: <string>
662	Self test bypassed
663	<string> out of range
664	Video mode has timed out
665	Target protect, check horizontal
666	System self calibration recommended due to temperature change
667	Self calibration is not recommended until 30 minutes after power up
668	Trigger external arm requires control board upgrade

Programming Examples

Introduction

This section provides some program examples to show how routines can be used to perform acquisitions, read the waveform preamble information for scaling data, acquire and scale waveform data, and graph the data on a computer terminal. In addition, some background information on waveform transfer formats and data transfers is provided.

The program listings provided in this section are written in BASIC for IBM PC, XT, AT, and 386-compatible microcomputers and HP 200/300 Series computers. IBM-compatibles require the National Instruments™ GPIB Interface Card with drivers and Microsoft QuickBASIC 4.5. HP computers require the HP 200/300 Series BASIC language.

The remainder of this section describes the integration of the SCD Series into the 7912AD/HB Series systems.

Waveform Data Formats

Each digitized waveform point is represented as an 11-bit number in absolute binary format. Waveform data can be transferred to the controller in one of three different formats (set by the WFTX command): Indefinite Length (IL), Definite Length (DL), and Tek Codes & Formats (TCF).

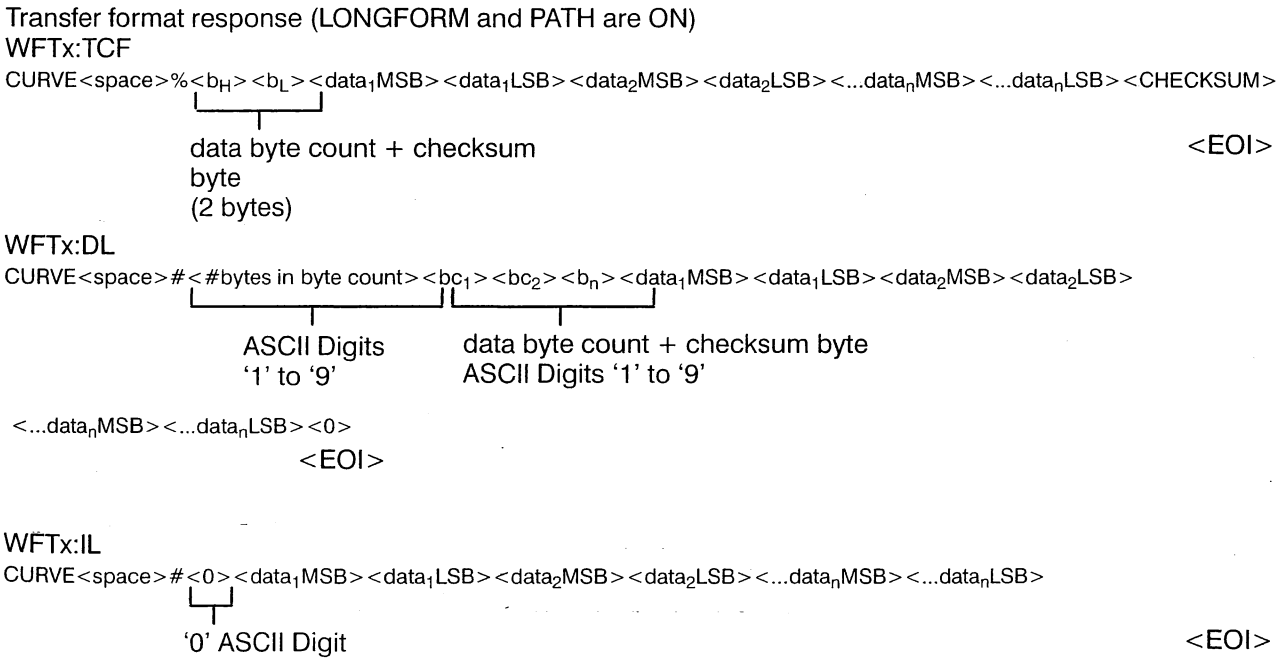


Figure 5-1: Data Transfer Formats Protocol

Table 5-1 illustrates the format of the linear and reference arrays when queried using the LINARRAY? and REFARRAY? commands. The number of data values varies depending on the signal type and intensity level. The number of data values can be up to 128 K words (16 bit) of data.

Table 5-1: Linear & Reference Waveform Data Format

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1	0	0	0	0	0	h	h	h	h	h	h	h	h	h	h
	0	v	v	v	v	v	v	v	v	v	d	d	d	d	d	d
	0	v	v	v	v	v	v	v	v	v	d	d	d	d	d	d
									.							
									.							
									.							
	0	v	v	v	v	v	v	v	v	v	d	d	d	d	d	d
	1	0	0	0	0	0	h	h	h	h	h	h	h	h	h	h
	0	v	v	v	v	v	v	v	v	v	d	d	d	d	d	d
	0	v	v	v	v	v	v	v	v	v	d	d	d	d	d	d
									.							
									.							
									.							
	0	v	v	v	v	v	v	v	v	v	d	d	d	d	d	d

Bit 15 = 1 indicates a horizontal point where hhhhhhhhhh (10 bits: 0..1023) is a binary number representing the horizontal position.

Bit 15 = 0 indicates a vertical position where vvvvvvvv (9 bits 0..511) is a binary number representing the vertical position and dddddd (6 bits 0..63) represents the charge intensity data.

Each format transfers the data differently as shown in Figure 5-1.

Waveform Data Transfers

The DATA statement specifies the source, length, and other parameters of the transfer. The DATA statement also provides a means of transferring only a portion of a record.

In order to select a certain record, the record number must be specified in a DATA statement. For example, to acquire data from only record 5, the following DATA statement would be sent to the digitizer:

```
DATA STRECORD:5;DATA CNTRECORD:1;DATA COUNT:0;CURVE?
```

A section of a waveform record can also be selected for transfer by specifying the starting sample point and the number of sample points to be transferred. To transmit 100 samples starting at sample point 32, the following DATA statement would be sent:

```
DATA STRECORD:5;DATA CNTRECORD:1;DATA START:32;DATA
COUNT:100;CURVE?
```

Record numbers and start points begin with 1. If the start/count combination exceeds the record length, a warning SRQ will be issued and the count will be corrected. If COUNT is set to 0, the instrument will transmit the entire record. If the selected records are not from the same acquisition, a warning SRQ will be generated.

The CURVE?, WAVEFRM?, and REPEAT? query commands initiate data transfers. CURVE? transfers only waveform data. WAVEFRM? transfers waveform preamble information first and then waveform data. REPEAT? causes the digitizer to repeat a programmed number of acquisitions and then transfer them to the controller after each acquisition.

Waveform Scaling Algorithms

Scaling of waveform data for graphing and analysis requires scaling data included in the waveform preamble. The scaling data along with the vertical and horizontal data points are used in the following algorithms.

```
YY = (PV - YOFF) * YMULT + YZERO
X = (PH - PT.OFF) * XINCR + XZERO
```

NOTE

XZERO is always 0. YOFF is always 127.

Sample WFMPRE? & SET? Transfers

The SCD waveform preamble contains the information needed to convert the binary waveform data from the digitizer into actual vertical and horizontal data for graphing, plotting, and analysis. The preamble information includes:

- the number of points in the waveform
- the vertical scale factor
- the sample interval
- the position of the first data point in the transfer relative to the trigger position
- the vertical offset
- timestamp

IBM QuickBasic Example

```
DECLARE SUB GRAPH.WFM (iwfm%(), xi#, numpt!)
DECLARE SUB SWAP.BYTES (test%(), numpt!)
DECLARE FUNCTION MIN% (iwfm%(), MINLOC!)
DECLARE FUNCTION MAX% (iwfm%(), MAXLOC!)
DECLARE SUB GETWFM (dig!, iwfm%(), MODE$, flag!)
DECLARE SUB GETSCALE (dig!, rec!, numpt!, ym#, yz#, yo#, xi#, flag!)
DECLARE FUNCTION GETANS$ (mes$)
DECLARE SUB PRESS.ANY.KEY (mes$)
DECLARE SUB GPIB.WRITE (dig!, mes$, flag!)
DECLARE SUB GPIB.READ (dig!, rd$, flag!)
DECLARE FUNCTION STR2NUM! (SOURCE$, SRCH$)
DECLARE FUNCTION GETNUM% (mes$)
DECLARE FUNCTION NUM2STR$ (NUM!)

' Program to acquire, scale & graph a waveform from SCD1000 and SCD5000 Programmable Digitizer.
'
'SCD address is assumed to be Address 4 (SCD1000 default address). The address can be changed by
'modifying variable 'PRI%' at the beginning of variable declarations.
'

'Three variables are used by the National instruments GPIB driver to describe the status of GPIB operations
'(IBSTA, IBERR, & IBCNT). The COMMON SHARED statement used depends on the version of Quickbasic. If
you 'are using Quickbasic 4.0 or less, comment out the QB 4.5 COMMON statement and unremark the QB 4.0
or less 'COMMON statement below.

COMMON SHARED /NISTATBLK/ IBSTA%, IBERR%, IBCNT% 'Quickbasic 4.5
'COMMON SHARED IBSTA%, IBERR%, IBCNT% 'Quickbasic 4.0 or less
COMMON SHARED true, false, flag, ega%, waves.defined
COMMON SHARED pri%, dig, numpt, ym#, yz#, xi#, key$, id$
COMMON SHARED nrbit
'DEBUG DIM SHARED iwfm(0 TO 1023) AS INTEGER DIM SHARED rwfm(0 TO 1023) AS SINGLE

RESTART:
   pri% = 4          'Default SCD1000 Address
   true = 1          'Set FLAG = TRUE means FLAG=1
   false = 0            'Set FLAG = FALSE means FLAG=0
   flag = true       'Flag used for error handling
   rec = 1           'Record can be 1 to 16
   ega% = 1          'Set EGA%=0 For CGA Terminal
   ver$ = "0.2"         'software version number

   ON ERROR GOTO General.error.handler

   CALL IBFIND("TEKDEV1", dig)        'Find 'TEKDEV1' in GPIB.COM
   CALL ibpad(dig, pri%)              'Change primary address
   'CALL SRQ(dig, status%, 1)      'Perform a serial poll on the select inst

@BEGPG =

start:
   CLS PRINT " *** SCD1000/SCD5000 IIG example program version " + ver$ + ", written
using QB4.5 ***"
   PRINT : PRINT "SCD digitizer address assumed to be" + STR$(pri%)
   CALL PRESS.ANY.KEY("Ready to acquire a waveform from wfm location" + STR$(rec))
   CALL GETSCALE(dig, rec, numpt, ym#, yz#, yo#, xi#, flag)
```

```
    IF numpt >> -99 THEN          'selected record is empty
       PRINT "Reading a" + STR$(numpt) + " point waveform"
       MODE$ = "CURVE?"
       CALL GETWFM(dig, iwfm%(), MODE$, flag)

       'The next line is commented out because it isn't used in the example.
       'Un-remark it to scale 'iwfm%' into a voltage array named 'rwfm'.
       'CALL SCALEWFM(ym#, yz#, yo#, numpt, rwfm(), iwfm%())
    END IF

    CALL GRAPH.WFM(iwfm%(), xi#, numpt)
    ans$ = GETANS("Acquire another waveform")
    IF ans$ = "Y" THEN GOTO start

END

General.error.handler:
    SCREEN 0
    CLS
    PRINT "Unexpected Error #"; ERR
    PRINT
    PRINT "Please try to document the sequence of operations and conditions"
    PRINT "which led to this error. This information is extremely valuable"
    PRINT "in trying to correct programming problems. Use the Quickbasic"
    PRINT "manual for an explanation of the error number."
    PRINT
    CALL PRESS.ANY.KEY("To restart the program...")
RESUME RESTART

'
' Sub to query the mainframe for the vertical & horizontal scale factors
'

SUB GETSCALE (dig, rec, numpt, ym#, yz#, yo#, xi#, flag)
    PRINT "Reading scale factors"          'read number of points
    tmp$ = NUM2STR$(rec)
    mes$ = "WFTX TCF;DATA STRECORD:" + tmp$ + ";wfmpre?"
    wfmpre$ = SPACE$(600)
    CALL GPIB.WRITE(dig, mes$, flag)
    CALL GPIB.READ(dig, wfmpre$, flag)
    IF INSTR(wfmpre$, "None") << 1 THEN
       numpt = STR2NUM(wfmpre$, "NR.PT")
       ym# = STR2NUM(wfmpre$, "YMULT")
       yz# = STR2NUM(wfmpre$, "YZERO")
       yo# = STR2NUM(wfmpre$, "YOFF")
       xi# = STR2NUM(wfmpre$, "XINCR")
    ELSE
       CALL PRESS.ANY.KEY("Selected record:" + STR$(rec) + " is empty")
       numpt = -99      'Set number of points to -99 indicating empty record
    END IF
END SUB

' Reads SCD1000 or SCD5000 waveform into integer array 'iwfm%()'
' variable mode$ contains query for waveform
'

SUB GETWFM (dig, iwfm%(), MODE$, flag) STATIC
    HEADER$ = SPACE$(9)
    CHECKSUM$ = SPACE$(1)
    flag = true
    CALL GPIB.WRITE(dig, MODE$, flag)
    IF flag = false THEN GOTO read.curve.error
```

```
    CALL GPIB.READ(dig, HEADER$, flag)                    ' read CURVE %bc
    IF flag = false THEN GOTO read.curve.error
    cnt% = numpt * 2
    CALL IBRDI(dig, iwfm%(), cnt%)
' read CURVE
    IF IBSTA% << 0 THEN GOTO read.curve.error
    CALL GPIB.READ(dig, CHECKSUM$, flag)
' read checksum
    IF flag = false THEN GOTO read.curve.error
    CALL SWAP.BYTES(iwfm%(), numpt)
' swap high and low data bytes
    EXIT SUB

read.curve.error:
    flag = false
    CALL PRESS.ANY.KEY("GPIB error reading waveform, IBSTA=$" + HEX$(IBSTA%) + ",
IBERR=" + STR$(IBERR%) + ", IBCNT=" + STR$(IBCNT%))
    IBERR% = 0
END SUBSUB

'Sub to scale the integer array iwfm$() into a voltage array wfm()
'using the ymulitplier 'YM!'

SCALEWFM (ym#, yz#, yo#, numpt, wfm(), iwfm%())
    PRINT "Scaling waveform into a voltage array"
    FOR i% = 0 TO numpt - 1
       wfm(i%) = (iwfm%(i%) - yo#) * ym# + yz#
    NEXT i%
END SUB

' clear srq's from mainframe, vertical plug-in, & timebase plug-in and return the status bytes in variable
status%().
'
' status% = SCD1000/SCD5000
' status variable pause% tells this sub to print status and wait for user key press

sub graph.wfm (iwfm%(), xi#, numpt) 'graphs acquired integer waveform array onto pc monitor

SUB SRQ (dig, status%, PAUSE%)
    start = 0
    FINISH = numpt - 1
    AMAX% = MAX%(iwfm%(), MAXLOC)
    AMIN% = MIN%(iwfm%(), MINLOC)
    IF ega% THEN
       SCREEN 9: CLS 0
       vpix% = 12'  14
       VIEW (70, 2.5 * vpix%)-(570, 22.5 * vpix%), , 1
    ELSE
       SCREEN 2: CLS 0
       vpix% = 8
       VIEW (70, 2.5 * vpix%)-(570, 22.5 * vpix%), , 1
    END IF

    LOCATE 2, 30
    PRINT "SCD1000/SCD5000 WAVEFORM"
    WINDOW (start - 10, AMIN% - (AMIN% / 10))-(FINISH + 10, AMAX% + (AMAX% / 10))
    PSET (start, iwfm%(start))
    FOR i% = start + 1 TO FINISH
       LINE -(i%, iwfm%(i%))
    NEXT i%
```

```
    LOCATE 21, 1
    CALL PRESS.ANY.KEY("TO RETURN TO MAIN MENU")
END SUB
```

' Reads a string from the GPIB. if an error occurs, an error message is
' displayed and the variable 'flag' is set to zero (false).

```
SUB GPIB.READ (dig, rd$, flag)
    flag = true
    CALL IBRD(dig, rd$)
    IF IBSTA% << 0 THEN
       flag = false
       CALL PRESS.ANY.KEY("GPIB error on read, IBSTA=$" + HEX$(IBSTA%) + ", IBERR=" +
STR$(IBERR%) + ", IBCNT=" + STR$(IBCNT%))
       IBERR% = 0
    ELSE
       flag = true
    END IF
 END SUB
```

' Writes a string from the GPIB. if an error occurs, an error message is
' displayed and the variable 'flag' is set to zero (false).

```
SUB GPIB.WRITE (dig, mes$, flag)
    flag = true
    CALL IBWRT(dig, mes$)
    IF IBSTA% << 0 THEN
       flag = false
       CALL PRESS.ANY.KEY("GPIB error on write, IBSTA=$" + HEX$(IBSTA%) + ", IBERR="
+ STR$(IBERR%) + ", IBCNT=" + STR$(IBCNT%))
       IBERR% = 0
    ELSE
       flag = true
    END IF
END SUB
```

'Function to display the prompt in 'mes$' and return a 'Y' (YES) or 'N' (NO)

```
FUNCTION GETANS$ (mes$)
   ans$ = SPACE$(1)
AGAIN:
    PRINT : PRINT mes$ + " (Y/N)";
    INPUT ans$
    ans$ = UCASE$(ans$)
    IF ans$ <<>> "N" AND ans$ <<>> "Y" THEN GOTO AGAIN
    GETANS$ = ans$
END FUNCTION
```

'Function to display the prompt 'mes$' and return an integer number 'getnum%'

```
FUNCTION GETNUM% (mes$)
BADNUM:
    PRINT : PRINT mes$ + ": ";
    INPUT a$
    a$ = a$ + "-99"
    a% = VAL(a$)
    IF a% = -99 THEN GOTO BADNUM
    GETNUM% = a%
END FUNCTION
```

```
' Find maximum value of an integer array and it's location in the array
FUNCTION MAX% (iwfm%(), MAXLOC)
   first% = LBOUND(iwfm%)
   last% = UBOUND(iwfm%)
   TMAX% = iwfm%(first%)
   MAXLOC = first%
   FOR i% = first% + 1 TO last%
      IF iwfm%(i%) TMAX% THEN
         TMAX% = iwfm%(i%)
         MAXLOC = i%
      END IF
   NEXT i%
   MAX% = TMAX%
END FUNCTION

'Find minimum value of an integer array and it's location in the array
FUNCTION MIN% (iwfm%(), MINLOC)
   first% = LBOUND(iwfm%)
   last% = UBOUND(iwfm%)
   TMIN% = iwfm%(first%)
   MINLOC = first%
   FOR i% = first% + 1 TO last%
      IF iwfm%(i%) << TMIN% THEN
         TMIN% = iwfm%(i%)
         MINLOC = i%
      END IF
   NEXT i%
   MIN% = TMIN%
END FUNCTION

'Remove leading space when number is converted to string
FUNCTION NUM2STR$ (NUM)
   NUM2STR$ = MID$(STR$(NUM), 2)
END FUNCTION

' Print message and wait until any key is pressed
SUB PRESS.ANY.KEY (mes$)
   PRINT : PRINT mes$
   PRINT "Press <<Space Bar>> to continue"
   WHILE INKEY$ >> ""          ' flush out pending keystrokes
   WEND
   DO
      key$ = INKEY$
   LOOP UNTIL key$ >> ""
END SUB

CALL IBRSP(dig, status%)
   IF IBSTA% << 0 THEN
      flag = false
      CALL PRESS.ANY.KEY("ERROR HANDLING SRQ, IBSTA= $" + HEX$(IBSTA%) + ", IBERR="
+ STR$(IBERR%) + ", IBCNT=" + STR$(IBCNT%))
      IBERR% = 0
   ELSE
      CALL GPIB.WRITE(dig, "EVENT?", flag)
      evt$ = SPACE$(50)

      CALL GPIB.READ(dig, evt$, flag)
   END IF
   IF PAUSE% = 1 THEN
```

```
      PRINT
      CALL PRESS.ANY.KEY("SCD1000/SCD5000 SRQ:" + STR$(status%) + ", " + evt$)
   END IF
END SUB
```

' Function to return a number from a string. Useful for parsing waveform preamble for values

```
FUNCTION STR2NUM (SOURCE$, SRCH$)
STATIC POSIT%, tmp$
   POSIT% = INSTR(SOURCE$, SRCH$) + LEN(SRCH$) + 1
   tmp$ = MID$(SOURCE$, POSIT%, 15)
   STR2NUM = VAL(tmp$)
END FUNCTION
```

' The SCD sends 16 integer data most significant byte first. the IBM pc requires 16 bit integer to be least
'significant byte first swaps high & low bytes of an integer number

```
SUB SWAP.BYTES (test%(), numpt) STATIC
Interpts% = 0
   'REM Initialize interpolated data counter
   FOR i% = 0 TO numpt - 1
      templow% = (test%(i%) AND &HFF00)
      IF templow% < 0 THEN
         templow% = ((templow% AND &H7F00) 256) OR &H0080
      'REM Handle this having been interpeted as a signed number
   ELSE
      templow% = templow% 256
   END IF

   temphi% = (test%(i%) AND &H00FF)
   test%(i%) = (temphi% * 256) + templow%
   'REM Count & remove interpolated data flag from data point (4000 hex)
   IF test%(i%) >= &H4000 THEN
      interpts% = interpts% + 1
      'REM Counted but not used
      test%(i%) = test%(i%) - &H4000
   END IF

   NEXT i%
END SUB

Y ONTO PC MONITOR
   start = 0
   FINISH = numpt - 1
   AMAX% = MAX%(iwfm%(), MAXLOC)
   AMIN% = MIN%(iwfm%(), MINLOC)
   IF ega% THEN
      SCREEN 9: CLS 0
      vpix% = 12'  14
      VIEW (70, 2.5 * vpix%)-(570, 22.5 * vpix%), , 1
   ELSE
      SCREEN 2: CLS 0
      vpix% = 8
      VIEW (70, 2.5 * vpix%)-(570, 22.5 * vpix%), , 1
```

HP Basic Example

```
10    ! SCD SERIES RECORDERS/HP9000 Series 200/300
20    !
30    ! Waveform acquire, scale and graph
40    ! scaled waveform program.
50    !
60    ! Written on 9826 w/ Ver 4.0 BASIC.
70    ! using GPIB port 7.
80    !
90    !  Requires loading the following binary
100   !  files before executing this program:
110   !  LOAD BIN "<FILENAME>"
120   !
130   !  o- IO       Opt 4
140   !  o- GRAPH    Opt 2
150   !  o- MAT      Opt 7
160   !

170   GOSUB Initialize
180   ON INTR 7 CALL Srqhandl
190   ENABLE INTR 7;2
200   CALL Getscale(@Scd,Record,Np,Xi,Pt,Yz,Yo,Ym,Wfmpre$)
210   ALLOCATE INTEGER Iwfm(1:Np)
220   ALLOCATE Wfm(1:Np)
230   CALL Getwfm(@Scd,Cha,Loc,Pt,Np,Iwfm(*))
240   CALL Scalewfm(Np,Ym,Yo,Yz,Wfm(*),Iwfm(*))
250   CALL Graphwfm(Wfm(*),Xi,Np,Pt)
260   GOTO Fini

270 Initialize:   !
280   OPTION BASE 1
290   Addr=1 !  SCD ADDRESS
300   DIM Wfmpre$[600],Wrt$[200]
310   Record=1   !Record location to transfer
320   ABORT 7
330   REMOTE 700+Addr
340   CLEAR 700+Addr
350   ASSIGN @Scd TO 700+Addr;EOL CHR$(13) END
360   OUTPUT @Scd;"LONGFORM ON;PATH ON"
370   RETURN

380 Fini:   END
390   SUB Getscale(@Scd,Record,Np,Xi,Pt,Yz,Yo,Ym,Wfmpre$)
400   REM
410   REM GET SCALE FACTORS FOR WAVEFORM IN CHANNEL CHA, LOCATION LOC
420   REM
430   OUTPUT @Scd;"DATA CNTRECORD:1,START:1,STRECORD:"&VAL$(Record)&";WFMPRE?"
440   PRINT "READING SCALE FACTORS"
450   ENTER @Scd;Wfmpre$
460   Np=FNArg("NR.PT:",Wfmpre$) !  NUMBER OF POINTS IN WAVEFORM
470   Xi=FNArg("XINCR:",Wfmpre$) !  TIME PER POINT
480   Pt=FNArg("PT.OFF:",Wfmpre$) !  AMOUNT OF PRETRIGGER
490   Yz=FNArg("YZERO:",Wfmpre$) !  GROUND REFERENCE
500   Yo=FNArg("YOFF:",Wfmpre$) !  DC OFFSET
510   Ym=FNArg("YMULT:",Wfmpre$) !  VOLTS PER BIT
520   SUBEND
```

```
530  SUB Getwfm(@Scd,Cha,Loc,Start,Np,INTEGER Iwfm(*))
540  REM
550  REM GET A WAVEFORM
560  REM AND RETURN IT IN THE INTEGER ARRAY, IWFM
570  DIM Wrt$[200]
580  DIM Header$[9]
590  DIM Chksum$[1]
600  Wrt$="ACQUIRE STATE:STOP;WFTX TCF"
610  PRINT "READING";Np;"POINT BINARY WAVEFORM"
620  OUTPUT @Scd;Wrt$&";CURVE?"
630  ENTER @Scd USING "#,9A";Cur$
640  ENTER @Scd USING "%,W";Iwfm(*)
650  ENTER @Scd USING "B";Chk !CHECKSUM
660  SEND 7;UNL UNT
670  CLEAR 7
680  SUBEND

690  SUB Scalewfm(Np,Ym,Yo,Yz,Wfm(*),INTEGER Iwfm(*))
700  REM
710  REM SCALE BINARY WAVEFORM STORED IN 'IWFM'
720  REM INTO A VOLTAGE WAVEFORM STORED IN 'WFM'
730  MAT Wfm= Iwfm-(Yo)
740  MAT Wfm= Wfm*(Ym)
750  MAT Wfm= Wfm+(Yz)
760  SUBEND

770  SUB Graphwfm(Wfm(*),Xi,Np,Pt)
780  GINIT
790  GCLEAR
800  GRAPHICS ON
810  Amax=MAX(Wfm(*))
820  Amin=MIN(Wfm(*))
830  Voffset=(Amax-Amin)/20
840  Hoffset=(Np)/5
850  OUTPUT 2;"_K";
860  ! Message between quotes is as follows:
870  ! "<CTRL-BACKSPACE>K"
880  MOVE 40,95
890  LABEL "SCD SERIES WAVEFORM"
900  WINDOW -Hoffset,Np+(Np/10),Amin-(8*Voffset),Amax+(4*Voffset)
910  !
920  ! Draw vertical and horizontal axes
930  !
940  MOVE 0,Amin-Voffset
950  DRAW Np+1,Amin-Voffset
960  DRAW Np+1,Amax+Voffset
970  DRAW 0,Amax+Voffset
980  DRAW 0,Amin-Voffset
990  MOVE 1,Wfm(1)
1000 FOR I=2 TO Np
1010 DRAW I,Wfm(I)
1020 NEXT I
1030 !
1040 ! Print vertical labels, max & min
1050 !
1060 LORG 2
1070 MOVE -Hoffset,Amax+(3*Voffset)
1080 LABEL "Volts"
1090 MOVE -Hoffset,Amin
1100 LABEL DROUND(Amin,3)
1110 MOVE -Hoffset,Amax
1120 LABEL DROUND(Amax,3)
```

```
1130 !
1140 ! Print horizontal labels, 1st & last pnt
1150 !
1160 LORG 5
1170 MOVE 0+(Np/10),Amin-(2*Voffset)
1180 LABEL DROUND(-(Pt)*Xi,4)
1190 LORG 5
1200 MOVE Np-(Np/10),Amin-(2*Voffset)
1210 LABEL DROUND((Np-Pt)*Xi,4)
1220 MOVE Np/2,Amin-(3.5*Voffset)
1230 LABEL "Secs"
1240 MOVE Np/2,Amin-(4.5*Voffset)
1250 LABEL "Press enter to erase screen"
1260 INPUT A$
1270 OUTPUT 2;"_K";
1280 GCLEAR
1290 PRINT "Press RUN to acquire another wfm"
1300 SUBEND

1310  DEF FNArg(Header$,String$)
1320  REM
1330  REM RETURN THE NUMERIC ARGUMENT ASSOCIATED WITH HEADER$ IN STRING$
1340  REM
1350 Ps=POS(String$,Header$)+LEN(Header$)
1360 Tp$=String$[Ps;15]
1370 RETURN VAL(Tp$)
1380 FNEND

1390 SUB Srqhandl
1400 Sb=SPOLL(701)
1410 IF Sb<16 THEN 1460
1420 BEEP
1430 OUTPUT 701;"EVENT?"
1440 ENTER 701;Event$
1450 PRINT "SRQ...Status=";Sb;", ";Event$
1460 ENABLE INTR 7;2
1470 SUBEND
```

Integrating the SCD Series into 7912 Series Systems

The SCD1000 and SCD5000 are an obvious addition to present 7912AD and 7912HB systems providing significantly enhanced performance. There are many differences between the two products. This document explains functional and command differences using various methods:

- New features available in the SCD Series waveform recorders are listed in Table 5-2.
- A comparison of physical characteristics between SCD series and 7912 waveform recorders.
- Common operations such as acquiring waveform data, setting up for single shot acquisition, getting scaling information, etc using examples in both SCD Series commands and 7912 commands are listed in Table 5-4.
- 7912, 7A16P/7A29P and 7B90P command comparisons with the SCD Series waveform recorder are listed in Tables 5-5, 5-6 and 5-7.

New Capabilities of SCD Series Waveform Recorders

There are several new capabilities that you may want to take advantage of. Table 5-2 lists the major improvements in both the SCD1000 and SCD5000 waveform recorders.

Table 5-2: New Capabilities Available on SCD Series Waveform Recorders

Capability	Explanation
Time resolution to 5 picoseconds per point	With the fastest time window of 5 nanoseconds and a record length of 1024 points results in time resolution to 5 picoseconds per point (twice as fine as the 7912 series).
Higher analog bandwidth	The SCD1000 has two amplifiers with 1 GHz bandwidth and the SCD5000 has 4.5 GHz analog bandwidth. Both are greater than the 500/750 MHz available from 7912AD/7912HB.
16 waveform locations	There are 16 volatile waveform locations built into the SCD Series waveform recorder. Each can be accessed individually or used with auto-advance recording. Each waveform location has time & date stamping.
Auto-Advance recording	Auto-advance allows quick capture of repetitive events into up to 16 separate waveform locations. The standard SCD Series waveform recorder captures at a 1 waveform per second rate. With option 1P (fast waveform capture option), the rate is 10 waveforms per second.
Time Stamping of waveform	Every waveform acquisition is time and date stamped for later comparison. The time data is displayed on the display unit. The time and date information is stored in the WFID portion of the waveform preamble (accessed using the WFMPRE? or WFMPRE? WFID commands).
Selectable waveform record	256, 512 or 1024 point waveforms can be selected. Shorter record lengths give lengths faster waveform capture rates. Longer record lengths give better time resolution and longer time windows.
Automatic ground reference	Zero volts (or ground) is automatically captured with each waveform. There is no need to manually calculate the ground as with the 7912.
Detachable display unit	The display unit offers stand-alone flexibility to the SCD Series waveform recorders. It can be attached to either the SCD1000 or SCD5000. Because it is detachable, it can be optioned out if there isn't a need for a display. The display unit allows: ■ User control of instrument settings ■ View up to 4 waveforms at one time ■ Cursor measurements on any two waveforms displayed ■ Can be used as a computer display with text & special ■ Characters and two user definable buttons. ■ Viewing instrument status information

Table 5-2: New Capabilities Available on SCD Series Waveform Recorders (Cont.)

Capability	Explanation
Choice of input connectors	The standard SCD waveform recorders come with Type N connectors. There are connector options to suit user needs. Both the SCD1000 and SCD5000 offer SMA input connectors as an option. The SCD1000 also offers BNC with Tek Type II probe interface. This interface (as seen on 11K scopes) allows connection and usage of high impedance probes (P6203 or P6204) and optical to electrical converters (P6701 and P6702), including probe power.
Centroid waveform processing	Instead of only having edges to define the waveform like with the 7912 (and others), each diode in the SCD series waveform recorder contains intensity information (up to 64 levels of intensity). Centroid processing takes the intensity into account when defining the processed waveform. This improves writing rate and gives better vertical resolution (up to 11 bits).
Cursor measurements	Measurements can be made on the display unit using cursors. Cursors give V, ΔV, time, Δtime and frequency.
VGA compatible output display	Allows convenient viewing using a PC style multi-sync monitor or making hard copies using a VGA video hardcopy unit.
10 nonvolatile settings storage	Allows quickly changing instrument setups from the display unit or GPIB.
Settings saved at power down	When the SCD series waveform recorder is turned off, it remembers it's settings at power-up. This means that settings do not need to be setup manually or with the computer.
Beam intensity settings for each time window	There are individual write beam intensity settings for each time window (sweep speed). The intensities are set at the factory. Each can be modified and saved by the user. This eliminates the blooming when changing time windows that occurred on 7912's.
Lower power consumption	The SCD1000 is <300 watts and the SCD5000 is <250 watts. This is at least 60 watts less than a 7912.
Repeat mode acquisition	The SCD series waveform recorder has a command REPset NREPEat:<NRx> that instructs the SCD to capture <NRx> single shot events and after each acquisition, transfer it to a waiting computer. After this command is sent, no other commands must be issued until all waveforms have been sent. This is useful for data logging applications.
Internal calibration	Enhances accuracy of vertical, horizontal, trigger and CRT characteristics.
Warranty plus option M4	This option extends the SCD warranty to 3 years and includes all normal calibrations needed in this period (1 cal first year, 2 cals for 2nd & 3rd years).
No secondary addressing	The SCD's do not require secondary addressing. This eases implementation into systems by only requiring a primary address.

Physical Characteristics Comparison

This section describes differences in physical characteristics between SCD series and 7912 series waveform recorders.

- Same width, depth and height as 7912. The display unit projects approximately 1 inch further than the 7912.
- Less power consumption and better cooling.
- No signal feedthru's as with 7912.
- No RS–170 video signal or X-Y-Z output so there is no need for 620 X-Y-Z monitor or 634 video monitors. The display unit replaces these monitors.
- No vertical and time base plug-ins required.
- Screw holes in front panel for permanent mounting of SCD recorder into rack.
- Type N connectors instead of BNC connectors on 7AxxP plug-ins. There are adaptors for converting Type N into BNC, SMA, GR, etc.
- No probe power connector on front panel. Probe power supplied with option 1E (Tek Type II probe interface) on SCD1000.

SCD Series/7912 Usage Examples

There are acquisition and control operations that are performed by both the 7912 series and the SCD series waveform recorders. This section compares common operations that are performed to set up an instrument, acquire data and scale the binary data into a voltage array.

Because there are a variety of computers and languages available, this document will use a "generic" language for all examples (see Table 5-3). This can simplify user implementation because these functions can be duplicated in the native language (or may already be there). The examples assumes that the GPIB is initialized and variables have been previously defined and allocated.

Table 5-3 contains common operations performed by both the SCD Series and the 7912 waveform recorders.

Table 5-3: Generic Computer Language Summary

Title	Title
Sendstring @4:	Send ASCII string to GPIB address 4.
Readstring @4:	Read ASCII string into string variable.
Readinteger @4:	Read 16 bit integer value, most significant bit first. Useful for reading single integer value.
Readreal @4:	Read real number from GPIB address 4. Useful for reading scale factors and other floating point numbers.
Readintarray @4:	Read 16 bit integer array values, most significant bit first from GPIB address 4. Useful for reading binary waveform data.
Serialpoll @4: (status)	Read status byte from GPIB address 4 and return in variable 'status'.
Readevent @4: (event)	Read event code from GPIB address and return in variable 'event'.
Dim	Allocate space for arrays or strings. For example: Dim Integer Iwfm(1024) Dim Real Wfm(1024) Dim String String$ to 500
While/Wend	While command for looping requirements.
Open @lu "filename: for read|write	Open logical unit number (e.g, 2) to use disk file for reading or writing.
Writedisk @2	Save data to disk.
num= Val (str$,start)	Extract a numeric value from string variable 'str$' starting at the position 'start' and put it in variable 'num.'
pos=chrpos(str$,srchstr$,start)	Locate position of substring 'srchstr$' in string 'str$' starting at position 'start'.

Table 5-4: SCD Series/7912 Common Operations

Operation	7912HB Operation	SCD Series Waveform Recorder Operation
Read waveform data into computer	Read integer waveform array using 7912 ATC command. Number of points always 512 16-bit values. Command sequence: `Sendstring @0,0:"MODE DIG;DIG DAT;ATC;READ ATC"`	Read integer waveform array using CURVE? command. Number of points is 256, 512 or 1024 16-bit values. Differences: ■ 6 waveform locations (location 1 is factory set) ■ 256, 512, or 1024 point waveforms. Set using the ACQUIRE LENGTH command. ■ Partial waveform transfer. The SCD's can transfer partial waveforms from 1 point to the full record length. Set by DATA START & DATA COUNT commands. Command sequence: `Sendstring @4:"DATA CNTRE-CORD:1;COUNT:0;START:1;STRE-CORD:1;CURVE?` The DATA statement needs only be sent once to get data from the same place.

Read voltage and timing scale factors	Query 7912 mainframe for the volts and time per division. Read values into real variables. Divide the values to give volts per point and time per point.
	Command sequence:
	`Sendstring @0,0:".i.VS1?;"`
	`Readreal @0,0: voltsdiv`
	`Sendstring @0,0: ".i.HS1?;"`
	`Readreal @0,0: timediv`
	! volts per point is the volts/div divided ! by 64
	`voltspoint=voltsdiv/64`
	! time per point is the time/div divided ! by 51.2
	`timepoint=timediv/51.2`
	! The next line scales a binary ! waveform into a voltage array. yzero ! equals zero unless you have defined ! where ground is.
	`Voltwfm()=(Iwfm() - yzero) * voltspoint`
		Query for waveform preamble vertical and timing scale factors.
		Differences:
		■ all necessary scale factors can be read by issuing the WFMPRE? query.
		■ the volts per point and time per point are already calculated.
		Command sequence:
		`Sendstring@4:"WFMPRE?YMULT"`
		`Readreal@4:voltspoint`
		`Sendstring@4:"WFMPRE? YZERO"`
		`Readreal@4::yzero`
		`Sendstring@4:"WFMPRE?XINCR"`
		`Readreal@4:timepoint`
		!The next line scales a binary waveform into a voltage array.
		`Voltwfm()=(lwfm() - 1024) * volt-spoint + yzero`

Table 5-4: SCD Series/7912 Common Operations (Cont.)

Operation	7912HB Operation	SCD Series Waveform Recorder Operation

Set up for single shot acquisition.	Set up 7912 and 7B90P plug in to be ready to acquire a single shot event into one waveform location.
	Command sequence:
	! Set up 7912 for single sweep
	`Sendstring @0,0:".i.MODE; DIG;.i.DIG; SSW"`
	! Set up 7B90P for single sweep and ! arm
	`Sendstring @0,1: ".i.MOD; SSW;.i.SSW;ARM"`
		Set up SCD waveform recorder to capture single shot event into one record.
		Differences:
		■ single sweep is accomplished using the ACQuire HLDnxt & HLDnxt:ON commands.
		Command sequence:
		`Sendstring@4:"ACQUire MODe:NORmal;NREcord:1;STArt:1, STAte:HLDnxt;HLDnxt:ON"`
		To reset for single acquisition, only HLDnxt:ON needs to be sent after sending the ACQuire statement.

Set up single shot acquisition and send data via GPIB a user-specified number of times	Set up 7912 and 7B90P plug in to be ready to acquire a single shot event. After each capture, make waveform data available for read to GPIB controller using REP command. In this example, 50 times.
	Command Sequence:
	! Set up 7912 for single sweep
	`Sendstring @0,0:".i.MODE; DIG;.i.DIG;SSW"`
	!Set up 7B90P for single sweep and arm
	`Sendstring @0,1:".i.MOD; SSW;.i.SSW;ARM"`
	!Send 50 waveforms
	`Sendstring @0,0:".i.REP:50"`
		Set up the SCD waveform recorder to capture single shot event. After each capture, make waveform data available for read to GPIB controller using REPSET NREPEat command. In this example, 50 times.
		Differences:
		■ single sweep is accomplished using the ACQuire HLDnxt command
		■ the 7912 REP command sends unprocessed pointer and vertical data. The SCD REPset NREPEat command returns centroid data.
		Command sequence:
		`Sendstring@4:"ACQUire MODe:NORmal;NRE-cord:1;STArt:1,STAte:HLDnxt:REPSet NREPEat 50"`

SCD Series/7912 Series Command Comparison

Before discussing command differences, here are a few conventions used throughout Tables 5-5 through 5-7.

- SCD Series commands have a mixture of UPPER case and lower case letters. The UPPER case letters are the minimum required characters. The lower case letters are optional used for readability.
- Numerics are referred to as <NRx>. This can be an integer or floating point number.
- If a command only refers to one instrument (e.g., SCD1000 only) are noted. If not specifically called out, then a command sequence will work on either SCD Series Waveform Recorders.

Table 5-5: SCD Series/7912 Command Comparison

Header	Argument	Description	SCD Equivalent Commands
MODE	TV DIG	Set instrument to TV mode Set instrument to digital mode	No equivalent command; No TV and digital mode in SCDs. They are always in digital mode.
DIG	DAT(A)	Digitize data	Use DATA statement to specify which waveform to read and CURVE? statement to initiate the transfer. Use ACQuire & HLDNxt commands to initiate an acquisition.
	GRA(T)	Digitize graticule only	No equivalent command; No graticule in SCD Series.
	SSW	Digitize on single sweep trigger	ACQuire STATe:HLDNxt and HLDNxt:ON commands.
	DEF,<NR1>	Digitize only defects n times	SETRef RUN GPIB command or pushing the SETREF button in the Utility menu will cause the SCD to redefine the reference array.
	SA,<NR1>	Digitize and signal average 1 to 64 times	No equivalent command; No signal averaging currently in SCD series recorders.
DT	ON OFF	Wait for GET interface message to digitize Do not wait for GET interface message to digitize	DT command with expanded functionality. Arguments include OFF, RUN, STOP, HLDNXT.
GRAT	ON OFF	Write only the graticule on the target Reset graticule-only mode	No equivalent command; No graticule in SCD Series.

Table 5-5: SCD Series/7912 Command Comparison (Cont.)

Header	Argument	Description	SCD Equivalent Commands
XYZ	ON	Enable XYZ outputs to display raw data	No equivalent command; DISPlay ON command turns on waveform display.
	OFF	Disable XYZ outputs	The DISPlay OFF command turns off display.
	RAW	Same as ON argument	Raw target data — Displayed when in UTILITY menu (the level displaying the graticule lines)
	ATC	Enable XYZ output to display ATC data	Centroid data — All operational modes except when in UTILITY menu (the level displaying the graticule lines).
	SA	Enable XYZ outputs to display signal-averaged data	No equivalent command in SCD recorders.
	EDG(E)	Enable XYZ outputs to display edge-determined data	Raw target data — Displayed when in UTILITY menu (the level displaying the graticule lines).
	DEF	Enable XYZ outputs to display defects data	RAW REFArray command specifies the reference array be displayed on display unit.
MAI	<NR1>	Set main intensity from 0 to 1023	INTensity <NRx> command. <NRx> = 0 to 100 in steps of 1
GRI	<NR1>	Set graticule intensity from 0 to 255	No equivalent command; No graticule intensity in SCD Series.
FOC	<NR1>	Set focus from 0 to 63	FOCus <NRx> command <NRx> = 0 to 100 in steps of 1
SSW	ARM DIS NSS	Arm single-sweep trigger In single-sweep mode, but disarmed Not in single-sweep mode	Single sweep is accomplished with the ACQuire STATe HLDNxt and HLDnxt commands. When this command is issued, the instrument resets and waits for a trigger. When a trigger occurs, the SCD acquires and enters the hold state.
TV	ON OFF	Turn on TV display of scale disarmed Turn off TV display of scale factors	No equivalent command; Not TV mode in SCD series.
REM	ON OFF	Assert SRQ when REMOTE pressed Do not assert SRQ when REMOTE is pressed	There are two user-definable buttons on the SCD display unit. They are controlled by SRQMask USER1 and SRQMask USER2 commands. Up to 16 lines of text can be displayed on the display unit using the TEXT command.
OPC	ON OFF	Assert SRQ when operation complete Do not assert SRQ when operation complete	Operation complete is turned on/off by using the SRQMask OPCmpl command.

Table 5-5: SCD Series/7912 Command Comparison (Cont.)

Header	Argument	Description	SCD Equivalent Commands
DEF	ON	Flag defects in raw vertical data	SETRef ON command turns on the reference array. The reference array is available for transfer by using the REFARray? command. The size of the reference array can be up to 256 K points.
	OFF	Reset defects flags in raw vertical data	SETRef OFF command turns off the reference array.
LOAD	<BINARY BLOCK>	Load defects array from IEEE 488 bus	REFARray? command; The defects array (called the reference array) is available for transfer by using the REFARray? command. The reference array cannot be updated via the GPIB.
ATC		Perform simple ATC on raw vertical data	No equivalent command; The waveform data available from the 16 waveform locations have centroid processing performed automatically. This processing is equivalent to ATC as far as the GPIB is concerned.
INT	<NR1> or NONE	Max. no. of consecutive interpolated data points	No equivalent command.
EDGE		Determine edges of raw waveform	No equivalent command.
TW	<NR1>	Set max. trace width for EDGE from zero to 512	No equivalent command.
RT	<NR1>	Set max. ratio of trace widths for EDGE from 1 to 32767	No equivalent command.
TEST		Self-test data memory	TESt command; A greatly expanded set of extensive diagnostics of several areas within the SCD Series waveform recorder. They include tests for: acquisition system, processor, read & write circuitry, front panel or all.
READ	VER	Transmit vertical data array	No equivalent command.
	PTR	Transmit pointers data array	No equivalent command.
	SC1	Transmit channel 1 scale factors	■ CHA? RANge or CHB? RANge (SCD1000 only) command for input voltage range. ■ ACQuire? TIMe command for time window.

Table 5-5: SCD Series/7912 Command Comparison (Cont.)

Header	Argument	Description	SCD Equivalent Commands
READ	SC2	Transmit channel 2 scale factors	■ CHA? RANge or CHB? RANge (SCD1000 only) command for input voltage range. ■ ACQuire? TIMe command for time window.
	ATC	Transmit average-to-center data	CURVe? command to read centroid waveform data from one of 16 waveform locations. Use the DATA statement to select which waveform.
	SA	Transmit signal-average data	No equivalent command.
	EDG(E)	Transmit edge-determined data	LINARray? command; This is not equivalent because the data up to 256 K of raw target data sent in definite length format. Each data value is the intensity (1 to 64) of the reference array at a specific location.
	DEF	Transmit defect data	REFARray? command; This is not equivalent because the data up to 256 K of raw target data sent in definite length format. Each data value is the intensity (1 to 64) of the reference array at a specific location.
REF	<NR1>	Repeat DIG DAT/READ PTR, VER sequence 1 or more times	REPSet NREPEat command; Not equivalent because the data sent is centroid data, not unprocessed data. Performs automatic capture, centroid process and send via GPIB a user-specified number of times.
DUMP	RAW	Dump raw data memory area	LINARRAY? command; This is not equivalent because the data up to 256 K of raw target data is sent in definite length format. Each data value is the intensity (1 to 64) of a linear array at a specific location.
	PR	Dump processed data memory area	No equivalent command.
VS1	<NR3> or NONE	Scale factor for vertical channel 1	CHA? RANge command
VS2	<NR3> or NONE	Scale factor for vertical channel 2	CHA? RANge command
HS1	<NR3> or NONE	Scale factor for horizontal channel 1	ACQuire? TIMe command
HS2	<NR3> or NONE	Scale factor for horizontal channel 2	ACQuire? TIMe command

Table 5-5: SCD Series/7912 Command Comparison (Cont.)

Header	Argument	Description	SCD Equivalent Commands
VU1	<CHARAC-TERS>	Units for vertical channel 1	WFMPRE? YUNit command will return the units of the vertical data.
VU2	<CHARAC-TERS>	Units for vertical channel 2	WFMPRE? YUNit command will return the units of the vertical data.
HU1	<CHARAC-TERS>	Units for horizontal channel 1	WFMPRE? XUNit command will return the units of the horizontal scaling.
HU2	<CHARAC-TERS>	Units for horizontal channel 2	WFMPRE? XUNit command will return the units of the horizontal scaling.
ERR	<NR1> or NONE	Code for error indicated in last status byte reported	EVEnt? or ALLEV? commands; Return event code giving specifics about SRQ. EVQty? command returns the number of events in the buffer.
SRQ	NULL	Service request code (7912HB provides no other response)	SRQMask command sets up various conditions for issuing SRQ's. RQS command turns SRQ capability on or off.
ID?	<CHARAC-TERS>	Identity of instrument	ID? command; Returns the identity of the SCD Series Waveform Recorder.
SET?	<MESSAGE UNITS>	Setting of programmable functions (header is omitted)	SET? command; Will learn all programmable functions within SCD Series Waveform Recorder. This string can be returned at any time to set up instrument. The LONgform ON/OFF command reduces the length of the ASCII setting string. There is also LLSET? command for fast binary transfers. The PATh { ON|OFF } command selects if the header and link are sent. If off, only the argument is sent. There are 10 nonvolatile settings storage locations in the SCD Series waveform recorder. Accessed using SAVe and RECAll commands.

Table 5-6: SCD Series/7A16P & 7A29P Command Comparison

Header	Argument	Description	SCD Equivalent Commands
INP	A	Input is from A connector	VMOde CHA command (SCD1000 only)
	B	Input is from B connector	VMOde CHB command (SCD1000 only) There is also VMODe ADD and CHx INVert for algebraic addition and subtraction of channels A & B (SCD1000 only).
RIN	HI	High (1 MΩ) input impedance is selected	No equivalent command. If using option 1E (Tek type II probe interface) with SCD1000, high impedance probes like the Tektronix P6203 and P6204 can be used.
	LOW	Low (50 Ω) input impedance is selected	No equivalent command. SCD series Waveform Recorders are 50 Ω.
BW	LIM	Limited bandwidth (20 MHz) is selected	No equivalent command; No bandwidth limit.
	FUL	Full bandwidth is selected	
CPL	AC	Input is AC coupled	CHA COUPling:AC or CHB COUPling:AC
	DC	Input is DC coupled	CHA COUPling:DC or CHB COUPling:DC
	GND	Input is grounded	CHA COUPling:OFF or CHB COUPling:OFF
CPL?	OVL	7A29P ONLY Overload is returned if input is in OVERLOAD condition. (Query only). Returns an error if used in a set command.	No equivalent command
POL	NOR	Amplifier polarity is normal	CHA INVert:OFF or CHB INVert:OFF (SCD1000 only)
	INV	Amplifier polarity is inverted	CHA INVert:ON or CHB INVert:ON (SCD1000 only)
V/D	<NRx>	Volts/Division is set to argument must be a number in the range of 0.01 to 1 in a 1–2–5 sequence. V/D 0 means probe is on IDENTIFY.	CHA RANge:<NRx> or CHB RANge:<NRx> (SCD1000 only) <NRx>= 100E–3 to 10 in 1–2–5 sequence No probe id return value
POS	<NRx>	Vertical position of trace (from center screen) is set to <NRx>; range is −10.22 to +10.24 is 0.02 steps For example, POS 2 corresponds to +2.00 div above center.	CHA OFFSet:<NRx> or CHB OFFSet:<NRx> (SCD1000 only) <NRx>= ±2.5 times vertical range (volts) ±250 % in steps of 1 CHA TYPEOffset:{VOLTs|PERcent} sets the units for offset

Table 5-6: SCD Series/7A16P & 7A29P Command Comparison (Cont.)

Header	Argument	Description	SCD Equivalent Commands
VAR	OFF	Variable off — deflection factors are calibrated	No equivalent command
	ON	Variable on — deflection factors are uncalibrated	
PRB?	X1	1X or unencoded probe is present on selected input	No equivalent command
	X10	10X probe is present on selected input	
	X100	100X probe is present on selected input	
	ID	Returned when probe ID button is pressed	
ID?		Returns the plug-in type; for example TEK/7A29P,V77.1,F1.0	ID? command; Returns the identity of the SCD Series Waveform Recorder.
SET?	<MESSAGE UNITS>	Setting of programmable functions (header is omitted)	SET? command; Will learn all programmable functions within SCD Series Waveform Recorder. This string can be returned at any time to set up instrument. The LONGFORM ON/OFF command reduces the length of the ASCII settings string. There is also LLSET? command for fast binary transfers. There are 10 nonvolatile settings storage locations in the SCD Series Waveform Recorder. Accessed using SAVE and RECALL commands.

Table 5-7: SCD Series/7B90P Command Comparison

Header	Argument	Description	SCD Equivalent Commands
MOD	PPA	Peak-to-Peak auto trigger mode is selected	TRIgger MODe:AUTo command
	NOR	Normal triggering mode is selected	TRIgger MODe:NORmal command
	SSW	Single-Sweep Triggering mode is selected	ACQuire STATe:HLDNxt and HLDNxt:ON commands
CPL	AC	Trigger signal is AC coupled	TRIgger COUPling:AC command
	DC	Trigger signal is DC coupled	TRIgger COUPling:DC command
	LFR	Trigger signal is AC coupled with low frequency roll off	No equivalent command
	HFR	Trigger signal is AC coupled with high frequency roll off	No equivalent command
SRC	INT	Trigger source is internal	TRIgger SOUrce command. Choices of source are CHA, CHB and ADD (SCD1000 only). In SCD5000 choices are INT or EXT.
	LIN	Trigger source is the line voltage	No equivalent command
	EXT	Trigger source is external input	TRIgger SOUrce:EXT command
	E10	Trigger source is external input attenuated by 10	No equivalent command
T/D	<NR3>	Time/Division is set to <NRx>; range is 5E−10 to 5E−1 in 1−2−5 sequence. Query returns <NR3> value.	ACQuire TIMe:<NRx> command <NRx>= 5 E−9 to 100 E−6 in 1−2−5 sequence. The SCD Series waveform recorder is programmed using time window (total time) rather than time per division. Time window = Time/div * 10
MAG	ON	Sweep magnifier is turned on (10X)	No equivalent command
	OFF	Sweep magnifier is turned off (1X)	
POS	<NR2>	Horizontal position of sweep is set to <NRx>. Range is −6.4 to +6.39 in 0.0125 steps (80 steps/division). Query returns <NR2>	TRIgger DELay:<NRx> command <NRx>= 0 to (5 * time window) (seconds) 0 to 500% (percent) TRIgger TYPEdelay:SECond|PERcent command selects units for setting trigger delay. With zero trigger delay, there is ≈2.5 ns of pretrigger information.
HOL	<NRx>	Trigger holdoff period is <NRx>; range is 0 to 63 uncalibrated	No equivalent command

Table 5-7: SCD Series/7B90P Command Comparison (Cont.)

Header	Argument	Description	SCD Equivalent Commands
EOS	ON	End-of-sweep SRQ signal is enabled	SRQMask OPCmpl:ON command will issue an operation complete SRQ at end of acquisition.
	OFF	End-of-sweep SRQ signal is disabled	
TRI	ON	Trigger light is on (Read-only; TRI? returns TRI ON/OFF)	No equivalent command
	OFF	Trigger light is off (Read-only)	
SSW	ARM	Single sweep is armed. A GET (Group Execute Trigger) IEEE–488 universal command has the same effect.	Single sweep is accomplished with the ACQuire STATe HLDNxt and HLDNXt:{ON|OFF} commands. When these commands are issued, the instrument resets and waits for a trigger. When a trigger occurs, the SCD acquires and enters the hold state.
	DIS	Single Sweep is disarmed (Read-only; SSW? returns SSW ARM/DIS)	
ID?		Query only; Returns the plug-in type Example Response: Tek/7B90P,V77.1,LLL	ID? command; Returns the identity of the SCD Series Waveform Recorder.
SET?	<MESSAGE UNITS>	Setting of programmable functions (header is omitted)	SET? command; Will learn all programmable functions within SCD Series Waveform Recorder. This string can be returned at any time to set up instrument. The LONGFORM ON/OFF command reduces the length of the ASCII settings string. There is also LLSET? command for fast binary transfers. There are 10 nonvolatile settings storage locations in the SCD Series Waveform Recorder. Accessed using SAVE and RECALL commands.

Option 2F

High Speed Data Output (HSDO)

Option 2F provides nonvolatile storage (battery backup) of Linear Array Data and is an alternative to GPIB output for Curve and Linear Array Data.

HSDO data encoding is absolute binary. The Option 2F interface allows SCD1000 and 5000 instruments to transfer waveform data encoded in 16-bit words at a 2 MHz maximum rate. With the GPIB command HSDO FORmat, the 16-bit words may be transferred 1 byte at a time, high byte first (MSB). The HSDO port is output enabled with the GPIB HSDO STATe and HSDO DUMp: CONTinuous commands, or by a combination of internal and external switch settings.

The HSDO port provides high speed data transfer to the Tektronix 9503/9504 Fast Data Cache system and to interfaces such as Digital Equipment Corporation DRV 11–B or Hewlett Packard GPIO (see special cabling requirements in Tables NO TAG and 6-2). Most computers, including MS-DOS based, support this type of parallel port interface.

Connectors

HSDO	(High Speed Data Output)
Type	DB 37 Mating connector: 3M 3357–9237 Cable: 3M 3659/37
Signal Level	TTL
ACQ-CONT	(Acquire Continuous Data)
Type	BNC
Signal Level	TTL

Dump Continuous Mode

Dump Continuous repeatedly outputs acquisition data from the HSDO port until specifically inactivated, or a new acquisition is initiated. This allows external control of the acquisition process. Dump Continuous transfers Linear Array data exclusively, and is activated by the GPIB command HSDO DUMp: CONTinuous. This mode may be implemented manually (by a certified service technician only) as follows. Set DIP switch 2 on the instrument rear panel to closed; set DIP switch 2 to closed and DIP switch 3 to open on the Processor board. If Dump Continuous is activated manually, the instrument will power-up in the Hold Next acquisition state; the HSDO transfer mode will be 1 (Handshake).

When activated, Dump Continuous mode loops continuously, transferring the contents of the Linear Array over the HSDO port. The loop may be interrupted by setting the states on two HSDO pins. Setting HS STAT 3 (pin 28) low causes an acquisition to occur followed by continuous transmission of data; setting HS STAT 2 (pin 27) low restarts continuous transmission of data from its start point.

The rear panel ACQ-CONT BNC allows the same control as HSDO connector HS STAT 3 (pin 28), permitting acquisition control from another source.

Two output pins provide the Controller with digitizer and HSDO transmission status, HS CNTL 0 (pin 20) and HS CNTL 1 (pin 21). HS CNTL 0 is high when data is to be read from the digitizer. HS CNTL 1 toggles for each new data transmission.

The following is an example of how the HS STAT and HS CNTL pins may be used to retransmit or force an acquisition while in Continuous Dump mode.

To Retransmit:

1. Halt HSDO port handshake data
2. Note the state of HS CNTL 1 (pin 21)
3. Set HS STAT 2 (pin 27) low
4. Monitor HS CNTL 1, await toggle state
5. Following HS CNTL 1 toggle, return HS STAT 2 high
6. Resume HSDO port handshake data output

To Reacquire:

1. Halt HSDO port handshake data
2. Set HS STAT 3 (pin 28) low
3. Wait for HS CNTL 0 (pin 20) to transition high (digitizer enabled, ACQUire STATe: HLDNxt)
4. Set HS STAT 3 high
5. Wait for HS CNTL 0 to transition low (acquisition complete, ACQUire STATe: STOP) and HS CNTL 1 (pin 21) to toggle. The instrument is ready for data transmission.

6. Resume HSDO port handshake data output

When an acquisition is initiated (and before the trigger), HS STAT 3 ignores requests for data retransmission and acquisition aborts. Retransmission requests are not accepted until acquisition is complete.

Handshake Transfer Mode

Two HSDO port communication protocols are available, Handshake and Synchronous.

Transfer Mode 1 (Handshake) — Data is output to an external device (the external device initiates the transfer). This mode is full handshake compatible with Digital Equipment Corporation DRV 11–B and Hewlett Packard GPIO.

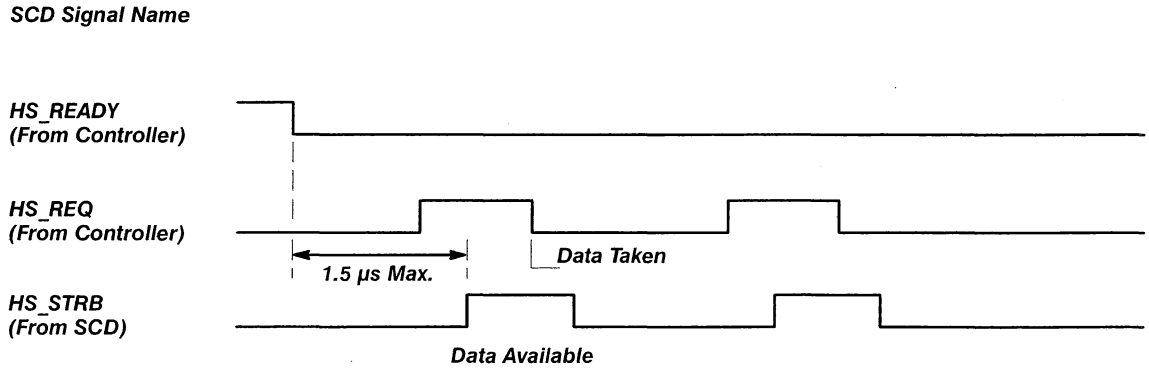


Figure 6-1: Handshake Mode Timing Diagram

Transfer Mode 2 (Synchronous) — Data is transferred to a custom interface. The interface accepts data on a clock edge generated by the SCD. No handshake is used.

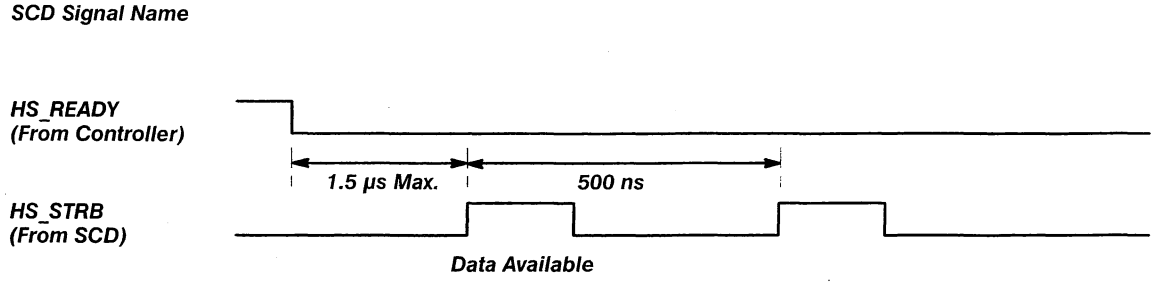


Figure 6-2: Synchronous Mode Timing Diagram

GPIB Commands

Option 2F adds the following GPIB commands:

Header	Link	Argument	Description
HSDO	DUMp	OFF	Sets the HSDO dump mode to off.
		CONTinuous	Repeatedly outputs acquired data until a new acquisition is requested; following a new acquisition, the new data is repeatedly output, etc. Refer also to the Option 2F alternate interface description in this manual.
			Factory setting: OFF
			Example: HSDO DUMp: CONTinuous
			Interactions: HSDO STATE forced to OFF HSDO MODE forced to 1 (Handshake)
	FORmat:	BYTE	HSDO interface outputs one 8-bit byte at a time.
		WORd	All 16 bits of the HSDO interface used for output. Transfers 2 bytes (one waveform data point) at a time. Byte output is in the LSB of the interface, MSB first.
			Factory setting: WORd
			Example: HSDO FORmat: WORd
	MODe:	<NRx>	NRx = 1: selects HSDO Handshake mode for data output to external devices (DRV 11–B and GPIO).
			NRx = 2: selects HSDO Synchronous mode for data output to external devices. Data is transferred to a custom interface accepting data on a clock edge generated by the SCD. No handshake used.
			Limits: 1 or 2
			Factory setting: 1
			Example: HSDO MODe: 1
	STATe:	OFF	Inactivates the HSDO (the GPIB port is used for waveform transfers).
		ALL	The HSDO outputs all memory regardless of valid data quantity (262144 bytes fixed data length).
		VALid	The HSDO outputs only valid data (variable data length depends on information acquired).
			Factory setting: OFF
			Example: HSDO STATe: VALid

GPIB Query Commands

Option 2F adds the following query selections:

Header	Link	Argument	Description
HSDO?			Returns all HSDO settings. Example: HSDO? Response: HSDO STATE: OFF, MODE: 1, FORMAT: WORD, LENGTH: 12555, DUMP: OFF
	DUMp		Returns HSDO Dump settings. Example: HSDO? DUMp Response: HSDO DUMP: OFF
	FORmat		Returns HSDO Format settings. Example: HSDO? FORmat Response: HSDO FORMAT: WORD
	LENgth		Query only; returns the length of HSDO port data in bytes. If HSDO: STATe is set to VALid, the length of data sent is returned. Example: HSDO? LENgth
	MODe		Returns the HSDO Mode setting. Example: HSDO? MODe Response: HSDO MODE: 1
	STATe		Returns the HSDO State setting. Example: HSDO? STATe Response: HSDO STATE: OFF, MODE: 1, FORMAT: WORD, LENGTH: 12555, DUMP: OFF

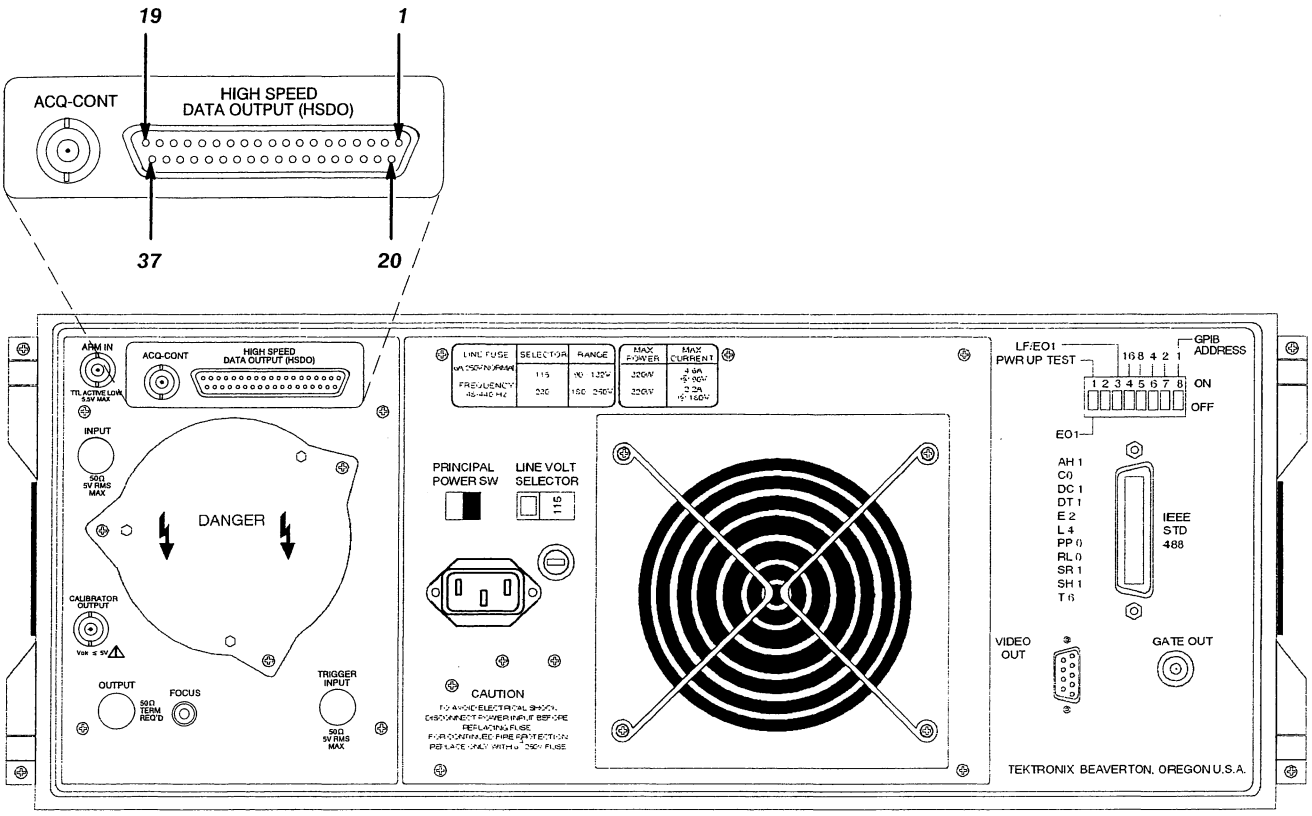


Figure 6-3: Location of HSDO Rear Panel Connector (SCD1000)

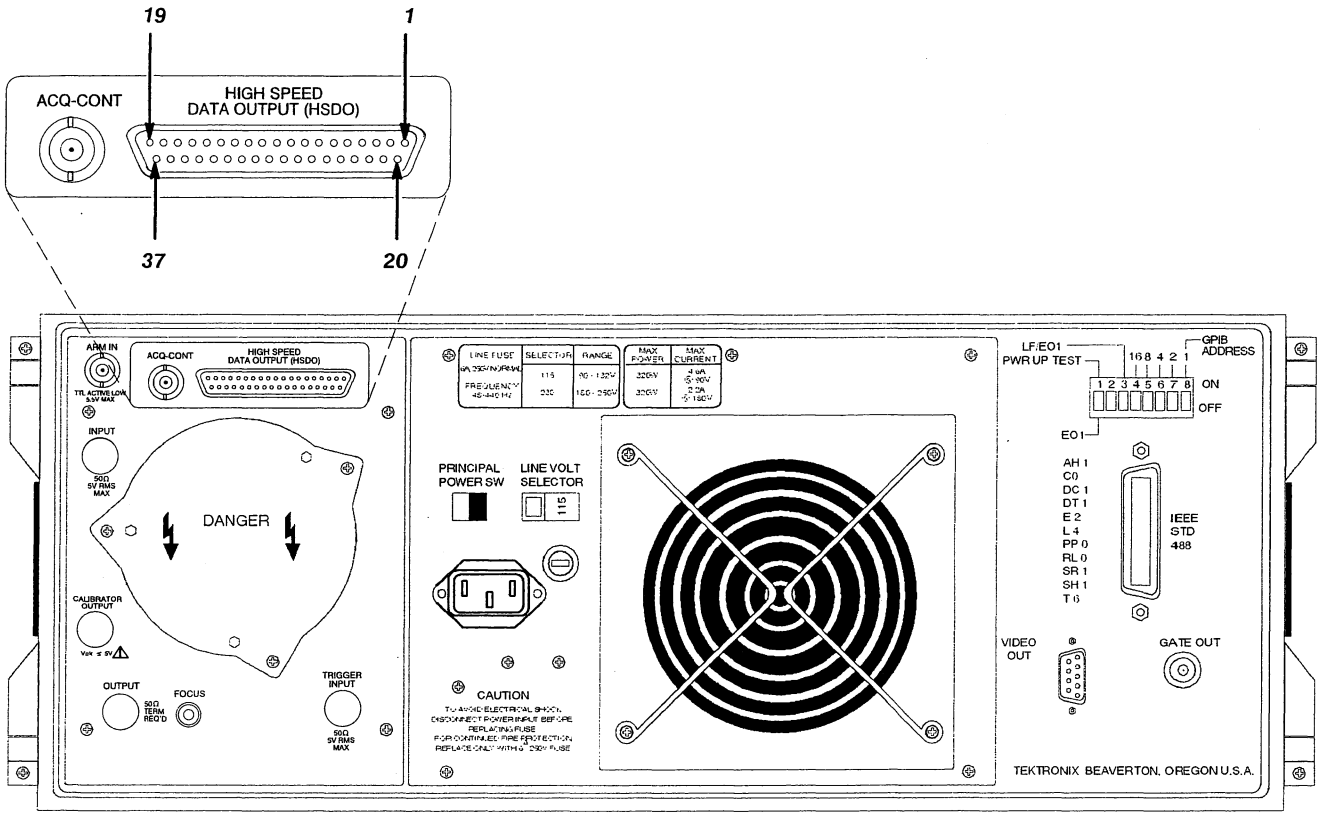


Figure 6-4: Location of HSDO Rear Panel Connector (SCD5000)

Table 6-1: SCD Series to DEC DRV11-B Cabling

SCD Pin	Signal Name	Direction (Pin to Pin)	DRV11-B Pin	Signal Name	Description
1	HSDO0	→	J2-UU	00 IN H	LOW BYTE DATA
2	HSDO1	→	J2-SS	01 IN H	LOW BYTE DATA
3	HSDO2	→	J2-PP	02 IN H	LOW BYTE DATA
4	HSDO3	→	J2-MM	03 IN H	LOW BYTE DATA
5	HSDO4	→	J2-KK	04 IN H	LOW BYTE DATA
6	HSDO5	→	J2-HH	05 IN H	LOW BYTE DATA
7	HSDO6	→	J2-EE	06 IN H	LOW BYTE DATA
8	HSDO7	→	J2-CC	07 IN H	LOW BYTE DATA
9	HSDO8	→	J2-DD	08 IN H	HIGH BYTE DATA
10	HSDO9	→	J2-FF	09 IN H	HIGH BYTE DATA
11	HSDO10	→	J2-JJ	10 IN H	HIGH BYTE DATA
12	HSDO11	→	J2-LL	11 IN H	HIGH BYTE DATA
13	HSDO12	→	J2-NN	12 IN H	HIGH BYTE DATA
14	HSDO13	→	J2-RR	13 IN H	HIGH BYTE DATA
15	HSDO14	→	J2-TT	14 IN H	HIGH BYTE DATA
16	HSDO15	→	J2-VV	15 IN H	HIGH BYTE DATA
17	HSREQ	←	J2-B	BUSY H	Request for next word
18	HSREADY	←	J1-F	READY H	LOW indicates DRV11 is ready
19	HS_STRB	→	J1-B	CYCLE REQ H	Latch data on rising edge
20	HS_CNTL 0	→	J1-L	STATUS A	Contains mode of SCD Series instrument
21	HS_CNTL 1	→	J1-R	STATUS B	Contains mode of SCD Series instrument
22	VCC	→	J2-F	A00H	Tied to +5 V through a 1 K resistor
23	HSSYSRES	→	J2-D	ATTN H	Terminates DMA transfers
24	C1	→	J2-T	C1 H	Must be HIGH for DATA transfer to DRV11. Driven high by SCD Series instrument.
25	C0	→	J2-N	C0 H	Must be LOW for DATA transfer to DRV11. Connected to ground by SCD Series instrument.
26	HS_STAT 0	←	J2-V	FNCT 1 H	Not used

Table 6-1: SCD Series to DEC DRV11-B Cabling (Cont.)

SCD Pin	Signal Name	Direction (Pin to Pin)	DRV11–B Pin	Signal Name	Description
27	HS_STAT 1	←	J2–R	FNCT 2 H	See Continuous Dump mode
28	HS_STAT 2	←	J2–LK	FNCT 3 H	See Continuous Dump mode
29	HSOK	→			Not used
30	HSATTN	←			Not used
31	DGND				Logic Ground
32	DGND				Logic Ground
33	DGND				Logic Ground
34	DGND				Logic Ground
35	DGND				Logic Ground
36	VCC		J2–J	WC INC EN H	
37	VCC		J2–J	BA INC EN H	

Table 6-2: SCD Series to HP GPIO Cabling

SCD Pin	Signal Name	Direction (Pin to Pin)	GPIO Pin	Signal Name	Description
1	HSDO0	→	42	DI0	LOW BYTE DATA
2	HSDO1	→	41	DI1	LOW BYTE DATA
3	HSDO2	→	40	DI2	LOW BYTE DATA
4	HSDO3	→	39	DI3	LOW BYTE DATA
5	HSDO4	→	38	DI4	LOW BYTE DATA
6	HSDO5	→	37	DI5	LOW BYTE DATA
7	HSDO6	→	36	DI6	LOW BYTE DATA
8	HSDO7	→	35	DI7	LOW BYTE DATA
9	HSDO8	→	34	DI8	HIGH BYTE DATA
10	HSDO9	→	33	DI9	HIGH BYTE DATA
11	HSDO10	→	32	DI10	HIGH BYTE DATA
12	HSDO11	→	31	DI11	HIGH BYTE DATA
13	HSDO12	→	30	DI12	HIGH BYTE DATA
14	HSDO13	→	29	DI13	HIGH BYTE DATA
15	HSDO14	→	28	DI14	HIGH BYTE DATA
16	HSDO15	→	27	DI15	HIGH BYTE DATA
17	HSREQ	←	19	PCTL	LOW means not ready. HIGH means request.
18	HSREADY	←		DGND	Not used. From SCD Series instrument pin 35.
19	HS_STRB	→	44	PFLG	Latch data; clear PCTL on rising edge.
20	HS_CNTL 0	→	47	STI0	Contains mode of SCD Series instrument
21	HS_CNTL 1	→	48	STI1	Contains mode of SCD Series instrument
22	VCC	→			Not used
23	HSSYSRES	→	46	EIR	Interrupt host
24	C1	→			Not used
25	C0	→			Not used
26	HS_STAT 0	←	22	CTL0	Not used
27	HS_STAT 1	←	23	CTL1	See Continuous Dump mode

Table 6-2: SCD Series to HP GPIO Cabling (Cont.)

SCD Pin	Signal Name	Direction (Pin to Pin)	GPIO Pin	Signal Name	Description
28	HS_STAT 2	←			See Continuous Dump mode
29	HSOK	→	45	PSTS	The HSDO mode is ready and OK
30	HSATTN	←	21	PRESET	Not used
31	DGND	↔	1	LOGIC GND	
32	DGND	↔	18	LOGIC GND	
33	DGND	↔	24	LOGIC GND	
34	DGND	↔	26	LOGIC GND	
35	DGND	↔	49	LOGIC GND	
36	VCC	↔			
37	VCC	↔			
			43	SAFETY GND	
			25	SAFETY GND	

ASCII & GPIB Code Chart

Table 7-1: The ASCII & GPIB Code Chart

	0	1	2	3	4	5	6	7
0	0 **NUL** 0 0	20 **DLE** 10 16	40 0 **space** 20 32	60 16 **0** 30 48	100 0 **@** 40 64	120 16 **P** 50 80	140 0 **‘** 60 96	160 16 **p** 70 112
1	1 GTL **SOH** 1 1	21 LLO **DC1** 11 17	41 1 **!** 21 33	61 17 **1** 31 49	101 1 **A** 41 65	121 17 **Q** 51 81	141 1 **a** 61 97	161 17 **q** 71 113
2	2 **STX** 2 2	22 **DC2** 12 18	42 2 **”** 22 34	62 18 **2** 32 50	102 2 **B** 42 66	122 18 **R** 52 82	142 2 **b** 62 98	162 18 **r** 72 114
3	3 **ETX** 3 3	23 **DC3** 13 19	43 3 **#** 23 35	63 19 **3** 33 51	103 3 **C** 43 67	123 19 **S** 53 83	143 3 **c** 63 99	163 19 **s** 73 115
4	4 SDC **EOT** 4 4	24 DCL **DC4** 14 20	44 4 **$** 24 36	64 20 **4** 34 52	104 4 **D** 44 68	124 20 **T** 54 84	144 4 **d** 64 100	164 20 **t** 74 116
5	5 PPC **ENQ** 5 5	25 PPU **NAK** 15 21	45 5 **%** 25 37	65 21 **5** 35 53	105 5 **E** 55 69	125 21 **U** 55 85	145 5 **e** 65 101	165 21 **u** 75 117
6	6 **ACK** 6 6	26 **SYN** 16 22	46 6 **&** 26 38	66 22 **6** 36 54	106 6 **F** 46 70	126 22 **V** 56 86	146 6 **f** 66 102	166 22 **v** 76 118
7	7 **BEL** 7 7	27 **ETB** 17 23	47 7 **’** 27 39	67 23 **7** 37 55	107 7 **G** 47 71	127 23 **W** 57 87	147 7 **g** 67 103	167 23 **w** 77 119
8	10 GET **BS** 8 8	30 SPE **CAN** 18 24	50 8 **(** 28 40	70 24 **8** 38 56	110 8 **H** 48 72	130 24 **X** 58 88	150 8 **h** 68 104	170 24 **x** 78 120
9	11 TCT **HT** 9 9	31 SPD **EM** 19 25	51 9 **)** 29 41	71 25 **9** 39 57	111 9 **I** 49 73	131 25 **Y** 59 89	151 9 **i** 69 105	171 25 **y** 79 121
A	12 **LF** A 10	32 **SUB** 1A 26	52 10 ***** 2A 42	72 26 **:** 3A 58	112 10 **J** 4A 74	132 26 **Z** 5A 90	152 10 **j** 6A 106	172 26 **z** 7A 122
B	13 **VT** B 11	33 **ESC** 1B 27	53 11 **+** 2B 43	73 27 **;** 3B 59	113 11 **K** 4B 75	133 27 **[** 5B 91	153 11 **k** 6B 107	173 27 **{** 7B 123
C	14 **FF** C 12	34 **FS** 1C 28	54 12 **,** 2C 44	74 28 **<** 3C 60	114 12 **L** 4C 76	134 28 **** 5C 92	154 12 **l** 6C 108	174 28 **|** 7C 124
D	15 **CR** D 13	35 **GS** 1D 29	55 13 **–** 2D 45	75 29 **=** 3D 61	115 13 **M** 4D 77	135 29 **]** 5D 93	155 13 **m** 6D 109	175 29 **}** 7D 125
E	16 **SO** E 14	36 **RS** 1E 30	56 14 **.** 2E 46	76 30 **>** 3E 62	116 14 **N** 4E 78	136 30 **^** 5E 94	156 14 **n** 6E 110	176 30 **~** 7E 126
F	17 **SI** F 15	37 **US** 1F 31	57 15 **/** 2F 47	77 UNL **?** 3F 63	117 15 **O** 4F 79	137 UNT **_** 5F 95	157 15 **o** 6F 111	177 **rubout** 7F 127
	ADDRESSED COMMANDS	UNIVERSAL COMMANDS	LISTEN ADDRESSES		TALK ADDRESSES		SECONDARY ADDRESSES OR COMMANDS	

KEY octal 25 PPU **NAK** hex 15 21 — GPIB code, ASCII character, decimal