

Errata

Title & Document Type: 54200A/D Programming Section of Operation Manual

Manual Part Number: 54200-90001

HP References in this Manual

This manual may contain references to HP or Hewlett-Packard. Please note that Hewlett-Packard's former test and measurement, semiconductor products and chemical analysis businesses are now part of Agilent Technologies. We have made no changes to this manual copy. The HP XXXX referred to in this document is now the Agilent XXXX. For example, model number HP8648A is now model number Agilent 8648A.

About this Manual

We've added this manual to the Agilent website in an effort to help you support your product. This manual provides the best information we could find. It may be incomplete or contain dated information, and the scan quality may not be ideal. If we find a better copy in the future, we will add it to the Agilent website.

Support for Your Product

Agilent no longer sells or supports this product. You will find any other available product information on the Agilent Test & Measurement website:

www.tm.agilent.com

Search for the model number of this product, and the resulting product page will guide you to any available information. Our service centers may be able to perform calibration if no repair parts are needed, but no other support from Agilent is available.

SECTION 8

PROGRAMMING THE 54200A/D

8-1. INTRODUCTION

This section provides instructions to allow you to program the 54200A/D. The 54200A/D is a fully HP-IB bus programmable instrument. Menu functions (except protect status on memories), remote-only functions, and front panel functions (except the line switch and cursor movements) are programmable through the HP-IB.

8-2. COMMAND STRUCTURE

The 54200A/D programming commands are divided into three types: system commands, subsystem selectors, and subsystem commands. A programming command tree is shown in figure 8-1, and a programming command cross-reference is shown in table 8-1.

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

Subsystem selectors are commands that switch from one subsystem to another and are valid at any time with one important exception: certain subsystem selector commands also look like subsystem command parameters. For example the keywords CHANNEL, GRAPH, and TRIGGER are used both as subsystem selectors and as command parameters. Care must be taken to follow the correct rules of syntax and punctuation to avoid creating errors or unwanted results in the program.

The nine subsystems are:

- Acquire
- Channel
- Display
- Graph
- Measure
- State (54200D only)
- Timebase
- Trigger
- Waveform

Subsystem commands are legal only when the subsystem has been previously selected and control a particular portion of the oscilloscope, for example the timebase. Only one subsystem may be selected at a time, and any command which is not a system command or a subsystem selector is assumed to be a command for the selected subsystem.

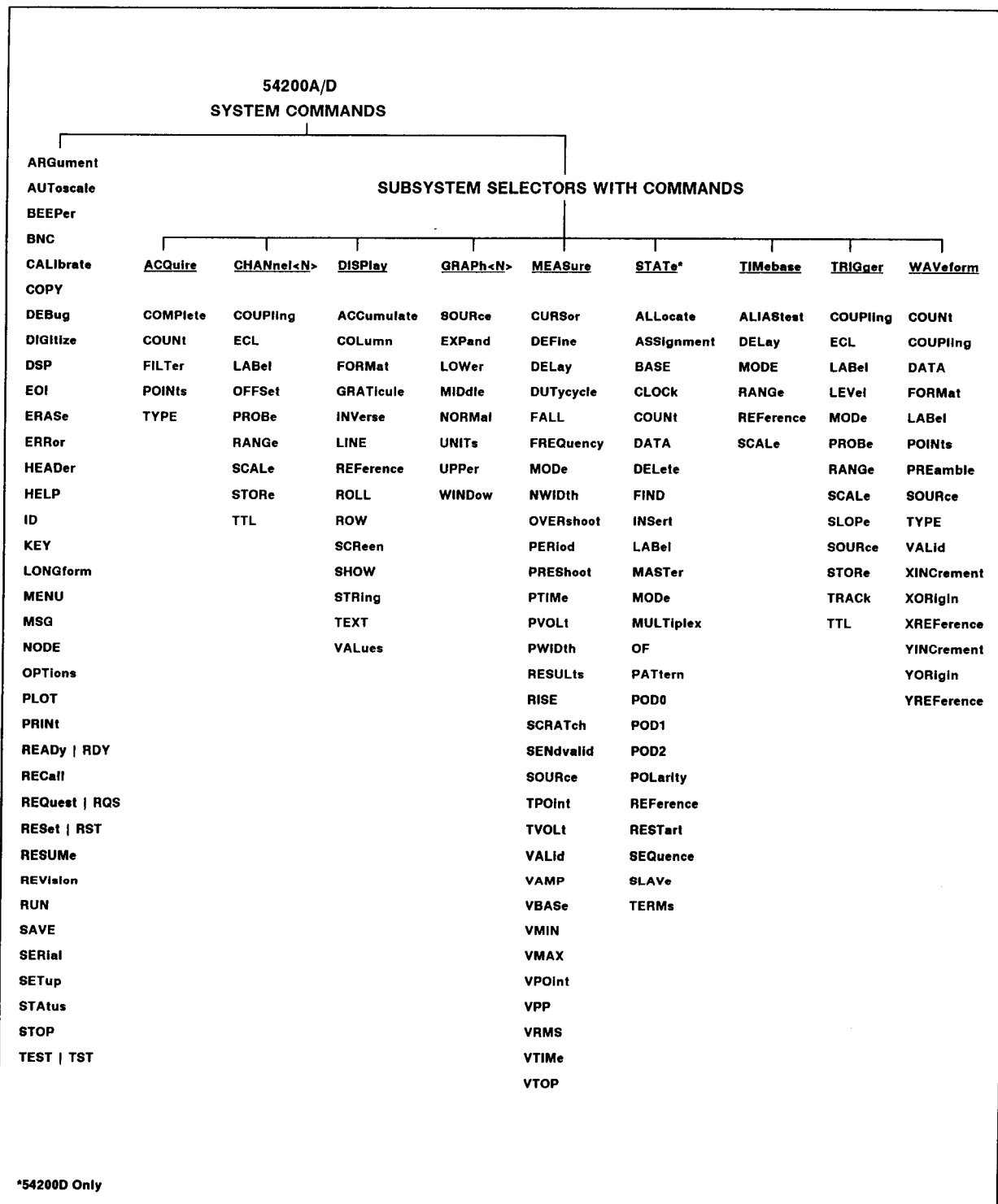


Figure 8-1. Programming Command Tree

Table 8-1. Alphabetical Command Cross-Reference

| COMMAND | WHERE USED | COMMAND | WHERE USED | COMMAND | WHERE USED |
|-------------------|--------------------|----------------------|--------------------|-------------------|--------------------|
| ACCumulate | Display subsystem | LABel | Channel subsystem | SAVE | System command |
| ACQuire | Subsystem selector | LABel | State subsystem | SCALE | Channel subsystem |
| ALIAStest | Timebase subsystem | LABel | Trigger subsystem | SCALE | Timebase subsystem |
| ALLocate | State subsystem | LABel | Waveform subsystem | SCALE | Trigger subsystem |
| ARGument | System command | LEVel | Trigger subsystem | SCRATch | Measure subsystem |
| ASSignment | State subsystem | LINE | Display subsystem | SCReen | Display subsystem |
| AUToscale | System command | LONGform | System command | SENdvalid | Measure subsystem |
| | | LOWer | Graph subsystem | SEQuence | State subsystem |
| BASE | State subsystem | | | SERial | System command |
| BEEPer | System command | MASTer | State subsystem | SETup | System command |
| BNC | System command | MEASure | Subsystem selector | SHOW | Display subsystem |
| | | MENU | System command | SLAVE | State subsystem |
| CALibrate | System command | MIDdle | Graph subsystem | SLOPe | Trigger subsystem |
| CHANnel | Subsystem selector | MODE | Timebase subsystem | SOURce | Graph subsystem |
| CLOCK | State subsystem | MODE | Measure subsystem | SOURce | Measure subsystem |
| COLUMN | Display subsystem | MODE | State subsystem | SOURce | Trigger subsystem |
| COMPLete | Acquire subsystem | MODE | Trigger subsystem | SOURce | Waveform subsystem |
| COPY | System command | MSG | System command | STATe | Subsystem Selector |
| COUNT | State subsystem | MULTiplex | State subsystem | STATus | System command |
| COUNT | Acquire subsystem | | | STOP | System command |
| COUNT | Waveform subsystem | NODE | System command | STORe | Channel subsystem |
| COUpling | Channel subsystem | NORMal | Graph subsystem | STORe | Trigger subsystem |
| COUpling | Trigger subsystem | NWIDth | Measure subsystem | STRing | Display subsystem |
| COUpling | Waveform subsystem | | | | |
| CURSor | Measure subsystem | OF | State subsystem | TERMs | State subsystem |
| | | OFFSet | Channel subsystem | TEST TST | System command |
| DATA | State subsystem | OPTions | System command | TEXT | Display subsystem |
| DATA | Waveform subsystem | OVERshoot | Measure subsystem | TIMEbase | Subsystem selector |
| DEBUg | System command | | | TPOint | Measure subsystem |
| DEFine | Measure subsystem | PATtern | State subsystem | TRACK | Trigger subsystem |
| DELAy | Measure subsystem | PERiod | Measure subsystem | TRIGger | Subsystem selector |
| DELAy | Timebase subsystem | PLOT | System command | TTL | Channel subsystem |
| DELeTe | State subsystem | POD0 | State subsystem | TTL | Trigger subsystem |
| DIGitize | System command | POD1 | State subsystem | TVOLt | Measure subsystem |
| DISPlay | Subsystem selector | POD2 | State subsystem | TYPE | Acquire subsystem |
| DSP | System command | POINts | Acquire subsystem | TYPE | Waveform subsystem |
| DUTYcycle | Measure subsystem | POINts | Waveform subsystem | | |
| | | POLarity | State subsystem | UNITs | Graph subsystem |
| ECL | Channel subsystem | PREamble | Waveform subsystem | UPPer | Graph subsystem |
| ECL | Trigger subsystem | PREShoot | Measure subsystem | | |
| EOI | System command | PRINt | System command | VALid | Measure subsystem |
| ERASe | System command | PROBe | Channel subsystem | VALid | Waveform subsystem |
| ERRor | System command | PROBe | Trigger subsystem | VALues | Display subsystem |
| EXPand | Graph subsystem | PTIME | Measure subsystem | VAMP | Measure subsystem |
| | | PVOLT | Measure subsystem | VBASE | Measure subsystem |
| FALL | Measure subsystem | PWIDth | Measure subsystem | VMAX | Measure subsystem |
| FILTer | Acquire subsystem | | | VMIN | Measure subsystem |
| FIND | State subsystem | RANGE | Channel subsystem | VPOint | Measure subsystem |
| FORMat | Display subsystem | RANGE | Timebase subsystem | VPP | Measure subsystem |
| FORMat | Waveform subsystem | RANGE | Trigger subsystem | VRMS | Measure subsystem |
| FREQuency | Measure subsystem | READy RDY | System command | VTIME | Measure subsystem |
| | | RECall | System command | VTOP | Measure subsystem |
| GRAPH | Subsystem selector | REFeRence | Display subsystem | | |
| GRATicule | Display subsystem | REFeRence | State subsystem | WAVeform | Subsystem selector |
| | | REFeRence | Timebase subsystem | WINDow | Graph subsystem |
| HEADer | System command | REQuEst RQS | System command | | |
| HELP | System command | REStArt | State subsystem | XINCrement | Waveform subsystem |
| | | RESULts | Measure subsystem | XORigin | Waveform subsystem |
| ID | System command | RESUMe | System command | XREFeRence | Waveform subsystem |
| INSert | State subsystem | RESeT RST | System command | | |
| INVerse | Display subsystem | REVisiOn | System command | YINCrement | Waveform subsystem |
| | | RISE | Measure subsystem | YORigin | Waveform subsystem |
| KEY | System command | ROLL | Display subsystem | YREFeRence | Waveform subsystem |
| | | ROW | Display subsystem | | |
| | | RUN | System command | | |

8-3. COMMAND ABBREVIATIONS

Every command and every alpha parameter has at least two forms, a short form and a long form, although in some cases these are identical. The longform throughout this manual is shown with leading upper case letters and trailing lower case letters. The shortform is only the upper case letters. The short form is obtained by truncating the longform using the following rule:

If the longform has more than 4 characters,
then if the 4th character is a vowel or the same as the 3rd character,
then truncate to 3 characters,
else truncate to 4 characters.

Example - LONGFORM abbreviates to LONG and is shown as LONGform
REFERENCE abbreviates to REF and is shown as REFERENCE
STATE abbreviates to STAT and is shown as STATE

In the case where two or more short forms would be identical, all but one is changed slightly to differentiate between the commands. For example RESET is abbreviated to RES, RESUME is abbreviated to RESUM, and RESTART is abbreviated to REST. In the command descriptions that follow, each command is given in both long and short forms.

Some commands also have industry standard forms, such as RQS for REQUEST and RDY for READY. These are also valid forms in the 54200A/D instruction set.

The commands may be executed using either long or short form. When a query is executed and the HEADER command is set ON, the answer returned to the controller will be returned in the long form format if the LONGform command has been set ON and will be returned in short form format if the LONGform command has been set OFF. If the HEADER command is set OFF, neither long nor short form format will be returned to the controller.

8-4. ALPHA AND NUMERIC ARGUMENTS

Most of the programming commands that require parameters can use either alpha or numeric arguments as their parameters. For these commands, the syntax of the command gives the relationship between the numeric and alpha arguments.

For example if the syntax of a command is:

```
SCReen { [ OFF | 0 ]  
         [ ON  | 1 ] }
```

executing SCREEN OFF is the same as SCREEN 0 *and*

executing SCREEN ON is the same as SCREEN 1

The commands may be executed using either alpha or numeric arguments. When a query is executed, the answer returned to the controller will be returned in alpha format if the ARGument command has been set to ALPHa and will be returned in numeric format if the ARGument command has been set to NUMeric.

8-5. NOTATION CONVENTIONS AND DEFINITIONS

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

- < > Angular brackets enclose descriptive words or characters that are used to symbolize a program code parameter or an HP-IB command. For example, <A> represents the ASCII character 'A'.
- ::= "is defined as". For example, <A> ::= indicates that <A> can be replaced by in any statement containing <A>.
- { } When several items are enclosed by braces, one, and only one of these elements must be selected.
- | "or": Indicates a choice of exactly one element from a list. For example, <A>| indicates <A> or but not both.
- ... An ellipsis (trailing dots) is used to indicate that the preceding element may be repeated one or more times.
- [] Square brackets indicate that the enclosed items are optional parameters.
- !X!N An exclamation mark used in this form indicates a parameter may be repeated from X times up to N times.

The following definitions are used:

- d ::= A single ASCII numeric character, 0-9
- n ::= A single ASCII non-zero, numeric character, 1-9
- <LF> ::= ASCII linefeed (decimal 10)
- <CR> ::= ASCII carriage return (decimal 13)
- <SP> ::= ASCII space (decimal 32)

8-6. SYNTAX DIAGRAMS

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

8-7. PROGRAMMING AIDS

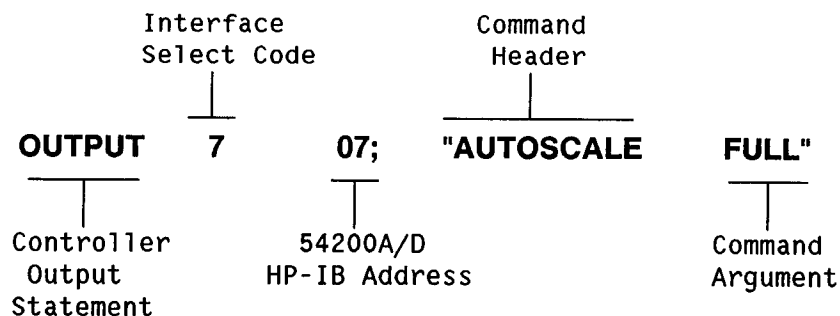
The system DEBug command helps those who are unfamiliar with the command syntax to find programming errors. When DEBug ON is issued, each command is shown on the 54200A/D display as the command is received over the HP-IB. The commands are defined as either inputs or outputs and appear within the < > symbols with imbedded carriage returns (CR) and line feeds (LF). The form in which commands are displayed on the 54200A/D is dependent on the current status of the system LONGform, HEADer, and ARGument commands. Spelling and syntax errors can be quickly located and corrected by stepping through a program. If an error occurs, the error is highlighted with an inverse video cursor on the 54200A/D display. An error message is then displayed, followed by message text, on the status line of the 54200A/D.

The system HELP? command returns the entire command set of the 54200A/D organized by subsystem. Refer to the System Command portion of this section for detailed information on this aid.

8-8. PROGRAMMING EXAMPLES

All programming examples used in this section are given in HP Series 200 Desktop Computer enhanced BASIC programming language. For all examples, the 54200A/D interface select/address code is set to 707.

The following diagram shows how the commands are sent to the 54200A/D:



8-9. DATA OUTPUT (QUERY) FORMAT TO CONTROLLER

If a query command (command followed by "?") is sent to the 54200A/D, a response message is generated and sent back to the controller the next time the 54200A/D is addressed to talk.

The command header will be returned if HEADer command has been set ON and will not be returned if set OFF.

The command argument will be returned as an alpha argument if ARGument command has been set to ALPha and will be returned as a numeric argument if set to NUMeric.

Headers and alpha arguments will be returned in the longform if LONGform command has been set ON and will be returned in the shortform if set OFF.

8-10. INPUT DATA FORMAT FROM CONTROLLER

Input Data messages contain a string of device dependent commands (program commands) and an End-of-String (EOS) message. The program codes within a Data message are executed after the EOS message is received. The following format rules must be observed for all input Data messages:

- A linefeed (<LF>) or an EOI are used as the End-of-String (EOS) message. Each Data message must be terminated by a <LF> or by asserting the EOI bus signal line with the last byte in the message.
- The carriage return character (<CR>) is not required before <LF>. Preceding <LF>, <CR> is treated as "no operation" and may be repeated as many times as permitted by the maximum string length limitation.
- When several program commands are sent in a Data message, a semicolon (preferred), or space must be used to separate each program command from each other.
- Multiple arguments for a command must be separated by commas.
- The maximum length of a Data message is 256 characters (including: semicolons, commas, <SP>, <CR>, and <LF>), except those messages that contain binary data blocks. Binary data blocks must begin within the first 256 bytes of the data message.

8-11. Program Order Considerations

Commands are interpreted and setups are changed in the 54200A/D as they are received and found to be syntactically correct. Commands preceding an error in multi-command messages are executed up to the point where the error is detected. This provides consistent operation whether command are sent one per message or several per message.

8-12. Program Command Format

Program commands consist of a header followed by a parameter field. Headers may be of a Long or Short (abbreviated) form. This allows the programmer to use full words or abbreviations for program commands. Command arguments can be words, or numbers in most cases. This allows one programmer to generate readable code, and allows another programmer to generate code that is more efficient in the use of space in his computer.

Program Code Parameters may be of four types:

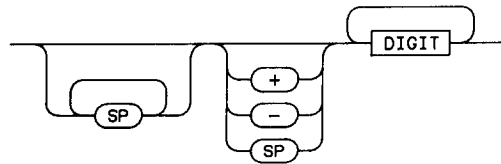
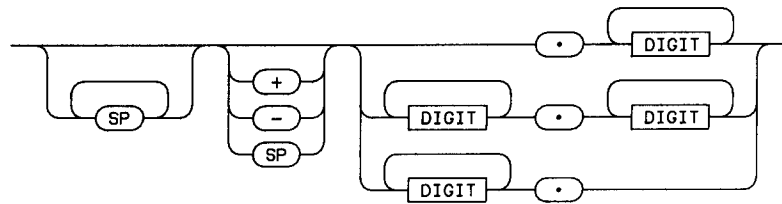
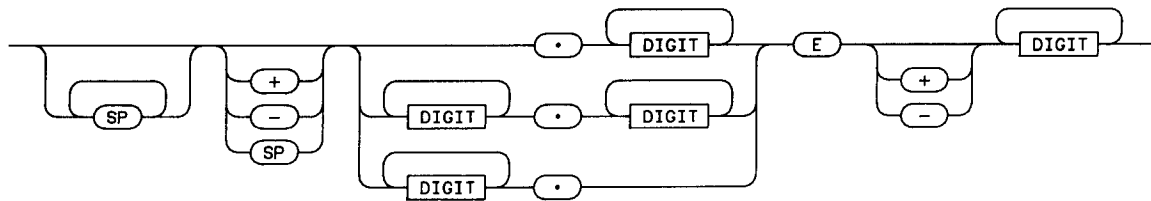
- Strings** - Any collection of ASCII characters excluding quotation marks (decimal 34) surrounded by quotation marks.
- Blocks** - A block of data in formats as defined by IEEE Std 728-1982.
- #A** - This format is a binary block with the format: <#><A><length word><DAB...DAB>. The length word is a 16 bit binary integer representing the number of DABs. DABs are the data bytes themselves. <#> and <A> are ASCII bytes.
- #Y** - This format is an ASCII representation of a binary number with the format: <#><Y><DAB...DAB>. All bytes are ASCII.
- #O** - This format is an ASCII representation of an octal number with the format: <#><O><DAB...DAB>. All bytes are ASCII.
- #H** - This format is an ASCII representation of a hexadecimal number with the format: <#><H><DAB...DAB>. All bytes are ASCII.
- Numeric** - Any integer (NR1 format), decimal (NR2 format), or exponential value (NR3 format). The character <E> or <e> may be used to delimit the mantissa of exponential parameters. Spaces are not allowed between the <+>, <->, <E>, <.>, or between digits. Refer to figure 8-2 for syntax diagrams of NR1, NR2, and NR3 numeric representations.
- Alpha** - Some commands require or allow alpha arguments such as "ON" or "OFF". These arguments are ASCII strings that start with an alpha character and are followed by printable character except a <SP>, <,>, <,>, <#>, <">, or <_>.

8-13. General Program Command Format Rules

The general rules of program command format are:

- The 54200A/D sends and receives Data messages in standard 7-bit ASCII code.
- The instrument responds equally to upper and lower case characters.
- The instrument responds equally to longform and shortform command headers and alpha arguments.
- The instrument responds equally to alpha and numeric arguments.
- Parameter fields containing multiple parameters require a comma (,) to delimit individual parameters.
- Alpha enumerated arguments must be separated from the command header with a space; numeric enumerated arguments and queries (?) do not require a space.

Errors in Data messages syntax are trapped and can be reported via the HP-IB. For details about detecting and reporting format errors refer to the system "STatus?" and "ERRor?" programming commands.

Integer Format NR1**Decimal Format NR2****Exponential Format NR3**

85420011

Figure 8-2. Numeric Formats NR1, NR2, and NR3 used in Command Arguments

SYSTEM COMMANDS

System commands provide general utility functions useful in many subsystems, or special functions which are simple to specify. Note the hierarchical structure of the system and subsystem commands that was shown in Programming Command Tree, Figure 8-1.

SUBSYSTEM SELECTORS

The nine subsystem selectors are commands which allow access to subsystem functions. All have a query form which provides a means to interrogate all settings in that subsystem. Subsystem selectors are:

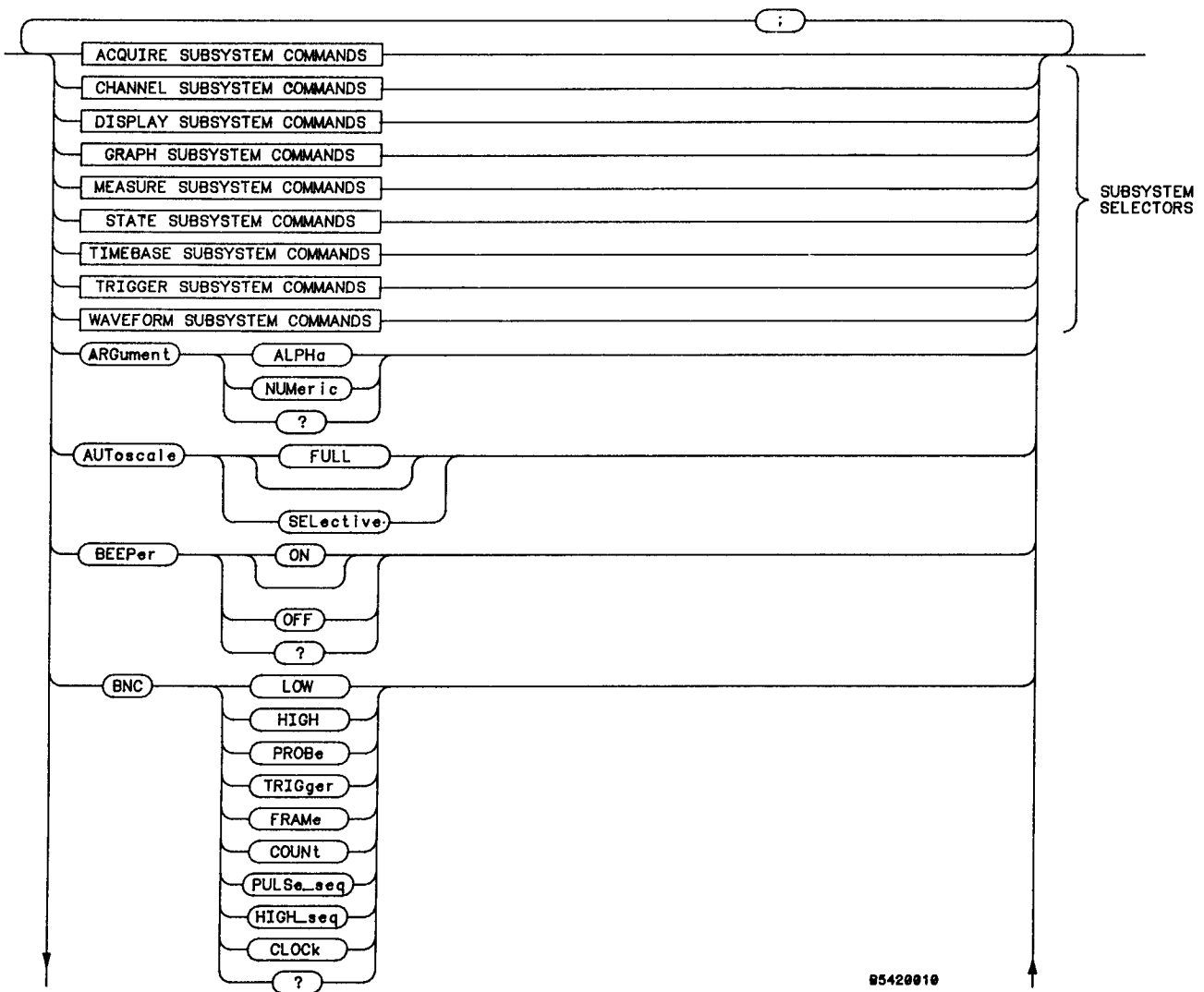
ACQuire
CHANnel<N>
DISPlay
GRAPh<N>
MEASure
STATe
TIMEbase
TRIGger
WAVEform

These commands are detailed in the corresponding subsystem.

SYSTEM COMMANDS:

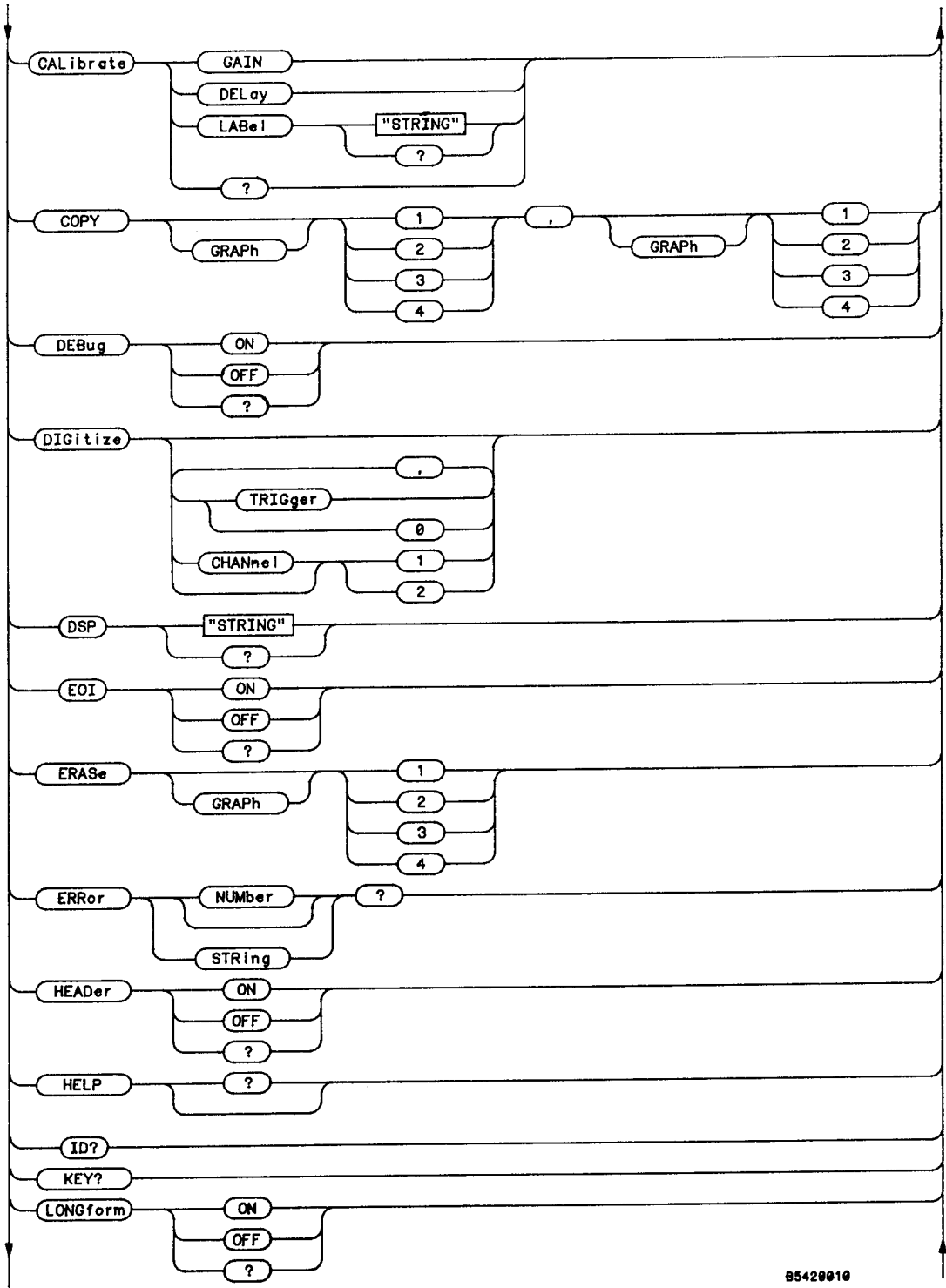
| | |
|-----------|---------------|
| ARGument | MSG |
| AUToscale | NODE |
| BEEPer | OPTions |
| BNC | PLOT |
| CALibrate | PRINT |
| COPY | READY RDY |
| DEBug | RECall |
| DIGitize | REQuest RQS |
| DSP | RESet RST |
| EOI | RESUME |
| ERASe | REVisiOn |
| ERRor | RUN |
| HEADer | SAVE |
| HELP | SERial |
| ID | SETup |
| KEY | STATus |
| LONGform | STOP |
| MENU | TEST TST |

SYSTEM COMMANDS



System Command Syntax Diagram

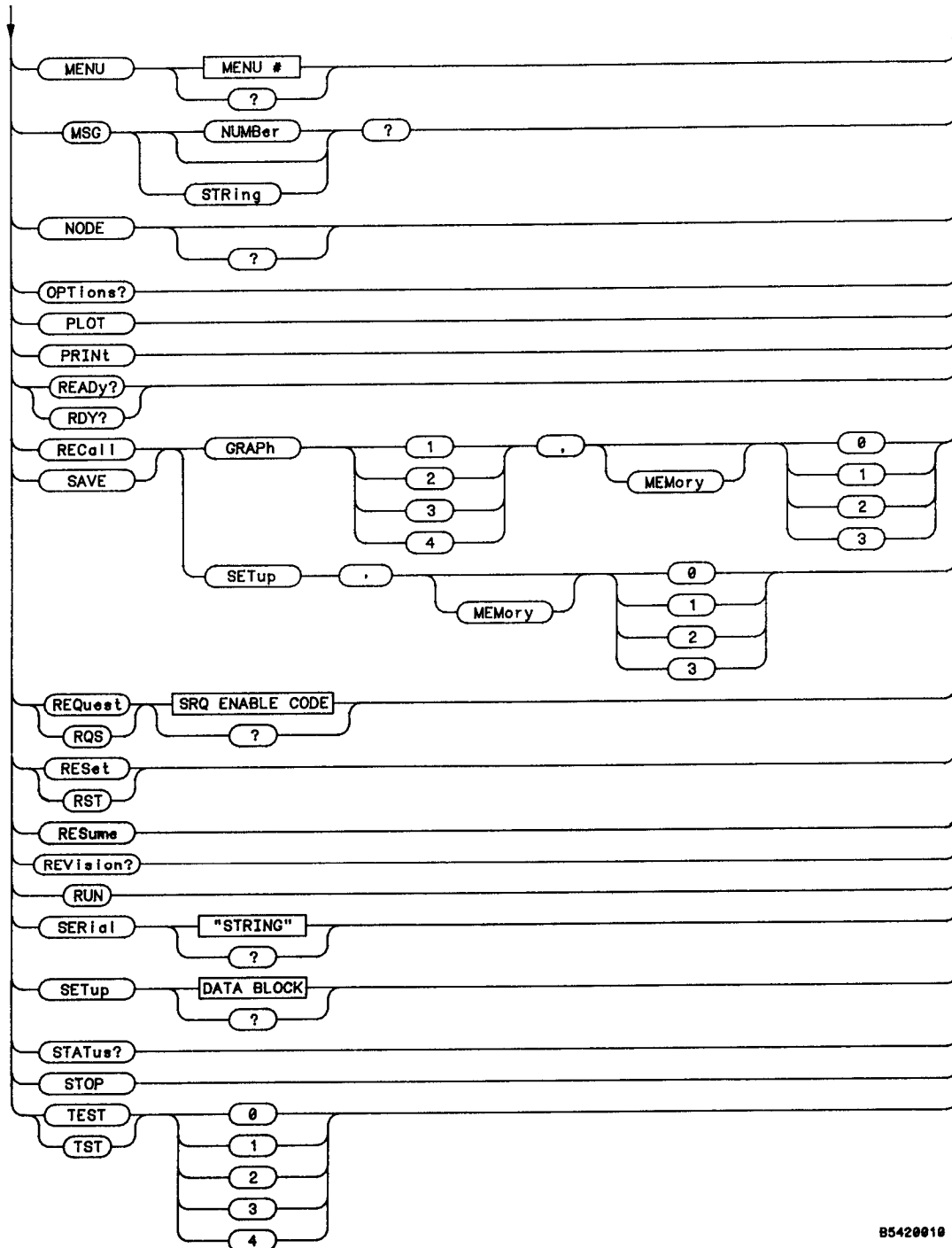
SYSTEM COMMANDS



85420010

System Command Syntax Diagram (Cont)

SYSTEM COMMANDS



B5420010

System Command Syntax Diagram (Cont)

SYSTEM COMMANDS

ARGument

Command sets output mode (instrument response to a query) for commands that have both alpha and numerical arguments. If set to alpha, the arguments are returned in alpha format and follow the same abbreviation rules as commands. If set to numeric, the arguments are returned in numerical format. This does not affect input data messages to the 54200A/D -- arguments may be input in either alpha or numerical form regardless of how the ARGument command is set. Query returns current argument mode.

Command Syntax: ARGument { [ALPha | 1]
[NUMeric | 0] }

Example: OUTPUT 707;"ARGUMENT NUMERIC"

Query Syntax: ARGument ?

Returned Format : [ARGument]<argument><crlf>

Example: OUTPUT 707;"ARGUMENT?"
ENTER 707;Argument\$
PRINT Argument\$

AUToscale

Command which performs autoscale per CHANnel, TIMEbase, and TRIGger subsystem specifications if SELECTIVE is specified. If FULL is specified or no argument is specified, a full autoscale (period and all inputs) is performed. Autoscale rearranges graph sources such that signals found (chan1 or chan2) are presented on the first graphs (1, and 2 if two inputs have signals). Any sources not autoscaled or autoscaled with no input signals are displayed on subsequent graphs. The remaining graphs are turned off unless they contain memory waveforms. Active sources are displayed in normal mode (i.e., not expanded) and will be displayed on no more than one graph after autoscale.

Command Syntax: AUToscale [FULL | 0]
[SElective | 1]

Example: OUTPUT 707;"AUTOSCALE SELECTIVE"

SYSTEM COMMANDS

BEEPer

Commands sets beeper mode. If no argument is passed, a beep is sounded and the beeper is turned on. Query returns the current beeper mode.

Command Syntax: BEEPer [OFF | 0]
[ON | 1]

Example: OUTPUT 707;"BEEP OFF"

Query Syntax: BEEPer ?

Returned Format : [BEEPer]<argument><crLf>

Example: OUTPUT 707;"BEEPER ?"
ENTER 707;Beep\$
PRINT Beep\$

BNC

Command sets the output mode of the rear-panel BNC. Query returns the current mode of rear-panel BNC.

Command Syntax: BNC { [LOW | 0]
[HIGH | 1]
[PROBe | 2]
[TRIGger | 3]
[FRAMe | 4]
[COUNT | 5]
[PULSE_SEQ | 6] (54200D only)
[HIGH_SEQ | 7] (54200D only)
[CLOCK | 8] } (54200D only)

Example: OUTPUT 707;"BNC PROBE"

Query Syntax: BNC ?

Returned Format: [BNC]<argument><crLf>

Example: OUTPUT 707;"BNC ?"
ENTER 707;Bnc\$
PRINT Bnc\$

SYSTEM COMMANDS

CALibrate

If GAIN or DELay is specified, 54200A/D performs a self cal. If LABEL is specified, 54200A/D accepts a quoted string up to 64 characters in length that may be used to indicate when the cal was performed, when the next cal should be done, etc. The rear-panel cal switch must be set the unprotected position for all of these, otherwise an error is generated.

Note

GAIN requires all front panel inputs to 54200A/D be disconnected before being executed or an erroneous cal will be performed. DELay requires rear-panel BNC be connected to INPUT 1 or an erroneous cal will be performed.

Command Syntax: CALibrate { [GAIN | 0]
[DELay | 1]
[LABEL <string> | 2 <string>] }

Example: OUTPUT 707;"CAL LABEL ""CAL PERFORMED XX/XX/XX"""

Query Syntax: CALibrate [LABEL] ?

If the LABEL argument is specified, 54200A/D returns the current cal label:

Returned Format : [CALibrate]< 64 character quoted string><crlf>

Example: DIM Cal\$[80]
OUTPUT 707;"CALIBRATE LAB ?"
ENTER 707;Cal\$
PRINT Cal\$

If no argument is sent, then 54200A/D outputs calibration data:

| | |
|--------------------------------|-------------------|
| Returned Format : [CALibrate] | |
| <NR1>,!17!<NR1><crLf> | (Chan 1 gain) |
| <NR1>,!17!<NR1><crLf> | (Chan 2 gain) |
| <NR1>,!17!<NR1><crLf> | (Trig 1 gain) |
| <NR1>,!17!<NR1><crLf> | (Trig 2 gain) |
| <NR1>,!17!<NR1><crLf> | (Trig Ext gain) |
| <NR1>,!11!<NR1><crLf> | (Chan 1 offset) |
| <NR1>,!11!<NR1><crLf> | (Chan 2 offset) |
| <NR1>,!05!<NR1><crLf> | (Trig 1 offset) |
| <NR1>,!05!<NR1><crLf> | (Trig 2 offset) |
| <NR1>,!05!<NR1><crLf> | (Trig Ext offset) |
| <NR1><crLf> | (I offset) |
| <NR1><crLf> | (I max) |
| <NR1><crLf> | (I spurious) |
| <NR3><crLf> | (Delay cal) |
| <NR1><crLf> | (Trig 1 adjust) |
| <NR1><crLf> | (Trig 2 adjust) |

SYSTEM COMMANDS

CALibrate (cont)

```
Example: OUTPUT 707;"HEADER ON;LONGFORM ON"
          DIM Cal$(800)
          OUTPUT 707;"CAL ?"
          ENTER 707 USING "-K";Cal$
          PRINT USING "K";Cal$
```

A sample of the print output of this program sequence would look like:

```
CALIBRATE
201, 99, 51, 198, 97, 50, 198, 70, 22, 201, 99, 51, 198, 97, 50, 198, 70, 22
209, 102, 52, 208, 102, 52, 205, 73, 21, 209, 102, 52, 208, 102, 52, 205, 73, 21
233, 119, 66, 231, 117, 63, 229, 87, 32, 233, 119, 66, 231, 117, 63, 229, 87, 32
236, 119, 65, 229, 119, 64, 233, 86, 32, 236, 119, 65, 229, 119, 64, 233, 86, 32
236, 119, 65, 230, 117, 64, 230, 87, 33, 236, 119, 65, 230, 117, 64, 230, 87, 33
255, 250, 255, 245, 255, 248, 255, 237, 255, 244, 255, 216
1, 109, 1, 58, 1, 87, 0, 238, 1, 56, 0, 11
0, 14, 0, 13, 0, 15
0, 13, 0, 19, 0, 14
255, 254, 255, 244, 0, 14
33
75
0
28.20E-09
0
0
```

SYSTEM COMMANDS

COPY

Command which copies data from first graph specified to second graph specified.

Command Syntax: COPY <copy specification>

<copy specification> ::= [GRAPH]<graph #>, [GRAPH]<graph #>

<graph #> ::= { 1 | 2 | 3 | 4 }

Example: OUTPUT 707;"COPY GRAPH 1,GRAPH 2"

DEBug

The system DEBug command helps those who are unfamiliar with the command syntax to find programming errors. When DEBug ON is issued, each command is shown on the 54200A/D display as the command is received over the HP-IB. The commands are defined as either inputs or outputs and appear within the < > symbols with imbedded carriage returns (CR) and line feeds (LF). The form in which commands are displayed on the 54200A/D is dependent on the current status of the system LONGform, HEADer, and ARGument commands. Spelling and syntax errors can be quickly located and corrected by stepping through a program. If an error occurs, the error is highlighted with an inverse video cursor on the 54200A/D display. An error message is then displayed, followed by message text, on the status line of the 54200A/D.

Query returns the current debug mode.

Command Syntax: DEBug { [ON | 1]
[OFF | 0] }

Example: OUTPUT 707;"DEB ON"

Query Syntax: DEBug ?

Returned Format : [DEBug]<argument><crlf>

Example: OUTPUT 707;"DEBUG?"
ENTER 707;Debug\$
PRINT Debug\$

SYSTEM COMMANDS

DIGitize

Upon receipt of the DIGitize command, all graphs are turned off and acquisition is stopped. DIGitize automatically assigns channels to their default graphs for convenience of measurements following acquisition. The DIGitize command then causes 54200A/D to start running, as if the RUN command had been sent.

When each requested source completes its acquisition, it is "turned off" (no longer acquired).

When all are acquired, acquisition is stopped as though the STOP command had been issued and the message "Acquisition Complete" is displayed in the status line.

Note

This command does not have a finite number of arguments, therefore the last argument cannot be punctuated with a ",".

If no argument is included with DIGitize, then the current graph sources determine what is acquired. If no active sources are assigned to any graphs, the default is made as follows:

- If one graph is displayed, channel 1 is assigned to graph 1.
- If two graphs are displayed, channel 1 is assigned to graph 1 and channel 2 is assigned to graph 2.
- If four graphs are displayed, channel 1 is assigned to graph 1, channel 2 is assigned to graph 2, and trigger view is assigned to graph 3.
- All other graphs are set to off.

Command Syntax : DIGitize [<acquire spec>[,<acquire spec>]!0!N!]

```
<acquire spec> ::= { [ TRIGger      | 0  ]
                     [ CHANnel1    | 1  ]
                     [ CHANnel2    | 2  ] }
```

Example: OUTPUT 707;"DIGITIZE TRIGGER,CHANNEL1"

SYSTEM COMMANDS

DSP

Command places string up to 32 characters long in the message area of the 54200A/D display. Priority is higher than normal advisories, lower than errors. The null string ("") clears the DSP message. Query returns advisory string currently in status line.

Command Syntax: DSP <quoted string>

Example : OUTPUT 707;"DSP ""Set offset, then press CONTINUE"""

Query Syntax: DSP ?

Returned Format: [DSP]<string><crlf>

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

EOI

Command sets EOI mode for data outputs. If on, then EOI is asserted with the last data byte sent. Command does not affect response to EOI on input data messages. Query returns the current status of EOI.

Command Syntax: EOI { [ON | 1]
[OFF | 0] }

Example: OUTPUT 707;"EOI OFF"

Query Syntax: EOI ?

Returned Format : [EOI]<argument><crlf>

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

ERASe

Command erases specified graph. If the graph contains a memory waveform, the graph is turned off. If the graph contains an active source (channel or trigger) then the waveform display is erased and the acquisition count for that waveform is set to "0" (no data acquired yet).

Command Syntax: ERASe [GRAPH] { 1 | 2 | 3 | 4 }

Example: OUTPUT 707;"ERASE GRAPH 1"

SYSTEM COMMANDS

ERRor

Query which responds with the error code for the first error condition since ERRor?, device clear, RESet, instrument power-on, or RQS<mask spec>. If STRing is specified, a quoted string up to 32 characters long is returned. If NUMber is specified or no argument is sent, only the error number is returned. All errors will halt parsing. Refer to system MSG query for other types of messages.

Query Syntax: ERRor [NUMber | 0]
[STRing | 1] ?

Returned Format : [ERRor][<NR1><crlf> | <string><crlf>]

Example: OUTPUT 707;"ERROR STRING ?"
ENTER 707;String\$
PRINT String\$

Device Dependent Error Messages (Primarily from local operation):

| <u>ERROR NO.</u> | <u>ERROR STRING</u> |
|------------------|----------------------------------|
| 3 | Register Write Protected |
| 4 | Not Allowed In This Menu |
| 5 | Graphics Must Be On |
| 8 | Use [NEXT] [PREV] Keys |
| 9 | Numeric Entry Required |
| 10 | Use Hex Keys |
| 11 | Use Alphanumeric Keys |
| 13 | Requires Correction First |
| 15 | DON'T CARE Not Allowed |
| 16 | Use 0 or 1 |
| 17 | Use 0, 1, or DON'T CARE |
| 18 | Use 0 thru 7 |
| 19 | Use 0 thru 7 or DON'T CARE |
| 20 | Use 0 thru 3 |
| 21 | Use 0 thru 3 or DON'T CARE |
| 22 | Value is Too Large |
| 23 | CHS Not Allowed |
| 24 | Use CHS Key |
| 25 | Value Not Allowed |
| 30 | Maximum INSERT's Used |
| 31 | Use 0 thru 9 |
| 39 | Turn Lines Off First |
| 40 | Disable Accumulate First |
| 43 | Must Be in Talk Only |
| 49 | Value Out of Range |
| 54 | FAILED Test <N> / FAILED Cal <N> |
| 58 | No Hardcopy Device |
| 70 | Cal RAM Write Protected |
| 89 | Local Lockout In Effect |
| 90 | Key Not Allowed in Remote |

SYSTEM COMMANDS

ERRor (cont)

Standardized HP-IB Error Messages:

| <u>ERROR NO.</u> | <u>ERROR STRING</u> |
|------------------|----------------------------|
| -100 | In Controller Command |
| -110 | Invalid Header |
| -119 | Command Header Expected |
| -120 | In Numeric Argument |
| -121 | Numeric Data Expected |
| -123 | Numeric Overflow |
| -125 | Invalid Representation |
| -129 | Missing Argument |
| -130 | In Non-numeric Argument |
| -131 | Character Data Expected |
| -132 | String Data Expected |
| -133 | Block Data Expected |
| -134 | String Too Long |
| -135 | Block Length |
| -136 | Block Checksum |
| -137 | Invalid Argument |
| -139 | Missing Argument |
| -142 | Too Many Arguments |
| -143 | Missing Argument Delimiter |
| -150 | Query Expected |
| -151 | Query Not Permitted |
| -200 | Execution Not Possible |
| -211 | Settings Conflict |
| -212 | Number Out of Range |
| -231 | Input Buffer Overflow |
| -233 | Output Buffer Empty |
| -300 | Hardware Failure |
| -302 | System Fault |
| -311 | RAM Failure |
| -312 | RAM Data Loss |
| -313 | CAL Factor Loss |
| -321 | ROM Checksum |
| -322 | Firmware Incompatible |
| -330 | Power-up Failed |

SYSTEM COMMANDS

HEADer

Command sets command header echo mode for query responses; query responses will only return the command header when HEADer is set ON. Query form of this command returns the current echo mode.

Command Syntax: HEADer {[ON | 1]
[OFF | 0]}

Example: OUTPUT 707;"HEADER ON"

Query Syntax: HEADer ?

Returned Format : [HEADer]<argument><crLf>

Example: OUTPUT 707;"HEADER ?"
ENTER 707;Header\$
PRINT Header\$

HELP

Query which returns all commands in tabular form organized by subsystem. The command arguments are not output. "?" optional for this query.

Query Syntax: HELP [?]

Returned Format: [HELP]<string with imbedded <crLf's> >

Example: OUTPUT 707;"EOI ON"
DIM Help\$[2000]
OUTPUT 707; "HELP?"
ENTER 707 USING "-K"; Help\$
PRINT USING "K"; Help\$

ID

Query which returns model identification string, HP54200A or HP54200D.

Query Syntax: ID ?

Returned Format: [ID]"HP54200A"<crLf> or
[ID]"HP54200D"<crLf>

Example: OUTPUT 707;"ID ?"
ENTER 707;Id\$
PRINT Id\$

SYSTEM COMMANDS

KEY

Query which returns the code of the first key pressed in the key buffer. A key queue of length 10 is held. If more than 10 keys have been pressed, the most recent key is the last in the buffer.

Query Syntax: KEY ?

Returned Format: [KEY]<key code><crlf>

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

| CODE | KEY | CODE | KEY | CODE | KEY |
|------|---------------|------|--------------------|------|------------------------------|
| 1 | NEXT[]/FINE | 34 | CHS | 67 | B |
| 2 | PREV[]/FINE | 35 | Not assigned | 68 | C |
| 3 | ← FIELD | 36 | Not assigned | 69 | D |
| 4 | → FIELD | 37 | sec/Volt | 70 | E |
| 5 | ↑ FIELD | 38 | msec/mV | 71 | F |
| 6 | ↓ FIELD | 39 | usec | 72 | G |
| 7 | ← CURSOR | 40 | nsec | 73 | H |
| 8 | → CURSOR | 41 | Not assigned | 74 | I |
| 9 | ↑ CURSOR | 42 | Not assigned | 75 | J |
| 10 | ↓ CURSOR | 43 | . | 76 | K |
| 11 | SYSTEM MENU | 44 | SAVE | 77 | L |
| 12 | STATUS MENU | 45 | RECALL | 78 | M |
| 13 | DISPLAY MENU | 46 | COPY | 79 | N |
| 14 | TRIG MENU | 47 | ERASE | 80 | O |
| 15 | CHAN MENU | 48 | Not assigned | 81 | P |
| 16 | TIME MENU | 49 | 0 | 82 | Q |
| 17 | RUN/STOP | 50 | 1 | 83 | R |
| 18 | RESUME | 51 | 2 | 84 | S |
| 19 | AUTO SCALE | 52 | 3 | 85 | T |
| 20 | ECL | 53 | 4 | 86 | U |
| 21 | HARDCOPY | 54 | 5 | 87 | V |
| 22 | TTL | 55 | 6 | 88 | W |
| 23 | CLEAR ENTRY | 56 | 7 | 89 | X |
| 24 | INSERT | 57 | 8 | 90 | Y |
| 25 | DELETE | 58 | 9 | 91 | Z |
| 26 | LOCAL | 59 | Not assigned | 92 | RISE MEASUREMENT |
| 27 | CURSOR VALUES | 60 | ←EDGE | 93 | FALL MEASUREMENT |
| 28 | DON'T CARE | 61 | EDGE → | 94 | DUTY MEASUREMENT |
| 29 | CURSOR X | 62 | FREQ MEASUREMENT | 95 | DELAY MEASUREMENT |
| 30 | CURSOR O | 63 | PERIOD MEASUREMENT | 96 | V _{amp} MEASUREMENT |
| 31 | CURSOR X&O | 64 | +WIDTH MEASUREMENT | 97 | V _{max} MEASUREMENT |
| 32 | CURSOR NORM | 65 | -WIDTH MEASUREMENT | 98 | V _{min} MEASUREMENT |
| 33 | CURSOR EXPAND | 66 | A | 99 | V _{rms} MEASUREMENT |
| | | | | 100 | BLUE SHIFT KEY |

SYSTEM COMMANDS

LONGform

Command sets longform mode for instrument responses to queries. If OFF then command headers and alpha arguments are sent from 54200A/D in the abbreviated form, else the whole word is output. This does not affect input data messages to the 54200A/D -- headers and arguments may be input to the 54200A/D in either long or short form regardless of how the LONGform command is set. Query returns the LONGform status.

Command Syntax: LONGform { [ON | 1]
[OFF | 0] }

Example: OUTPUT 707;"LONG 1"

Query Syntax: LONGform ?

Returned Format : [LONGform]<argument><crlf>

Example: OUTPUT 707;"LONGFORM?"
ENTER 707;Long\$
PRINT Long\$

MENU

Command displays the desired menu. Menus 41 and 42 are available for the 54200D only. The 54200A treats 41 or 42 as parameter errors. Query returns the current menu selected.

Command Syntax: MENU { [0] (None/measurement answers)
[10] (System Peripherals)
[11] (System Calibration)
[12] (System Self Tests)
[20] (Status Configuration)
[21] (Status Measurements)
[22] (Status Memory)
[30] (Display)
[40] (Trigger Analog)
[41] (Trigger State Assignment)
[42] (Trigger State Sequence)
[50] (Channel 1)
[51] (Channel 2)
[60] (Timebase)
[70] } (Text-see Display subsystem TEXT command)

Example: OUTPUT 707;"MENU 40"

Query Syntax: MENU ?

Returned Format: [MENU]<NR1><crlf>

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

SYSTEM COMMANDS

MSG

Query which responds with the message code for the first advisory since MSG query, device clear, RESet, instrument power-on, or RQS<mask spec>. If STRing is specified, a quoted string up to 32 characters long is returned. If NUMber is specified or no argument is sent, only the message number is returned. Text, Status, Warning, and Wait messages but not ERRor messages are returned. Refer to system ERRor query for error messages. The MSG query clears the advisory bit in the status byte (if set and no SRQ is pending).

Query Syntax: MSG [NUMber | 0]
[STRing | 1] ?

Returned Format : [MSG][<NR1><crlf> | <string><crlf>]

Example: OUTPUT 707;"MSG NUMBER?"
ENTER 707;Message
PRINT Message

Device Dependent Message Codes:

| <u>MSG NO.</u> | <u>MESSAGE STRING</u> |
|----------------|---------------------------------|
| 1 | Power-Up Complete |
| 6 | Expansion Limit Reached |
| 7 | No Waveform Exists |
| 12 | Press STOP Key First |
| 14 | End of Waveform |
| 26 | INSERT to Add Sequence Term |
| 27 | INSERT to Add OR'd Pattern |
| 28 | INSERT to Add New Label |
| 29 | One "*" Required For Each Label |
| 32 | Each Label Must Have a Name |
| 33 | All Names Must Be Unique |
| 34 | At Least 1 Clock Edge Required |
| 35 | Delay Rounded |
| 36 | Enter 55 to 99 % |
| 37 | Enter 1 to 45 % |
| 38 | DELETE to Remove Field |
| 41 | INSERT to Add AND'd Pattern |
| 42 | Only One Label Allowed |
| 44 | Signal Aliased |
| 45 | Command Ignored |
| 46 | Coupled Function Changed |
| 47 | Offset/Trig Will Change |
| 48 | INSERT-Proceed Else CLEAR ENTRY |
| 50 | Enter Cal 0 thru 9 |
| 51 | Enter Test 0 thru 2 |
| 52 | No Edge Found |

SYSTEM COMMANDS

MSG (cont)

| <u>MSG NO.</u> | <u>MESSAGE STRING</u> |
|----------------|-------------------------------|
| 55 | Hardcopy in Progress |
| 56 | Acquired Frame <N> (See Note) |
| 57 | Hardcopy Device Down |
| 59 | No Ext Trig Auto |
| 60 | Hardcopy Aborted |
| 61 | < Not defined > |
| 62 | < Not defined > |
| 63 | Waiting to Send Data |
| 64 | Waiting to Receive Data |
| 65 | < Not defined > |
| 66 | Executing Autoscale |
| 67 | Exceeds Graph Scale |
| 68 | Enter Address 0 thru 30 |
| 69 | Default Complete |
| 71 | < Not defined > |
| 72 | Armed by State Trigger |
| 73 | Trigger On State Only |
| 74 | Enter Graph 1 thru 4 |
| 75 | Enter Memory 0 thru 3 |
| 76 | Auto Scale Disabled |
| 77 | Offset/Trig Rounded |
| 78 | Threshold Limited |
| 79 | Enter Value and Units |
| 80 | Signal Not Found |
| 81 | < Not defined > |
| 82 | Acquisition Stopped |
| 83 | Acquisition Complete |
| 84 | No Trigger Found |
| 85 | Waiting for Trigger |
| 87 | SHIFT Selects Alphabet |
| 88 | < Not defined > |

Note

Frame number returned may not agree with count displayed when message was generated.

Standardized HP-IB Warning Messages:

| <u>MSG NO.</u> | <u>MESSAGE STRING</u> |
|----------------|-----------------------|
| -122 | Number Rounded |
| -230 | Transmission Aborted |

SYSTEM COMMANDS

NODE

Command that resets 54200A/D parser to SYStem node. If query, 54200A/D returns currently selected node (system or subsystem).

Command Syntax: NODE

Example: OUTPUT 707;"NODE"

Query Syntax: NODE ?

Returned format: [NODE]{

| | |
|----------|---|
| SYStem | 0 |
| CHANnel | 1 |
| TRIGger | 2 |
| TIMEbase | 3 |
| DISPlay | 4 |
| GRAPH | 5 |
| ACQuire | 6 |
| WAVEform | 7 |
| MEASure | 8 |
| STATe | 9 |

}}<crLf>

Example: OUTPUT 707;"NODE ?"
 ENTER 707;Node\$
 PRINT Node\$

OPTions

Query which returns what options are installed in 54200A/D. There currently are no options.

Query Syntax: OPTions ?

Returned Format : [OPTions]{NONE | 0 }<crLf>

Example: OUTPUT 707;"OPTIONS?"
 ENTER 707;Option\$
 PRINT Option\$

SYSTEM COMMANDS

PLOT

Command sets hardcopy device to plotter and does hardcopy dump of 54200A/D display in a format acceptable to an HPGL plotter as soon as the 54200A/D is next addressed to talk.

Command Syntax: PLOT

Example:

```

210  CLEAR 707          ! Clear interface buffers.
220  OUTPUT 707;"PLOT"  ! Starts plotter buffering.
230  SEND 7;UNT UNL     ! Clears bus, set ATN line at controller true.
240  SEND 7;LISTEN 5    ! Tells plotter at address 5 to listen.
250  SEND 7;TALK 7      ! Sets 54200A/D to talk mode.
260  SEND 7;DATA        ! Sets ATN line at controller to false
270                      ! so data can be transferred.
280  WAIT 50            ! Wait 50 seconds for transfer to finish
290  !                  Note: If programming, use the SRQ capabilities
300  !                  of the 54200A/D to determine if the transfer
310  !                  is complete. Attempting to program the
320  !                  54200A/D while making a hardcopy dump will
330  !                  cause errors.
340  !

```

PRINT

Command sets hardcopy device to printer and does hardcopy dump of 54200A/D display in a format acceptable to a printer compatible with the HP RASTER GRAPHICS STANDARD as soon as the printer is next addressed to talk.

Command Syntax: PRINT

Example:

```

210  CLEAR 707          ! Clear interface buffers.
220  OUTPUT 707;"PRINT" ! Starts print buffering.
230  SEND 7;UNT UNL     ! Clears bus, set ATN line at controller true.
240  SEND 7;LISTEN 1    ! Tells printer at address 1 to listen.
250  SEND 7;TALK 7      ! Sets 54200A/D to talk mode.
260  SEND 7;DATA        ! Sets ATN line at controller to false
270                      ! so data can be transferred.
280  WAIT 25            ! Wait 25 seconds for transfer to finish
290  !                  Note: If programming, use the SRQ capabilities
300  !                  of the 54200A/D to determine if the transfer
310  !                  is complete. Attempting to program the
320  !                  54200A/D while making a hardcopy dump will
330  !                  cause errors.
340  !

```

SYSTEM COMMANDS

READY | RDY

Query which returns an integer equivalent to the 54200A/D ready register (ready byte of the status word). The ready register is dynamic and indicates the current status of the instrument. The ready register is defined as follows:

Query Syntax: { READY | RDY } ?

Returned Format: [READY]<NR1><crlf>

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

The Ready Register

| BIT | MASK WEIGHT | READY BIT CONDITION |
|-----|-------------|---|
| 7 | 128 | Not used, always 0. |
| 6 | 64 | Cal = High indicates that self calibration has completed execution. |
| 5 | 32 | Test = High indicates that the requested self test has completed execution. |
| 4 | 16 | Hard = Hardcopy complete - High indicates that the last byte of printer or plotter dump has been sent and received |
| 3 | 8 | Data = Data available - High indicates that something is in the buffer waiting to be read. |
| 2 | 4 | Acq = Acquisition complete - High indicates that all waveforms are acquired. |
| 1 | 2 | Trig = Triggered - High indicates that the instrument is receiving triggers. Can only be set while running; if stopped, reflects status previous to stopping. |
| 0 | 1 | Parse = Parse complete - High indicates that the last command has been completely parsed. |

SYSTEM COMMANDS

RECall

Command displays a waveform from 54200A/D memory on the specified graph or, restores the specified instrument setup. Refer to the SAVE command to save waveforms or setups.

Command Syntax: RECall { [GRAPH<graph#>,[MEMory]<memory#>]
[[SETup,][MEMory]<memory#>] }

<graph#> ::= { 1 | 2 | 3 | 4 }
<memory#> ::= { 0 | 1 | 2 | 3 }

Examples: OUTPUT 707;"RECALL GRAPH 1,2" *is equivalent to*
OUTPUT 707;"RECALL GRAPH 1,MEMORY 2"

OUTPUT 707;"RECALL SETUP,MEMORY 0" *is equivalent to*
OUTPUT 707;"RECALL SETUP 0" *and*
OUTPUT 707;"RECALL 0"

REQuest | RQS

The REQuest command sends an SRQ enable code which is an integer representing the binary weighted values of the condition bits in the ready mask and the RQS mask.

The ready mask determines what ready conditions cause the ready bit in the status byte to be set. The RQS mask determines what conditions will cause an SRQ to be issued.

Setting the SRQ enable code clears any pending SRQ, as well as all errors, messages and keys awaiting query.

Bits 0, 1, 6 are don't cares and are always set to zero.

Query returns the current SRQ enable code.

Command Syntax: { REQuest | RQS }<SRQ enable code>

Example: OUTPUT 707;"RQS 56"

Query Syntax: { REQuest | RQS } ?

Returned Format: [REQuest]<SRQ enable code><cr><lf>

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

SYSTEM COMMANDS

REQuest | RQS (cont)

The ready mask is defined as follows:

The Ready Byte

| BIT | MASK WEIGHT | READY BIT CONDITION |
|-----|-------------|---|
| 15 | 32768 | Not used, always 0. |
| 14 | 16384 | Cal = High indicates that self calibration has completed execution. |
| 13 | 8192 | Test = High indicates that the requested self test has completed execution. |
| 12 | 4096 | Hard = Hardcopy complete - High indicates that the last byte of printer or plotter dump has been sent and received |
| 11 | 2048 | Data = Data available - High indicates that something is in the buffer waiting to be read. |
| 10 | 1024 | Acq = Acquisition complete - High indicates that all waveforms are acquired. |
| 9 | 512 | Trig = Triggered - High indicates that the instrument is receiving triggers. Can only be set while running; if stopped, reflects status previous to stopping. |
| 8 | 256 | Parse = Parse complete - High indicates that the last command has been completely parsed. |

SYSTEM COMMANDS

REQuest | RQS (cont)

The RQS mask is defined as follows:

The Status Byte

| BIT | MASK WEIGHT | STATUS BIT CONDITION |
|--|-------------|--|
| 7 | 128 | MSG = High indicates that a message was displayed on the status line of the display. A MSG query is used to determine the message code. |
| 6 | 64 | RQS = Not used, always 0. |
| 5 | 32 | ERR = Error - High indicates an error occurred. An ERRor query is used to determine error code. |
| 4 | 16 | RDY = Ready - High indicates the instrument is ready. This is based on the ready mask. A RDY query is used to determine condition. |
| 3 | 8 | LCL = Local switch or power cycle - High indicates that the instrument has been switched to local from the front panel or that the power was cycled off then on again. |
| 2 | 4 | FPS = Front panel service request - High indicates a front panel key has been pressed. A KEY query is used to determine the key code. |
| 1 | 2 | PWR = Not used, always 0. |
| 0 | 1 | RQC = Not used, always 0. |
| <p>Notes: 1. To set the RQS bit and SRQ bus control line true, the condition must be enabled in the RQS mask.</p> <p>2. If no condition is enabled, the 54200A/D can not set the SRQ bus control line nor the RQS bit true. However, bits 2-5 and 7 of the status byte are set to indicate which conditions have occurred.</p> | | |

SYSTEM COMMANDS

RESet | RST

The 54200A/D has three distinct and non-overlapping levels of reset capabilities; two are HP-IB bus commands and one is a 54200A/D system command.

1. IFC (HP-IB bus command)
 - Per IEEE-488, resets only the device interface functions (T, TE, L,LE, and C) as defined.
2. DCL or SDC (HP-IB bus commands)
 - Resets the parser state so that no subsystem is enabled.
 - Clears device status word (system STATUS command), and KEY, MSG, and ERROR queues.
 - Clears all IEEE-488 input and output buffers.
 - DOES NOT affect any device variables handled by system RST command, in particular:
 - WAVEform FORMat
 - LONGform
 - EOI
 - HEADer
 - ARGument
 - RQS
3. RESet (RST) 54200A/D system command.
 - Resets parser state so that no subsystem is enabled.
 - Clears device status word (system STATUS command), and KEY, MSG, and ERROR queues.
 - Basically the same as key-down power up, except no system tests are performed or corresponding failure default actions.
 - Changes these command variables:
 - WAVEform FORMat = WORD
 - EOI = ON
 - LONGform = OFF
 - HEADer = OFF
 - ARGument = NUMeric
 - RQS = 32512 (Ready bits all true except bit 15, status bits all false)
 - Sets GRAPH, TRIGGER, and WAVEform SOURCE parameters to CHANNEL 1
 - Sets MEASURE SOURCE parameter to GRAPH1

Command Syntax: RESet | RST

Example: OUTPUT 707;"RST"

When RESet system command has been executed, the condition of the 54200A/D is as follows:

| <u>CONDITION</u> | <u>SETTING</u> |
|------------------|----------------|
| TIMEBASE: | |
| Sweep mode | Auto |
| Time Range | 10 us |
| Reference | Center |
| Delay | 0.0 s |
| Autoscale | Period |
| Alias detect | On |

SYSTEM COMMANDS

RESet | RST (cont)

| <u>CONDITION</u> | <u>SETTING</u> |
|----------------------|---------------------|
| CHANNELS: | |
| Range | 5.0 V |
| Offset | 0.0 V |
| Store mode | Normal |
| Labels | " " |
| Coupling | DC |
| Probe Attenuation | 1:1 |
| Autoscale | Enabled |
| TRIGGER: | |
| Mode | Analog only |
| Range | Track (5.0 V) |
| Source | Channel 1 |
| Level | 0.0 V |
| Slope | Positive |
| Store mode | Normal |
| Label | Blank |
| Autoscale | Enabled |
| DISPLAY: | |
| Graticule | Frame |
| # of graphs | 2 |
| Graph sources | |
| 1 | Channel 1 |
| 2 | Channel 2 |
| Reference lines | Off |
| Accumulate mode | Disabled |
| MEASUREMENTS: | Standard |
| OTHER: | |
| Running | True |
| HP-IB address/mode | Not changed |
| Hardcopy device | Printer |
| Rear BNC | Constant low |
| Beeper | On |
| Setup Labels | Blank |
| Stored: | |
| Setups | Unprotected |
| Waveforms | Cleared |
| Data filter | On |
| STATE: | |
| All settings | See menus next page |

SYSTEM COMMANDS

```

Trigger [ State ] -----Status: No Trigger Found-----
                                     TALK ONLY
Define [ Assignment ]           Trigger Mode [ Analog Only ]

                                     Clock
                                     JKL
State Mode [ Normal ]           [ ↑.. ] -----
Pod 2      Pod 1      Pod 0
Multiplexing [ Off ]           [ TTL ] [ TTL ] [ TTL ]
                                8.....0 8.....0 8.....0
Label      Pol
  A        [+ ] [*****] [*****] [*****]

```

Trigger State Assignment Menu after RESet

```

Trigger [ State ] -----Status: No Trigger Found-----
                                     TALK ONLY
Define [ Sequence ]           Trigger Mode [ Analog Only ]

In Sequence,
  find 00001 Occurrences of           [or ] Trigger Teri
      [any state]
  then Do Nothing

Sequence Restart on no state

Label>      A
Base > [ HEX ]

a  [XXXXXXXX]
b  [XXXXXXXX]
c  [XXXXXXXX]
d  [XXXXXXXX]

```

Trigger State Sequence Menu after RESet

SYSTEM COMMANDS

RESUMe

Command causing 54200A/D to resume running. This differs from the RUN command in that averaging or enveloping continue with the data already obtained.

Command Syntax: RESUMe

Example: OUTPUT 707;"RESUMe"

REVision

Query which returns the revision date of 54200A/D firmware.

Query Syntax: REVision ?

Returned Format : [REVision]<date code><crlf>

<date code>::=<NR1> (YYWW : where YY is years since 1960
and WW is week of that year)

Example: OUTPUT 707;"REVISION?"
ENTER 707;Rev\$
PRINT Rev\$

RUN

Command causing 54200A/D to start running. Same as pressing the 54200A/D front-panel RUN key. Waveforms are initialized to count of 0 (no data) and when terminal count is satisfied the ACQ bit is set in the ready byte (see READy command). Differs from the DiGitize command in that acquisition is not halted when terminal count is satisfied. RUN does not disturb graph source selection or expansion, so is very useful for specialized measurements.

Command Syntax: RUN

Example: OUTPUT 707;"RUN"

SYSTEM COMMANDS

SAVE

Command saves the waveform or setup in the specified memory. If the memory is protected, an error is generated and SAVE is not executed. Protection can be cleared via the RESet command, but cannot be set via HP-IB. Refer to RECall command to recall waveforms and setups.

Command Syntax: SAVE {[GRAPh<graph#>],[MEMory]<memory#>]
[[SETup,] [MEMory]<memory#>]}

<graph#> ::= { 1 | 2 | 3 | 4 }
<memory#> ::= { 0 | 1 | 2 | 3 }

Examples: OUTPUT 707;"SAVE GRAPH 1,2" *is equivalent to*
OUTPUT 707;"SAVE GRAPH 1,MEMORY 2" *and*

OUTPUT 707;"SAVE SETUP,MEMORY 0" *is equivalent to*
OUTPUT 707;"SAVE SETUP 0" *and*
OUTPUT 707;"SAVE 0"

SERial

Command programs the serial number of the 54200A/D. Any string is accepted up to 10 characters. If less than 10 characters, it is padded with spaces to length 10. A checksum is made and the value is stored in non-volatile RAM. The rear-panel cal switch must be set to the not protected position, otherwise SERial is a query-only command. Query returns 54200A/D serial number.

Command Syntax: SERial <string>

Example: OUTPUT 707;"SERIAL ""2511A01219"""

Query Syntax: SERial ?

Returned Format : [SERial]<serial number code><cr|f>

Example: OUTPUT 707;"SERIAL?"
ENTER 707;Serial\$
PRINT Serial\$

SYSTEM COMMANDS

SETup

Command sets up 54200A/D according to learn string. Query returns 54200A/D learn string. Block length is 984 bytes. Block format #A is discussed in paragraph 8-12 at the front of this section.

Query Syntax: SETup ?

Returned Format: [SETup] <block type A>

Example: DIM Learn\$[1000]
OUTPUT 707;"HEADER OFF;SETUP?"
ENTER 707 USING "-K";Learn\$

Command Syntax: SETup <#A>,<number of bytes in learn string>
<learn string>

Example: OUTPUT 707 USING "#,K";"SETUP ",Learn\$

STatus

The instrument status word is a 16-bit binary word which is returned as an integer, and contains information about the instrument conditions that set the ready bit in the status byte and/or generate a Require Service message. The upper 8 bits of the status word are known collectively as the ready byte, while the lower 8 bits correspond to the status byte sent during a serial poll. The STatus query is used to read the status word representing the current status of the 54200A/D. Unlike the response to a serial poll, the conditions are dynamic, not latched. Therefore the status response reflects current status.

A companion 16 bit word, the request mask, is used to specify both those conditions in the ready byte that set the ready bit in the status byte, and those conditions in the status byte that generate a Require Service message. The bits in the request mask have the same meanings as those in the instrument status word. The ready bit in the status byte is set when all of the conditions corresponding to bits in the ready mask are true at the same time. This bit is actually set on the transition of the last condition to become true. The REQuest system command is used to specify the request mask.

Query Syntax: STatus ?

Returned Format: [STatus]<NR1>

Example: OUTPUT 707;"STATUS?"
ENTER 707;Status\$
PRINT Status\$

SYSTEM COMMANDS

STatus (cont)

The ready byte of the 16-bit status word is defined as follows:

*The Upper Byte of the Status Word
(The Ready Byte)*

| BIT | MASK WEIGHT | READY BIT CONDITION |
|-----|-------------|---|
| 15 | 32768 | Not used, always 0. |
| 14 | 16384 | Cal = High indicates that self calibration has completed execution. |
| 13 | 8192 | Test = High indicates that the requested self test has completed execution. |
| 12 | 4096 | Hard = Hardcopy complete - High indicates that the last byte of printer or plotter dump has been sent and received |
| 11 | 2048 | Data = Data available - High indicates that something is in the buffer waiting to be read. |
| 10 | 1024 | Acq = Acquisition complete - High indicates that all waveforms are acquired. |
| 9 | 512 | Trig = Triggered - High indicates that the instrument is receiving triggers. Can only be set while running; if stopped, reflects status previous to stopping. |
| 8 | 256 | Parse = Parse complete - High indicates that the last command has been completely parsed. |

SYSTEM COMMANDS

STAtus (cont)

The status byte of the 16-bit status word is defined as follows:

*The Lower Byte of the Status Word
(The Status Byte)*

| BIT | MASK WEIGHT | STATUS BIT CONDITION |
|-----|-------------|--|
| 7 | 128 | MSG = High indicates that a message was displayed on the status line of the display. A MSG query is used to determine the message code. |
| 6 | 64 | RQS = Requesting service - High indicates that this instrument requested service. |
| 5 | 32 | ERR = Error - High indicates an error occurred. An ERROR query is used to determine error code. |
| 4 | 16 | RDY = Ready - High indicates the instrument is ready. This is based on the ready mask. A RDY query is used to determine condition. |
| 3 | 8 | LCL = Local switch or power cycle - High indicates that the instrument has been switched to local from the front panel or that the power was cycled off then on again. |
| 2 | 4 | FPS = Front panel service request - High indicates a front panel key has been pressed. A KEY query is used to determine the key code. |
| 1 | 2 | PWR = Not used, always 0. |
| 0 | 1 | RQC = Not used, always 0. |

SYSTEM COMMANDS

STOP

Command which causes 54200A/D to stop running.

Command Syntax: STOP

Example: OUTPUT 707;"STOP"

TEST | TST

Command which executes the specified self test. Failures are reported as errors and may be queried with ERRor?.

Note

Inputs must be properly configured before the TEST command is issued:

TEST 0 - No special configuration required.

TEST 1 - All input signals must be disconnected.

TEST 2 - Connect rear panel BNC to INPUT 1 BNC.

TEST 3 - Connect rear panel BNC to INPUT 2 BNC.

TEST 4 - Connect rear panel BNC to EXT TRIG BNC.

Command Syntax: {TEST | TST} <test number>

| | |
|---------------------|---------------------------------------|
| <test number> ::= 0 | (Self test 0: CPU/Memory) |
| 1 | (Self test 1: Acquisition/Trigger) |
| 2 | (Self test 2: Channel 1 Input) |
| 3 | (Self test 2: Channel 2 Input) |
| 4 | (Self test 2: External Trigger Input) |

Example: OUTPUT 707;"TEST 4"

SYSTEM COMMANDS

Notes

ACQUIRE SUBSYSTEM

ACQuire

The acquire subsystem allows the store mode for channel 1, channel 2, and trigger to be set the same with two commands, TYPE and COUNT. The POINTs and COMPlete commands are included to provide compatibility between HP 54200A/D programs and HP 54100A/D programs. The TYPE and COUNT functions may be selectively programmed by using the STORe command of the channel and trigger subsystems.

The ACQuire command selects acquire subsystem as the destination for the commands that follow.

The ACQuire query responds with the settings of the acquire subsystem. Store mode and completion criteria for all active sources are set to be the same as those in effect for channel 1.

Command Syntax: ACQuire

Example: OUTPUT 707;"ACQUIRE"

Query Syntax: ACQuire ?

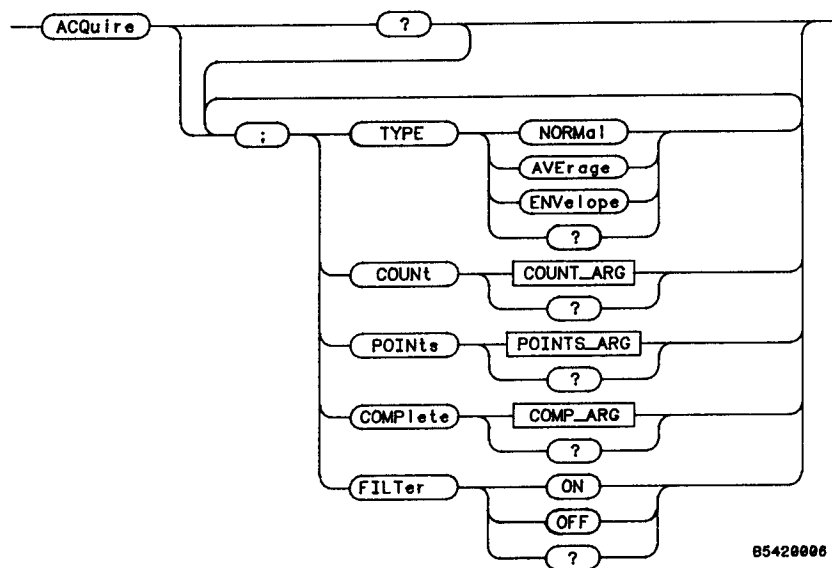
```
Returned Format: [ ACQuire <crLf> ]
                  [ TYPE ]<argument><crLf>
                  [ POINTs ]<NR1><crLf>
                  [ COUNT ]<NR1><crLf>
                  [ COMPlete ]<NR1><crLf>
                  [ FILTer ]<argument><crLf>
```

```
Example: DIM Acquire$[70]
          OUTPUT 707;"ACQUIRE?"
          ENTER 707 USING "-K";Acquire$
          PRINT USING "K";Acquire$
```

ACQUIRE SUBSYSTEM

ACquire Commands:

COMPlete
COUNT
FILTer
POINTs
TYPE



ACquire Subsystem Syntax Diagram

ACQUIRE SUBSYSTEM

COMPlete

Command does nothing, must be a valid integer. Query returns the completion criteria, 100%.

Command Syntax: COMPlete <NR1>

Example: OUTPUT 707;"COMPLETE 100"

Query Syntax: COMPlete ?

Returned Format: [COMPlete] <NR1><crLf>

Example: OUTPUT 707;"COMPLETE?"
ENTER 707;Complete\$
PRINT Complete\$

COUNT

If store mode is type average, command signifies both the terminal count and the number of averages (sample weighting factor). For the other store modes, command determines the terminal count, that is, the number of frames which were acquired. For normal store mode the count parameter must be 1. For average mode, the COUNT must be exactly 4, 16, 64, or 256. For envelope, COUNT parameter can be 10 to 10,000.

All active sources are set the same.

Query returns the count factor as described above. Channel 2 and trigger type and count are set the same as channel 1 so that the response is consistent for all sources.

Command Syntax: COUNT <NR1>

Example: OUTPUT 707;"COUNT 1"

Query Syntax: COUNT ?

Returned Format: [COUNT]<NR1><crLf>

Example: OUTPUT 707;"COUNT?"
ENTER 707;Count\$
PRINT Count\$

ACQUIRE SUBSYSTEM

FILTer

Command sets the data filter mode. The data filter is a 3-point smoothing, low-pass function that is run on the waveform data to generate 7-bit data when the time range is 5 μ s or longer. On faster time ranges, where a data interpolator is used, 7-bit data is produced without a data filter.

The FILTer query returns the data filter mode.

A slight speed enhancement is obtained by turning the filter off when acquiring averaged waveforms. The only benefit of the filter on this type of waveform is the low-pass characteristic of the filter, since averaging produces greater than 7-bit resolution.

The filter should be turned off when the absolute magnitude of individual data points is important to the measurement, such as when performing a V_{max} measurement and there are fewer than three samples on the waveform peak.

The filter should also be turned off when measuring a deliberately aliased signal, such as in swept frequency response measurements.

Command Syntax: FILTer { [ON | 1]
[OFF | 0] }

Example: OUTPUT 707;"FILTER ON"

Query Syntax: FILTer ?

Returned Format: [FILTer]<argument><crlf>

Example: OUTPUT 707;"FILTER?"
ENTER 707;Filter\$
PRINT Filter\$

ACQUIRE SUBSYSTEM

POINTS

Command does nothing, but must be a valid integer. Query returns the number of data points acquired, 1001.

Command Syntax: POINTs <NR1>

Example: OUTPUT 707;"POINTS 1001"

Query Syntax: POINTs ?

Returned Format: [POINTs] <NR1><crlf>

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

TYPE

Command sets the data type for all active sources and makes the count setting of channel 2 and trigger the same as channel 1.

Query responds with the data type of channel 1 and makes channel 2 and trigger the same type and count so that the response is consistent for all sources.

Command Syntax: TYPE {[NORMa1 | 1]
 [AVErage | 2]
 [ENVelope | 3]}

Example: OUTPUT 707;"ACQUIRE; TYPE NORMAL"

Query Syntax: TYPE ?

Returned Format: [TYPE]<argument><crlf>

Example: OUTPUT 707;"TYPE?"
 ENTER 707;Type\$
 PRINT Type\$

ACQUIRE SUBSYSTEM

Notes

CHANNEL SUBSYSTEM

CHANnel

The channel subsystem controls all "Y axis" functions, including selecting the type of data to be acquired. Channel 1 and Channel 2 are independently programmable for all functions.

The CHANnel command selects channel 1 or channel 2 as the destination for the commands that follow.

The CHANnel query responds with all of the settings for the specified channel.

Command Syntax: CHANnel { 1 | 2 }

Example: OUTPUT 707;"CHANNEL 1"

Query Syntax: CHANnel { 1 | 2 } ?

Returned Format: [CHANnel]<NR1><crLf>
 [PROBe]<NR3><crLf>
 [RANGe]<NR3><crLf>
 [OFFSet]<NR3><crLf>
 [COUPling]<argument><crLf>
 [LABe1]< 10 chars quoted ><crLf>
 [STORe]<argument>,<NR1><crLf>
 [SCALe]<argument><crLf>

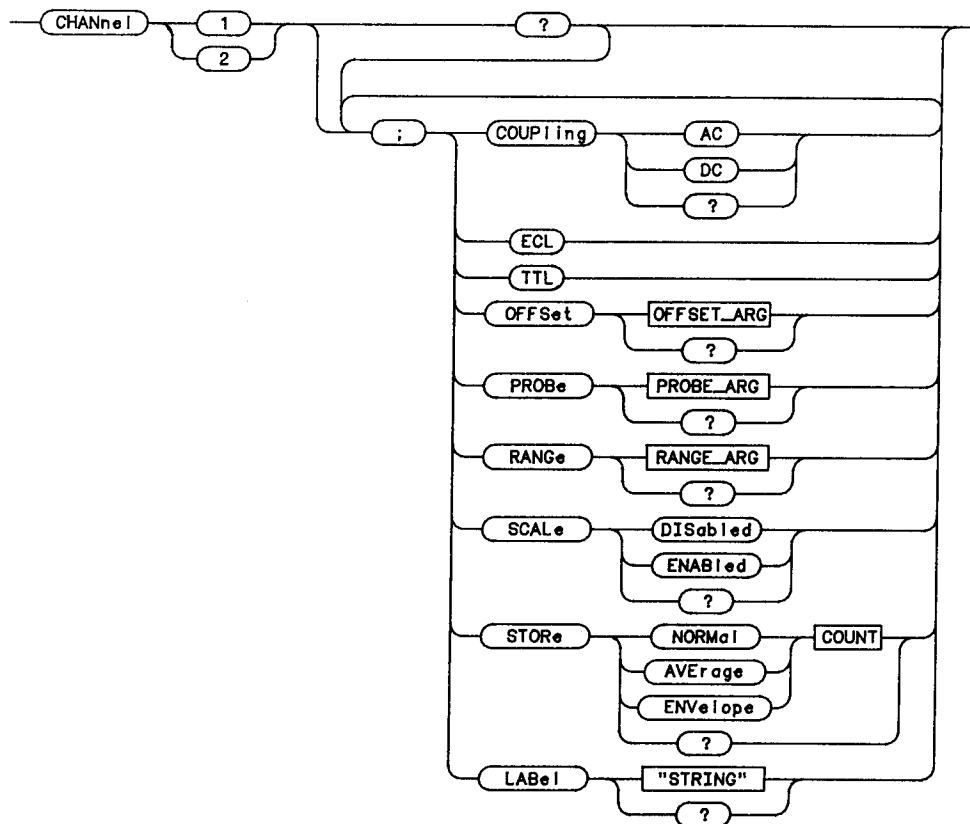
Example: DIM Chan\$[150]
 OUTPUT 707;"CHANNEL 1?"
 ENTER 707 USING "-K";Chan\$
 PRINT USING "K";Chan\$

CHANNEL SUBSYSTEM

CHANnel Commands:

COUPling
ECL
TTL
LABel
OFFSet
PROBe
RANGe
SCALE
STORe

Channel commands can be sent to either channel 1 or channel 2 and they are order dependent as follows: Range limits are determined by the current probe attenuation factor. The value limits for offset are determined by the current range selection. So to completely specify channel settings, program PROBe, then RANGe, and then OFFSet.



85420001

CHANnel Subsystem Syntax Diagram

CHANNEL SUBSYSTEM

COUPling

Command selects the input coupling for the selected channel. Query responds with the selected coupling.

Command Syntax: COUPling {[DC | 1]
[AC | 2]}

Example: OUTPUT 707;"COUPLING DC"

Query Syntax: COUPling ?

Returned Format: [COUPling]<argument><crlf>

Example: OUTPUT 707;"COUPLING?"
ENTER 707;Coupling\$
PRINT Coupling\$

ECL

Command presets the selected channel for ECL as follows:

```
COUPling = DC
OFFSet   = -1.00 V
RANGe    = 2.0 V
SCALE    = DISabled
```

The SCALE (autoscale) function is disabled so that a selective autoscale may be performed, thus preserving the channel settings.

Command Syntax: ECL

Example: OUTPUT 707;"ECL"

TTL

Command presets the selected channel for TTL as follows:

```
COUPling = DC
OFFSet   = 2.50 V
RANGe    = 6.0 V
SCALE    = DISabled
```

The SCALE (autoscale) function is disabled so that a selective autoscale may be performed, thus preserving the channel settings.

Command Syntax: TTL

Example: OUTPUT 707;"TTL"

CHANNEL SUBSYSTEM

LABel

Command sets the user label for selected channel. Query responds with the user label for selected channel.

Command Syntax: LABel <10 character string>

Example: OUTPUT 707;"LABEL ""TEST 1"""

Query Syntax: LABel ?

Returned Format:: [LABel]< 10 characters quoted ><crlf>

Example: OUTPUT 707;"LABEL?"
ENTER 707;Label\$
PRINT Label\$

OFFSet

Command sets channel offset. Offset is ± 2 V for 40 mV to 390 mV channel range and ± 20 V for 400 mV to 40 V channel range.

Query responds with the channel offset setting.

Command Syntax: OFFSet <NR3>

Example: OUTPUT 707;"OFFSET 400E-3"

Query Syntax: OFFSet ?

Returned Format: [OFFSet]<NR3><crlf>

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

CHANNEL SUBSYSTEM

PROBe

Command sets channel probe factor. Must be 1, 2, 5, 10, 20, 50 or 100. Query responds with channel probe factor.

Command Syntax: PROBe <NR1>

Example: OUTPUT 707;"PROBE 10"

Query Syntax: PROBe ?

Returned Format: [PROBe]<NR3><crlf>

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

RANGe

Command sets channel range. Query responds with current range setting. The allowable range settings (with 2-digit resolution) are based on the PROBe setting and are as follows:

| <u>PROBE</u> <u>FACTOR</u> | <u>VALID</u> <u>RANGE</u> <u>SETTING</u> |
|----------------------------|--|
| 1:1 | 40 mV to 40 V |
| 2:1 | 80 mV to 80 V |
| 5:1 | 200 mV to 200 V |
| 10:1 | 400 mV to 400 V |
| 20:1 | 800 mV to 800 V |
| 50:1 | 2 V to 2000 V |
| 100:1 | 4 V to 4000 V |

Command Syntax: RANGe <NR3>

Example: OUTPUT 707;"RANGE 5.5E0"

Query Syntax: RANGe ?

Returned Format: [RANGe]<NR3><crlf>

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

CHANNEL SUBSYSTEM

SCALE

Command sets autoscale mode for channel. Query responds with current autoscale mode.

Command Query: SCALE {[DISAbled | 0]
[ENABled | 1]}

Example: OUTPUT 707;"SCALE ENABLED"

Query Syntax: SCALE ?

Returned Format: [SCALE]<argument><crlf>

Example: OUTPUT 707;"SCALE?"
ENTER 707;Scale\$
PRINT Scale\$

STORE

Command sets the store mode. If average mode is selected, count determines both the terminal count and the number of averages (sample weighting factor). For the other store modes, count determines the terminal count, that is, the number of frames which must be acquired until the waveform is considered ready for analysis.

For normal store mode the count parameter is optional; if sent it must be 1. For average mode, the count must be exactly 4, 16, 64, or 256. For envelope, count can be 10 to 10,000.

The STORE function may also be programmed using the ACQUIRE subsystem TYPE and COUNT commands.

Query responds with the current store mode for the addressed channel.

Command Syntax: STORE < type >,< count >

< type > ::= {[NORMAl | 1]
AVERage | 2]
ENVELOpe | 3]}

< count > ::= <NR1>

Query Syntax: STORE ?

Returned Format: [STORE]<argument>,<NR1><crlf>

Example: OUTPUT 707;"STORE?"
ENTER 707;Store\$
PRINT Store\$

DISPLAY SUBSYSTEM

DISPlay

The display subsystem provides access to 54200A/D functions which control how waveforms are presented on the CRT. It contains commands to select whether the cursor values display is on. Also, a group of commands is provided that allow users to write text messages to screen.

The text display is available when the text menu is on and allows up to 4 lines of text to be displayed in the user definable text area. Five lines of text can be held in memory, the fifth line is not displayed unless the display is rolled up or down.

Text is volatile, it is set to " " at power on. However, once written, the user may display it with "TEXT ON" or display the previous menu by sending "TEXT OFF". The system command "MENU 70" is equivalent to "TEXT ON".

The DISPlay command selects display subsystem as the destination for the commands that follow.

The DISPlay query responds with the settings of the display subsystem.

Command Syntax: DISPlay

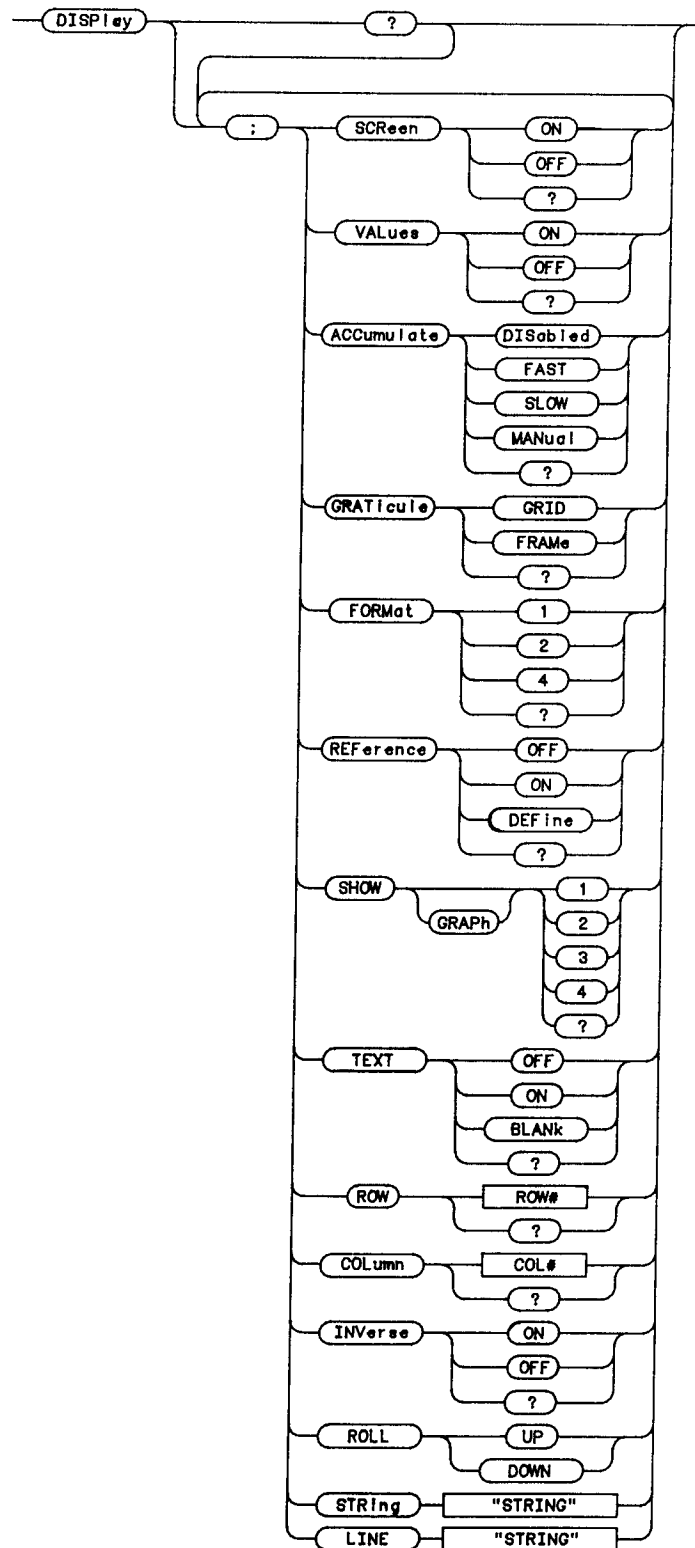
Example: OUTPUT 707;"DISPLAY"

Query Syntax: DISPlay ?

```
Returned Format: [ DISPlay ]<crLf>
                  [ FORMat ]<NR1><crLf>
                  [ GRATicule ]<argument><crLf>
                  [ ROW ]<NR1><crLf>
                  [ COLumn ]<NR1><crLf>
                  [ SCReen ]<argument><crLf>
                  [ TEXT ]<argument><crLf>
                  [ INVerse ]<argument><crLf>
                  [ REFeRence ]<argument><crLf>
                  [ ACCumulate ]<argument><crLf>
                  [ VALues ]<argument><crLf>
                  [ SHOW ] GRAPh<NR1><crLf>
```

```
Example: DIM Display$[160]
          OUTPUT 707;"DISPLAY?"
          ENTER 707 USING "-K";Display$
          PRINT USING "K";Display$
```

DISPLAY SUBSYSTEM



B5420005

DISPlay Subsystem Syntax Diagram

DISPLAY SUBSYSTEM

DISPlay Commands:

ACCumulate
 COL
 ROW
 FORMat
 GRATicule
 INVerse
 LINE
 ROLL
 REFerence
 SCReen
 SHOW
 STRing
 TEXT
 VALues

ACCumulate

Command sets the accumulate mode. If enabled, waveform data is displayed accumulated as it is acquired. If fast is selected, data is erased every 16 frames. Slow erases every 64 frames. Manual doesn't erase (use system ERASe command).

Query returns the current accumulate mode selection.

Command Syntax: ACCumulate {[DISabled | 0]
 [FAST | 1]
 [SLOW | 2]
 [MANual | 3]}

Example: OUTPUT 707;"ACCUMULATE DISABLED"

Query Syntax: ACCumulate ?

Returned Format: [ACCumulate]<argument><CrLf>

Example: OUTPUT 707;"ACCUMULATE?"
 ENTER 707;Accum\$
 PRINT Accum\$

Example: OUTPUT 707;"COLUMN 42"

Returned Format: [COlumn]<NR1><crlf>

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

Command sets the specified row as the target for LINE or STRing command. This does not affect the current COLUMN command specification. Query returns the current row number.

Command Syntax: ROW { 1 | 2 | 3 | 4 }

Example: OUTPUT 707;"ROW 3"

Returned Format: [ROW]<NR1><crlf>

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

DISPLAY SUBSYSTEM

FORMat

Command sets the number of graphs displayed; number must be 1, 2, or 4. Query returns the number of graphs displayed.

Command Syntax: FORMat { 1 | 2 | 4 }

Example: OUTPUT 707;"FORMAT 2"

Query Syntax: FORMat ?

Returned Format: [FORMat]<NR1><crLf>

Example: OUTPUT 707;"FORMAT?"
 ENTER 707;Format\$
 PRINT Format\$

GRATicule

Command sets the graticule type on the display graph. Query returns the graticule type displayed.

Command Syntax: GRATicule { [GRID | 1]
 [FRAME | 2] }

Example: OUTPUT 707;"GRATICULE GRID"

Query Syntax: GRATicule ?

Returned Format: [GRATicule]<argument><crLf>

Example: OUTPUT 707;"GRATICULE?"
 ENTER 707;Grat\$
 PRINT Grat\$

INVerse

Command sets inverse video display on or off. After INVerse ON is received, all subsequent text is displayed in inverse until INVerse OFF is received.

Command Syntax: INVerse { [OFF | 0]
 [ON | 1] }

Example: OUTPUT 707;"INVERSE OFF"

DISPLAY SUBSYSTEM

LINE

Command displays the string starting at the current row and column specification. If the column limit (63) is reached, the text is wrapped to the next line irrespective of word boundaries. At the end of the string, the remainder of the current row is set to " " and row is initialized to the next row and column is set to 0. If column limit is reached on row 4, the display is rolled up and the new line 4 is set to " " before the write operation continues. Refer to the STRing command for similar application.

Command Syntax: LINE < any quoted string >

Example: OUTPUT 707;"LINE ""Enter text on this line"""

REFeRence

Command sets the reference lines display mode. In define mode, when display menu is on, the graph levels are displayed by the graphs. Query returns the reference lines mode.

Command Syntax: REFeRence { [OFF | 0]
[ON | 1]
[DEFIne | 2] }

Example: OUTPUT 707;"REFERENCE OFF"

Query Syntax: REFeRence ?

Returned Format: [REFeRence]<argument><crlf>

Example: OUTPUT 707;"REFERENCE?"
ENTER 707;Ref\$
PRINT Ref\$

ROLL

Causes the text lines to roll up or down. Current row and column text pointers are unaffected.

Command Syntax: ROLL { [UP | 1]
[DOWN | 0] }

Example: OUTPUT 707;"ROLL DOWN"

DISPLAY SUBSYSTEM

SCReen

Command controls the 54200A/D display. ON turns normal display on, which has no effect if the display is already on; the last 54200A/D menu is restored. OFF turns everything except line 1 and 2 of the display off, resulting in faster operation when running since in this mode waveform data is not written to graph displays. Accumulated displays are not generated with screen off. This command cancels system command DEBug display.

Query returns the current screen status.

Command Syntax: SCReen { [ON | 1]
[OFF | 0] }

Example: OUTPUT 707;"SCREEN ON"

Query Syntax: SCReen ?

Returned Format: [SCReen]<argument><crlf>

Example: OUTPUT 707;"SCREEN?"
ENTER 707;Screen\$
PRINT Screen\$

SHOW

Command selects which graphs scale factors are shown in the scale line; value must be 1, 2, 3, or 4. If higher than maximum graph currently displayed then value is rounded to that graph number with no error.

Query returns graph number shown on current scale-factor line of display.

Command Syntax: SHOW [GRAPh] { 1 | 2 | 3 | 4 }

Example: OUTPUT 707;"SHOW GRAPH 4"

Query Syntax: SHOW ?

Returned Format: [SHOW] GRAPh <NR1><crlf>

Example: OUTPUT 707;"SHOW?"
ENTER 707;Show\$
PRINT Show\$

DISPLAY SUBSYSTEM

STRing

Command displays the string starting at the current row and column specification. If the column limit (63) is reached, the text is wrapped to the next line irrespective of word boundaries. At the end of the string, the row and column is set to the next available number. If column limit is reached on row 4, the display is rolled up and the new line 4 is set to " " before the write operation continues. Refer to the LINE command for similar application.

Command Syntax: STRing < any quoted string >

Example: OUTPUT 707;"STRING ""Press INSERT to continue"""

TEXT

TEXT OFF command sets menu previously displayed before text was turned on (no effect if text isn't on). TEXT ON saves current menu and puts on text menu. TEXT BLANK clears text memory, sets COLUMN=1 and ROW=1, but doesn't affect what menu is displayed.

Query returns text on/off status.

Command Syntax: TEXT {[OFF | 0]
[ON | 1]
[BLANK | 2]}

Example: OUTPUT 707;"TEXT ON"

Query Syntax: TEXT ?

Returned Format: [TEXT]<argument><crlf>

Example: OUTPUT 707;"TEXT ON"

DISPLAY SUBSYSTEM

VALues

Command controls the display of the cursor values readout. If off, the labels for the graphs are displayed.

Query returns the values mode.

Command Syntax: VALues { [OFF | 0]
[ON | 1] }

Example: OUTPUT 707;"VALUES ON"

Query Syntax: VALues ?

Returned Format: [VALues]<argument><crlf>

Example: OUTPUT 707;"VALUES?"
ENTER 707;Values\$
PRINT Values\$

DISPLAY SUBSYSTEM

Notes

GRAPH SUBSYSTEM

GRAPh

The graph subsystem allows the user to define the graph source and to define graph specific measurement criteria when measurement mode is user defined. X-axis windowing can also be done using the graph subsystem.

The GRAPh command selects the graph that is the destination for the graph commands that follow.

The GRAPh query responds with all of the settings for the specified graph.

Command Syntax: GRAPh { 1 | 2 | 3 | 4 }

Example: OUTPUT 707;"GRAPh 1"

Query Syntax: GRAPh { 1 | 2 | 3 | 4 } ?

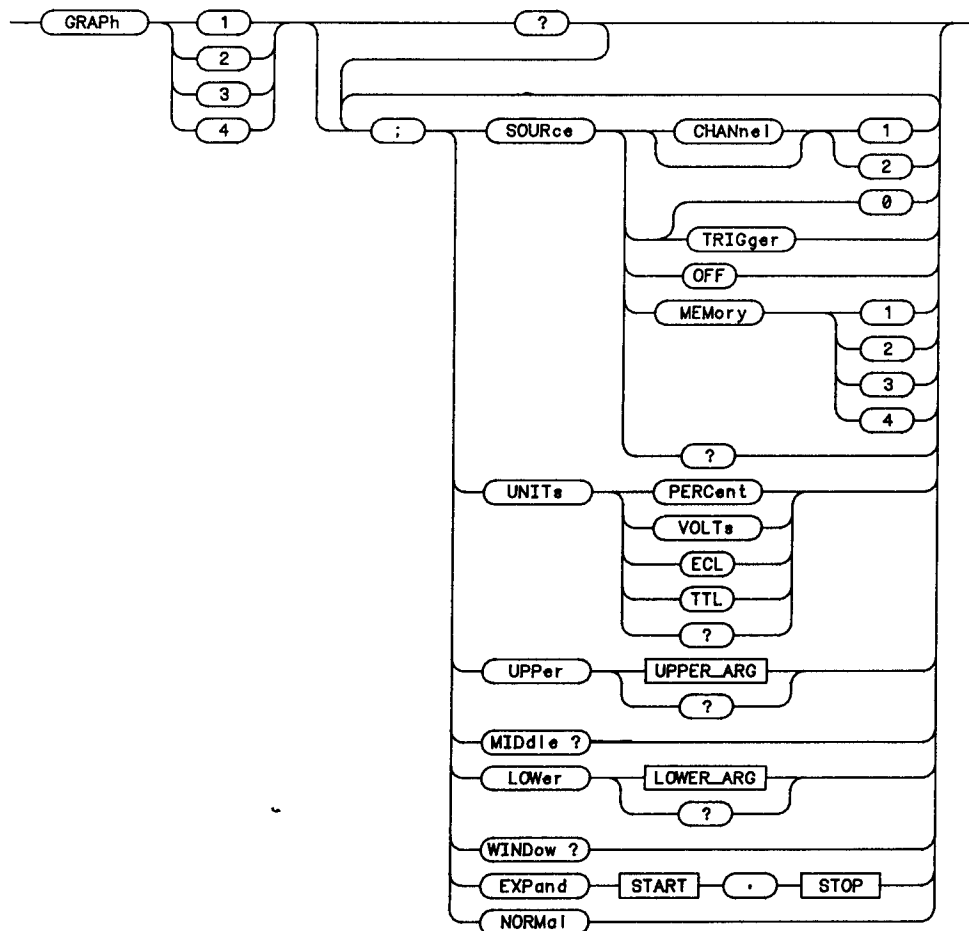
Returned Format: [GRAPh<NR1><crlf>]
 [SOURce]<source spec><crlf>
 [UNITs]<argument><crlf>
 [UPPer]{<NR1>|<NR3>}<crlf>
 [LOWer]{<NR1>|<NR3>}<crlf>

Example: DIM Graph\$[70]
 OUTPUT 707;"GRAPh 2?"
 ENTER 707 USING "-K";Graph\$
 PRINT USING "K";Graph\$

GRAPH SUBSYSTEM

GRAPH Commands:

SOURce
UNITs
UPPer
LOWer
MIDdle
WINDow
EXPand
NORMal



85420007

GRAPH Subsystem Syntax Diagram

GRAPH SUBSYSTEM

SOURce

Command selects the data source for the currently addressed graph. Query returns the data source displayed on the addressed graph.

Command Syntax: SOURce {[TRIGger | 0]
 [CHANnel1 | 1]
 [CHANnel2 | 2]
 [OFF |]
 [MEMory {0|1|2|3}]}

Example: OUTPUT 707;"SOURCE CHANNEL1"

Query Syntax: SOURce ?

Returned Format : [SOURce]{[CHANnel1]
 [CHANnel2]
 [TRIGger]
 [OFF]
 [MEMory0]
 [MEMory1]
 [MEMory2]
 [MEMory3]}

Example: OUTPUT 707;"SOURCE?"
 ENTER 707;Source\$
 PRINT Source\$

GRAPH SUBSYSTEM

UNITs

Command sets the measurement threshold units when user defined measurement mode is selected. Each units selection has its own settings for upper, middle, and lower.

Query returns the units selection in the user-defined mode for the addressed graph.

Note

If the 54200A/D is currently in standard measurement mode, the query will still return the user-defined UNITs selection.

Command Syntax: UNITs {[PERCent | 0]
[VOLTs | 1]
[TTL | 2]
[ECL | 3]}

Example: OUTPUT 707;"UNITs PERCENT"

Query Syntax: UNITs ?

Returned Format: [UNITs]<argument><crlf>

Example: OUTPUT 707;"UNITs?"
ENTER 707;Units\$
PRINT Units\$

UPPer

Command sets upper measurement threshold if UNITs has been set to VOLTs or PERCent. The upper limit is preset for ECL and TTL. If UNITs has been set to PERCent, the allowable upper limit is from 55% to 99%.

Query returns the value of the upper measurement threshold.

Command Syntax: UPPer <upper limit>

Example: OUTPUT 707;"UPPER 85" (assuming UNITs has been set to PERCent)

Query Syntax: UPPer ?

Returned Format: [UPPer] {[<NR1>] (if percent)
[<NR3>] (if volts)
[<NR2>]} <crlf> (if TTL or ECL)

Example: OUTPUT 707;"UPPER?"
ENTER 707;Upper\$
PRINT Upper\$

```
Example:  OUTPUT 707;"MIDDLE?"
          ENTER 707;Middle$
          PRINT Middle$
```

```
Example:  OUTPUT 707;"LOWER?"
          ENTER 707;Lower$
          PRINT Lower$
```

GRAPH SUBSYSTEM

WINDow

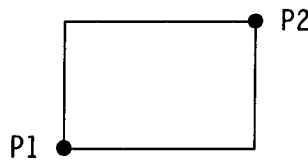
Query which returns a pair of (X,Y) coordinates describing the time and voltage limits of the graph display. Effectively these are P1 and P2 coordinates as used in plotter or HP Series 200 Controller window commands.

Query Syntax: WINDow ?

Returned Format: [WINDow]<NR3>,<NR3>,<NR3>,<NR3><crlf>

where the first NR3 pair is the P1 (time,volts) and
the second NR3 pair is the P2 (time,volts).

Example: DIM Window\$[50]
OUTPUT 707;"WINDOW?"
ENTER 707;Window\$
PRINT Window\$



EXPand

The graph expand function uses time rather than cursor location as the expansion criteria. The command sets the time at the starting point of the graph and at the stopping point of the graph. The graph must contain valid data or an error is generated.

The start time must be greater than or equal to the time of the first point of the waveform. The stop time must be less than or equal to the time of the last point. Minimum time (Stop - Start) must be ≥ 54 sample intervals, i.e., the minimum number of points displayed must be ≥ 55 .

The WINDow command is useful to determine the min and max values for start and stop times respectively.

Command Syntax: EXPand <start time>,<stop time>

Example: OUTPUT 707;"EXPAND 50E-9,125E-9"

NORMal

Command sets graph to show entire waveform, unexpanded.

Command Syntax: NORMal

Example: OUTPUT 707;"NORMAL"

MEASURE SUBSYSTEM

MEASure

The measure subsystem provides access to the front panel measurement functions and also provides some special functions particularly useful in systems applications.

The MEASure command selects the measure subsystem for the commands that follow.

The MEASure query responds with the settings of the measure subsystem.

Command Syntax: MEASure

Example: OUTPUT 707;"MEASURE"

Query Syntax: MEASure ?

```
Returned Format: [ MEASure <crLf> ]
                  [ SOURce ] <source spec><crLf>
                  [ MODE ]<argument><crLf>
                  [ SENDvalid ]<argument><crLf>
                  [ DEFine]{DElay      | 0 },<meas spec><crLf>
                  [ DEFine]{NWIDth     | 1 },<meas spec><crLf>
                  [ DEFine]{PWIDth     | 2 },<meas spec><crLf>
                  [ DEFine]{ENVELOpe   | 3 },<meas spec><crLf>
```

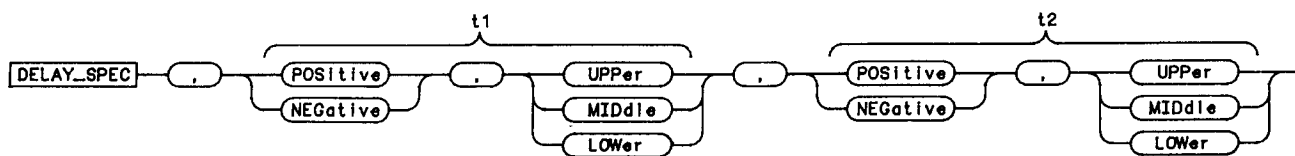
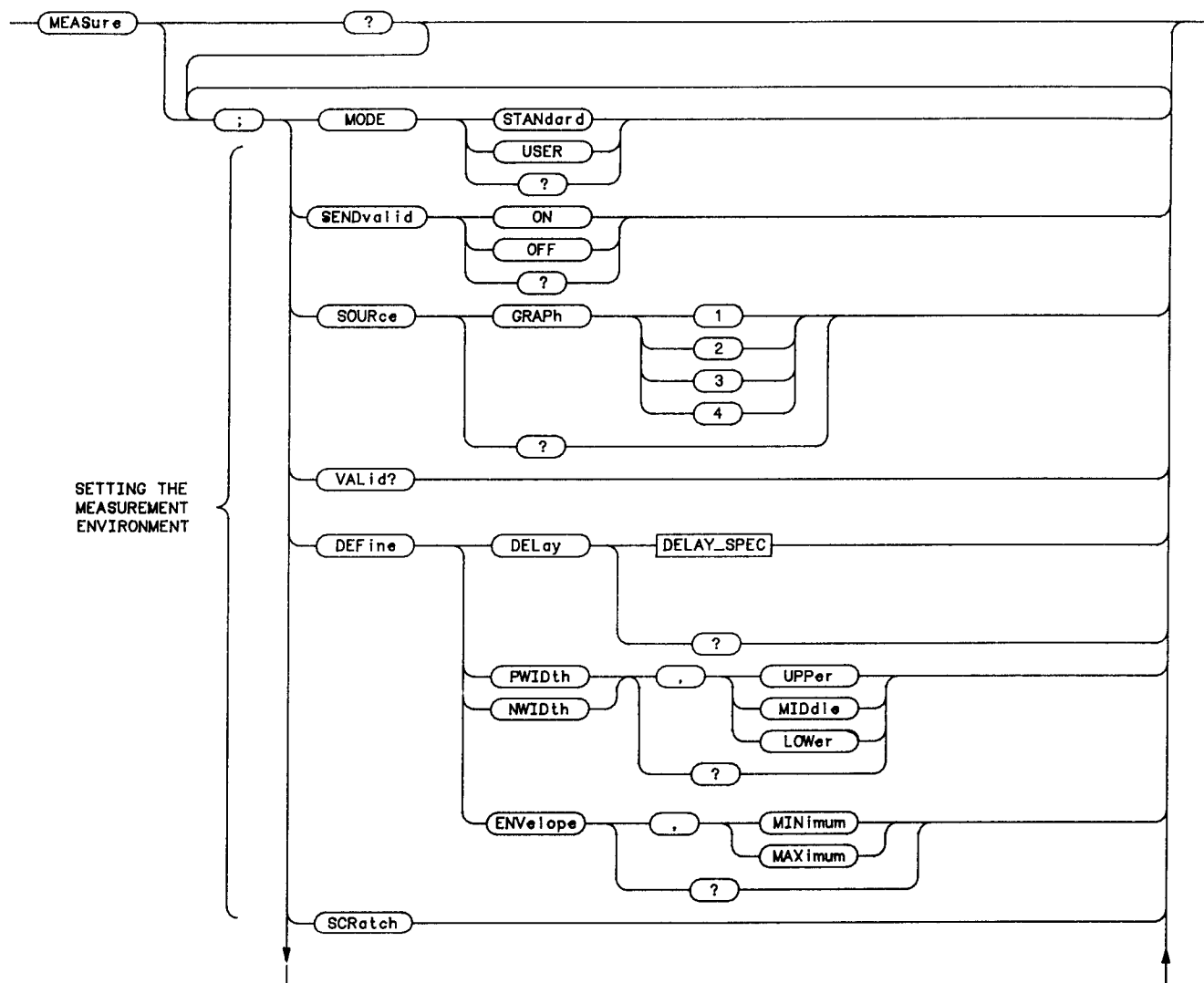
```
Example: DIM Measure$[180]
         OUTPUT 707;"MEASURE?"
         ENTER 707 USING "-K";Measure$
         PRINT USING "K";Measure$
```

MEASURE SUBSYSTEM

MEASure Commands:

| | | |
|--|---|--|
| MODE SENDvalid SOURce VALid DEFine SCRatch | } | Setting the Measurement Environment |
| RESULTs | | Getting Answers |
| FREQuency PERiod PWIDTH NWIDTH RISE FALL DUTycycle VAMP VPP VMAX VMIN VRMS VTOP VBASe OVERshoot PREShoot DELay | } | Defined Measurements |
| PTIME VTIME PVOLt TVOLt TPoint VPoint CURSor | } | Fundamental Measurements |

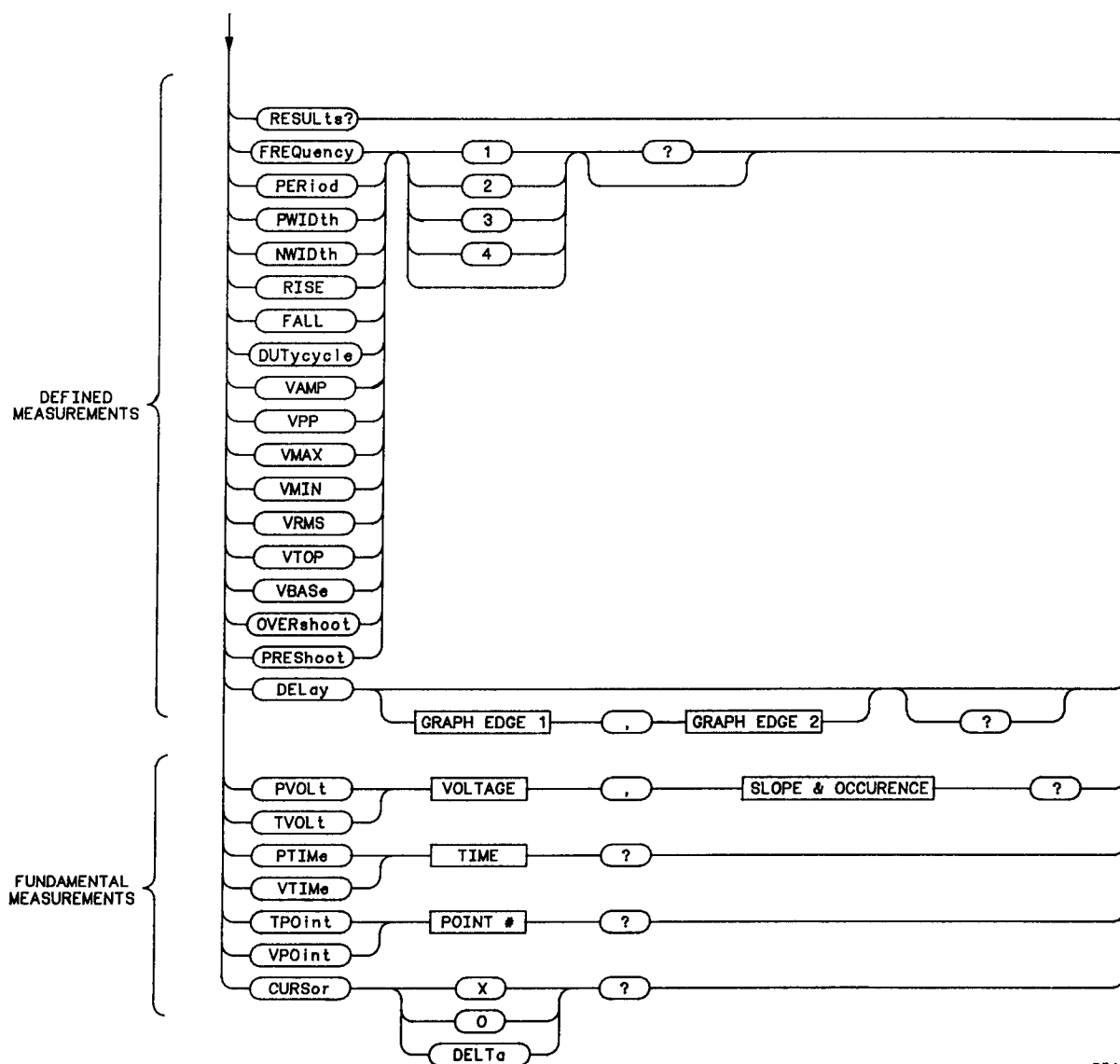
MEASURE SUBSYSTEM



85420009

MEASURE Subsystem Syntax Diagram

MEASURE SUBSYSTEM



85420000

MEASure Subsystem Syntax Diagram (Cont)

MEASURE SUBSYSTEM

SETTING THE MEASUREMENT ENVIRONMENT

54200A/D provides two basic modes for measurements, standard and user defined. In standard mode, reference levels are preset and measurement definitions are fixed. In user defined mode, reference levels may be individually selected for each graph and certain measurement definitions may be customized to better suit your application.

Graph reference thresholds are set in the graph subsystem.

All measurements are based upon the time voltage windows defined by the four graphs; measurements always reference graphs, not channels or memories directly. By expanding a graph, you can select a region of the waveform data to measure.

For graphs containing waveforms that are actively being acquired, the measurement occurs when the waveform is completely acquired, that is, the count parameter has been satisfied.

MODE

Command sets the measurement mode (definitions and thresholds). User mode definitions are set by the MEASure DEFINE command. USER mode thresholds are set by the GRAPH subsystem UNITS, UPPER, and LOWER commands.

Query returns the current measurement mode.

Command Syntax: MODE { [STANDARD | 0]
[USER | 1] }

Example: OUTPUT 707;"MODE STANDARD"

Query Syntax: MODE ?

Returned Format: [MODE]<argument><crlf>

Example: OUTPUT 707;"MODE?"
ENTER 707;Mode\$
PRINT Mode\$

MEASURE SUBSYSTEM

SENDvalid

If command SENDvalid is ON, an optional parameter is sent by the 54200A/D with measurement RESULTS queries that qualifies the result. This parameter indicates why an invalid result occurred or that the measurement result is good data. The optional validity parameter returned with the RESULTS query is:

```
<validity> ::= { [ GOOD_DATA      | 1 ]
                  [ NO_WAVEFORM   | 0 ]
                  [ SCALING?      | -1 ]
                  [ THRESHOLD?    | -2 ]
                  [ CLIPPED        | -3 ]
                  [ NOT_FOUND     | -6 ] }
```

The SENDvalid query returns the current SENDvalid selection, ON or OFF.

Command Syntax: SENDvalid { [ON | 1]
[OFF | 0] }

Example: OUTPUT 707;"SENDVALID ON"

Query Syntax: SENDvalid ?

Returned Format: [SENDvalid]<argument><crLf>

Example: OUTPUT 707;"SENDVALID?"
ENTER 707;Send\$
PRINT Send\$

SOURce

Command selects the graph to be addressed for the measurements that follow. This graph will be the default for measurement commands as well as the one used for the VALid query.

Query returns the currently addressed graph.

Command Syntax: SOURce [GRAPH]{ 1 | 2 | 3 | 4 }

Example: OUTPUT 707;"SOURCE GRAPH 1"

Query Syntax: SOURce ?

Returned Format:: [SOURce] GRAPH<NR1><crLf>

Example: OUTPUT 707;"SOURCE?"
ENTER 707;Source\$
PRINT Source\$

MEASURE SUBSYSTEM

VALid

Query which returns the valid status for the currently addressed graph. The VALid arguments are the same as defined by SENDvalid.

Query Syntax: VALid ?

```
Returned Format: [VALid] {[ GOOD_DATA      | 1 ]
                        [ NO_WAVEFORM      | 0 ]
                        [ SCALING?         | -1 ]
                        [ THRESHOLD?       | -2 ]
                        [ CLIPPED          | -3 ]
                        [ NOT_FOUND        | -6 ]} <crlf>
```

```
Example:  OUTPUT 707;"VALID?"
          ENTER 707;Valid$
          PRINT Valid$
```

DEFine

When the measurement MODE command has been set to USER mode, definitions for customizing measurements can be set using the DEFine command.

The DEFine query returns the currently defined user definitions.

```
Command Syntax:  DEFine {[ DELay      | 0 ]
                        [ PWIDTH      | 1 ]
                        [ NWIDTH      | 2 ]
                        [ ENVELOPE    | 3 ]} ,<measurement specification>
```

The PWIDTH measurement is made from the first rising threshold defined by the measurement specification to the next falling threshold. The NWIDTH measurement is made from the first falling threshold defined by the measurement specification to the next rising threshold. For both width measurements,

```
<measurement specification> ::= {[ MIDDLE | 0 ]
                                   [ UPPER  | 1 ]
                                   [ LOWER  | 2 ]}
```

```
Example:  OUTPUT 707;"DEFINE PWIDTH,MIDDLE"
```

MEASURE SUBSYSTEM

DEFine (cont)

For DELay, both edge polarity and threshold level is definable,

```
<measurement specification> ::= <polarity>,<level>,<polarity>,<level>
```

```
<polarity> ::= {[ POSitive | 0 ]
                 [ NEGative | 1 ]}
```

```
<level> ::= {[ MIDdle | 0 ]
               [ UPPer | 1 ]
               [ LOWer | 2 ]}
```

Example: OUTPUT 707;"DEFINE DELAY,POSITIVE,UPPER,POSITIVE,UPPER"

DEFine ENVELOpe allows the selection of the minimum waveform or the maximum waveform for the basic measurement commands VTIME, VPOINT, TVOLT, and PVOLT. For ENVELOpe,

```
<measurement specification> ::= {[ MINimum | 0 ]
                                   [ MAXimum | 1 ]}
```

Example: OUTPUT 707;"DEFINE ENVELOPE,MAXIMUM"

Query Syntax: DEFine {[DELay]
 [PWIDth]
 [NWIDth]
 [ENVELOpe]} ?

Returned Format: [DEFine] {[DELay | 0]
 [NWIDth | 1]
 [PWIDth | 2]
 [ENVELOpe | 3]} ,<meas spec><crlf>

Example: DIM Define\$[60]
 OUTPUT 707;"DEFINE DELAY?"
 ENTER 707;Define\$
 PRINT Define\$

SCRatch

Command clears the voltage and time measurement lists from the display.

Command Syntax: SCRatch

Example: OUTPUT 707;"SCRATCH"

MEASURE SUBSYSTEM

EXECUTING MEASUREMENTS

The two modes of execution of measurements in the 54200A/D are immediate and queued.

- Immediate mode is performed when a measurement is specified as a query. The measurement is performed as the command is parsed and the results are entered into the output buffer.
- In queued mode, the measurement is added to the display list. The queued measurements are performed as a part of the normal acquisition cycle and results are displayed on screen. The results may be read over HP-IB by using the RESULTS query.

In addition to the defined measurement functions on the front panel, two other groups of fundamental measurements exist. The cursor values may be read over HP-IB and a set of general purpose time, voltage, and point measurements are provided.

Note

The measurement will not be completed until the the parameter set up by the COUNT command in the ACQUIRE subsystem or the STORE command in the CHANNEL or TRIGGER subsystem has been satisfied. If the store mode is average, 4 to 256 acquisitions must be made before the measurement is complete (default is 4). If the store mode is envelope, 10 to 10000 acquisitions must be made before the measurement is complete (default is 100).

RESULTS

RESULTS is a query which causes the 54200A/D to output the list of active measurements.

Query Syntax: RESULTS ?

Returned Format: [RESULTS]<# of measurements><crLf>
 [<Time measurement list>]
 [<Volt measurement list>]

<# of measurements> ::= integer from 0 to 6

<Time measurement list> and <Volt measurement list> ::=

[<measurement name>]<value>[,<validity>]<crLf>
 [<measurement name>]<value>[,<validity>]<crLf>
 [<measurement name>]<value>[,<validity>]<crLf>

MEASURE SUBSYSTEM

RESULTS (cont)

Number of answers returned in Time measurement list can be from 0 to 3.

Number of answers returned in Volt measurement list can be from 0 to 3.

Validity parameter only returned if SENDvalid command is set ON.

```
<validity> ::= {[ GOOD_DATA      | 1 ]
                 [ NO_WAVEFORM   | 0 ]
                 [ SCALING?      | -1 ]
                 [ THRESHOLD?    | -2 ]
                 [ CLIPPED       | -3 ]
                 [ NOT_FOUND     | -6 ] }
```

```
Example: DIM Results$[200]
         OUTPUT 707;"RESULTS?"
         ENTER 707 USING "-K";Results$
         PRINT Results$
```

Defined Measurements

The following measurements are defined for 54200A/D:

```
<measurement name> ::= FREquency
                        PERiod
                        PWIDth
                        NWIDth
                        RISE
                        FALL
                        DUTycycle
                        VAMP
                        VPP
                        VMAX
                        VMIN
                        VRMS
                        VTOP
                        VBASe
                        OVERshoot
                        PREShoot
                        DELay
```

MEASURE SUBSYSTEM

Defined Measurements (cont)

With the exception of DELay, all follow the syntax:

Command/Query Syntax: <measurement name>[<graph number>][?]

DELay is specified with:

Command/Query Syntax: DELay [<graph of first edge>,<graph of second edge>][?]

If no graph is specified, the default graph is measured.

If the command form is specified, the measurement is added to the appropriate measurement list (time or volt) on the 54200A/D display and the answers are returned to the controller using the RESULTS query. If more than 3 measurements are specified for either the time or volt list, only the last 3 are retained (performed) and any others are discarded.

Example: OUTPUT 707;"FREQUENCY 1" or
OUTPUT 707;"DELAY 1,2"

If the query form is specified, the measurement is performed immediately and not displayed on the 54200A/D. The results are placed in the output buffer of the 54200A/D with the following format for non-delay measurements:

Returned Format: [<meas name><graph number>,<graph number>,<value>[,<validity>]<crLf>

Example: DIM Rise[30]
OUTPUT 707;"RISE?"
ENTER 707;Rise\$
PRINT Rise\$

and with the following format for delay measurements:

Returned Format:

[<meas name><graph number>,<graph number>,<value>[,<validity>]<crLf>

Example: DIM Delay\$[30]
OUTPUT 707;"DELAY 1,2 ?"
ENTER 707;Delay\$
PRINT Delay\$

Note

When any of these defined measurements are performed on envelope type waveforms, the minimum wave data is always used.

MEASURE SUBSYSTEM

Fundamental Measurements

The following measurements are basic voltage, time, and point queries that can be used to create custom measurements not handled by the defined measurements. All queries reference the current measurement source waveform.

PTIME
VTIME
PVOLT
TVOLT
TPoint
VPoint

PTIME

If point of time is requested, the nearest point preceding the requested time is returned.

Query Syntax: PTIME <time> ?

Returned Format: [PTIME]<NR1><crlf>

Example: DIM Ptime\$[25]
OUTPUT 707;"PTIME 500E-6 ?"
ENTER 707;Ptime\$
PRINT Ptime\$

VTIME

Query returns the interpolated voltage at the specified time.

Note

If the waveform is type envelope and user mode is selected for measurements, the user definition of min/max is used to select which waveform to measure.

Query Syntax: VTIME <time> ?

Returned Format: [VTIME]<NR3><crlf>

Example: DIM Vtime\$[25]
OUTPUT 707;"VTIME 2E-3 ?"
ENTER 707;Vtime\$
PRINT Vtime\$

MEASURE SUBSYSTEM

PVOLT

When point of voltage is requested, the waveform is searched from the first displayed point until an interval is found that contains the requested voltage.

The sign of <slope & occurrence> selects rising (+) or falling (-) edge. The magnitude of this parameter selects the occurrence count. For example, if <slope & occurrence>= -2, the waveform is searched for the second occurrence of the specified voltage on a negative slope. The nearest point preceding this interval is returned.

Note

If the waveform is type envelope and user mode is selected for measurements, the user definition of min/max is used to select which waveform to measure.

Query Syntax: PVOLT <voltage>,<slope & occurrence > ?

Returned Format: [PVOLT]<NR1><crlf>

Example: DIM Pvolt\$[25]
 OUTPUT 707;"PVOLT 200E-3,-3 ?"
 ENTER 707;Pvolt\$
 PRINT Pvolt\$

TVOLT

When time of voltage is requested, the waveform is searched from the first displayed point until an interval is found that contains the requested voltage. Then a linear interpolation is done to estimate the time the voltage was traversed.

The sign of <slope & occurrence> selects rising (+) or falling (-) edge. The magnitude of this parameter selects the occurrence count. For example, if <slope & occurrence>= -2, the waveform is searched for the second occurrence of the specified voltage on a negative slope.

Note

If the waveform is type envelope and user mode is selected for measurements, the user definition of min/max is used to select which waveform to measure.

Query Syntax: TVOLT <voltage>,<slope & occurrence > ?

Returned Format: [TVOLT]<NR3><crlf>

Example: DIM Tvolt\$[25]
 OUTPUT 707;"TVOLT 5.5,+2 ?"
 ENTER 707;Tvolt\$
 PRINT Tvolt\$

MEASURE SUBSYSTEM

TPoint

Query returns the time of the specified point (1 to 1001).

Query Syntax: TPoint <point number> ?

Returned Format: [TPoint]<NR3><cr><lf>

Example: DIM Tpoint\$[25]
OUTPUT 707;"TPOINT 267 ?"
ENTER 707;Tpoint\$
PRINT Tpoint\$

VPoint

Query returns the voltage of the specified point (1 to 1001).

Note

If the waveform is type envelope and user mode is selected for measurements, the user definition of min/max is used to select which waveform to measure.

Query Syntax: VPoint <point number> ?

Returned Format: [VPoint]<NR3><cr><lf>

Example: DIM Vpoint\$[25]
OUTPUT 707;"VPOINT 426 ?"
ENTER 707;Vpoint\$
PRINT Vpoint\$

MEASURE SUBSYSTEM

Cursor Values

Although the cursors are not programmable via the HP-IB, the cursor values may be read.

CURSor

If a query with X or O argument is executed, the 54200A/D returns an x,y pair representing the time and voltage at the point of the specified cursor.

If a query with the DELTa argument is executed, the 54200A/D returns an x,y pair representing the Δ time and Δ voltage between the points of the X and the O cursors.

The optional validity parameter is only returned if the SENDvalid command has been set to ON.

Query Syntax: CURSor { [DELTa | 0]
 [X | 1]
 [O | 2] } ?

Returned Format: [CURSor]<argument>,<Time>,<Voltage>[,<validity>]<crlf>

<Time> and <Voltage> are number type NR3

<validity> ::= { [GOOD_DATA | 1]
 [NO_WAVEFORM | 0]
 [SCALING? | -1]
 [THRESHOLD? | -2]
 [CLIPPED | -3]
 [NOT_FOUND | -6] }

Example: DIM Cursor\$[25]
 OUTPUT 707;"CURSOR DELTA ?"
 ENTER 707;Delta\$
 PRINT Delta\$

MEASURE SUBSYSTEM

Notes

STATE SUBSYSTEM

STATE

The state subsystem is available only on the 54200D and provides a means to program the state trigger function. The two major groupings of the state trigger commands are assignment and sequence.

- Assignment consists of defining the mode of state operation, and specifying the probe thresholds, labels and patterns.
- Sequence is the actual specification of the events that are needed before a state trigger is generated.

The STATE command selects the state subsystem as the destination for the commands that follow. The STATE query responds with all of the settings for the state subsystem.

Command Syntax: STATE

Example: OUTPUT 707;"STATE"

Query Syntax: STATE ?

Returned Format: [STATE <crLf>]
 [ASSignment <crLf>]
 [MODE]<argument><crLf>
 [MULTiplex]<argument><crLf> (normal mode only)
 < clock specification >
 [POD0]<NR2><crLf>
 [POD1]<NR2><crLf>
 [POD2]<NR2><crLf> (normal and qualified modes only)
 {< label specification >}!1!8! (maximum is 1 for bit fault modes)
 [SEquence <crLf>]
 [FIND] <sequence find list><crLf>
 [COUNT]<NR1><crLf>
 [OF]<sequence trigger list><crLf>
 [TERMs]<argument><crLf>
 [REStart]<sequence restart list><crLf>

<clock specification> ::=

For normal mode, multiplexing off:

[CLOCK] <argument>,<argument>,<argument><crLf>

For normal mode, multiplexing on or for qualified mode:

[MASTer] <argument>,<argument>,<argument><crLf>
 [SLAVE] <argument>,<argument>,<argument><crLf>

For missing or extra bit modes:

[DATA] <argument><crLf>
 [REFerence] <argument><crLf>

STATE SUBSYSTEM

STATE (cont)

Note

The following label commands apply to all defined labels. When used individually they refer to the currently addressed label.

<label specification> ::=

```
[LAbel ] <5 chars quoted> <crlf>
[POLarity ] <argument> <crlf>
[ALLocate ] <NR1>, <NR1>, <NR1> <crlf>
[BASE ] <argument> <crlf>
[PATtern ] A, <pattern expression> <crlf>
[PATtern ] B, <pattern expression> <crlf>
[PATtern ] C, <pattern expression> <crlf>
[PATtern ] D, <pattern expression> <crlf> (normal and qualified modes only)
```

```
<pattern expression> ::= { [ #H<hexadecimal expression> ]
                             [ <decimal expression> ]
                             [ #O<octal expression> ]
                             [ #Y<binary expression> ] }
```

```
<hexadecimal expression> ::= { 0|1|2|3|4|5|6|7|8|9|a|b|
                                c|d|e|f|A|B|C|D|E|F|X|$ }!1!7!
```

```
<decimal expression> ::= { 0|1|2|3|4|5|6|7|8|9|X|$ }!1!9!
```

```
<octal expression> ::= { 0|1|2|3|4|5|6|7|8|X|$ }!1!9!
```

```
<binary expression> ::= { 0|1|X }!1!27!
```

Note

"\$" valid for query only. An "X" specifies a don't care state, while a "\$" specifies an undefined state. "Don't cares" in one value base cannot always be translated to another base.

Example: DIM State\$[500]
 OUTPUT 707;"STATE?"
 ENTER 707 USING "-K";State\$
 PRINT USING "K";State\$

STATE SUBSYSTEM

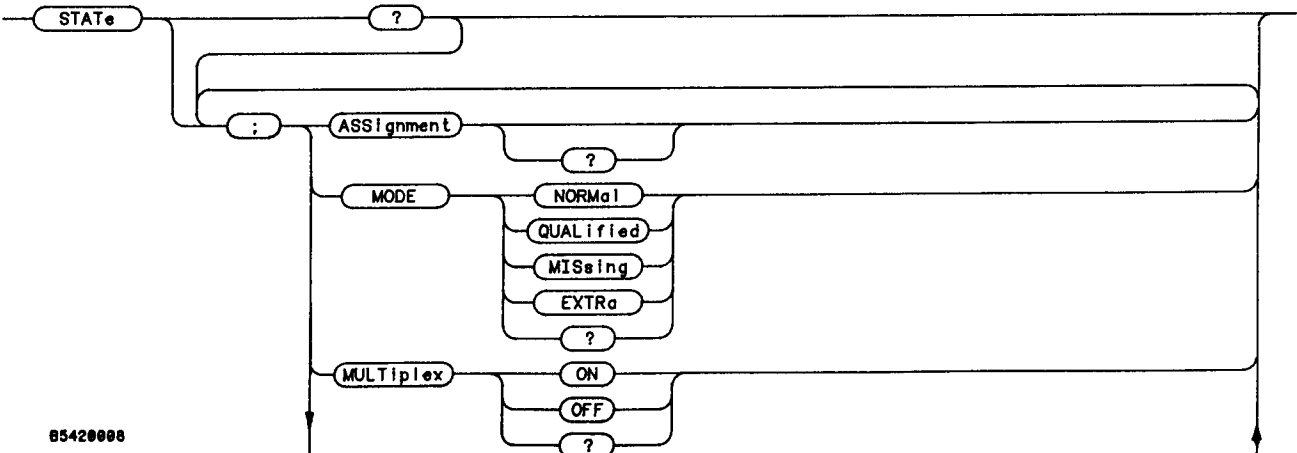
STATe Commands:

Assignment Commands

ASSignment
MODE
MULTiplex
CLOCK
MASTER
SLAVE
DATA
REFerence
POD0
POD1
POD2
LABel
INSert
DELeTe
POLarity
ALLocate
BASE
PATtern

Sequence Commands

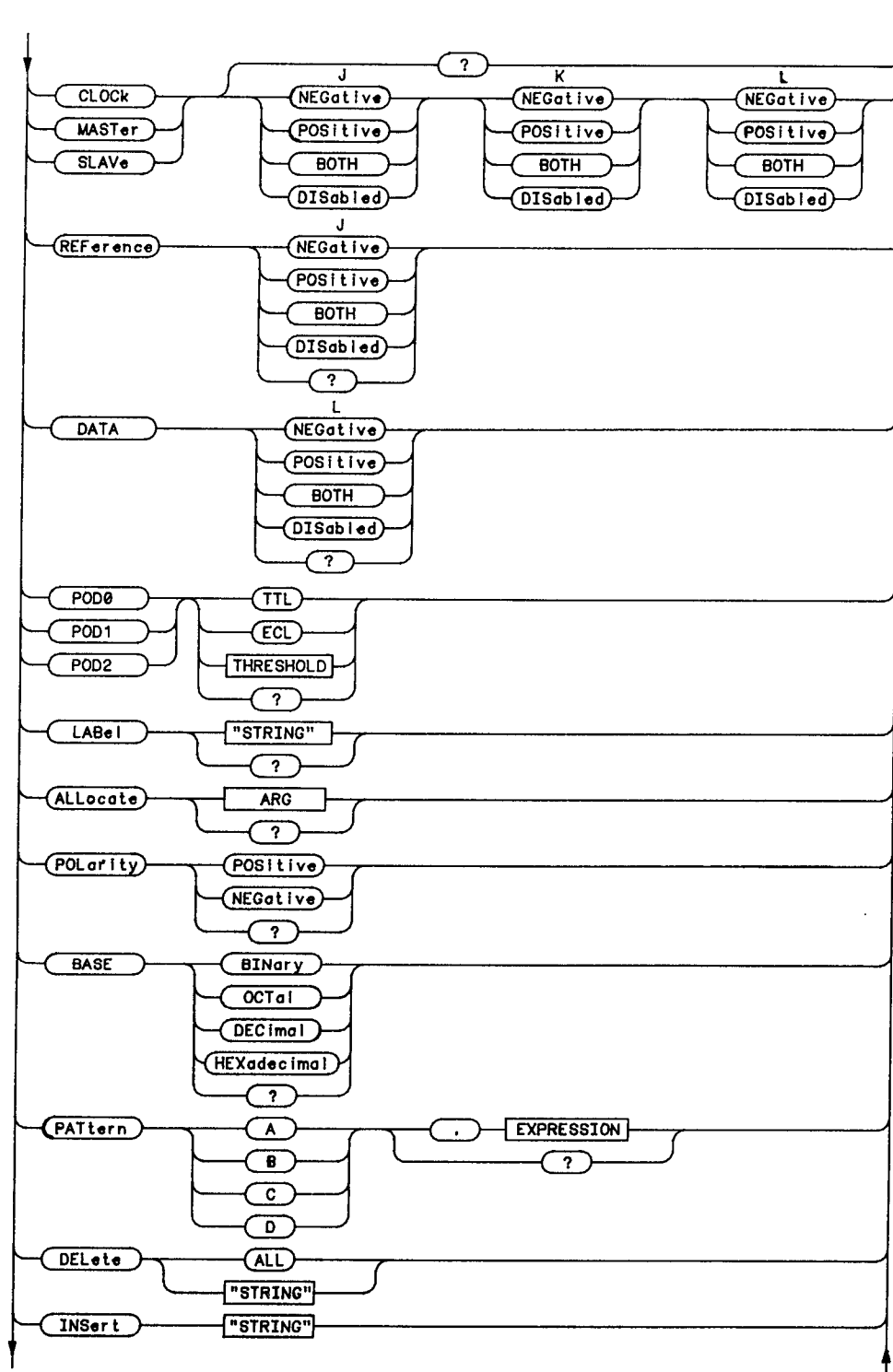
SEquence
FIND
COUNT
OF
TERMs
REStart



85420008

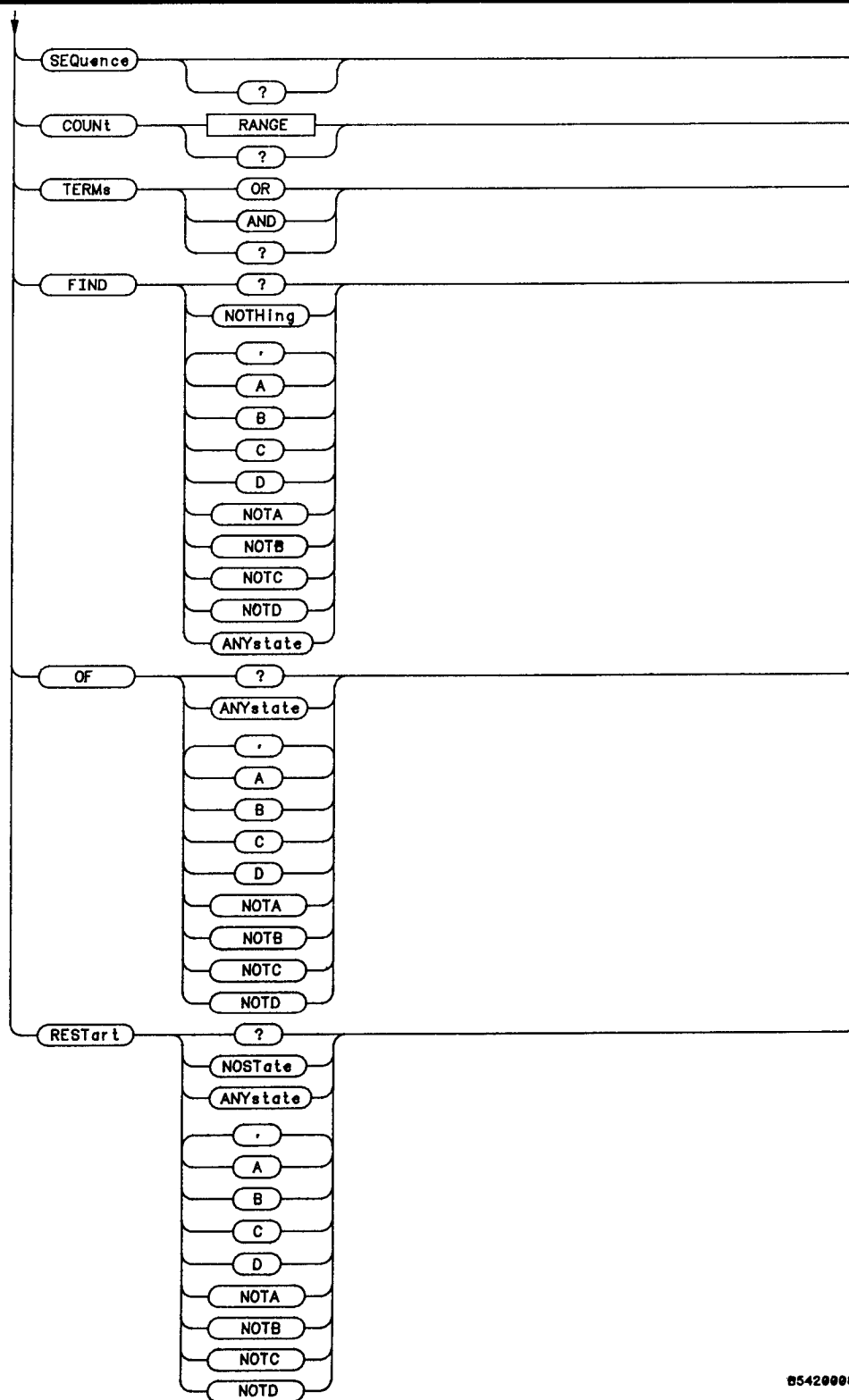
STATe Subsystem Syntax Diagram (Assignment)

STATE SUBSYSTEM



85420008

STATE Subsystem Syntax Diagram (Assignment cont)

STATE SUBSYSTEM

85420008

STATE Subsystem Syntax Diagram (SEQUENCE)

STATE SUBSYSTEM

Assignment Commands

ASSignment

Command enables state assignment queries to be returned intact.
Query returns the current assignment settings.

Command Syntax: ASSignment

Example: OUTPUT 707;"ASSIGNMENT"

Query Syntax: ASSignment ?

Returned Format: [STATE <crLf>]
 [ASSignment <crLf>]
 [MODE]<argument><crLf>
 [MULTiplex]<argument><crLf> (normal mode only)
 < clock specification >
 [POD0]<NR2><crLf>
 [POD1]<NR2><crLf>
 [POD2]<NR2><crLf> (normal and qualified modes only)
 {< label specification >}!1!8! (maximum is 1 for bit fault modes)

Example: DIM Assign\$[200]
 OUTPUT 707;"ASSIGNMENT?"
 ENTER 707 USING "-K";Assign\$
 PRINT USING "K";Assign\$

MODE

Command sets the state trigger mode. Query returns the current mode selection.

Command Syntax: MODE {[NORMa1 | 0]
 [QUALified | 1]
 [MISSing | 2]
 [EXTRA | 3]}

Example: OUTPUT 707;"MODE QUALIFIED"

Query Syntax: MODE ?

Returned Format: [MODE]<argument><crLf>

Example: OUTPUT 707;"MODE?"
 ENTER 707;Mode\$
 PRINT Mode\$

STATE SUBSYSTEM

MULTiplex

Command defines multiplexing of clocks when state is in normal mode; other modes have predefined multiplexing. Query returns the multiplexing mode in effect when in normal mode.

Command Syntax: MULTiplex {[ON | 1]
[OFF | 0]}

Example: OUTPUT 707;"MULTIPLEX ON"

Query Syntax: MULTiplex ?

Returned Format: [MULTiplex]<argument><crLf>

Example: OUTPUT 707;"MULTIPLEX?"
ENTER 707;Mult\$
PRINT Mult\$

CLOCK

Command sets the clock polarities for the master clock used in the normal mode with multiplexing off. Query returns the master clock settings used in normal with multiplexing off.

Command Syntax: CLOCK <J clock spec>,<K clock spec>,<L clock spec>

< any clock spec >::= {[NEGative | 0]
POSitive | 1]
BOTH | 2]
DISabled | 3]}

Example: OUTPUT 707;"CLOCK POSITIVE,NEGATIVE,POSITIVE"

Query Syntax: CLOCK

Returned Format: [CLOCK]<J clock spec>,<K clock spec>,<L clock spec><crLf>

Example: DIM Clock\$[50]
OUTPUT 707;"CLOCK?"
ENTER 707;Clock\$
PRINT Clock\$

STATE SUBSYSTEM

MASTer

Command sets the clock polarities for the master clock used in normal mode with multiplexing on or in qualified mode. Query returns the master clock settings used in normal mode with multiplexing off and qualified modes.

Command Syntax: MASTer <J clock spec>,<K clock spec>,<L clock spec>

| | | | | | | |
|-----------------------|---|---|----------|--|---|---|
| < any clock spec >::= | { | [| NEGative | | 0 |] |
| | | [| POSitive | | 1 |] |
| | | [| BOTH | | 2 |] |
| | | [| DISabled | | 3 |] |

Example: OUTPUT 707;"MASTER NEGATIVE,NEGATIVE,NEGATIVE"

Query Syntax: MASTer ?

Returned Format: [MASTer]<J clock spec>,<K clock spec>,<L clock spec><crlf>

Example: DIM Master\$[50]
 OUTPUT 707;"MASTER?"
 ENTER 707;Master\$
 PRINT Master\$

SLAVe

Command sets the clock polarities for the slave clock used in normal multiplex mode or qualified mode. Query returns the slave clock settings used in normal multiplex and qualified modes.

Command Syntax: SLAVe <J clock spec>,<K clock spec>,<L clock spec>

| | | | | | | |
|-----------------------|---|---|----------|--|---|---|
| < any clock spec >::= | { | [| NEGative | | 0 |] |
| | | [| POSitive | | 1 |] |
| | | [| BOTH | | 2 |] |
| | | [| DISabled | | 3 |] |

Example: OUTPUT 707;"SLAVE POSITIVE,POSITIVE,POSITIVE"

Query Syntax: SLAVe ?

Returned Format: [SLAVe]<J clock spec>,<K clock spec>,<L clock spec><crlf>

Example: DIM Slave\$[50]
 OUTPUT 707;"SLAVE?"
 ENTER 707;Slave\$
 PRINT Slave\$

STATE SUBSYSTEM

DATA

Command sets the clock polarity for the serial data clock in extra or missing bit modes. Query returns the serial data clock setting used in extra or missing bit modes.

Command Syntax: DATA <L clock spec>

| | | | | | | |
|---------------------|---|---|----------|--|---|---|
| < L clock spec >::= | { | [| NEGative | | 0 |] |
| | | [| POSitive | | 1 |] |
| | | [| BOTH | | 2 |] |
| | | [| DISabled | | 3 |] |
| | | } | | | | |

Example: OUTPUT 707;"DATA BOTH"

Query Syntax: DATA ?

Returned Format: [DATA]<L clock spec><cr><lf>

Example: OUTPUT 707;"DATA?"
 ENTER 707;Data\$
 PRINT Data\$

REFERENCE

Command sets the clock polarity for the reference cell clock used in extra or missing bit modes. Query returns the reference cell clock setting used in extra or missing bit modes.

Command Syntax: REFERENCE <J clock spec>

| | | | | | | |
|---------------------|---|---|----------|--|---|---|
| < J clock spec >::= | { | [| NEGative | | 0 |] |
| | | [| POSitive | | 1 |] |
| | | [| BOTH | | 2 |] |
| | | [| DISabled | | 3 |] |
| | | } | | | | |

Example: OUTPUT 707;"REFERENCE POSITIVE"

Query Syntax: REFERENCE ?

Returned Format: [REFERENCE]<J clock spec><cr><lf>

Example: OUTPUT 707;"REFERENCE?"
 ENTER 707;Ref\$
 PRINT Ref\$

STATE SUBSYSTEM

POD

Command sets the threshold for the specified pod. TTL and ECL are presets. <NR2> has the range -9.9 to 9.9 volts. Query returns the settings for the specified pod. ECL and TTL are returned as voltage values.

Command Syntax: { POD0 | POD1 | POD2 } {[TTL]
[ECL]
[<NR2>]}

Example: OUTPUT 707;"POD0 TTL"

Query Syntax: { POD0 | POD1 | POD2 } ?

Returned Format: [POD0 | POD1 | POD2]<NR2><cr><lf>

Example: OUTPUT 707;"POD2?"
ENTER 707;Pod2\$
PRINT Pod2\$

LABel

The label command addresses a particular label up to five characters long as the destination for the label commands that follow. A valid label name cannot be all spaces (" "). If the label does not exist, then a new label is added to the list, unless the list is full. If all labels had been previously deleted, the dummy first label is replaced with the new label entered.

Label names are sensitive to leading spaces. If the label name is less than 5 characters, spaces are appended to pad to 5 characters. Thus "A" becomes "A ", and "A " is unique from " A".

When a new label is declared, its polarity is defaulted to positive, and all bits are allocated.

In normal and qualified modes, up to 8 labels may be used. In the bit fault modes, only one label is permitted. There is a unique label list held for each of the two cases.

Query returns the currently addressed label.

Command Syntax: LABEL <string>

Example: OUTPUT 707;"LABEL ""DATA"""

Query Syntax: LABel ?

Returned Format: [LABel]<string><cr><lf>

Example: OUTPUT 707;"LABEL?"
ENTER 707;Labe1\$
PRINT Labe1\$

STATE SUBSYSTEM

BASE

Command Selects the base in which the pattern resource terms are displayed for the addressed label and sets the base for query responses. Binary is the most useful in many cases since any pattern of don't cares is expressible without "\$".

Query returns the selected display base.

Command Syntax: BASE { [BINary | 0]
[OCTal | 1]
[DECimal | 2]
[HEXadecimal | 3] }

Example: OUTPUT 707;"HEXADECIMAL"

Query Syntax: BASE ?

Returned Format: [BASE]<argument><cr lf>

Example: OUTPUT 707;"BASE?"
ENTER 707;Base\$
PRINT Base\$

DElete

Command which deletes either all labels or the label specified. If all labels are deleted, a temporary place holding label is written to the first label location since front panel operation requires at least one label to be displayed. However, this label is replaced when a new label is defined with the LABEL command or the INSERT command. The place holder is considered a valid label if any of its attributes are changed or after reset or power cycle, and it is used for triggering.

Command Syntax: DElete { [ALL
[<5 characters quoted>] }

Example: OUTPUT 707;"DELETE ""DATA """

INSert

Command allows a new label to be added to the list directly following the currently addressed label. Similar in action to the label command except the label name must be unique, that is, the insert command cannot be used to merely address a label. Refer to the LABEL command for more details.

Command Syntax: INSert <5 characters quoted>

Example: OUTPUT 707"INSERT" Clk ""

STATE SUBSYSTEM

PATtern

Command defines the selected resource term for the addressed label. Pattern D is not allowed for missing or extra bit modes.

Query returns the specified pattern expression for the addressed label.

Command Syntax: PATtern <which pattern>,<pattern expression>

<which pattern> ::= { A|B|C|D }

<pattern expression> ::= { [#H<hexadecimal expression>]
[<decimal expression>]
[#O<octal expression>]
[#Y<binary expression>] }

<hexadecimal expression> ::= { 0|1|2|3|4|5|6|7|8|9|a|b|
c|d|e|f|A|B|C|D|E|F|X|\$ }!1!7!

<decimal expression> ::= { 0|1|2|3|4|5|6|7|8|9|X|\$ }!1!9!

<octal expression> ::= { 0|1|2|3|4|5|6|7|8|X|\$ }!1!9!

<binary expression> ::= { 0|1|X }!1!27!

Note

"\$" valid for query only. An "X" specifies a don't care state, while a "\$" specifies an undefined state. "Don't cares" in one value base cannot always be translated to another base.

Example: OUTPUT 707;"PATTERN A,#Y01110001"

Query Syntax: PATtern <which pattern> ?

Returned Format: [PATtern] <which pattern>,<pattern expression><crlf>

Example: DIM Pattern\$[50]
OUTPUT 707;"PATTERN?"
ENTER 707;Pattern\$
PRINT Pattern\$

STATE SUBSYSTEM

Sequence Commands

SEquence

Command enables sequence queries to be returned intact.
Query returns the current sequence settings.

Command Syntax: SEquence

Example: OUTPUT 707;"SEQUENCE"

Query Syntax: SEquence ?

Returned Format: [SEquence <crLf>]
[FIND] <sequence find list><crLf>
[COUNT]<NR1><crLf>
[OF]<sequence trigger list><crLf>
[TERMs]<argument><crLf>
[REStart]<sequence restart list><crLf>

Example: DIM Seq\$[200]
OUTPUT 707;"SEQUENCE?"
ENTER 707;Seq\$
PRINT Seq\$

STATE SUBSYSTEM

FIND

Command defines the sequence terms that must be found before the trigger term. There can be 0, 1, 2, or 3 sequence terms. "FIND NOTHING" specifies the 0 term case. Terms "D" and "NOTD" are not permitted in missing bit and extra bit modes.

Query returns the sequence term list.

Command Syntax: FIND < sequence find list >

```
< sequence find list >:= {[ NOTHING | 0 ]
                           [ <list item>[,<list item>[,<list item>]]]}
```

```
<list item>:= {[ ANYState | 1 ]
                [ A       | 2 ]
                [ B       | 3 ]
                [ C       | 4 ]
                [ D       | 5 ]
                [ NOTA    | 6 ]
                [ NOTB    | 7 ]
                [ NOTC    | 8 ]
                [ NOTD    | 9 ]}
```

Example: OUTPUT 707;"FIND ANYSTATE"

Query Syntax: FIND

Returned Format: [FIND]<sequence find list><cr><lf>

Example: OUTPUT 707;"FIND"
 ENTER 707;Find\$
 PRINT Find\$

COUNT

Command programs the trigger term occurrence count. Range is 1..59999.

Query returns the occurrence count.

Command Syntax: COUNT < NR1 >

Example: OUTPUT 707;"COUNT 154"

Query Syntax: COUNT ?

Returned Format: [COUNT]<NR1><cr><lf>

Example: OUTPUT 707;"COUNT?"
 ENTER 707;Count\$
 PRINT Count\$

STATE SUBSYSTEM

OF

Command defines the trigger term. 1 to 4 terms may be used in the compound expression. Only one term may be specified if using "ANYState" list item. The "TERMs" command defines whether the terms "or" or "and". This command is not valid for missing bit and extra bit modes.

Query returns the compound trigger term.

Command Syntax: OF < sequence trigger list >

```
<sequence trigger list> ::=
    <list item>[,<list item>[,<list item>[,<list item>]]]
```

```
<list item> ::= { [ ANYState | 1 ]
                  [ A       | 2 ]
                  [ B       | 3 ]
                  [ C       | 4 ]
                  [ D       | 5 ]
                  [ NOTA    | 6 ]
                  [ NOTB    | 7 ]
                  [ NOTC    | 8 ]
                  [ NOTD    | 9 ] }
```

Query Syntax: OF ?

Returned Format: [OF] <sequence trigger list><crLf>

Example: DIM Of\$[50]
 OUTPUT 707;"OF?"
 ENTER 707;Of\$
 PRINT Of\$

TERMs

Command defines whether to "or" or "and" the trigger terms.
 Query returns the "and" or "or" mode for the trigger terms.

Command Syntax: TERMs { [OR | 0]
 [AND | 1] }

Example: OUTPUT 707;"TERMs AND"

Query Syntax: TERMs ?

Returned Format: [TERMs]<argument><crLf>

Example: OUTPUT 707;"TERMs?"
 ENTER 707;Terms\$
 PRINT Terms\$

STATE SUBSYSTEM

REStart

Defines the restart expression. 1 to 4 terms may be used in a compound expression. Only one term may be specified if using "NOSTate" or "ANYState" list items. The "D" and "NOTD" terms are not permitted in missing bit or extra bit modes.

Query returns the restart expression.

Command Syntax: REStart <sequence restart list>

<sequence restart list>::=

<list item>[,<list item>[,<list item>[,<list item>]]]

| | | | | |
|------------------|----------|--|----|---|
| <list item>::= { | ANYState | | 1 | } |
| | A | | 2 | } |
| | B | | 3 | } |
| | C | | 4 | } |
| | D | | 5 | } |
| | NOTA | | 6 | } |
| | NOTB | | 7 | } |
| | NOTC | | 8 | } |
| | NOTD | | 9 | } |
| | NOSTate | | 10 | } |

Example: OUTPUT 707;"RESTART B,NOTC"

Query Syntax: REStart ?

Returned Format: [REStart]<sequence restart list><crlf>

Example: DIM Restart\$[50]
 OUTPUT 707;"RESTART?"
 ENTER 707;Restart\$
 PRINT Restart\$

STATE SUBSYSTEM

Notes

TIMEBASE SUBSYSTEM

TIMEbase

The timebase group controls all "X axis" oscilloscope functions.

The TIMEbase command selects timebase as the destination for the commands that follow.

The TIMEbase query responds with all of the settings for the timebase.

Command Syntax: TIMEbase

Example: OUTPUT 707;"TIMEBASE"

Query Syntax: TIMEbase ?

Returned Format : [TIMEbase]<cr lf>
[MODE]<argument><cr lf>
[RANGE]<NR3><cr lf>
[DELay]<NR3><cr lf>
[REFerence]<argument><cr lf>
[ALIAStest]<argument><cr lf>
[SCALE]<argument><cr lf>

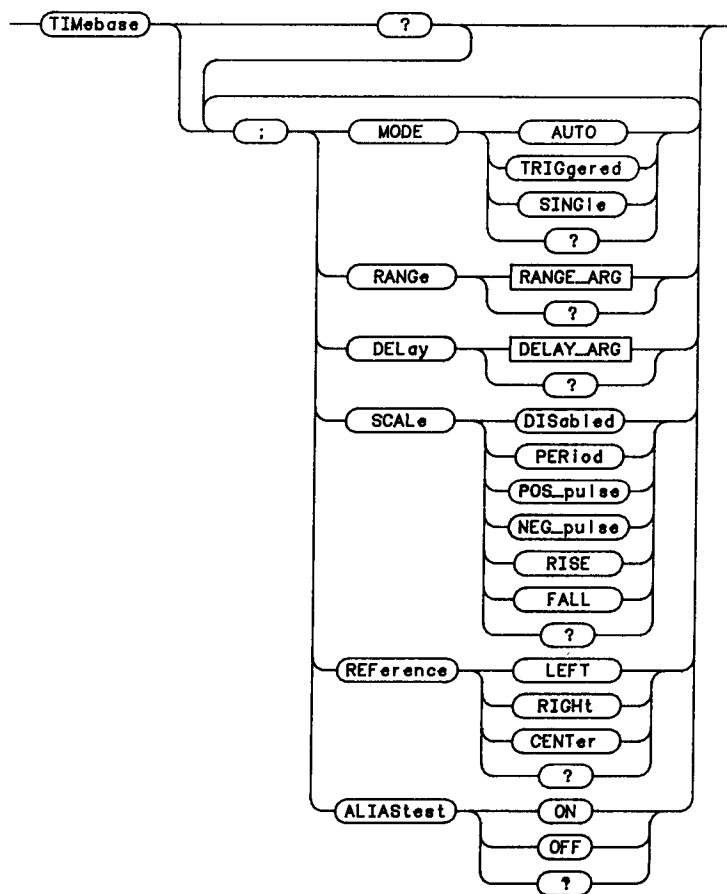
Example: DIM Time\$[130]
OUTPUT 707;"TIMEBASE?"
ENTER 707 USING "-K";Time\$
PRINT USING "K";Time\$

TIMEBASE SUBSYSTEM

TIMEbase Commands:

MODE
RANGE
DElay
REFerence
ALIAStest
SCALE
RATE

Timebase commands are order dependent in that delay limits are determined by the current range setting.



85420003

TIMEbase Subsystem Syntax Diagram

TIMEBASE SUBSYSTEM

MODE

Command sets the timebase sweep mode.

Query returns the current sweep mode.

Command Syntax: MODE { [AUTO | 0]
 [TRIGgered | 1]
 [SINGle | 2] }

Example: OUTPUT 707;"AUTO"

Query Syntax: MODE ?

Returned Format: [MODE]<argument><crlf>

Example: OUTPUT 707;"MODE?"
 ENTER 707;Mode\$
 PRINT Mode\$

RANGE

Command sets timebase range (s/div * 10 div). Range is from 50 ns to 10 s in a 1-2-5 sequence. If a range is entered that is not in this sequence, the number will be rounded up to the next larger range.

Query returns the current range setting.

Command Syntax: RANGE <NR3>

Example: OUTPUT 707;"RANGE 50E-3"

Query Syntax: RANGE ?

Returned Format:: [RANGE]<NR3><crlf>

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

TIMEBASE SUBSYSTEM

DELaY

Command sets time to current reference point from trigger. Delay range is defined as follows.

| <u>Time Range</u> | <u>Pre-Trigger Delay Range</u> | <u>Post-Trigger Delay Range</u> |
|--------------------|--------------------------------|---------------------------------|
| 50 ns to 5 μ s | Up to 5 μ s | Up to 1 ms |
| 10 μ s to 10 s | Up to 1 screen diam. | Up to 260 screen diam. |

Query returns current delay setting.

Command Syntax: DELaY <NR3>

Example: OUTPUT 707;"DELAY 50E-6"

Query Syntax: DELaY ?

Returned Format: [DELaY]<NR3><crLf>

Example: OUTPUT 707;"DELAY?"
ENTER 707;Delay\$
PRINT Delay\$

REFerence

Command sets the reference point for delay such that delay specifies the time of point 1 (LEFT), point 501 (CENTer), or point 1001 (RIGHT).

Query returns the current delay reference point.

Command Syntax: REFerence { [LEFT | 0]
[CENTer | 1]
[RIGHT | 2] }

Example: OUTPUT 707;"REFERENCE CENTER"

Query Syntax: REFerence ?

Returned Format: [REFerence]<argument><crLf>

Example: OUTPUT 707;"REFERENCE?"
ENTER 707;Ref\$
PRINT Ref\$

TIMEBASE SUBSYSTEM

ALIAStest

Command sets alias test mode. If on, then a trigger is used to count the number of trigger crossings over the acquisition window to determine whether the fundamental of the trigger signal would generate an alias.

Query returns the currently selected alias test mode.

Command Syntax: ALIAStest { [DISabled | 0]
[ENABled | 1] }

Example: OUTPUT 707;"ALIASTEST ENABLED"

Query Syntax: ALIAStest ?

Returned Format: [ALIAStest]<argument><cr><lf>

Example: OUTPUT 707;"ALIASTEST?"
ENTER 707;Alias\$
PRINT Alias\$

SCALE

Command sets autoscale mode for timebase.

Query returns the current autoscale mode.

Command Syntax: SCALE { [DISabled | 0]
[PERiod | 1]
[POS_pulse | 2]
[NEG_pulse | 3]
[RISE | 4]
[FALL | 5] }

Example: OUTPUT 707;"SCALE PERIOD"

Query Syntax: SCALE ?

Returned Format: [SCALE]<argument><cr><lf>

Example: OUTPUT 707;"SCALE?"
ENTER 707;Scale\$
PRINT Scale\$

TIMEBASE SUBSYSTEM

RATE

Command sets time range in terms of sample rate. Sample period is range/1000, rate is 1/period. The rate is displayed on the 54200A/D time menu in the "Sampling @" field.

Query returns the current sample rate.

Command Syntax: RATE <NR3>

Example: OUTPUT 707;"RATE 10E7"

Query Syntax: RATE ?

Returned Format: [RATE]<NR3><crlf>

Example: OUTPUT 707;"RATE?"
ENTER 707;Rate\$
PRINT Rate\$

TRIGGER SUBSYSTEM

TRIGger

The trigger group controls all analog trigger functions, including selecting the type of data to be acquired for trigger view.

The TRIGger command selects trigger subsystem as the destination for the trigger commands that follow. Included are the analog trigger and the overall selection of trigger mode (analog, armed, state-only).

The TRIGger query responds with all of the settings for the analog trigger.

Command Syntax: TRIGger

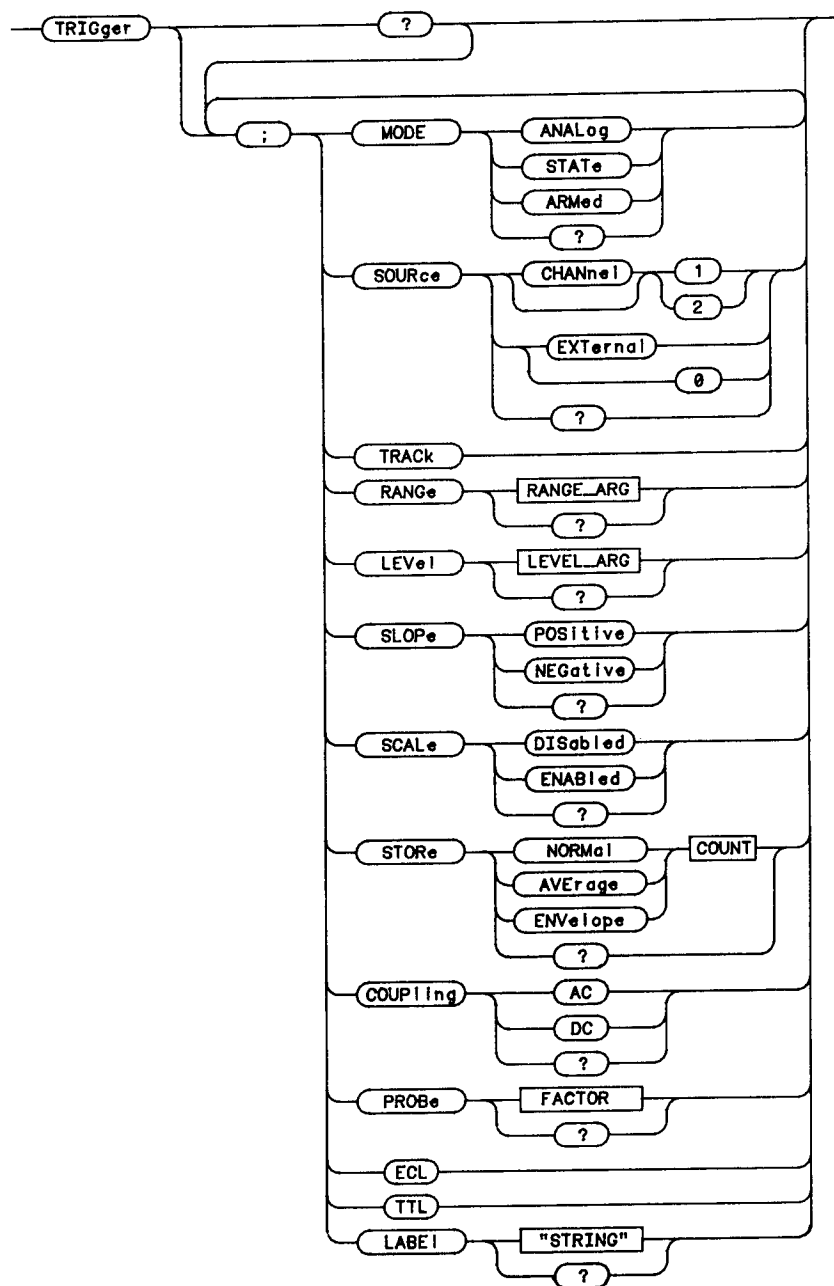
Example: OUTPUT 707;"TRIGGER"

Query Syntax: TRIGger ?

Returned Format: [TRIGger<cr>lf>]
 [MODE]<argument><cr>lf>
 [SOURce][CHANne1<NR1>| EXTernal]<cr>lf>
 [PROBe]<NR1><cr>lf>
 [RANGe]<NR3><cr>lf>
 [SLOPe]<argument><cr>lf>
 [LEVe1]<NR3><cr>lf>
 [COUPling]<argument><cr>lf>
 [LABe1]< 10 chars quoted ><cr>lf>
 [STORe]<argument>,<NR1><cr>lf>
 [SCALe]<argument><cr>lf>

Example: DIM Trig\$[200]
 OUTPUT 707;"TRIGGER?"
 ENTER 707 USING "-K";Trig\$
 PRINT USING "K";Trig\$

TRIGGER SUBSYSTEM



85420002

TRIGGER Subsystem Syntax Diagram

TRIGGER SUBSYSTEM

TRIGger Commands:

MODE
 SOURce
 PROBe
 RANGE
 LEVe1
 TRACk
 SLOPe
 COUP1ing
 SCALe
 STORe
 ECL
 TTL
 LABe1

Trigger commands are order dependent as follows: Range limits are determined by the current probe attenuation factor for external trigger source. The value limits for level are determined by the current range selection. So to completely specify trigger settings, program PROBe (if external source), then RANGE, and then LEVe1.

Separate level and range settings are held for each trigger source.

MODE

Command sets the trigger mode. Query returns the current trigger mode.

Command Syntax: MODE { [ANALog | 0]
 [STATe | 1]
 [ARMed | 2] }

Example: OUTPUT 707;"MODE ANALOG"

Query Syntax: MODE ?

Returned Format:: [MODE]<argument><crlf>

Example: OUTPUT 707;"MODE?"
 ENTER 707;Mode\$
 PRINT Mode\$

TRIGGER SUBSYSTEM

SOURce

Command sets the specified trigger source.
Query returns the selected trigger source.

Command Syntax: SOURce { [EXterna] | 0]
 [CHANne] 1 | 1]
 [CHANne] 2 | 2] }

Example: OUTPUT 707;"SOURCE CHANNEL 1"

Query Syntax: SOURce ?

Returned Format: [SOURce][CHANne] <NR1> | EXterna] <cr lf>

Example: OUTPUT 707;"SOURCE?"
 ENTER 707;Source\$
 PRINT Source\$

PROBe

Command sets trigger probe factor, valid only for external. Must be 1, 2, 5, 10, 20, 50 or 100.

Query returns the trigger probe factor. Valid for any source.

Command Syntax: PROBe <NR1>

Example: OUTPUT 707;"PROBE 10"

Query Syntax: PROBe ?

Returned Format: [PROBe] <NR3> <cr lf>

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

TRIGGER SUBSYSTEM

RANGE

Command sets trigger range. If internal source, then adjust mode is automatically selected. If external source, the allowable range settings (with 2-digit resolution) are based on the PROBe setting and are as follows:

| <u>PROBE</u> <u>FACTOR</u> | <u>VALID</u> <u>RANGE</u> <u>SETTING</u> |
|----------------------------|--|
| 1:1 | 40 mV to 40 V |
| 2:1 | 80 mV to 80 V |
| 5:1 | 200 mV to 200 V |
| 10:1 | 400 mV to 400 V |
| 20:1 | 800 mV to 800 V |
| 50:1 | 2 V to 2000 V |
| 100:1 | 4 V to 4000 V |

Query reponds with current range setting.

Command Syntax: RANGE <NR3>

Example: OUTPUT 707;"RANGE 5.5E0"

Query Syntax: RANGE ?

Returned Format: [RANGe]<NR3><crlf>

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

LEVeL

Command sets trigger level. Trigger level is ± 2 V for 40 mV to 390 mV range and ± 20 V for 400 mV to 40 V range.

Query returns the trigger level.

Command Syntax: LEVeL <NR3>

Example: OUTPUT 707;"LEVEL 350E-3"

Query Sytnax: LEVeL ?

Returned Format: [LEVeL]<NR3><crlf>

Example: OUTPUT 707;"LEVEL?"
 ENTER 707;Level\$
 PRINT Level\$

TRIGGER SUBSYSTEM

TRACK

Command which sets range to track mode if internal source selected. Not valid for external source.

Command Syntax: TRACK

Example: OUTPUT 707;"TRACK"

SLOPe

Command sets trigger slope. Query returns the selected slope.

Command Syntax: SLOPe { [POSitive | 1]
[NEGative | 0] }

Example: OUTPUT 707;"SLOPE NEGATIVE"

Query Syntax: SLOPe ?

Returned Format: [SLOPe]<argument><crlf>

Example: OUTPUT 707;"SLOPE?"
ENTER 707;Slope\$
PRINT Slope\$

COUPling

Command selects the input coupling for trigger, valid only for external. Query returns the selected coupling. Valid for any source.

Command Syntax: COUPling { [AC | 2]
[DC | 1] }

Example: OUTPUT 707;"COUPLING DC"

Query Syntax: COUPling ?

Returned Format:: [COUPling]<argument><crlf>

Example: OUTPUT 707;"COUPLING?"
ENTER 707;Coupling\$
PRINT Coupling\$

TRIGGER SUBSYSTEM

SCALE

Command sets autoscale mode for trigger.
Query returns current autoscale mode.

Command Syntax: SCALE {[DISabled | 0]
[ENABled | 1]}

Example: OUTPUT 707;"SCALE ENABLED"

Query Syntax: SCALE ?

Returned Format: [SCALE]<argument><crlf>

Example: OUTPUT 707;"SCALE?"
ENTER 707;Scale\$
PRINT Scale\$

STORE

Command sets the store mode. If average mode is selected, count determines both the terminal count and the number of averages (sample weighting factor). For the other store modes, count determines the terminal count, that is, the number of frames which must be acquired until the waveform is considered ready for analysis.

For normal store mode the count parameter is optional, if sent it must be 1. For average mode, the count must be exactly 4, 16, 64, or 256. For envelope, count can be 10 to 10,000.

Query returns the current store mode for trigger.

Command Syntax: STORE < type > < count >

< type > ::= {[NORMa1 | 1]
[AVErage | 2]
[ENVelope | 3]}

Example: OUTPUT 707;"STORE ENVELOPE,500"

Query Syntax: STORE ?

Returned Format: [STORE]<argument>,<NR1><crlf>

Example: OUTPUT 707;"STORE?"
ENTER 707;Store\$
PRINT Store\$

TRIGGER SUBSYSTEM

ECL

Command presets the trigger for ECL as follows:

```
COUPling = DC
OFFSet   = -1.00 V
RANGe    = 2.0 V
SCALE    = DISabled
```

The SCALE (autoscale) function is disabled so that a selective autoscale may be performed, thus preserving the trigger settings.

Command Syntax: ECL

Example: OUTPUT 707;"ECL"

TTL

Command presets the trigger for TTL as follows:

```
COUPling = DC
OFFSet   = 2.50 V
RANGe    = 6.0 V
SCALE    = DISabled
```

The SCALE (autoscale) function is disabled so that a selective autoscale may be performed, thus preserving the trigger settings.

Command Syntax: TTL

Example: OUTPUT 707;"TTL"

LABel

Command sets the user label for trigger view. Query returns the user label for trigger view.

Command Syntax: LABel < 10 characters quoted >

Example: OUTPUT 707;"LABEL ""Data 1"""

Query Syntax: LABel ?

Returned Format:: [LABel]< 10 chars quoted ><crlf>

```
Example: OUTPUT 707;"LABEL?"
         ENTER 707;Label$
         PRINT Label$
```


WAVEFORM SUBSYSTEM

WAVEform

The waveform subsystem provides access to waveform data. This includes active data from the channels and trigger as well as static data from the waveform memories.

For the active sources, the following commands are QUERY ONLY with the exception of FORMat. For memory waveforms (addressed directly or via a graph), values may be set or queried using DATA and PREAmble or the individual preamble function commands.

Selecting a graph as the waveform source has the benefit that a portion of the whole waveform may be read if the graph is expanded, that is, with a graph you can window the waveform and just read the data of interest.

The WAVEform command addresses the waveform subsystem as the destination of the following commands.

The WAVEform query responds with the current wave source and the validity and preamble function settings for that source.

Command Syntax: WAVEform

Example: OUTPUT 707;"WAVEFORM"

Query Syntax: WAVEform ?

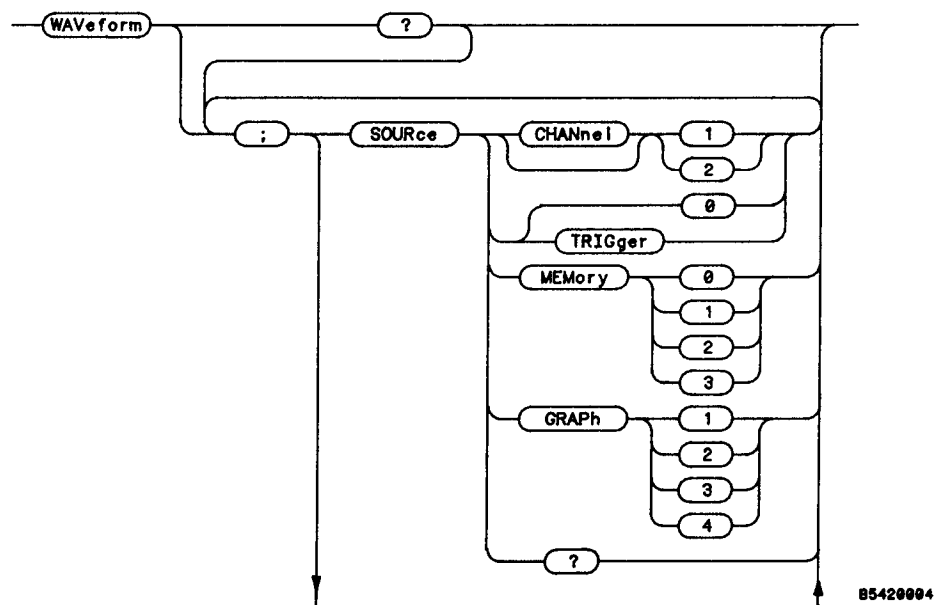
```
Returned Format: [ WAVEform <crLf> ]
[ SOURce ]<source spec><crLf>
[ VALid ]<argument><crLf>
[ FORMat ]<argument><crLf>
[ TYPE ]<argument><crLf>
[ POINTs ]<NR1><crLf>
[ COUNT ]<NR1><crLf>
[ XINCrement ]<NR3><crLf>
[ XORigin ]<NR3><crLf>
[ XREFerence ]<NR1><crLf>
[ YINCrement ]<NR3><crLf>
[ YORigin ]<NR3><crLf>
[ YREFerence ]<NR1><crLf>
[ COUPling ]<argument><crLf>
[ LABel ]<10 chars quoted><crLf>
```

```
Example: DIM Wave$(250)
         OUTPUT 707;"WAVEFORM?"
         ENTER 707 USING "-K";Wave$
         PRINT USING "K";Wave$
```

WAVEFORM SUBSYSTEM

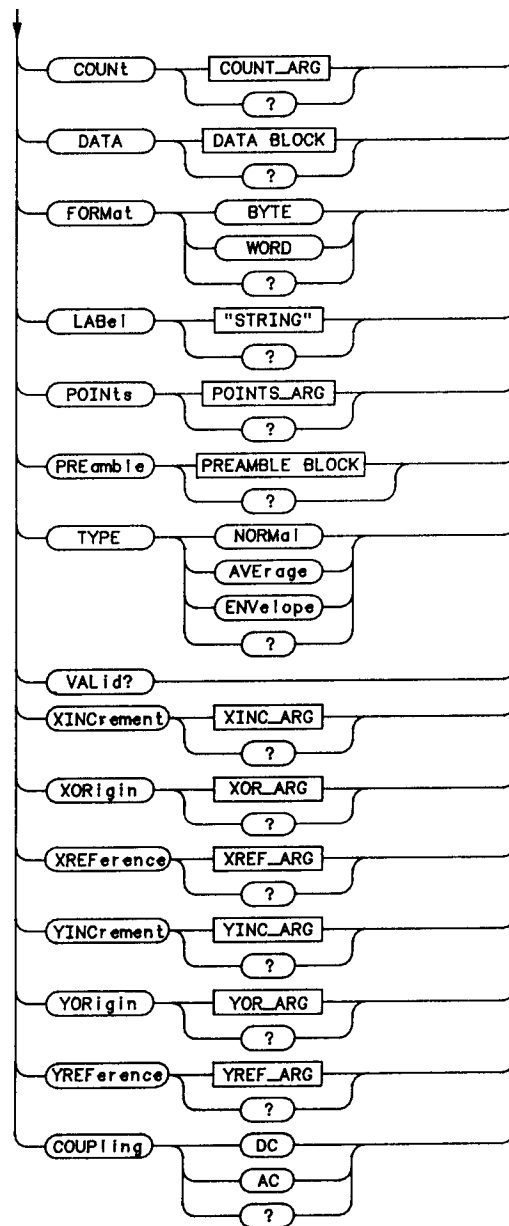
WAVEform Commands:

SOURce
COUPling
COUNT
DATA
FORMat
LABel
POINTs
PREAmble
TYPE
VALid
XINCrement
XORigin
XREFerence
YINCrement
YORigin
YREFerence



WAVEform Subsystem Syntax Diagram

WAVEFORM SUBSYSTEM



B5420004

WAVEform Subsystem Syntax Diagram (Cont)

WAVEFORM SUBSYSTEM

SOURce

Command sets the waveform source for the WAVEform commands that follow.

SOURce TRIGger – Sets trigger view (analog trigger) as the waveform addressed. Read only.

SOURce CHANnel{1|2} – Sets channel 1 or channel 2 as the waveform addressed. Read only.

SOURce GRAPh{1|2|3|4} – Sets the waveform displayed on graph 1, 2, 3, or 4 as the addressed waveform. Only data on display is accessible with DATA query. PREamble query returns points and x-origin adjusted for current expansion. Data can be written to graphs if the source is a memory, otherwise read only.

SOURce MEMory{0|1|2|3} – Sets memory 0, 1, 2, or 3 as the waveform addressed. Only memory waveforms are read/write.

Query returns the currently selected source.

Command Syntax: SOURce { [TRIGger | 0]
[CHANnel1 | 1]
[CHANnel2 | 2]
[GRAPh{1|2|3|4}]
[MEMory{0|1|2|3}] }

Example: OUTPUT 707;"SOURCE CHANNEL1"

Query Syntax: SOURce ?

Returned Format : [SOURce]{ [CHANnel1]
[CHANnel2]
[TRIGger]
[GRAPh1]
[GRAPh2]
[GRAPh3]
[GRAPh4]
[MEMory0]
[MEMory1]
[MEMory2]
[MEMory3] }

Example: OUTPUT 707;"SOURCE?"
ENTER 707;Source\$
PRINT Source\$

WAVEFORM SUBSYSTEM

COUPling

Command sets the coupling field in the waveform preamble.
Query returns the coupling for the waveform.

Command Syntax: COUPling { [DC | 1]
[AC | 2] }

Example: OUTPUT 707;"COUPLING AC"

Query Syntax: COUPling

Returned Format: [COUPling]<argument><crlf>

Example: OUTPUT 707;"COUPLING?"
ENTER 707;Coup\$
PRINT Coup\$

COUNT

Command sets the count parameter in the memory waveform preamble. If waveform is type average, count signifies both the terminal count and the number of averages (sample weighting factor). For the other store modes, count determines the terminal count, that is, the number of frames which were acquired.

The count parameter must be in the range 0 to 10000. To be meaningful, this value should be set as follows: For normal store mode the count parameter should be 1. For average mode, the count should be exactly 4, 16, 64, or 256. For envelope, count can be 1 to 10,000. If count is set to 0, the waveform will be considered invalid.

Query returns the count factor for the addressed waveform. 0 is returned for invalid waveforms.

Command Syntax: COUNT <NR1>

Example: OUTPUT 707;"COUNT 16"

Query Syntax: COUNT ?

Returned Format: [COUNT]<NR1><crlf>

Example: OUTPUT 707;"COUNT?"
ENTER 707;Count\$
PRINT Count\$

WAVEFORM SUBSYSTEM

DATA

54200A/D accepts a waveform data block if the waveform addressed is memory (or a graph displaying memory). Block length must match the defined store mode, format, and points settings. These parameters are set up by the PREamble command.

Query returns the addressed waveforms data points if it is a valid waveform. If running, and the waveform is being acquired, transmission occurs after the specified count has been reached. Invalid waveforms are sent as all 0's.

Command Syntax: DATA < Block type A >

Query Syntax: DATA ?

Returned Format : [DATA] #Abb< binary wave data >

where bb is a word (binary integer) giving the number of data bytes in <binary wave data>. Wave data is sent as either bytes or words based upon the current format specification.

The following example shows both the command and query form of DATA, and the steps required to enter preamble data required before loading waveform data into a memory.

```

1  OPTION BASE 1
1850 !
1860 ! Set up scope to accept waveform and preamble data
1870 ASSIGN @Fast TO 707;FORMAT OFF
1880 OUTPUT 707;"HEADER OFF ;EOI ON"
1890 OUTPUT 707;"ACQUIRE;TYPE NORMAL"
1900 !
1910 ! Acquire data
1920 OUTPUT 707;"DIGITIZE CHANNEL 1"
1930 !
1940 ! Set up waveform source
1950 OUTPUT 707;"WAVEFORM;SOURCE CHANNEL 1;FORMAT WORD"
1960 !
1970 ! Input waveform preamble to controller
1980 DIM Pre$(150)
1990 OUTPUT 707;"WAVEFORM PREAMBLE?"
2000 ENTER 707 USING "-K";Pre$
2010 !
2020 !Input waveform data to controller
2030 OUTPUT 707;"DATA?"
2040 ENTER 707 USING "#,2A,W";Header$,Bytes
2050 ! Header$ = #A
2060 ! Bytes = 2002
2070 Words=Bytes/2
2080 ALLOCATE INTEGER Wav(Words)
2090 ENTER @Fast;Wav(*)
2100 !
2110 ! Output waveform preamble and data to memory 3
2120 OUTPUT 707;"SOURCE MEMORY3"
2130 OUTPUT 707 USING "#,K";"PREAMBLE ",Pre$
2140 OUTPUT 707 USING "#,7A,W";"DATA #A",Bytes
2150 OUTPUT @Fast;Wav(*)
2160 !
2170 ! Display memory 3 on graph 2
2180 OUTPUT 707;"RECALL GRAPH 2, MEMORY 3"
2190 !

```

WAVEFORM SUBSYSTEM

FORMat

Command sets the data transmission mode for waveform data points. Data is transferred either as signed 8-bit (BYTE) or 16-bit (WORD), the sign bit is always positive (0). May be sent regardless of what type waveform is addressed.

Query returns the format specification in effect.

Command Syntax: FORMat { [BYTE | 1]
[WORD | 2] }

Example: OUTPUT 707;"FORMAT WORD"

Query Syntax: FORMat ?

Returned Format: [FORMat]<argument><crlf>

Example: OUTPUT 707;"FORMAT?"
ENTER 707;Format\$
PRINT Format\$

WAVEFORM SUBSYSTEM

LABel

Command sets the label field for the addressed waveform. Labels for the volatile waveforms are defaulted to " " at power-on or reset.

Query returns the label for the addressed waveform.

Command Syntax: LABel <10 characters quoted>

Example: OUTPUT 707;"LABEL ""Waveform 1"""

Query Syntax: LABel ?

Returned Format: [LABel] <10 characters quoted><crlf>

Example: OUTPUT 707;"LABEL?"
ENTER 707;Label\$
PRINT Label\$

POINTS

Command sets the number of points for a memory waveform. Must be in the range from 51 to 1001.

Query returns the number of data points in the addressed waveform. Value is 1001 for active sources, for graphs and memories may be 51 to 1001.

Command Syntax: POINTs<NR1>

Example: OUTPUT 707;"POINTS 1001"

Query Syntax: POINTs ?

Returned Format: [POINTs] <NR1><crlf>

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

WAVEFORM SUBSYSTEM

PREamble

Command sets the preamble for the addressed waveform if it is a memory or a graph displaying a memory. All items must be sent in exactly the prescribed order, delimited by commas.

Query returns the preamble block for the selected waveform.

In the examples given below, the PREamble query is shown first to show how the preamble can be stored in the controller. The PREamble command is then given to show how to re-enter the waveform preamble into a 54200A/D memory. Refer to the DATA command for more information.

Query Syntax: PREamble ?

```
Returned Format: [PREamble]
                 <format Parameter>,
                 <type Parameter>,
                 <points NR1>,
                 <count NR1>,
                 <xincrement NR3>,
                 <xorigin NR3>,
                 <xreference NR1>,
                 <yincrement NR3>,
                 <yorigin NR3>,
                 <yreference NR1>,
                 <coupling Parameter>,
                 <label 10 chars quoted><crLf>
```

```
Example: DIM Pre$[150]
        OUTPUT 707;"HEADER OFF"
        OUTPUT 707;"PREAMBLE?"
        ENTER 707;Pre$
        PRINT Pre$
```

Command Syntax: PREamble <preamble block>

```
<preamble block> ::= <format>,
                    <type>,
                    <points>,
                    <count>,
                    <xincrement>,
                    <xorigin>,
                    <xreference>,
                    <yincrement>,
                    <yorigin>,
                    <yreference>,
                    <coupling>,
                    <label><crLf>
```

```
Example: OUTPUT 707 USING "#,K";"PREAMBLE ",Pre$
```

WAVEFORM SUBSYSTEM

TYPE

Command sets the data type for memory waveforms.
Query returns the data type of the selected waveform.

Command Syntax: TYPE {[NORMa] | 1]
[AVErAge | 2]
[ENVelope | 3]}

Example: OUTPUT 707;"TYPE NORMAL"

Query Sytnax: TYPE ?

Returned Format: [TYPE]<argument><crlf>

Example: OUTPUT 707;"TYPE?"
ENTER 707;Type\$
PRINT Type\$

VALid

This command has no effect, but the parameter must be in range 0,1, or -3.

Query returns validity of data for the addressed waveform or graph. To be valid, count must be non-zero and equal to the terminal count specified in ACQuire, CHANnel or TRIGger subsystems.
WAVEform TYPE must be 1, 2, or 3.

Command Syntax: VALid {[GOOD_data | 1]
[NO_data | 0]
[CLIPped | -3]}<crLf>

Example: OUTPUT 707;"VALID GOOD_DATA"

Query Syntax: VALid ?

Returned Format: [VALid] {[GOOD_data | 1]
[NO_data | 0]
[CLIPped | -3]}<crLf>

Example: OUTPUT 707;"VALID?"
ENTER 707;Valid\$
PRINT Valid\$

WAVEFORM SUBSYSTEM

XINCrement

Command sets the time range and time per sample for memory waveform only.
NR3 is equal to time range ÷ 1000.

Query returns the x increment (time per sample) for any addressed waveform.

Command Syntax: XINCrement <NR3>

Example: OUTPUT 707;"XINCREMENT 200E-6"

Query Syntax: XINCrement ?

Returned Format: [XINCrement]<NR3><crLf>

Example: DIM Xinc\$[30]
OUTPUT 707;"XINCREMENT?"
ENTER 707;Xinc\$
PRINT Xinc\$

XORigin

Command sets the time of the first data point for memory waveforms only.

Query returns the time of the first data point (x origin) of any selected waveform.

Command Syntax: XORigin <NR3>

Example: OUTPUT 707;"XORIGIN 0E0"

Query Syntax: XORigin ?

Returned Format: [XORigin]<NR3><crLf>

Example: OUTPUT 707;"XORIGIN?"
ENTER 707;Xor\$
PRINT Xor\$

WAVEFORM SUBSYSTEM

XREference

Command sets the point at which time x origin exists for memory waveforms only. Must be 1.

Query returns the point number at which x origin exists for any selected waveform. Always 1.

Command Syntax: XREference 1

Example: OUTPUT 707;"XREFERENCE 1"

Query Syntax: REference ?

Returned Format: [XREference]<NR1><crlf>

Example: OUTPUT 707;"XREFERENCE?"
ENTER 707;Xref\$
PRINT Xref\$

YINCrement

Command sets the volts per q level and y axis range for the memory waveform only. This value is interpreted with the current format selection.

Query returns the y increment (volts per q level) for any addressed waveform. This value is scaled for the data format specification currently set.

Command Syntax: YINCrement <NR3>

Example: OUTPUT 707;"YINCREMENT 201.6E-6"

Query Syntax: YINCrement ?

Returned Format: [YINCrement]<NR3><crlf>

Example: DIM Yinc\$[30]
OUTPUT 707;"YINCREMENT?"
ENTER 707;Yinc\$
PRINT Yinc\$

WAVEFORM SUBSYSTEM

YORigin

Command sets the voltage at the midpoint of the voltage range, the same as offset for memory waveforms only.

Query returns the y origin of the selected waveform, the voltage at "center screen" for any selected waveform.

Command Syntax: YORigin <NR3>

Example: OUTPUT 707;"YORIGIN 2.3E0"

Query Syntax: YORigin ?

Returned Format: [YORigin]<NR3><crlf>

Example: OUTPUT 707;"YORIGIN?"
ENTER 707;Yorg\$
PRINT Yorg\$

YREference

Command has no effect but must be 62 (BYTE) or 15872 (WORD) depending upon FORMat selection.

Query returns the y reference value. This value is used to normalize q level values (the way waveforms are sent) prior to conversion to volts.

Command Syntax: YREference <NR1>

Example: OUTPUT 707;"YREFERENCE 15872"

Query Syntax: YREference ?

Returned Format: [YREference]<NR1><crlf>

Example: OUTPUT 707;"YREFERENCE?"
ENTER 707;Yref\$
PRINT Yref\$