Waveform Transfer Command Group

Waveform Transfer Overview

You use the commands in the Waveform transfer Command Group to transfer waveform data points to and from the instrument. Waveform data points are a collection of values that define a waveform. One data value usually represents one data point in the waveform record. When working with envelope waveforms, each data value is either the minimum or maximum of a min/max pair. Before you transfer waveform data, you must specify the data format, record length, and waveform source.

Data Formats

Acquired waveform data uses eight or more bits to represent each data point. The number of bits used depends on the acquisition mode specified when you acquired the data. Data acquired in SAMple or ENVelope mode uses eight bits per waveform data point. Data acquired in AVERage mode uses up to 14 bits per point.

The oscilloscope can transfer waveform data in either ASCII or binary format. You specify the format with the DATa:ENCdg command. The instrument uses signed, 4 byte integers and floating point values; it does not support unsigned floating point values.

ASCII data is represented by signed integer or floating point values. An example ASCII waveform data string may look like this:

```
CURVE<space>-110,-109,-110,-110,-109,-107,
-109,-107,-106,-105,-103,-100,-97,-90,-84,-80
```

Use ASCII to obtain more readable and easier to format output than binary. However, ASCII may require more bytes to send the same values than it does with binary. This may reduce transmission speeds.

Binary data can be represented by signed integer or floating point values. The range of the values depends on the byte width specified. When the byte width is one, signed integer data ranges from -128 to 127, and positive integer values range from 0 to 255. When the byte width is two, the values range from -32768 to 32767. When a MATH (or REF that came with a MATH) is involved, 32 bit floating point values are used that are four bytes in width.

The defined binary formats specify the order in which the bytes are transferred. The following are the four binary formats:

- RIBinary specifies signed integer data point representation with the most significant byte transferred first.
- SRIBinary is the same as RIBinary except that the byte order is swapped, meaning that the least significant byte is transferred first. This format is useful when transferring data to IBM-compatible PCs
- RFBinary specifies floating point data point representation with the most significant byte transferred first.
- SRFBinary is the same as RFBinary except that the byte order is swapped, meaning that the least significant byte is transferred first. This format is useful when transferring data to IBM-compatible PCs.

Waveform Data and Record Lengths

You can transfer multiple points for each waveform record. You can transfer a portion of the waveform or you can transfer the entire record. You can use the DATa:STARt and DATa:STOP commands to specify the first and last data points of the waveform record.

When transferring data into the instrument, you must first specify the record length of the destination waveform record. You do this with the WFMInopre:NR_Pt command. Next, specify the first data point within the waveform record. For example, when you set DATa:STARt to 1, data points will be stored starting with the first point in the record. The instrument will ignore the value set by DATa:STOP when reading in data. It will stop reading data when there is no more data to read or when it has reached the specified record length.

When transferring data from the instrument, you must specify the first and last data points in the waveform record. Setting DATa:STARt to 1 and DATa:STOP to the record length will always return the entire waveform.

Waveform Data Locations and Memory Allocation

The DATa:SOUrce command specifies the waveform source when transferring a waveform from the

oscilloscope. You can only transfer one waveform at a time. Waveforms sent to the instrument are always stored in one of the four reference memory locations. You use the DATa:DESTination command to specify a reference memory location.

Waveform Preamble

Each waveform that you transfer has an associated waveform preamble that contains information such as the horizontal scale, the vertical scale, and other settings in effect when the waveform was created. Refer to the individual WFMInpre and WFMOutpre commands for more information.

Scaling Waveform Data

Once you transfer the waveform data to the controller, you can convert the data points into voltage values for analysis using information from the waveform preamble.

Transferring Waveform Data from the Oscilloscope

You can transfer waveforms from the oscilloscope to an external controller using the following sequence:

1 Select the waveform source(s) using DATa:SOUrce.

2 Specify the waveform data format using DATa:ENCdg.

3 Specify the number of bytes per data point using WFMOutpre:BYT_Nr.

Note: MATH waveforms (and REF waveforms that came from a MATH) are always set to four bytes.

4 Specify the portion of the waveform that you want to transfer using DATa:STARt and DATa:STOP.

5 Transfer waveform preamble information using WFMOutpre?.

6 Transfer waveform data from the oscilloscope using CURVe?.

Transferring Waveform Data to the Oscilloscope

1 Specify waveform reference memory using DATa:DESTination.

2 Set WFMInpre:NR_Pt to equal the number of data points to be sent.

3 Specify the waveform data format using WFMInpre:ENCdg.

4 Specify the number of bytes per data point using WFMInpre:BYT_Nr.

5 Specify first data point in the waveform record using DATa:STARt.

6 Transfer waveform preamble information using WFMInpre.

Note: FastAcq waveforms cannot be returned to the scope via the CURVe command.

Waveform Transfer Commands

Command	Description
CURve?	Transfers waveform data from instrument specified by the DATa:SOUrce command
CURVe	Transfer waveform data to the instrument (reference memory location specified by DATa:DESTination)
DATa?	Returns waveform data format and location
DATa	Sets the waveform data format and location
DATa:DESTination?	Returns the reference memory location for storing waveform data sent to the oscilloscope
DATa:DESTination	Sets the reference memory location for storing waveform data sent to the oscilloscope
DATa:ENCdg?	Returns the format of outgoing waveform data
DATa:ENCdg	Sets the format of outgoing waveform data
DATa:SOUrce?	Returns the location of waveform data transferred from the instrument
DATa:SOUrce	Sets the location of waveform data transferred from the instrument
DATa:STARt?	Returns the starting point in waveform transfer
DATa:STARt	Sets the starting point in waveform transfer
DATa:STOP?	Returns the ending data point in waveform transfer
DATa:STOP	Sets the ending data point in waveform transfer
WAVFrm?	Returns a branch query containing waveform data in either binary or ASCII format, waveform formatting data, and the location of the waveform data source
WFMInpre?	Returns the waveform formatting specification to be applied to the next incoming CURVE command data
WFMInpre:BIT_Nr?	Returns the number of bits per binary waveform point
WFMInpre:BIT_Nr	Sets the number of bits per binary waveform point
WFMInpre:BN_Fmt?	Returns the format of binary data for the incoming waveform
WFMInpre:BN_Fmt	Sets the format of binary data for the incoming waveform
WFMInpre:BYT_Nr?	Returns the binary field data width for the first ordered waveform
WFMInpre:BYT_Nr	Sets the binary field data width for the first ordered waveform
WFMInpre:BYT_Or?	Returns the byte order of waveform points
WFMInpre:BYT_Or	Sets the byte order of waveform points

WFMInpre:ENCdg?	Returns the type of encoding for incoming waveform data
WFMInpre:ENCdg	Sets the type of encoding for incoming waveform data
WFMInpre:NR_Pt?	Returns the number of points in the transmitted waveform record
WFMInpre:NR_Pt	Sets the number of points in the transmitted waveform record
WFMInpre:PT_Fmt?	Returns the point format of incoming waveform data
WFMInpre:PT_Fmt	Sets the point format of incoming waveform data
WFMinpre:PT_OFF?	Returns the trigger point within the incoming waveform data
WFMinpre:PT_OFF	Sets the trigger point within the incoming waveform data
WFMinpre:WFID	Provided to allow a waveform extracted from the oscilloscope to be more easily sent back
WFMInpre:XINcr?	Returns the horizontal sampling interval between incoming waveform points
WFMInpre:XINcr	Sets the horizontal sampling interval between incoming waveform points
WFMInpre:XUNit?	Returns the horizontal units of the incoming waveform
WFMInpre:XUNit	Sets the horizontal units of the incoming waveform
WFMInpre:XZEro?	Returns the (sub-sample) time between the trigger sample and the occurrence of the actual incoming waveform trigger
WFMInpre:XZEro	Sets the (sub-sample) time between the trigger sample and the occurrence of the actual incoming waveform trigger
WFMInpre:YMUlt?	Returns the vertical scale factor, per digitizing level, of the incoming waveform points
WFMInpre:YMUlt	Sets the vertical scale factor, per digitizing level, of the incoming waveform points
WFMInpre:YOFf?	Returns the vertical position of the incoming waveform in digitizing levels
WFMInpre:YOFf	Sets the vertical position of the incoming waveform in digitizing levels
WFMInpre:YUNit?	Returns the vertical units of the incoming waveform
WFMInpre:YUNit	Sets the vertical units of the incoming waveform
WFMInpre:YZEro?	Returns the offset of the incoming waveform
WFMInpre:YZEro	Sets the offset of the incoming waveform
WFMOutpre?	Returns the waveform formatting data for the waveform specified by the DATA:SOURCE command
WFMOutpre:BIT_Nr?	Returns the number of bits per waveform

	point that outgoing waveforms contain
WFMOutpre:BIT_Nr	Sets the number of bits per waveform point that outgoing waveforms contain
WFMOutpre:BN_Fmt?	Returns the format of binary data for the waveform
WFMOutpre:BN_Fmt	Sets the format of binary data for the waveform
WFMOutpre:BYT_Nr?	Returns the data width for the waveform
WFMOutpre:BYT_Nr	Sets the data width for the waveform
WFMOutpre:BYT_Or?	Returns the byte order of waveform points
WFMOutpre:BYT_Or	Sets the byte order of waveform points
WFMOutpre:ENCdg?	Returns the type of encoding for outgoing waveforms
WFMOutpre:ENCdg	Sets the type of encoding for outgoing waveforms
WFMOutpre:NR_Pt?	Returns the number of points for the waveform transmitted in response to a CURVE? query
WFMOutpre:PT_Fmt?	Returns the point format for the waveform
WFMOutpre:PT_OFF?	Returns the trigger point relative to DATA:START for the waveform
WFMOutpre:PT_OR?	Returns whether the source waveform is DPO
WFMOutpre:WFId?	Returns a string describing the acquisition parameters for the waveform
WFMOutpre:XINcr?	Returns the horizontal sampling interval
WFMOutpre:XUNit?	Returns the horizontal units for the waveform
WFMOutpre:XZEro?	Returns the (sub-sample) time between the trigger sample and the occurrence of the actual waveform trigger
WFMOutpre:YMUlt?	Returns the vertical scale factor per digitizing level for the waveform
WFMOutpre:YOFf?	Returns the vertical offset in digitizing levels for the waveform
WFMOutpre:YUNit?	Returns the vertical units for the waveform
WFMOutpre:YZEro?	Returns the vertical offset for the waveform

CURVe

Description

This command transfers waveform data to and from the oscilloscope. Each waveform that is transferred has an associated waveform preamble that contains information such as data format and scale.

The CURVe? query transfers data from the instrument. The data source is specified by the DATa:SOUrce command. The first and last data points that are specified by the DATa:STARt and DATa:STOP commands.

The CURVe command transfers waveform data to the instrument. The data is stored in the reference memory location specified by DATa:DESTination, starting with the data point specified by DATa:STARt. Only one waveform can be transferred at a time. The waveform will only be displayed if the reference is displayed.

Group

Waveform Transfer

Related Commands

DATa:DESTination (see page 469), DATa:SOUrce (see page 472), DATa:STARt (see page 473), DATa:STOP (see page 474), WFMInpre? (see page 476), WFMInpre:BYT_Nr (see page 480), WFMOutpre? (see page 493)

Syntax 1

```
CURVe {<Block>|<asc curve>}
```

Syntax 2

```
CURVe?
```

Arguments

- `<Block>`

  This is the waveform data in binary format. The waveform is formatted as: `#<x><yyy><data><newline>`, where

  `<x>` is the number of y bytes. For example, if <yyy>=500, then <x>=3).

  `<yyy>` is the number of bytes to transfer. If width is 1, then all bytes on the bus are single data points. If width is 2, then all bytes on the bus are 2-byte pairs. If width is 4, then all bytes on the bus are 4-byte pairs. Use the WFMInpre:BYT_Nr command to set the width.

  `<data>` is the curve data.

  `<newline>` is a single byte new line character at the end of the data.

- `<asc curve>`

  This is the waveform data in ASCII format. The format for ASCII data is <NR1>[,<NR1>…], where each <NR1> represents a data point.

Example 1

```
CURVe?
```

This query with ASCII encoding, start and stop of 1 and 10 respectively, and a width set to 1 might return

```
:CURVE 61,62,61,60,60,-59,-59,-58,-58,-59
```

Example 2

```
CURVe <Block>
```

This command sets the format of the waveform data, transferred to and from the oscilloscope, to binary format.

DATa

Description

This command sets or queries the format and location of the waveform data that is transferred with the CURVe command.

Group

Waveform Transfer

Related Commands

CURVe (see page 467), DATa:STARt (see page 473), DATa:STOP (see page 474), DATa:ENCdg (see page 470)

Syntax 1

```
DATa {INIT|SNAp}
```

Syntax 2

```
DATa?
```

Argument

- `INIT`

  This initializes the waveform data parameters to their factory defaults.

- `SNAp`

  This sets DATa:STARt and DATa:STOP to match the current V Bar/Paired/Split cursor positions.

Example 1

```
DATa?
```

This query might return `:DATA:DESTINATION REF1:ENCDG RIBINARY;SOURCE CH1;START 1;STOP 500;WIDTH 1`

Example 2

```
DATa INIT
```

This command initializes the waveform data parameters to their factory defaults.

DATa:DESTination

Description

This command sets or queries the reference memory location for storing waveform data that is transferred into the oscilloscope by the CURVe command.

Group

Waveform Transfer

Related Commands

CURVe (see page 467)

Syntax 1

```
DATa:DESTination REF<x>
```

Syntax 2

```
DATa:DESTination?
```

Argument

- `REF<x>`

  This is the reference where the waveform will be stored. The reference number is specified by x, which ranges from 1 through 4.

Example 1

```
DATa:DESTination?
```

This query might return `:DATA:DESTINATION REF3`, indicating that reference 3 is the currently selected reference memory location for waveform data.

Example 2

```
DATa:DESTination REF1
```

This command requests that incoming waveform data be stored in reference 1.

DATa:ENCdg

Description

This command sets or queries the format of outgoing waveform data. This command is equivalent to setting WFMOutpre:ENCdg, WFMOutpre:BN_Fmt, and WFMOutpre:BYT_Or. Setting the DATa:ENGdg value causes the corresponding WFMOutpre values to be updated and visa versa.

Note: Values are constrained (for outbound data) to the format of the data specified by DATa:SOUrce.

Group

Waveform Transfer

Related Commands

WFMOutpre:ENCdg (see page 494), WFMOutpre:BN_Fmt (see page 496), WFMOutpre:BYT_Or (see page 497)

Syntax 1

```
DATa:ENCdg {ASCIi|FAStest|RIBinary|RPBinary|FPBinary|SRIbinary
|SRPbinary|SFPbinary}
```

Syntax 2

```
DATa:ENCdg?
```

Argument

- `ASCIi`

  This specifies the ASCII representation of signed INT, FLOAT. If ASCII is the value, then :BN_Fmt and :BYT_Or are ignored.

- `FAStest`

  This requests that the data be sent in the fastest possible manner consistent with maintaining accuracy and is interpreted with respect to the first waveform specified in the DATA:SOUrce list.

  :ENCdg will always be BIN, :BYT_Or will always be MSB, but :BN_Fmt and :BYT_Nr will depend on the first DATa:SOUrce waveform. :BN_Fmt will be RI unless the waveform is internally stored as a floating point number, in which case FP format will be used.

- `RIBinary`

  This specifies signed integer data-point representation with the most significant byte transferred first.

  When :BYT_Nr is 1, the range is from -128 through 127. When :BYT_Nr is 2, the range is from -32,768 through 32,767. When :BYT_Nr is 8, then the waveform being queried has been set to Fast Acquisition mode. Center screen is 0. The upper limit is one division above the top of the screen and the lower limit is one division below the bottom of the screen. This is the default argument.

- `RPBinary`

  This specifies the positive integer datapoint representation, with the most significant byte transferred first.

  When :BYT_Nr is 1, the range from 0 through 255. When :BYT_Nr is 2, the range is from 0 to 65,535. When :BYT_Nr is 8, then the waveform being queried has been set to Fast Acquisition mode. Center screen is 127. The upper limit is one division above the top of the screen and the lower limit is one division below the bottom of the screen.

- `FPBinary`

  This specifies the floating point (width = 4) data.

  The range is from -3.4×10^{38} to 3.4×10^{38}. Center screen is 0. The upper limit is one division above the top of the screen and the lower limit is one division below the bottom of the screen.

- `SRIbinary`

  This is the same as RIBinary except that the byte order is swapped, meaning that the least significant byte is transferred first. This format is useful when transferring data to IBM compatible PCs.

- `SRPbinary`

  This is the same as RPBinary except that the byte order is swapped, meaning that the least significant byte is transferred first. This format is useful when transferring data to IBM compatible PCs.

- `SFPbinary`

  This specifies floating point data in IBM format.

DATa and WFMOUTPRE Parameter Settings				
	WFMOUTPRE Settings			
DATa:ENCdg Setting	:ENCdg	:BN_Fmt	:BYT_Or	:BYT_NR
ASCii	ASC	N/A	N/A	1, 2, 4, 8
FAStext	BIN	RI/FP	MSB	1, 2, 4
RIBinary	BIN	RI	MSB	1, 2, 8
RPBinary	BIN	RP	MSB	1, 2, 8
FPBinary	BIN	FP	MSB	4
SRIbinary	BIN	RI	LSB	1, 2, 8
SRPbinary	BIN	RP	LSB	1, 2, 8
SFPbinary	BIN	FP	LSB	4

Example 1

```
DATa:ENCdg?
```

This query might return `:DATa:ENCDG SRPBINARY` for the format of the outgoing waveform data.

Example 2

```
DATa:ENGdg RPBinary
```

This command sets the data encoding format to be a positive integer where the most significant byte is transferred first.

DATa:SOUrce

Description

This command sets or queries the location of waveform data that is transferred from the oscilloscope by the CURVe? Query.

Group

Waveform Transfer

Related Commands

CURVe? (see page 467)

Syntax 1

```
DATa:SOUrce <wfm>[<,><wfm>]
```

Syntax 2

```
DATa:SOUrce?
```

Argument

- `<wfm>`

  This is the location of the waveform data that will be transferred from the oscilloscope to the controller.

Example 1

```
DATa:SOUrce?
```

This query might return `:DATA:SOURCE REF3`, indicating the source for the waveform data that is transferred using a CURVe? query.

Example 2

```
DATa:SOUrce CH1
```

This command specifies that the CH1 waveforms will be transferred in the next CURVe? query.

DATa:STARt

Description

This command sets or queries the starting data point for waveform transfer. This command allows for the transfer of partial waveforms to and from the oscilloscope.

Group

Waveform Transfer

Related Commands

DATa (see page 468), DATa:STOP (see page 474), WFMInpre:NR_Pt (see page 482), WFMOutpre:NR_Pt (see page 499)

Syntax 1

```
DATa:STARt <NR1>
```

Syntax 2

```
DATa:STARt?
```

Argument

- `<NR1>`

  This is the first data point that will be transferred, which ranges from 1 to the record length. Data will be transferred from <NR1> to DATa:STOP or the record length, whichever is less. If <NR1> is greater than the record length, the last data point in the record is transferred.

  When DATa:STOP is less than DATa:STARt, the stop value will equal DATa:STARt + (DATa:STARt - DATa:STOP). For example, if DATa:STARt = 30 and DATa:STOP = 20, then the range of data points for the waveform transfer will equal 30 through 40.

Example 1

```
DATa:STARt?
```

This query might return `:DATA:START 214`, indicating that this is the first waveform data point that will be transferred.

Example 2

```
DATa:STARt 10
```

This command specifies that the waveform transfer will begin with data point 10.

DATa:STOP

Description

This command sets or queries the last data point that will be transferred when using the CURVe? query. This command allows for the transfer of partial waveforms to the controller.

Note: When using the CURVe command, DATa:STOP is ignored and WFMInpre:NR_Pt is used.

Group

Waveform Transfer

Related Commands

CURVe (see page 467), DATa (see page 468), WFMInpre:NR_Pt (see page 482)

Syntax 1

```
DATa:STOP <NR1>
```

Syntax 2

```
DATa:STOP?
```

Argument

- `<NR1>`

  This is the last data point that will be transferred, which ranges from 1 to the record length. If <NR1> is greater than the record length, then data will be transferred up to the record length. If both DATa:STARt and DATa:STOP are greater than the record length, the last data point in the record is returned.

  If you always want to transfer complete waveforms, just set DATa:STARt to 1 and DATa:STOP to the maximum record length.

  When DATa:STOP is less than DATa:STARt, the stop value will equal DATa:STARt + (DATa:STARt - DATa:STOP). For example, if DATa:STARt = 30 and DATa:STOP = 20, then the range of data points for the waveform transfer will equal 30 through 40.

Example 1

```
DATa:STOP?
```

This query might return `:DATA:STOP 14900`, indicating that this is the last waveform data point that will be transferred.

Example 2

```
DATa:STOP 15000
```

This command specifies that the waveform transfer will stop at data point 15000.

WAVFrm?

Description

This query-only command returns WFMOutpre? and CURVe? data for the waveform as specified by the DATA:SOUrce command. This command is equivalent to sending both WFMOutpre? and CURVe?, with the additional provision that the response to WAVFrm? is guaranteed to provide a synchronized preamble and curve.

Group

Waveform Transfer

Related Commands

CURVe (see page 467), DATa:SOUrce (see page 472), WFMOutpre? (see page 493)

Syntax

```
WAVFrm?
```

Example

```
WAVFrm?
```

This query might return:

```
:WFMOUTPRE:BIT_NR 8;BN_FMT RI;BYT_NR 1;BYT_OR MSB;ENCDG
ASC;NR_PT 500;PT_FMT Y;PT_ORDER LINEAR;PT_OFF 0
;XINCR 400.0000E-12;XZERO 0.0000;XUNIT "s";YMULT
4.0000E-3;YOFF 0.0000;YZERO 0.0000;YUNIT "V";WFID "Ch1,
DC coupling, 100.0mV/div, 200.0ns/div, 5000 points,
Sample mode";:CURVE 51,50,51,48,51,48,50,49,51,49,51,48,
51,48,51,49,50,49,50,48,49,49,52,49,49,50,50,48,50,49,
49,49,49,49,50,47,49,47,50,48,49,48,50,48,49,47,49,48,
51,48,49,47,50,48,50,47,51,47,49,48,48,48,50,46,50,46,
48,45,48,47,49,47,49,48,49,48,49,45,49,47,48,46,48,48,
49,45,49,45,47,46,46,46,49,47,48,46,48,46,48,47,47,46,
47,47,48,46,48,45,48,46,47,46,47,47,46,45,46,45,47,47,
47,46,46,44,47,45,45,45,46,43,46,45,46,44,46,45,48,45,
45,44,46,45,46,45,45,45,45,43,45,44,46,44,47,44,44,44,
45,43,44,44,45,44,46,43,46,43,45,44,45,44,45,41,44,43,
45,42,44,43,44,43,44,42,43,42,44,41,44,41,44,42,43,42,
43,42,45,42,43,41,42,41,43,42,44,42,43,42,43,40,42,40,
42,40,42,41,43,41,42,41,42,39,41,41,42,42,43,41,42,39,
41,39,41,40,41,40,41,39,41,38,41,39,41,38,42,39,41,37,
40,38,40,38,39,38,39,38,41,39,39,38,40,37,38,37,38,38,
39,38,40,36,39,37,39,37,39,37,39,36,40,38,38,35,38,36,
38,37,39,37,38,37,38,37,37,36,38,35,38,36,37,36,38,36,
38,35,39,34,36,36,36,35,38,35,36,34,37,34,37,35,37,36,
37,35,37,35,36,34,36,33,36,35,36,34,36,33,34,33,34,34,
34,34,36,33,36,34,36,33,34,33,34,32,34,32,34,33,33,31,
33,32,34,32,33,32,32,31,33,31,31,32,33,31,32,31,32,30,
32,30,32,31,31,30,31,30,31,29,31,29,32,31,31,30,30,29,
30,29,30,30,31,29,30,28,30,29,31,28,30,28,30,29,29,27,
29,28,29,27,30,27,30,26,28,28,29,27,28,27,28,27,28,28,
28,27,28,27,28,26,28,26,28,26,26,25,27,25,28,27,28,26,
25,25,26,25,25,25,25,25,26,25,27,25,25,24,25,25,26,24,
26,24,24,23,25,24,24,22,25,23,24,23,24,22,24,22,23,22,
25,24,24,21,23,22,22,21,24,22,22,22,22,21,22,20,23,21,
21,20
```

WFMInpre?

Description

This query-only command returns the waveform formatting specification to be applied to the next incoming CURVe command data.

Group

Waveform Transfer

Related Commands

CURVe (see page 467), WFMOutpre? (see page 493)

Syntax

```
WFMInpre?
```

Example

```
WFMInpre?
```

This query might return `:WFMINPRE:BIT_NR 8;BN_FMT RI;BYT_NR 1;BYT_OR MSB;ENCDG BIN;NR_PT 500;PT_FMT Y;PT_OFF 0;XINCR 2.0000E-6;XZERO 1.7536E-6;XUNIT "s";YMULT 1.0000E-3;YOFF 0.0000;YZERO 0.0000;YUNIT "V"`

WFMInpre:ENCdg

Description

This command sets or queries the type of encoding for incoming waveform data.

Group

Waveform Transfer

Related Commands

WFMOutpre:ENCdg (see page 494)

Syntax 1

```
WFMInpre:ENCdg {ASC|BIN}
```

Syntax 2

```
WFMInpre:ENCdg?
```

Arguments

- `ASC`

  This specifies that the incoming data is in ASCII format.

- `BIN`

  This specifies that the incoming data is in a binary format whose further interpretation requires knowledge of BYT_NR, BIT_NR, BN_FMT, and BYT_OR.

Example 1

```
WFMInpre:ENCdg?
```

This query might return `:WFMINPRE:ENCDG BIN`, indicating that the incoming waveform data will be in binary format.

Example 2

```
WFMInpre:ENCdg ASC
```

This command sets the format of incoming waveform data to ASCII format.

WFMInpre:BN_Fmt

Description

This command sets or queries the format of binary data for incoming waveforms.

Group

Waveform Transfer

Related Commands

WFMOutpre:BN_Fmt (see page 496)

Syntax 1

```
WFMInpre:BN_Fmt {RI|RP|FP}
```

Syntax 2

```
WFMInput:BN_Fmt?
```

Arguments

- `RI`

  This specifies signed integer datapoint representation.

- `RP`

  This specifies positive integer datapoint representation.

- `FP`

  This specifies single-precision binary floating point representation.

Example 1

```
WFMInpre:BN_Fmt?
```

This query might return `:WFMINPRE:BN_FMT RI`, indicating that the incoming data is currently interpreted as signed integers.

Example 2

```
WFMInpre:BN_Fmt FP
```

This command specifies that incoming data will be interpreted as single-precision binary floating point.

WFMInpre:BYT_Or

Description

This command sets or queries which byte of binary waveform data is transmitted first for incoming waveform data when data points require more than one byte. This specification only has meaning when WFMInpre:ENCdg is set to BIN and WFMInpre:BYT_Nr is greater than 1.

Group

Waveform Transfer

Related Commands

WFMInpre:ENCdg (see page), WFMInpre:BYT_Nr (see page 480), WFMOutpre:BYT_Or (see page 497)

Syntax 1

```
WFMInpre:BYT_Or {LSB|MSB}
```

Syntax 2

```
WFMInpre:BYT_Or?
```

Arguments

- `LSB`

  This specifies that the least significant byte will be transmitted first.

- `MSB`

  This specifies that the most significant byte will be transmitted first.

Example 1

```
WFMInpre:BYT_Or?
```

This query might return `:WFMINPRE:BYT_OR LSB`, indicating that the least significant incoming CURVe data byte will be transmitted first.

Example 2

```
WFMInpre:BYT_Or MSB
```

This command sets the most significant incoming byte of incoming waveform data to be transmitted first.

WFMInpre:BYT_Nr

Description

This command sets or returns the binary field data width for the first ordered waveform, as specified by the DATa:DESTination command. This specification is only meaningful when WFMInpre:ENCdg is set to BIN and WFMInpre:BN_Fmt is set to either RI or RP.

Group

Waveform Transfer

Related Commands

DATa:DESTination (see page 469), WFMInpre:BN_Fmt (see page 478), WFMInpre:ENCdg (see page 477), WFMInpre:BIT_Nr (see page 481), WFMOutpre:BYT_Nr (see page 498)

Syntax 1

```
WFMInpre:BYT_Nr <NR1>
```

Syntax 2

```
WFMInpre:BYT_Nr?
```

Argument

- `<NR1>`

  This is the number of bytes per data point and can be 1, 2 (RI, RP) or 4 (FP).

Example 1

```
WFMInpre:BYT_Nr?
```

This query might return `:WFMINPRE:BYT_NR 2`, indicating that there are 2 bytes per incoming waveform data point.

Example 2

```
WFMInpre:BYT_Nr 1
```

This command sets the number of bytes per incoming waveform data point to 1, which is the default setting.

WFMInpre:BIT_Nr

Description

This command sets or returns the number of bits per binary waveform point for the waveform, as specified by the DATa:DESTination command. This specification is only meaningful when WFMInpre:ENCdg is set to BIN.

Group

Waveform Transfer

Related Commands

DATa:DESTination (see page 469), WFMInpre:ENCdg (see page 477), WFMInpre:BYT_Nr (see page 480), WFMOutpre:BIT_Nr (see page 495)

Syntax 1

```
WFMInpre:BIT_Nr <NR1>
```

Syntax 2

```
WFMInpre:BIT_Nr?
```

Argument

- `<NR1>`

  This number of bytes per data point can be 8, 16 (RI, RP) or 32 (FP).

Example 1

```
WFMInpre:BIT_Nr?
```

This query might return `:WFMINPRE:BIT_NR 8`, indicating that incoming RI or RP binary format data uses 8 bits per waveform point.

Example 2

```
WFMInpre:BIT_Nr 16
```

This command sets the number of bits per waveform point to 16, for incoming RI and RP binary format data.

WFMInpre:NR_Pt

Description

This command sets or returns the number of points that are in the transmitted waveform record.

Group

Waveform Transfer

Related Commands

WFMInpre:PT_Fmt (see page 483), WFMOutpre:NR_Pt? (see page 499)

Syntax 1

```
WFMInpre:NR_Pt <NR1>
```

Syntax 2

```
WFMInpre:NR_Pt?
```

Argument

- `<NR1>`

  If WFMInpre:PT_Fmt is set to Y, this is the number of data points; If WFMInpre:PT_Fmt is set to ENV, this is the number of max-min pairs.

Example 1

```
WFMInpre:NR_Pt?
```

This query might return `:WFMINPRE:NR_PT 8000`, indicating that there are 8000 data points in the transmitted waveform record.

Example 2

```
WFMInpre:NR_Pt 5000
```

This command specifies that 5000 data points will be expected.

WFMInpre:PT_Fmt

Description

This command sets or queries the point format of the incoming waveform data. Regardless of the argument used, the scale, offset, etc. are interpreted similarly. When ENV is used, waveform data interpreted over the min/max pair; when Y is used, it is interpreted over a single point.

Group

Waveform Transfer

Related Commands

WFMOutpre:PT_Fmt? (see page 500)

Syntax 1

```
WFMInpre:PT_Fmt {ENV|Y}
```

Syntax 2

```
WFMInpre:PT_Fmt?
```

Arguments

- `ENV`

  This specifies that the waveform is transmitted as maximum and minimum point pairs. Only Y values are explicitly transmitted. Absolute coordinates are given by:

  Xn = XZEro + XINcr (n-PT_Off)

  Ynmax = YZEro + YMUlt (ynmax - YOFf)

  Ynmin = YZEro + YMUlt (ynmin - YOFf)

- `Y`

  This specifies a normal waveform where one ASCII or binary data point is transmitted for each point in the waveform record. Only Y values are explicitly transmitted. Absolute coordinates are given by:

  Xn = XZEro + XINcr (N-PT_Off)

  Yn = YZEro + YMUlt (Yn - YOFf)

Example 1

```
WFMInpre:PT_Fmt ENV
```

This command sets the incoming waveform data point format to enveloped.

Example 2

```
WFMInpre:PT_Fmt?
```

This query might return `:WFMINPRE:PT_FMT ENV`, indicating that the waveform is transmitted as maximum and minimum point pairs.

WFMInpre:PT_Off

Description

This command specifies or returns the trigger point, within the waveform record, for the reference waveform specified by the DATa:DESTination command.

Group

Waveform Transfer

Related Commands

DATa:DESTination (see page 469), DATa:STARt (see page 473), WFMOutpre:PT_Off (see page 501)

Syntax 1

```
WFMInpre:PT_Off <NR1>
```

Syntax 2

```
WFMInpre:PT_Off?
```

Argument

- `<NR1>`

  This is (record length -1) to record length and is the position of the data point immediately following the actual trigger. <NR1> is expressed relative to DATa:STARt.

Example 1

```
WFMInpre:PT_Off 0
```

This command specifies that the trigger point is the first point in the waveform record, which is the default.

Example 2

```
WFMInpre:PT_Off?
```

This query might return `:WFMINPRE:PT_OFF 0`, indicating the incoming waveform trigger point.

WFMInpre:WFId

Description

This command (no query form) accepts but ignores the argument. This command is provided only to allow a waveform extracted from the oscilloscope to be more easily imported.

Group

Waveform Transfer

Related Commands

DATa:DESTination (see page 469), WFMInpre:BN_Fmt (see page 478), WFMInpre:ENCdg (see page 477), WFMOutpre:WFid (see page 503)

Syntax

```
WFMInpre:WFId <String>
```

Argument

- `<String>`

  This must be a valid 488.2 string (but the contents are ignored).

Example

```
WFMInpre:WFId "Ch1, DC coupling, 2.000V/div, 400.0ns/div, 500 points, Sample mode"
```

This is a syntactically correct command.

WFMInpre:XINcr

Description

This command sets or queries the horizontal interval between incoming waveform points in units specified by WFMInpre:XUNit.

Group

Waveform Transfer

Related Commands

WFMInpre:XUNit (see page 488), WFMOutpre:XINcr (see page 505)

Syntax 1

```
WFMInpre:XINcr <NR3>
```

Syntax 2

```
WFMInpre:XINcr?
```

Argument

- `<NR3>`

  This is the horizontal interval representation.

Example 1

```
WFMInpre:XINcr?
```

This query might return `:WFMINPRE:XINCR 1.0000E-3`, indicating that (assuming that WFMInpre:XUNit is set to "s") there is a 1 ms interval between incoming waveform points.

Example 2

```
WFMInpre:XINcr 3E-3
```

This command sets the interval between Incoming waveform points to 3 ms.

WFMInpre:XZEro

Description

This command sets or queries the (subsample) time between the trigger sample (designated by PT_OFF) and the occurrence of the actual trigger of the incoming waveform. This value is used to compute TTOFF for the incoming waveform, and is expressed in terms of WFMInpre:XUNit.

Group

Waveform Transfer

Related Commands

WFMInpre:PT_Off (see page 484), WFMInpre:XINcr (see page 486), WFMInpre:XUnit (see page 488), WFMOutpre:XZEro (see page 506)

Syntax 1

```
WFMInpre:XZEro <NR3>
```

Syntax 2

```
WFMInpre:XZEro?
```

Argument

- `<NR3>`

  This is value that ranges from -WFMInpre:XINcr to 0.

Example 1

```
WFMInpre:XZEro?
```

This query might return `:WFMINPRE:XZEro 7.5000E-6`, indicating that the trigger occurs 7.5 μs before the sample designated by WFMInpre:PT_Off.

Example 2

```
WFMInpre:XZEro 5.7E-6
```

This command specifies that the trigger actually occurred 5.7 μs before the sample designated by WFMInpre:PT_Off .

WFMInpre:XUNit

Description

This command sets or returns the horizontal units of the incoming waveform.

Group

Waveform Transfer

Related Commands

WFMOutpre:XUNit (see page 507)

Syntax 1

```
WFMInpre:XUNit <String>
```

Syntax 2

```
WFMInpre:XUNit?
```

Argument

- `<String>`

  This contains a maximum of three alpha characters that represent the horizontal unit of measure for the incoming waveform.

Example 1

```
WFMInpre:XUNit?
```

This query might return `:WFMINPRE:XUNIT "s"`, indicating that the horizontal units for the incoming waveform are expressed in seconds.

Example 2

```
WFMInpre:XUNit "Hz"
```

This command specifies that the horizontal units for the incoming waveform are hertz.

WFMInpre:YMUlt

Description

This command sets or queries the vertical scale factor (in units/digitizing level) for the reference waveform, specified by DATa:DESTination, upon a CURVe command.

Group

Waveform Transfer

Related Commands

DATa:DESTination (see page 469), WFMInpre:BYT_Nr (see page 480), WFMInpre:YUNit (see page 488)

Syntax 1

```
WFMInpre:YMUlt <NR3>
```

Syntax 2

```
WFMInpre:UMUlt?
```

Argument

- `<NR3>`

  This is the vertical scale factor per digitizing level of the incoming waveform points.

Example 1

```
WFMInpre:YMUlt?
```

This query might return `:WFMINPRE:YMULT 40.0000E-3`, indicating that the vertical scale is 40 mV/digitizing level (1V/div).

Example 2

```
WFMInpre:YMUlt 20E-3
```

This command specifies that (if WFMInpre:YUNit is "V" and WFMInpre:BYT_Nr is 1) the vertical scale is 20 mV/digitizing level (500 mV/div).

WFMInpre:YOFf

Description

This command sets or queries the vertical position of the incoming waveform in digitizing levels. Variations in this number are analogous to changing the vertical position of the waveform. For those formats in which WFMInpre:BYT_Nr is important (all nonfloating point formats), this command must take the location of the binary point implied by BYT_NR into consideration.

Group

Waveform Transfer

Related Commands

WFMInpre:BYT_Nr (see page 480), WFMInpre:YMUlt? (see page 489), WFMOutpre:YOFf? (see page 509)

Syntax 1

```
WFMInpre:YOFf <NR3>
```

Syntax 2

```
WFMInpre:YOFf?
```

Argument

- `<NR3>`

  This is the vertical offset in digitizing levels.

Example 1

```
WFMInpre:YOFf?
```

This query might return `:WFMINPRE:YOFF 25`, indicating the vertical position of the incoming waveform in digitizing levels.

Example 2

```
WFMInpre:YOFf 50
```

This command specifies that the zero reference for the incoming waveform is 50 digitizing levels (2 divisions) above the center of the data range.

WFMInpre:YUNit

Description

This command sets or queries the vertical units of the incoming waveform.

Group

Waveform Transfer

Related Commands

WFMOutpre:YUNit (see page 510)

Syntax 1

```
WFMInpre:YUNit <String>
```

Syntax 2

```
WFMInpre:YUNit?
```

Argument

- `<String>`

  This contains a maximum of three alpha characters that represent the vertical unit of measure for the incoming waveform.

Example 1

```
WFMInpre:YUNit?
```

This query might return `:WFMINPRE:YUNIT "s"`, indicating the vertical units for the incoming waveform are expressed in seconds.

Example 2

```
WFMInpre:YUNit "Pa"
```

This command specifies that the vertical units for the incoming waveform are pascal.

WFMInpre:YZEro

Description

This command sets or queries the offset of the incoming waveform in units specified by WFMInpre:YUNit. Variations in this number are analogous to changing the vertical offset of the waveform.

Group

Waveform Transfer

Related Commands

WFMInpre:YUNit (see page 491), WFMOutpre:YZEro (see page 511)

Syntax 1

```
WFMInpre:YZEro <NR3>
```

Syntax 2

```
WFMInpre:YZEro?
```

Argument

- `<NR3>`

  This is of the offset in YUNits,

Example 1

```
WFMInpre:YZEro?
```

This query might return `:WFMINPRE:YZEro 7.5000E-6`, indicating that the zero reference for the incoming waveform is 7.5 μV below the center of the data range (given that WFMInpre:YUNit is set to V).

Example 2

```
WFMInpre:YZEro 1.5E+0
```

This command specifies that the zero reference for the incoming waveform is 1.5 V below the center of the data range (given that WFMInpre:YUNit is set to V).

WFMOutpre?

Description

This query-only command returns the waveform formatting data for the waveform specified by the DATa:SOUrce command. The preamble components are considered to be of two types; formatting and interpretation. The formatting components are: ENCdg, BN_Fmt, BYT_Or, BYT_Nr, BIT_Nr. The interpretation components are derived from the DATa:SOUrce specified waveform.

Group

Waveform Transfer

Syntax

```
WFMOutpre?
```

Example

```
WFMOutpre:?
```

This query might return

```
:WFMOUTPRE:BIT_NR 8;BN_FMT RI
;BYT_NR 1;BYT_OR MSB;ENCDG BIN;NR_PT 500;PT_FMT Y
;PT_ORDER LINEAR;PT_OFF 0;XINCR 8.0000E-9
;XZERO 4.8794E-9;XUNIT "s";YMULT -2000.0000E-3
;YOFF -4999.9995E-3;YZERO 0.0000;YUNIT "V"
;WFID "Ch1, DC coupling, 2.000V/div, 400.0ns/div,
500 points, Sample mode"
```

WFMOutpre:ENCdg

Description

This command sets and queries the type of encoding for outgoing waveforms.

Group

Waveform Transfer

Related Commands

DATa:ENCdg (see page 470), WFMOutpre:BYT_Nr (see page 498), WFMOutpre:BYT_Or (see page 497), WFMOutpre:BIT_Nr (see page 495), WFMOutpre:BN_Fmt (see page 496)

Syntax 1

```
WFMOutpre:ENCdg {ASC|BIN}
```

Syntax 2

```
WFMOutpre:ENCdg?
```

Argument

- `ASC`

  This specifies that the outgoing data is to be in ASCII format. Waveforms internally stored as integers will be sent as <NR1> numbers while those stored as floats will be sent as <NR3> numbers.

- `BIN`

  This specifies that outgoing data is to be in a binary format whose further specification is determined by WFMOutpre:BYT_Nr, WFMOutpre:BIT_Nr, WFMOutpre:BN_Fmt and WFMOutpre:BYT_Or.

Example 1

```
WFMOutpre:ENCdg?
```

This query might return `:WFMOUTPRE:ENCDG BIN`, indicating that outgoing waveform data will be sent in binary format.

Example 2

```
WFMOutpre:ENCdg ASC
```

This command specifies that the outgoing waveform data will be sent in ASCII format.

WFMOutpre:BIT_Nr

Description

This command sets and returns the number of bits per waveform point that outgoing waveforms contain, as specified by the DATa:SOUrce command. Note that values will be constrained according to the underlying waveform data. This specification is only meaningful when WFMOutpre:ENCdg is set to BIN and WFMOutpre:BN_Fmt is set to either RI or RP.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472), WFMOutpre:BN_Fmt (see page 478), WFMOutpre:ENCdg (see page 494)

Syntax 1

```
WFMOutpre:BIT_Nr <NR1>
```

Syntax 2

```
WFMOutpre:BIT_Nr?
```

Argument

- `<NR1>`

  This number of bits per data point can be 8, 16, 32 or 64.

Example 1

```
WFMOutpre:BIT_Nr?
```

This query might return `:WFMOUTPRE:BIT_NR 8`, indicating that outgoing RI or RP binary; format data uses 8 bits per waveform point.

Example 2

```
WFMOutpre:BIT_Nr 16
```

This command sets the number of bits per waveform point to 16, for incoming RI and RP binary format data.

WFMOutpre:BN_Fmt

Description

This command sets or queries the format of binary data for outgoing waveforms specified by the DATa:SOUrce command.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472)

Syntax 1

```
WFMOutpre:BN_Fmt {RI|RP|FP}
```

Syntax 2

```
WFMOutput:BN_Fmt?
```

Arguments

- `RI`

  This specifies signed integer datapoint representation.

- `RP`

  This specifies positive integer datapoint representation.

- `FP`

  This specifies single-precision binary floating point.

Example 1

```
WFMOutpre:BN_Fmt?
```

This query might return `:WFMOUTPRE:BN_FMT RI`, indicating that the outgoing waveform data is currently in signed integer format.

Example 2

```
WFMOutpre:BN_Fmt FP
```

This command specifies that outgoing waveform data will be in single-precision binary floating point format.

WFMOutpre:BYT_Or

Description

This command sets or queries which byte of binary waveform data is transmitted first, during a waveform data transfer, when data points require more than one byte. This specification only has meaning when WFMOutpre:ENCdg is set to BIN.

Group

Waveform Transfer

Related Commands

WFMOutpre:ENCdg (see page 494)

Syntax 1

```
WFMOutpre:BYT_Or {LSB|MSB}
```

Syntax 2

```
WFMOutpre:BYT_Or?
```

Arguments

- `LSB`

  This specifies that the least significant byte will be transmitted first.

- `MSB`

  This specifies that the most significant byte will be transmitted first.

Example 1

```
WFMOutpre:BYT_Or?
```

This query might return `:WFMOUTPRE:BYT_OR LSB`, indicating that the least significant data byte will be transmitted first.

Example 2

```
WFMOutpre:BYT_Or MSB
```

This command sets the most significant outgoing byte of waveform data to be transmitted first.

WFMOutpre:BYT_Nr

Description

This command sets or returns the binary field data width for the waveform specified by the DATa:SOUrce command. Note that values will be constrained according to the underlying waveform data. This specification is only meaningful when WFMOutpre:ENCdg is set to BIN and WFMOutpre:BN_Fmt is set to either RI or RP.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472), WFMOutpre:BN_Fmt (see page 496), WFMOutpre:ENCdg (see page 494)

Syntax 1

```
WFMOutpre:BYT_Nr <NR1>
```

Syntax 2

```
WFMOutpre:BYT_Nr?
```

Argument

- `<NR1>`

  This is the number of bytes per data point and can be 1, 2, 4 or 8. A value of 1 or 2 bytes per waveform point indicates channel data; 4 bytes per waveform point indicate math data; 8 bytes per waveform point indicate pixel map (DPO) data.

Example 1

```
WFMOutpre:BYT_Nr?
```

This query might return `:WFMOUTPRE:BYT_NR 2`, indicating that there are 2 bytes per outgoing waveform data point.

Example 2

```
WFMOutpre:BYT_Nr 1
```

This command sets the number of bytes per outgoing waveform data point to 1, which is the default setting.

WFMOutpre:NR_Pt?

Description

This query-only command returns the number of points for the DATa:SOUrce waveform that will be transmitted in response to a CURVe? query.

Group

Waveform Transfer

Related Commands

CURVe? (see page 467), DATa:SOUrce (see page 472), DATa:STARt (see page 473), DATa:STOP (see page 474)

Syntax

```
WFMOutpre:NR_Pt?
```

Example

```
WFMOutpre:NR_Pt?
```

This query might return `:WFMOUTPRE:NR_PT 5000`, indicating that there are 5000 data points to be sent.

WFMOutpre:PT_Fmt?

Description

This query-only command returns the point format for the waveform specified by the DATa:SOUrce command. The format specifies a set of equations describing how the scale factors in the preamble are used to give meaning to the CURVe data points.

An error is reported if the DATa:SOUrce waveform does not exist.

Group

Waveform Transfer

Related Commands

CURVe? (see page 467), DATa:SOUrce (see page 472)

Syntax

```
WFMOutpre:PT_Fmt?
```

Example

```
WFMOutpre:PT_Fmt?
```

This query might return `:WFMOutpre:PT_Fmt ENV`, indicating that the waveform data are a series of max-min pairs.

WFMOutpre:PT_Off?

Description

This query-only command returns the trigger point relative to DATa:STARt for the waveform specified by the DATa:SOUrce command.

Note: This value is the point immediately following the actual trigger.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472), DATa:STARt (see page 473), WFMOutpre:XZEro? (see page 506)

Syntax

```
WFMOutpre:PT_Off?
```

Example

```
WFMOutpre:PT_Off?
```

This query might return `:WFMOUTPRE:PT_OFF 251`, specifying that the trigger actually occurred between points 250 and 251.

WFMOutpre:PT_OR?

Description

This query-only command specifies whether the source waveform is Fast Acquisition. A Fast Acquisition waveform is stored as a 200 (vertical) by 500 (horizontal) point bitmap. Each point represents display intensity for that screen location. Only CURVe? query functions are allowed on Fast Acquisition waveforms.

When the WFMOutpre:PT_OR query returns Column, this indicates that the source is a Fast Acquisition waveform (and that each of 500 possible horizontal columns being transmitted contains 200 vertical points). When the WFMOutpre:PT_OR? query returns Linear, this indicates that the source is not a Fast Acquisition waveform (and that each horizontal column being sent contains only one vertical point). Note that waveform points are transmitted in the following order: top to bottom, then left to right.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472)

Syntax

```
WFMOutpre:PT_OR?
```

Example 1

```
WFMOutpre:PT_OR?
```

This query might return `:WFMOUTPRE:PT_OR COL`, specifying that the waveform designated by the DATa:SOUrce waveform is a Fast Acquisition waveform.

Example 2

```
WFMOutpre:PT_OR?
```

This query might return `:WFMOUTPRE:PT_OR LINEAR` , specifying that the source pointed to is a non-Fast Acquisition waveform.

WFMOutpre:WFId?

Description

This query-only command returns a string describing several aspects of the acquisition parameters for the waveform specified by the DATa:SOUrce command.

An error is reported if the DATa:SOUrce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472)

Syntax

```
WFMOutpre:WFId?
```

Outputs

<string> comprises the following comma-separated fields:

Waveform Suffixes		
Field	**Description**	**Examples**
Source	Source identification string as it appears in the front panel scale factor readouts.	"Ch1-4" "Math1-3" "Ref1-4"
Coupling	A string describing the vertical coupling of the waveform (the Source 1 waveform in the case of Dual Waveform Math	"AC coupling" "DC coupling" "GND coupling"
Vert Scale	A string giving the vertical scale factor of the unzoomed waveform. The numeric portion will always be 4 digits. The examples cover all known internal units.	"100.0 mV/div" "20.00 dB/div" "45.00 deg/div "785.4 mrad/div" "500.0 uVs/div" "10.00 kV/s/div" "200.0 mV/div" "50.00 unk/div"
Horiz Scale	A string giving the horizontal scale factor of the unzoomed waveform. The numeric portion will always be four 4 digits. The examples cover all known internal units.	"100.0 ms/div" "10.00 kHz/div" "50.00 c/div"
Record Length	A string stating the number of waveform points available in the entire record. The numeric portion is given as an integer.	"500 points" "500000 points"
Acquisition Mode	A string describing the mode used to acquire the waveform.	"Sample mode" "Pk Detect mode" "Hi Res mode" "Envelope mode" "Average mode"
Primary Reference Offset	A string specifying the delta between the Primary Reference (typically, the A trigger) and the CURVe? zero reference location identified by a combination of PT_Off and XZEro in units of XUNits. For example, in Triggerable After Delay, this number would be the actual time between the A and B trigger. In the event that this number is not meaningful, the string will be exactly "0".	"57.2345 ms" "87.3 Hz" "0"

Example

```
WFMOutpre:WFId?
```

This query might return :WFMOUTPRE:WFID "Ch1, DC coupling, 100.0mVolts/div,500.0µs/div,500 points, Hi Res mode"

WFMOutpre:XINcr?

Description

This query-only command returns the horizontal point spacing in units of WFMOutpre:XUNit for the waveform specified by the DATa:SOUrce command. This value corresponds to the sampling interval.

An error is reported if the DATa:SOUrce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472), WFMOutpre:XUNit? (see page 507)

Syntax

```
WFMOutpre:XINcr?
```

Example

```
WFMOutpre:XINcr?
```

This query might return `:WFMOUTPRE:XINCR 10.0000E-6`, indicating that the horizontal sampling interval was 10 μs/point (500 μs/div).

WFMOutpre:XZEro?

Description

This query-only command returns the (subsample) time between the trigger sample (designated by PT_OFF) and the occurrence of the actual trigger for the waveform specified by the DATa:SOUrce command. This value is in units of WFMOutpre:XUNit.

An error is reported if the DATa:SOUrce waveform does not exist.

Note: During *stopped state operation* (i.e., ACQuire:STATE OFF), this is the only preamble that changes on each acquisition. If a query is run during *steady state operation* (i.e., all control changes have settled and triggers are arriving on a regular basis), the XZEro value of the *last* stopped state is returned.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472), WFMOutpre:XUNit? (see page 507)

Syntax

```
WFMOutpre:XZEro?
```

Example

```
WFMOutpre:XZEro?
```

This query might return `:WFMOUTPRE:XZERO 5.6300E-9`, indicating that the trigger actually occurred 5.63 ns before the trigger sample.

WFMOutpre:XUNit?

Description

This query-only command returns the horizontal units for the waveform specified by the DATa:SOUrce command.

An error is reported if the DATa:SOUrce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472)

Syntax

```
WFMOutpre:XUNit?
```

Example

```
WFMOutpre:XUNit?
```

This query might return `:WFMOUTPRE:XUNIT "Hz"`, indicating that the horizontal units for the waveform are measured in hertz.

WFMOutpre:YMUlt?

Description

This query-only command returns the vertical scale factor per digitizing level in units specified by the WFMOutpre:YUNit command for the waveform specified by the DATa:SOUrce command. For those formats in which WFMOutpre:BYT_Nr is important (all nonfloating point formats), WFMOutpre:YMUlt? must take the location of the binary point implied by BYT_NR into consideration.

An error is reported if the DATa:SOUrce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472), WFMOutpre:BYT_Nr (see page 498), WFMOutpre:YUNit? (see page 510)

Syntax

```
WFMOutpre:YMUlt?
```

Example

```
WFMOutpre:YMUlt?
```

This query might return `:WFMOUTPRE:YMULT 4.0000E-3`, indicating that the vertical scale for the corresponding waveform was 100 mV/div.

WFMOutpre:YOFf?

Description

This query-only command returns the vertical offset in digitized levels for the waveform specified by the DATa:SOUrce command. For those formats in which BYT_NR is important (all non-floating point formats), this command must take the location of the binary point implied by WFMOutpre:BYT_Nr into consideration.

An error is reported if the DATa:SOUrce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472), WFMOutpre:BYT_Nr (see page 498)

Syntax

```
WFMOutpre:YOFf?
```

Example

```
WFMOutpre:YOFf?
```

This query might return `:WFMOUTPRE:YOFF -50.0000E+0`, indicating that the position indicator for the waveform was 50 digitizing levels (2 divisions) below center screen.

WFMOutpre:YUNit?

Description

This query-only command returns the vertical units for the waveform specified by the DATa:SOUrce command.

An error is reported if the DATa:SOUrce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472)

Syntax

```
WFMOutpre:YUNit?
```

Example

```
WFMOutpre:YUNit?
```

This query might return `:WFMOUTPRE:YUNIT "dB"`, indicating that the vertical units for the waveform are measured in decibels.

WFMOutpre:YZEro?

Description

This query-only command returns the vertical offset in units specified by WFMOutpre:YUNit for the waveform specified by the DATa:SOUrce command.

An error is reported if the DATa:SOUrce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATa:SOUrce (see page 472), WFMOutpre:YUNit? (see page 510)

Syntax

```
WFMOutpre:YZEro?
```

Example

```
WFMOutpre:YUNit?
```

This query might return `:WFMOUTPRE:YZERO -100.0000E-3`, indicating that vertical offset was set to -100 mV.