14 Bit and 16 Bit
Digital Storage Oscilloscope

C-Class Oscilloscope
Models ZT412VXI

User's Manual: 0004-000053
Revision 2a

January 9, 2008

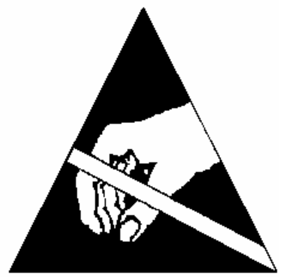
Contact

ZTEC Instruments
7715 Tiburon Street NE
Albuquerque, NM 87109

Telephone: (505) 342-0132
Fax: (505) 342-0222
Web Site: www.ztec-inc.com

ZTEC Instruments, Inc. welcomes your comments on this manual. All manuals are thoroughly reviewed before distribution. We are, however, grateful for any comments from our users which will further help to improve the content and quality of our documents.

Copyright



Handling Precautions for Electronic Devices Subject to Damage by Static Electricity

This instrument is susceptible to Electronic Static Discharge (ESD) damage. When transporting, place the instrument or module in conductive (anti-static) envelopes or carriers. Open only at an ESD-approved work surface. An ESD safe work surface is defined as follows:

- The work surface <u>must</u> be conductive and reliably connected to an earth ground with a safety resistance of approximately 250 kilohms.

- The surface <u>must</u> NOT be metal. A resistance of 30–300 kilohms per square inch is suggested.

Ground the frame of any line-powered equipment, chassis, test instruments, lamps, soldering irons, etc., directly to the earth ground. To avoid shorting out the safety resistance, ensure that the grounded equipment has rubber feet or other means of insulation from the work surface.

Avoid placing tools or electrical parts on insulators. Do NOT use any hand tool that can generate a static charge, such as a non-conductive plunger-type solder sucker. Use a conductive strap or cable with a wrist cuff to reliably ground to the work surface. The cuff <u>must</u> make electrical contact directly with the skin; do NOT wear it over clothing.

Note: Resistance between the skin and the work surface is typically 250 kilohms to 1 megohm using a commercially-available personnel grounding device.

Avoid circumstances that are likely to produce static charges, such as wearing clothes of synthetic material, sitting on a plastic-covered stool (especially when wearing woolen material), combing the hair, or making extensive pencil erasures. These circumstances are most significant when the air is dry.

When testing static sensitive devices, ensure DC power is ON before, during, and after application of test signals. Ensure all pertinent voltages are switched OFF while circuit boards or components are removed or inserted.

Revision History

Rev	Date	Section	Description
1	04-24-06	All	Initial Release
2	04-30-07	All	Bug fixes and C-Class release updates
2A	01-09-08	All	Clean up and spelling corrections

Table of Contents

List of Figures

List of Tables

Introduction

Description

The Model ZT412VXI (See Figure 1) is a 14-bit or 16-bit Digital Storage Oscilloscope. It is built around a high speed, deep memory sequencer and embedded TMS320VC5409 Digital Signal Processor (DSP) housed within a single-wide C-size VXIbus instrument. Together with the host processor and software, it provides a simple powerful way to capture and analyze wide bandwidth analog, IF, and low frequency RF signals.

Initiated by trigger events from both internal and external sources, the ZT412VXI digitizes signals in user-selectable record sizes. Analog signal processing allows selectable four channel operation, input impedance, AC or DC coupling, and input signal gain and offset. Waveform records are transferred from the digitizer into DSP memory, which can then be accessed from the VXIbus. Built-in DSP functions provide a variety of signal analysis and signal manipulation resources. All scope operations are controlled from an intuitive, software-based user interface running on the VXIbus host processor.

Product Options and Part Numbers

Part Number	Maximum Sample Rate	Sampling Resolution	Memory	Manual Naming
ZT412VXI-20	500 MS/s	14-bit	1 MSample	ZT412-2X
ZT412VXI-21	500 MS/s	14-bit	16 MSample	ZT412-2X
ZT412VXI-50	400 MS/s	16-bit	1 MSample	ZT412-5X
ZT412VXI-51	400 MS/s	16-bit	16 MSample	ZT412-5X

Table 1: ZT412VXI Product Options

Table 1 shows the product options for the ZT412VXI that define maximum sample rate, bandwidth and memory size. The first digit of the option field defines the maximum sample rate between 500 and 400 million samples-per-second (MS/s) and the analog-to-digital converter (ADC) sampling resolution of either 14 bits or 16 bits. The second digit of the option field selects the memory size of either 1 or 16 million samples (MS). Within this manual, the entire ZT412VXI product line shall hereafter be designated as ZT412. When referring to specific product options, this manual shall use the naming convention shown in the rightmost column of Table 1.

Figure 1: Photo of the ZT412VXI

Front Panel

A diagram of the ZT412VXI front panel is shown in Figure 2. Table 2 lists the front panel connector functions.

Label	**Description**
Channel 1	Channel 1 input signal (BNC Connector)
Channel 2	Channel 2 input signal (BNC Connector)
Channel 3	Channel 3 Input Signal (BNC Connector)
Channel 4	Channel 4 input signal (BNC Connector)
REF OUT	Reference Output (BNC Connector)
TRG IN	External Trigger Input (BNC Connector)
CLK IN	External Clock Input (BNC Connector)
ARM IN	External Arm Input (BNC Connector)

Table 2: ZT412VXI Front Panel Connectors

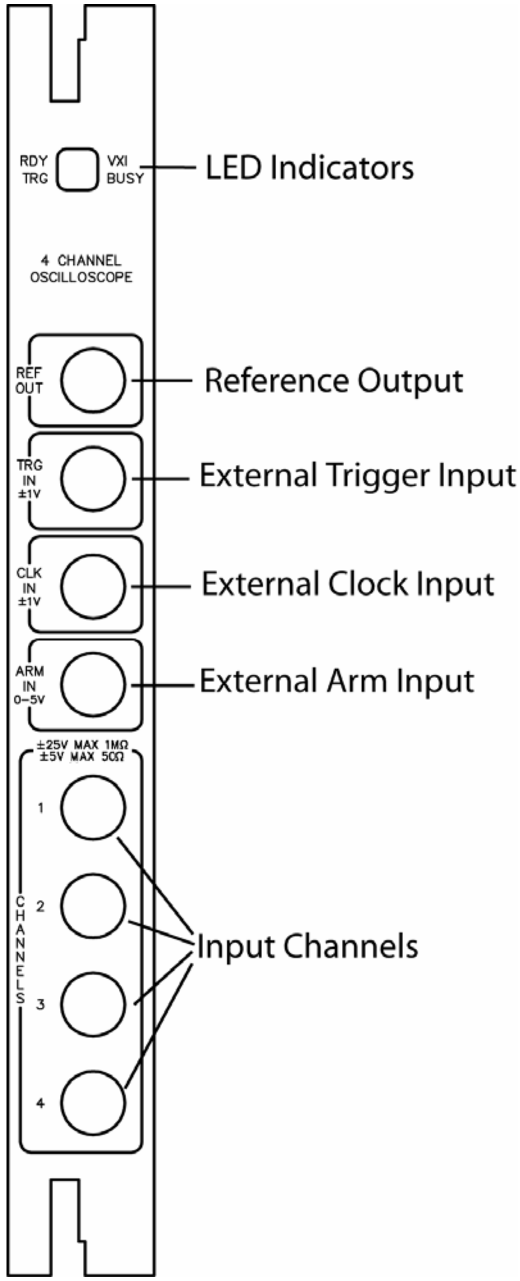


Figure 2: ZT412VXI Front Panel

Additional Resources

ZTEC Instruments, Inc. offers several hardware and software resources to use with the ZT4610 series. Please visit the website www.ztecinstruments.com for the latest information and versions. Detailed information is also available in the individual manuals. Resources include: ZScope™ C-Class soft front panel, instrument drivers, cables and probes.

Functionality and Operation

Functional Block Diagram

The functional blocks of the ZT412 are shown in Figure 3. Analog inputs are conditioned and digitized by the Input Channels. The acquisition time base record size and record placement with regard to the trigger event are configured by the Sweep Controls. Triggers are conditioned and selected by the Trigger and Arm Controls. Averager controls enable the instrument to capture the applied signal multiple times to create the resulting waveform record. Once captured, waveforms are manipulated and analyzed by the Calculate Controls. Waveform measurements, returning data from the ZT412, are handled by the Measure Controls. Status reporting and system utilities are handled by Utilities and Status Reporting. And finally, the data is stored for use and retrieval, and referenced by Waveform and Reference.

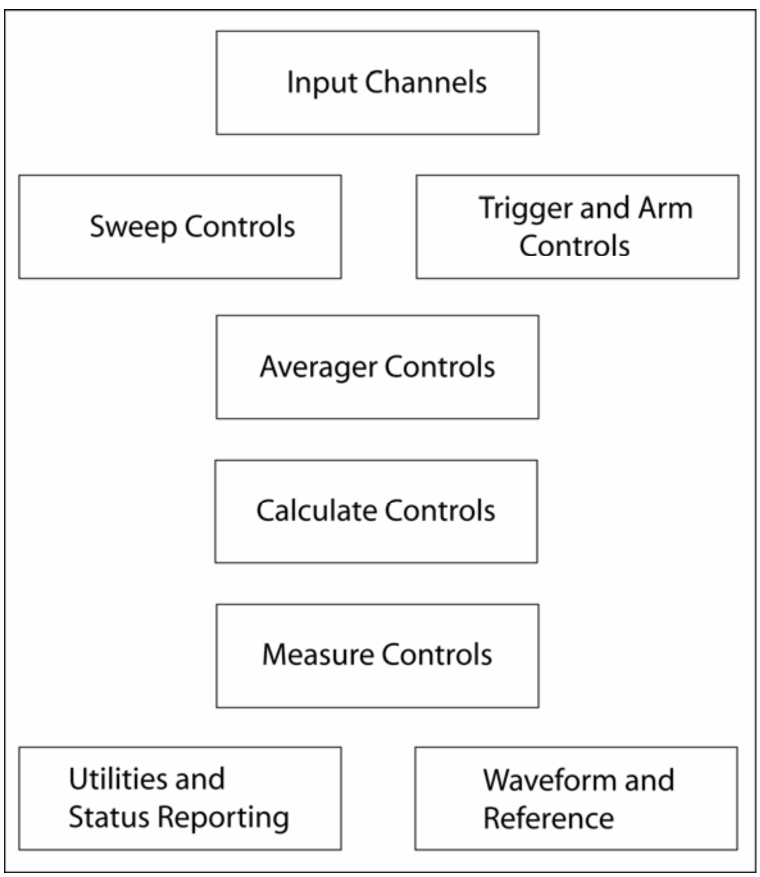


Figure 3: ZT412 Block Diagram

Data Flow

The data flow of the ZT412 is shown in Figure 4. Raw data is input through the four input channels (See *Input Channels* below). Next, the data flows to the Averager, which enables the instrument to acquire multiple waveforms to create the resulting waveform record. The ZT412 has four types of available acquisition: normal, average, envelope, and equivalent-time (See *Averager Controls*). New waveforms can now be mathematically created in the ZT412. The unit has two calculate channels, each capable of a 32K maximum waveform size (See *Calculate Controls*). Scalar measurements are possible using the *Measure Controls*. The ZT412 is capable of providing measurements using the following methods: Entire Waveform, Gated by Time, and Gated by Points. The ZT412 can save and download up to 4 reference waveforms. The reference waveforms, REF1–4, are stored in non-volatile Flash memory and are maintained when the unit is powered off. These waveforms are limited to record sizes of 32 kSamples.

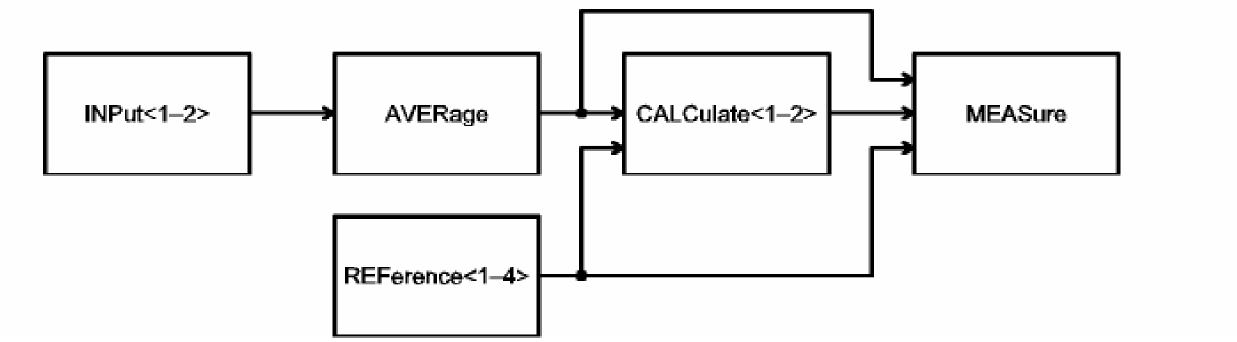


Figure 4: Data Flow

Input Channels

Product Option	1 Channel Maximum Sample Rate	2 Channel Maximum Sample Rate	Sampling Resolution
ZT412-2X	500 MS/s	250 MS/s	14-bit
ZT412-5X	400 MS/s	200 MS/s	16-bit

Table 3: Sample Rates and Bandwidths of Product Options

Input Channel Enable

The ZT412 allows capture of four input signals at rates from 10 kS/s up to one-half the maximum sample rate. Two interleaved channels can be sampled at the maximum sample rate. The maximum sample rate is 500 MS/s for the ZT412-2X and 400 MS/s for the ZT412-5X.

Input Signal Conditioning

The ZT412 provides signal conditioning to optimize input signal integrity. The analog bandwidth is DC to 250 MHz. User-configurable analog signal conditioning allows selection of input coupling, impedance, range, and offset.

Input Coupling

Input coupling can be selected as AC or DC (see the *Input Coupling Command*). Selecting AC causes a highpass filter to be inserted before the input amplifiers to limit the input signal frequency: 200 KHz for low impedance (50Ω) and 10 Hz for high impedance (1 MΩ). Selecting DC coupling bypasses the AC coupling filter.

Input Impedance

Input impedance can be selected as 50 Ω or as 1 MΩ (see the *Input Impedance Command*). Input load protection automatically switches from 50 Ω to 1 MΩ if voltages exceeding ±6 VDC are detected.

CAUTION

Do <u>not</u> apply signals having a peak value over ±25 VDC when using the high impedance (1MΩ) setting as unit damage may result.

Do <u>not</u> apply signals having a peak value over ±5 VDC when using the low impedance (50Ω) setting as unit damage may result.

Input Range

A variable gain input amplifier allows selection of voltage ranges up to 10 Volts peak-to-peak (Vpp) for the 50 Ω setting and up to 50 Vpp for the 1 MΩ setting (see the *Input Voltage Range Command*).

Note: The input voltage range and input impedance are interrelated. When changing the input impedance, always resend the input voltage range setting.

Input Offset

The *Input Voltage Offset Command* is used to set the specified input channel voltage offset. The offset range is shown in the following table:

Impedance	Range	Offset
1 MΩ	50 Vpp	0V
	25 Vpp	±12.5V
	10 Vpp	±5V
	5 Vpp	±5V
	2.5 Vpp	±5V
	1.25 Vpp	±5V
	0.5 Vpp	±5V
	0.25 Vpp	±5V
50Ω	10 Vpp	0V
	5 Vpp	±2.5V
	2 Vpp	±1V
	1 Vpp	±1V
	0.5 Vpp	±1V
	0.25 Vpp	±1V
	0.1 Vpp	±1V
	0.05 Vpp	±1V

Table 4: Input Voltage Range and Offset

Note: When setting the input voltage offset and range, an incompatible range and offset combination may occur if the commands are sent in the wrong order. In order to preclude setting an incompatible offset, set the offset to 0.0V before changing the range to the new setting.

Sweep Controls

The ZT412 provides a set of user-selectable sweep controls that enable the user to adjust the sample rate, timing, record size, and trigger position of the waveform capture process.

Record Size and Sampling Rate

Acquisition record size is specified in sample points. Valid sizes range from 100 points to the full digitizer memory size. Acquisition sample rates can be selected over a range from 10 kS/s to the maximum sample rate. The available rate selections are based on the traditional 1, 2.5, 4 and 5 steps. Four input signals can be sampled simultaneously at rates from 10 kS/s up to one-half the maximum sample rate. Two interleaved channels can be sampled at the maximum sample rate. The maximum sample rate is 500 MS/s for the ZT412-2X and 400 MS/s for the ZT412-5X. The time duration of the sample record can be calculated by dividing the number of points by the sample rate or by multiplying the number of points by the sample interval.

Time Base Reference Clock

The ZT412 supports flexible time base reference configurations. The 10 MHz time base reference is used to synchronize all internal timing including the sampling clock for the digitizer. The source of the time base reference is selectable between an internal temperature-compensated crystal oscillator (TCXO) and the VXIbus backplane CLK10 reference signal. The time base frequency must be 10 MHz ±100 ppm. The internal TCXO reference provides ±2.5 ppm frequency accuracy. The reference oscillator source is selected using the *Reference Oscillator Source Command*.

Internal and External Sampling Clock

The ZT412 supports flexible ADC sampling clock configurations. An internal sampling clock is generated by a phase-locked loop that is locked to the 10 MHz time base reference. The external sampling clock may be used to replace the onboard sampling clock for external synchronization or to achieve a sampling rate that cannot be specified by using the onboard clock. When two channels are enabled, the sampling occurs at the applied external frequency. When all channels are enabled, the sampling occurs at one-half the applied external frequency. The front panel sampling clock input has a clock rate range of 40 MHz to 500 MHz, a maximum input of ±5 V (no damage), and an input signal level of 500 mVpp to 1 Vpp (sine or square wave). The input is AC coupled into 50Ω, with an impedance accuracy of ±2%.

When using an external sample clock, the external source must be present before sending the *Clock Source Command*. Also, because the number of enabled channels affects the sample rate, the channel enable configuration must be set before sending the *Clock Source Command*. The external clock frequency must be entered using the *Clock Frequency Command* to properly set up the acquisition timing parameters. If the external clock frequency changes, the new frequency must be entered after the external clock has settled at the new frequency.

Record Length and Sweep

The record length and corresponding record sweep time are controlled using the *Sweep Points Command/Query* and *Sweep Time Query*. Record lengths can range from 100 Samples up to 8 MSamples per channel, or up to 16 MSamples/channel (using 2 channels interleaved). Memory options for the ZT412 are 2 MSamples and 32 MSamples total.

The ZT412 provides two sweep modes: automatic and normal. Automatic mode enables automatic triggering in absence of a trigger event. This mode will wait the sweep time plus 40 ms before it auto triggers. Normal mode will wait indefinitely for a trigger event before capturing data. Sweep mode is configured using the *Sweep Mode Command.*

Sweep Reference Scenarios

The ZT412 provides a flexible trigger to record timing adjustment that enables pre-trigger, post-trigger, or delayed trigger. The following figure depicts five sweep reference scenarios.

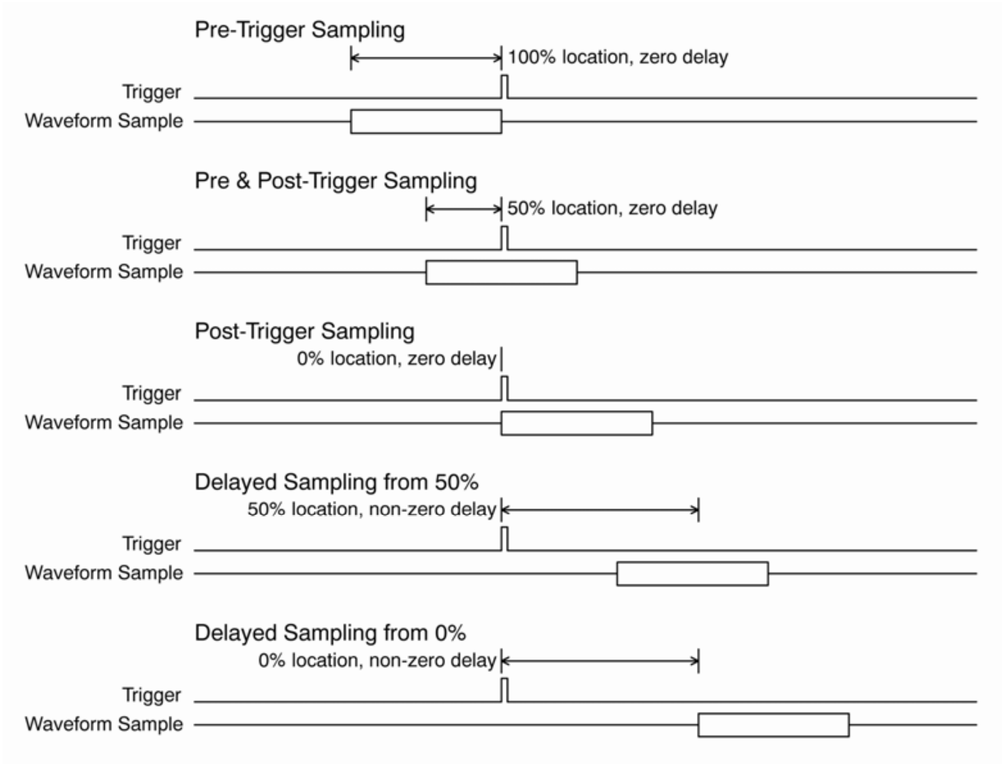


Figure 5: ZT412 Sweep Reference Scenarios

The trigger location within the waveform can be programmed between 0.0 (0%, start of waveform) and 1.0 (100%, end of waveform) using the *Sweep Offset Reference Command*. A timing delay between this reference location and the trigger event is also programmable using the *Sweep Offset Time Command*. This timing delay adjusts the trigger to a reference position in the positive time direction. Positive values move the end-of-capture further from the trigger event and consequently move the offset reference to the left. This allows the waveform capture to be delayed long after the trigger event.

The maximum cumulative delay between the trigger event and the end of the waveform record is 655 seconds. The cumulative delay is defined as:

Cumulative delay = (1 – Sweep Offset Location * Sweep Time) + Sweep Offset Time

Note: A trigger delay of 0.0 seconds causes the trigger position to be set by the offset reference location only, forcing the trigger to lie within the waveform.

Averager Controls

The ZT412 Averager controls enable the instrument to acquire multiple waveforms to create the resulting waveform record.

Averager Considerations

The following are considerations of using the averager controls:

- When averaging is enabled on the ZT412, only the final output waveform is retained; the raw, un-averaged data is not available.

- When the average is disabled, (NORMAL acquisition mode), waveforms are passed through without modification.

- The number of waveforms averaged and the average operation mode can be selected (See the *Average Count Command* and *Average Type Command* respectively).

- When averaging is enabled, it affects all active input channels.

Acquisition Types

There are four types of acquisition that can take place on the ZT412: Normal, Average, Envelope, and Equivalent-Time.

- In Normal mode, a single waveform is captured.

- In Scalar Average mode, waveform points from consecutive acquisitions are averaged together to produce the final displayed waveform. In Scalar Average mode, only two of the four inputs can be captured simultaneously. In Scalar Average mode, only one channel per channels 1-2 and one channel per channels 3-4 may be enabled.

- In Envelope mode, the minimum and maximum waveform points from multiple acquisitions are combined to form a waveform (an envelope) that shows minimum and maximum changes over time. In Envelope mode, only two of the four inputs can be captured simultaneously. In Envelope mode, only one channel per channels 1-2 and one channel per channels 3-4 may be enabled.

- In Equivalent-Time mode, a picture of a repetitive waveform is constructed by capturing a little bit of information from each repetition. This enables waveforms to be reconstructed at equivalent-time sample rates greater than the real-time sample rate (See Figure 6). Because the points appear randomly along the waveform, it is important to note that an entire waveform may not be constructed unless there are sufficient repetitions. Also, the number of points per point (selectable from 2 to 100) can be set to increase the resolution of the waveform. (See the *Average Equivalent Time Points Command*).

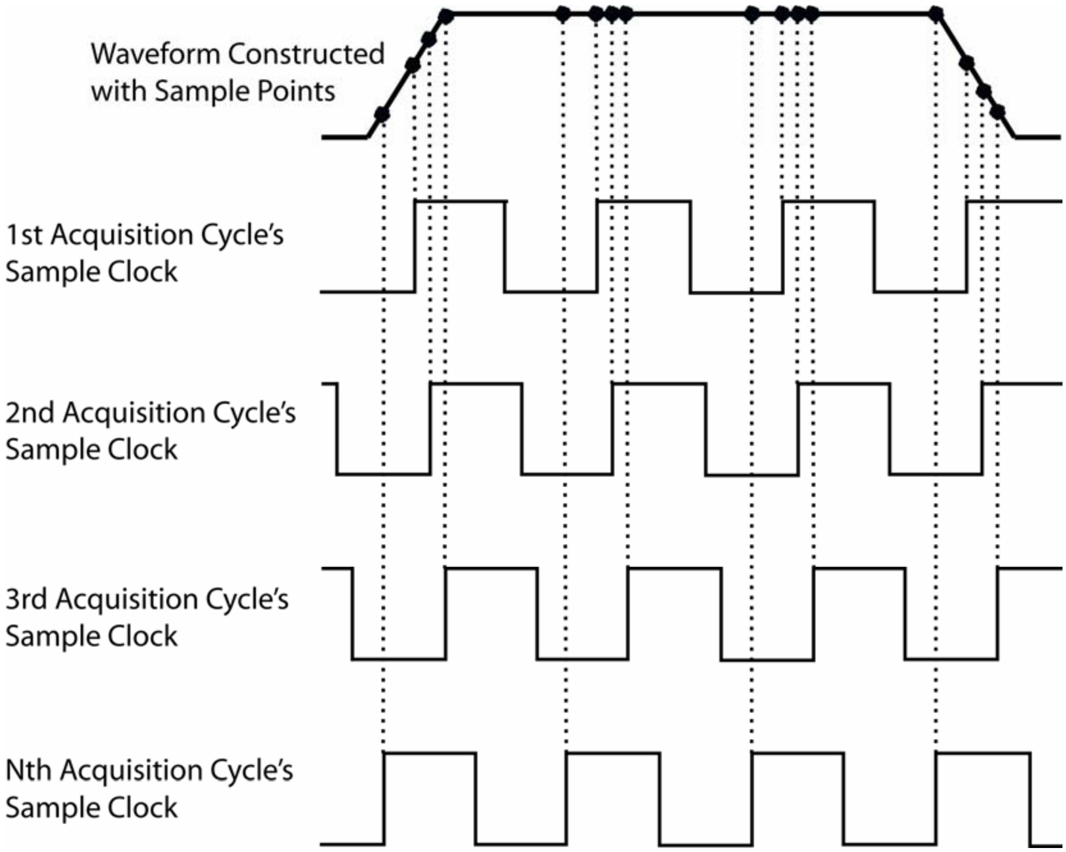


Figure 6: Equivalent Time Acquisition

Trigger and Arm Controls

The trigger and arm controls stabilize repeating waveforms and allow capture of single-shot waveforms.

Trigger Initiate Model

The ZT412 uses an arm-trigger model to control data acquisition. All acquisition cycles are started using the *Initiate Command*. Upon receiving an "initiate", the ZT412 will sequence into the "wait for arm" state. When the arm source goes active or if the arm source is set to immediate, the ZT412 will sequence into the "wait for trigger" state. When a trigger event is detected, the ZT412 will capture a waveform. The trigger loop will cycle for a selected number of times, saving the waveform associated with each pass. When the requested number of trigger loops has completed, the ZT412 will sequence back to the idle state. An *Abort Command* or *Reset Command* will immediately stop the capture sequence and return the instrument to the idle state from any other state.

The following figure shows a diagram of the trigger initiate model based on trigger mode. It shows the arm source, trigger source, and Initiate.

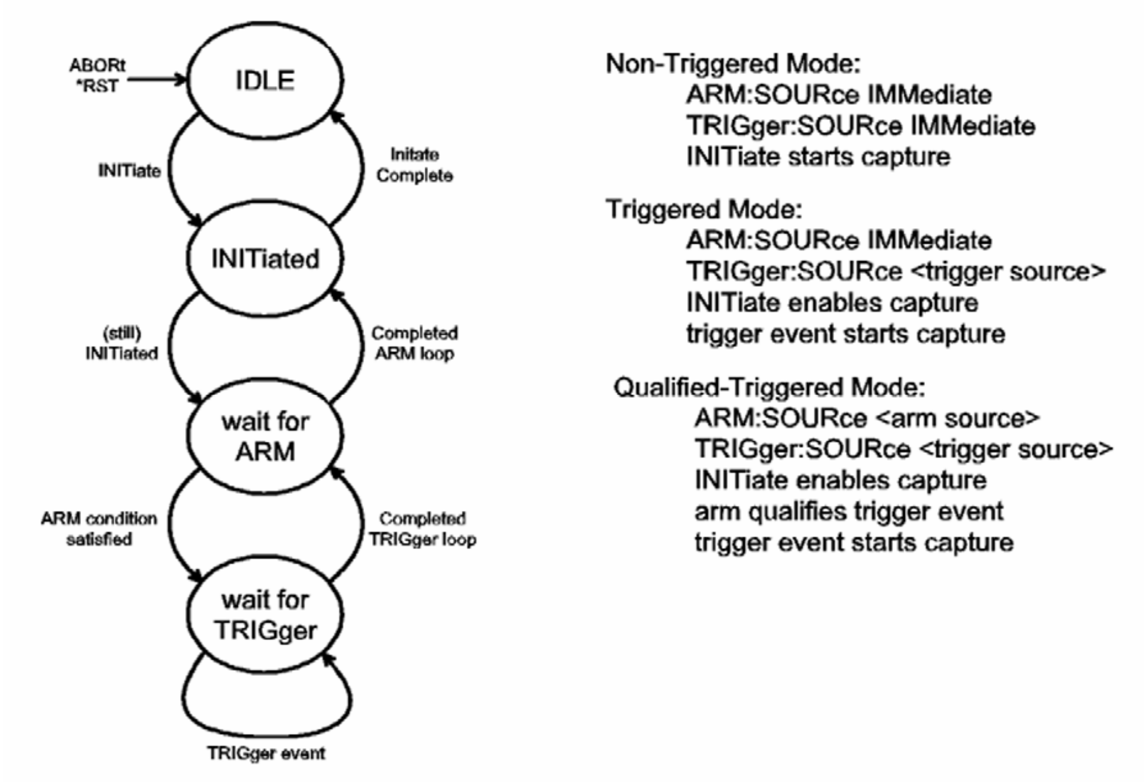


Figure 7: Trigger Initiate Model

Trigger Processing

The ZT412 accepts triggers from the following sources:

- Channels 1 to 4 (BNC)
- External Trigger (BNC)
- External Arm (BNC)
- ECLTRG0-1
- TTLTRG0-7*
- Pattern
- Software

Figure 8 shows a diagram of the ZT412 trigger processing.

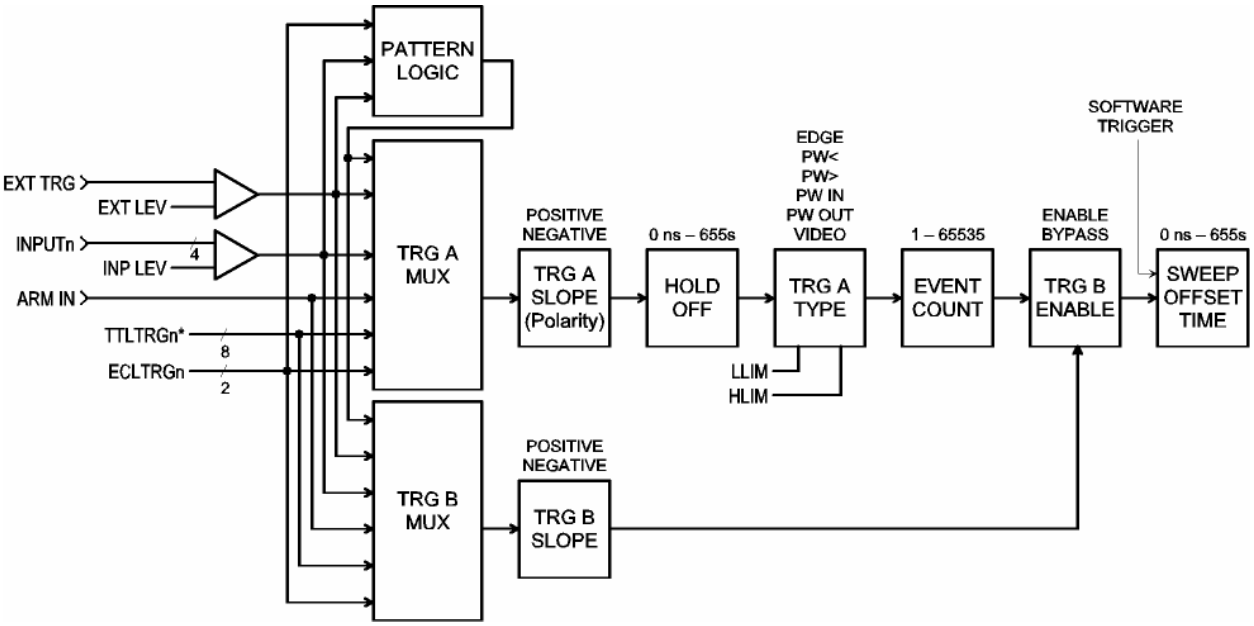


Figure 8: Trigger Processing

Trigger Types

Several types of triggers are used with the ZT412, including software, edge, pulse width, pattern, video, and event. Each type of trigger uses a different configure trigger function.

Software Trigger

Software triggers occur when a software command is used to force a trigger event to continue acquisition, regardless of the selected trigger source, type, or polarity. If manual (software) trigger source is selected, the software trigger must be used to cause a trigger event. An *Operation Complete Query* can not be used in conjunction with software triggering.

Edge Trigger

An edge trigger occurs when a signal crosses a specified trigger threshold. Specify the slope as either positive (on the rising edge) or negative (on the falling edge) to the trigger. Edge triggering is possible on all trigger sources.

Pulse Width Trigger

A pulse width trigger occurs when a signal triggers on a pulse width greater than a set limit, less than a set limit, between two set limits, or outside of two set limits. The pulse width range is 20 ns to 655 seconds with a resolution of 10 ns. Pulse width triggering is possible on all trigger sources.

Pattern Trigger

A pattern trigger occurs when a set pattern is matched TRUE or FALSE. Sources for the pattern are Channels 1 to 4, External Trigger, External Arm, and ECLTRG0–1. The three states for a pattern match are HIGH, LOW, or DO NOT CARE. Trigger polarity affects pattern match (positive polarity) or pattern not match (negative polarity). For example, a pattern trigger could be set up to trigger only when Channel 1 is LOW, Channel 2 is HIGH, ECLTRG0 is LOW, and the External Trigger is HIGH.

Video Trigger

A video trigger occurs when the ZT412 finds valid video signal synchronization. The ZT412 includes a mode for triggering on NTSC (60 Hz), PAL (50 Hz), and SECAM (50 Hz) format video standards, as well as triggering on a specific video line number and a specific video field.

Event Trigger

An event trigger enables the counting of multiple trigger events before completion of each acquisition cycle. The ZT412 allows a range of 1 to 65535 trigger events, where each trigger event is qualified by the selected source, type, polarity, or slope.

Trigger B

Trigger B enables edge triggering on a second trigger source after all Trigger A conditions are satisfied. Trigger A detection must complete before the Trigger B detector or sweep offset timer.

Arm

Each trigger must be qualified by an associated arm state condition. The arm polarity can be positive or negative. Arm sources include the following:

- External Arm input (BNC)
- External Trigger input (BNC)
- ECLTRG0–1
- TTLTRG0–7*
- Software

Trigger Timestamp

The trigger timestamp captures the time of the trigger event. This timestamp has a one-second period with a 100 ns resolution. With timestamps, it is possible to correlate multiple records or even multiple acquisitions. For example, a timestamp can be used to determine the amount of time between acquisitions.

Outputs

Trigger Outputs

The ZT412 can drive signals over any combination of the ten backplane trigger outputs (TTLTRG0–7* and ECLTRG0–1). Each output can be independently configured with unique source and enable controls.

Trigger Output sources include the following:

- Trigger event
- Arm event
- OPC event that occurs when all ZT412 operations are complete using the *Operation Complete Command*.
- Constant level (high or low)

Reference Output

The ZT412 provides a multi-function front-panel reference output signal source, REF OUT. The reference output can be selected from the following sources:

- a precision +8V voltage reference
- a 500 Hz TTL probe compensation output
- the selected 10 MHz TTL reference oscillator output
- the TTL Trigger event
- the TTL Arm event
- a 10 ns TTL pulse at 1 ms repetition interval
- a constant TTL low signal (reference output disabled)

Calculate Controls

The ZT412 can create new waveforms mathematically. The unit has two calculate channels, each capable of a 32K maximum waveform size. Sources include the following:

- 4 Input Channels
- 4 Reference Channels
- 2 Calculation Channels

Calculations are processed in channel order (i.e. Calculation Channel 1 can act on Calculation Channel 1, Calculation Channel 2 can act on Calculation Channel 2, or Calculation Channel 2 can act on Calculation Channel 1). Examples of calculations include:

- Filter Measurements (Time Domain Transform)
- Two Channel Measurements
- Math Measurements
- FFT Measurements (Frequency Domain Transform)
- Limit and Mask Testing

Calculate Functions

The following are the calculate functions:

Add

Use the *Calculate Add Command* to add the waveforms from the two sources.

Subtract

Use the *Calculate Subtract Command* to subtract the waveform from one source from the other source.

Multiply

Use the *Calculate Multiply Command* to multiply the waveforms from the two sources.

Copy

Use the *Calculate Copy Command* to copy the waveform from the source to the calculation channel.

Invert

Use the *Calculate Invert Command* to invert the source waveform.

Integral

Use the *Calculate Integral Command* to calculate the integral of the source waveform and place the result into its output.

Derivative

Use the *Calculate Derivative Command* to create a waveform that shows the rate of change of the source waveform. The derivative equation is:

$$y(i) = \frac{x(i) - x(i-1)}{\triangle t}$$

Absolute Value

Absolute value establishes an absolute value of the source waveform. All negative values are converted to positive.

Limit Test

This conducts a limit test on the waveform (See Chapter 4 for all of the limit test commands). Limit testing is the ability to compare an active signal with user-defined vertical and horizontal tolerances (test conditions) applied to measurements. Test conditions are established and an

Initiate Continuous Command is given to initiate the waveform continuously. If the active waveform exceeds the test conditions, it is a failure and the following actions occur:

- The waveform is stored into memory
- Measurement statistics are recorded including the Minimum, Maximum, Average, Failure Count, Total Count, and the Most Recent Measurement.
- The test may be stopped or run continuously.

Note: Do not perform a limit test where the calculation source and destination use the same calculation channel.

Mask Test

This conducts a mask test on a waveform (See Chapter 4 for all of the mask test commands).

A mask test is a type of limit test performed point-by-point on a waveform, determining whether an acquired signal meets a given set of criteria. It consists of an upper boundary (Ref_Max) and lower boundary (Ref_Min) where the captured waveform must not cross. These masks are typically defined by industry standards or user-defined limits, but both the references and the waveform must be the same length. The signal is first captured by the unit and then compared to the limit mask to verify whether it falls between the given limits. If any part of the waveform falls outside the mask, the software counts a failure.

Note: Do not perform a mask test where the calculation source and destination use the same calculation channel.

Frequency Transform

This conducts a Fast Fourier Transform (FFT) on a waveform (Use the *Calculate Transform Frequency Command* and the *Calculate Transform Frequency Window Command*).

The Fast Fourier Transform process mathematically converts the standard time-domain signal into its frequency components, thus providing spectrum analysis capabilities. Being able to quickly look at the signal frequency components and spectrum shape is a powerful research and analysis tool. FFT is an excellent troubleshooting aid for:

- Testing impulse response of filters and systems
- Measuring harmonic content and distortion in systems
- Identifying and locating noise and interference sources
- Analyzing vibration
- Analyzing harmonics in 50 and 60 Hz power lines

FFT results in power spectrum data in units of RMS voltage (V_{RMS}) represented as signed 16-bit values. The sample size is always a power of 2 (2^N). The following table shows the sample range, FFT size, and approximate computation time:

Sample Range	FFT Size	Approximate Computation Time in seconds
100–128	64	0.01s
129–256	128	0.02s
257–512	256	0.03s
513–1024	512	0.06s
1025–2048	1024	0.12s
2049–4096	2048	0.25s
4097–8192	4096	0.53s
8193–16384	8192	1.13s
16385–32768	16384	2.35s

Table 5: FFT Sample Range and Size

Four FFT windows (Rectangular, Hamming, Hanning, and Blackman) are available to match an analyzed signal. The Rectangular window is the best choice for non-periodic events such as transients, pulses, and one-shot acquisitions. The Hamming, Hanning, and Blackman windows are better choices for periodic signals. The following table provides a comparison of the four window types.

FFT Window	Characteristics	Used For
Rectangular	Best frequency, worst magnitude resolution. This is essentially the same as no window.	• Transients or bursts where the signal levels before and after the event are nearly equal • Equal-amplitude sine waves with frequencies that are very close • Broadband random noise with a relatively slow varying spectrum
Blackman	Best magnitude, worst at resolving frequencies	• Single frequency waveforms to look for higher order harmonics
Hamming	Better frequency, poorer magnitude resolution than Rectangular. Slightly better frequency resolution than Hanning.	• Sine, periodic, and narrowband random noise. • Transients or bursts where the signal levels before and after the event are significantly different
Hanning	Better frequency, poorer magnitude resolution than Rectangular.	• Sine, periodic, and narrowband random noise. • Transients or bursts where the signal levels before and after the event are significantly different

Table 6: FFT Window Comparison

Time Domain Transform

This conducts a low-pass filter Time Transform on a waveform (Use the *Calculate Transform Time Command*). The number of filter length data points (Range of 2 to 40) used to calculate the Time Transform is set in the second-order Infinite Impulse Response (IIR) filter (*Use the Calculate Transform Time Points Command*). A higher filter length point number causes a lower cutoff frequency for the filter.

Measure Controls

Waveform measurements offer a method for returning scalar measurement data from the ZT412 instead of the multi-point waveforms. The instrument can process a time-domain or frequency-domain waveform, and return many types of scalar measurement results.

Time Domain Measurements

The following list describes the measurements that can be performed upon time-domain waveforms:

Average	Average value of the entire captured waveform.
Alternating Current (AC)	The AC RMS characteristic of the signal subtracts the DC Average before computing the RMS.
AC High-Precision	The AC RMS level of the signal with more precision for use with waveform records having more than 8-bit resolution, such as averaged waveforms. The added precision requires approximately 10X processing time. The added precision is most noticeable when there is a non-zero input offset setting.
Amplitude	The low-to-high voltage amplitude of the applied signal.
Cycle Average	The average level of the first cycle of the selected waveform source.
Cycle Frequency	The frequency of the first cycle of the waveform.
Cycle Period	The period of the first cycle of the waveform.
Cycle RMS	The AC voltage RMS for one cycle of the waveform, measured from mid-point to mid-point.
Direct Current (DC)	The DC RMS level of the signal.
DC High-Precision	The DC RMS level of the signal with more precision for use with waveform records having more than 8-bit resolution, such as averaged waveforms. The added precision requires approximately 10X processing time. The added

	precision is most noticeable when there is a non-zero input offset setting.
Fall Crossing Time	The time of the selected falling edge of a waveform crossing the middle reference threshold measured from the start of the waveform. The edge number is selectable.
Fall Overshoot	The difference between the low level and the negative peak level of a signal as it transitions from its high state to its low state, expressed as a ratio of waveform amplitude.
Fall Preshoot	The difference between the high level and the maximum level of a signal as it transitions from its high state to its low state, expressed as a ratio of the waveform amplitude.
Fall Time	The time it takes the falling edge of a pulse to go from the upper reference threshold to the lower reference threshold.
Frequency	The frequency of the signal. All cycles in the entire capture window are used.
High	The high signal level.
Low	The low signal level.
Maximum	The maximum value of the waveform.
Minimum	The minimum value of the waveform.
Negative Duty Cycle	The ratio of negative width to period.
Negative Width	The negative width is expressed in seconds from the first falling edge reference to the next rising edge reference. The same reference is used for the rising and falling edges. The threshold is defined as the mid voltage level, or midway between high and low levels.
Peak-To-Peak	The peak-to-peak voltage or maximum to minimum voltage of the signal.
Period	Measures the period of the signal (1/frequency) using all cycles in the entire capture window.
Phase	Measures the phase of a periodic signal at the start of the Waveform in radians.
Positive Duty Cycle	The ratio of positive width to period.
Positive Width	The positive width expressed in seconds from the first rising edge reference to the next falling edge reference. The same reference is used for the rising and falling

	edges. The threshold is defined as the mid voltage level, or midway between high and low levels.
Rise Crossing Time	The time of the rising edge of a waveform as it crosses the middle reference threshold measured from the start of the waveform. The edge number is selectable.
Rise Overshoot	The difference between the high level and the positive peak level of a signal as it transitions from its low state to its high state, expressed as a ratio of waveform amplitude.
Rise Preshoot	The difference between the low level and the negative peak level of a signal as it transitions from its low state to its high state, expressed as a ratio of the waveform amplitude.
Rise Time	Measures the time for the leading edge of a pulse to rise from its lower reference threshold to its upper reference threshold. The edge number is selectable.
Time of Minimum	The time at which the first occurrence of the minimum voltage occurs.
Time of Maximum	The time at which the first occurrence of the maximum voltage occurs.

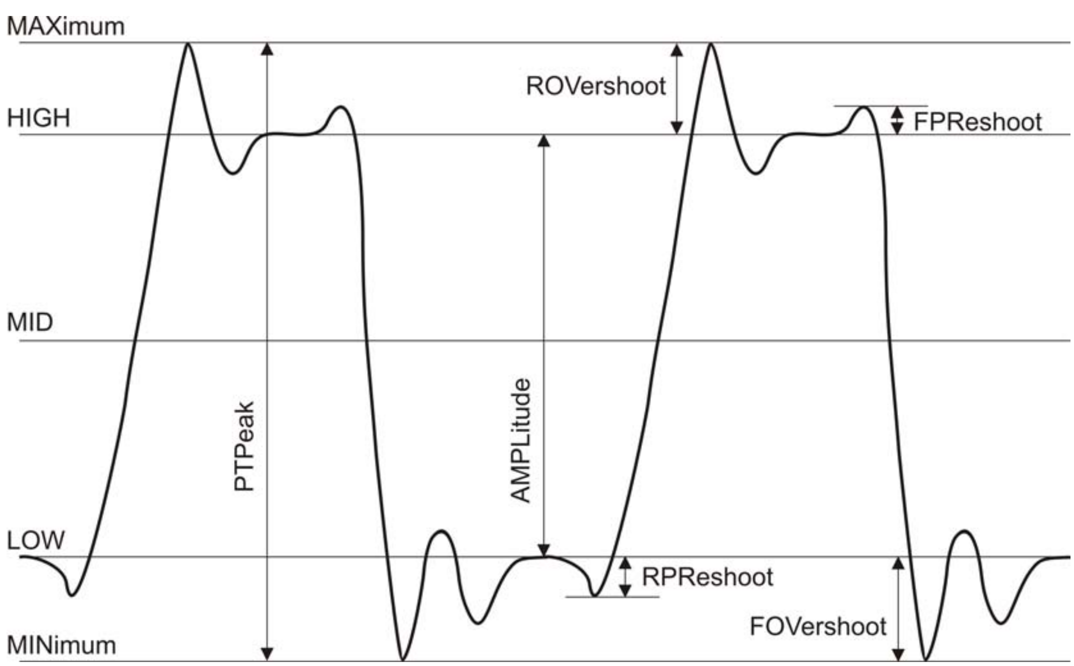


Figure 9: Vertical-Axis Measurement Terminology

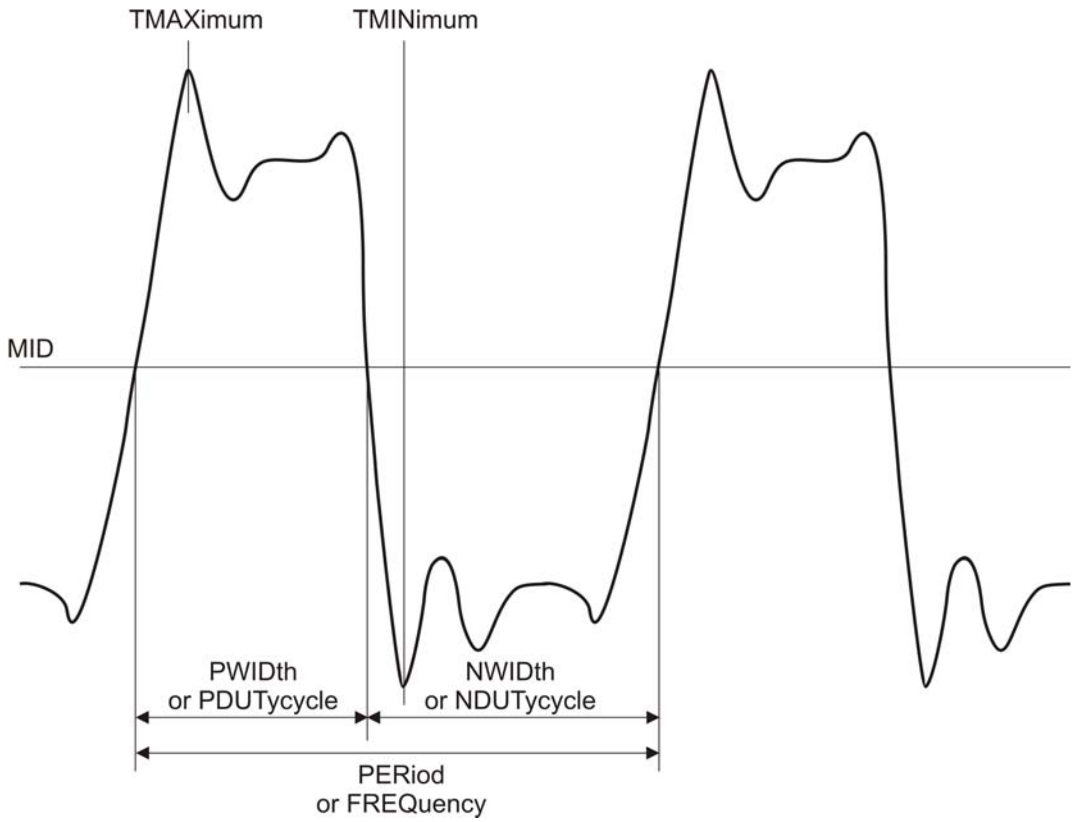


Figure 10: Horizontal-Axis Measurement Terminology

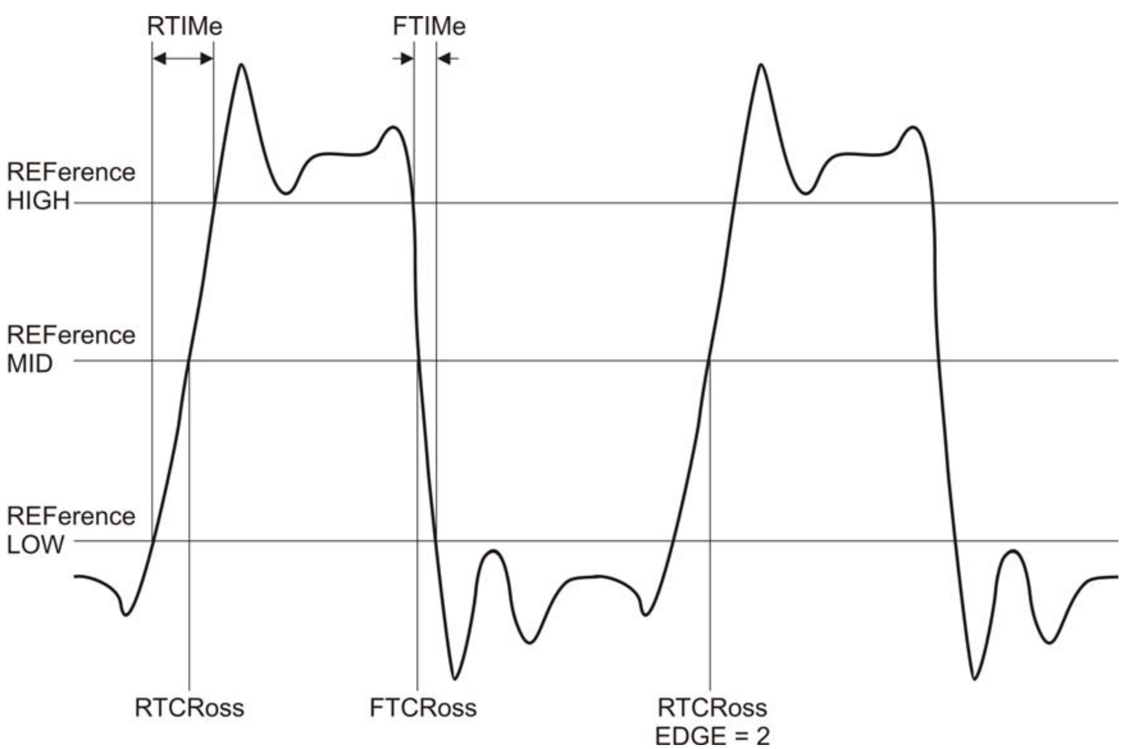


Figure 11: Edge Measurement Terminology

Frequency-Domain Measurements

Figure 12shows an example of frequency-domain measurement terminology. The following describes the measurements that can be performed upon frequency-domain waveforms, such as an FFT waveform in a calculate channel:

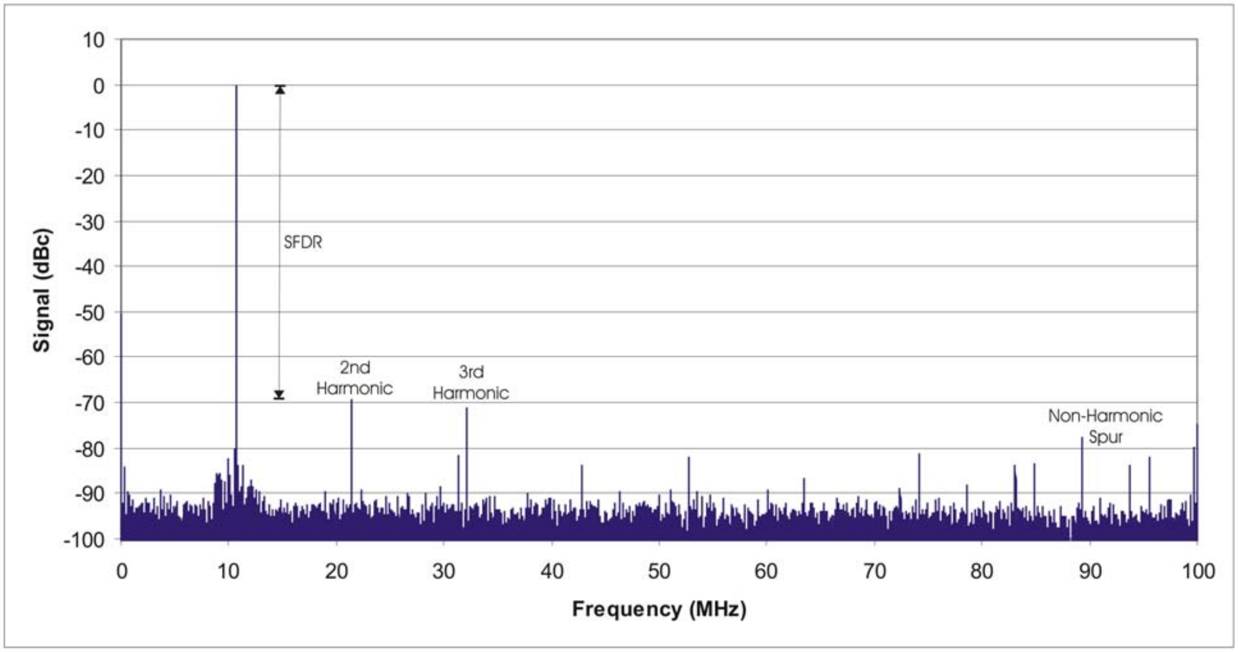


Figure 12: Frequency-Domain Measurement Terminology

Signal-to-Noise Ratio	Signal-to-Noise Ratio (SNR) is the ratio of the RMS amplitude of the input signal fundamental to the RMS amplitude of the sum of all non-harmonic noise sources. The input signal is assumed to be a perfect single-frequency sinusoidal signal. All signal components other than the input signal fundamental are considered to be harmonic distortion or noise. SNR does NOT include the first nine (second through tenth-order) harmonics as noise. This measurement is expressed in decibels relative to carrier (dBc) and is a positive value.
Total Harmonic Distortion	Total Harmonic Distortion (THD) is the ratio of the RMS amplitude of the sum of the first nine (second through tenth-order) harmonics to the RMS amplitude of the input signal fundamental. The input signal is assumed to be a perfect single-frequency sinusoidal signal. All signal components other than the input signal fundamental are considered to be harmonic distortion or noise. This measurement is expressed in decibels relative to carrier (dBc) and is a negative value.

Spurious-Free Dynamic Range	Spurious-Free Dynamic Range (SFDR) is the ratio of the RMS amplitude of the input signal fundamental to the RMS amplitude of the largest spurious signal. The spurious signal can be either a harmonic or non-harmonic of the input signal fundamental. The input signal is assumed to be a perfect single-frequency sinusoidal signal. All signal components other than the input signal fundamental are considered to be spurious signals. This measurement is expressed in decibels relative to carrier (dBc) and is a positive value.
Signal-to-Noise and Distortion	Signal-to-Noise and Distortion Ratio (SINAD) is the ratio of the RMS amplitude of the input signal fundamental to the RMS amplitude of the sum of all noise and distortion sources. The input signal is assumed to be a perfect single-frequency sinusoidal signal. All signal components other than the input signal fundamental are considered to be harmonic distortion or noise. SINAD is equivalent to the RMS sum of SNR and THD. This measurement is expressed in decibels relative to carrier (dBc) and is a positive value.
Effective Number of Bits	Effective Number of Bits (ENOB) provides a measure of the input signal dynamic range as if the signal were converted with an ideal analog-to-digital converter (ADC). ENOB provides the number of bits of an ideal ADC that would result in quantization noise equivalent to the sum of all input signal noise and distortion sources. ENOB is directly related to SINAD by the following equation: $ENOB = (SINAD - 1.763) / 6.02$ This measurement is expressed in bits and is a positive value.

Invalid Measurements

The ZT412 returns an invalid measurement code (9.99999E+37) whenever it encounters an invalid measurement condition. The following describes types of invalid measurement conditions.

Voltage Measurements

Voltage measurements (such as *Measure High Voltage Query*) where there is an over-voltage condition with the applied voltage exceeding the input range of the ADC will return the invalid measurement code (See Figure 13.

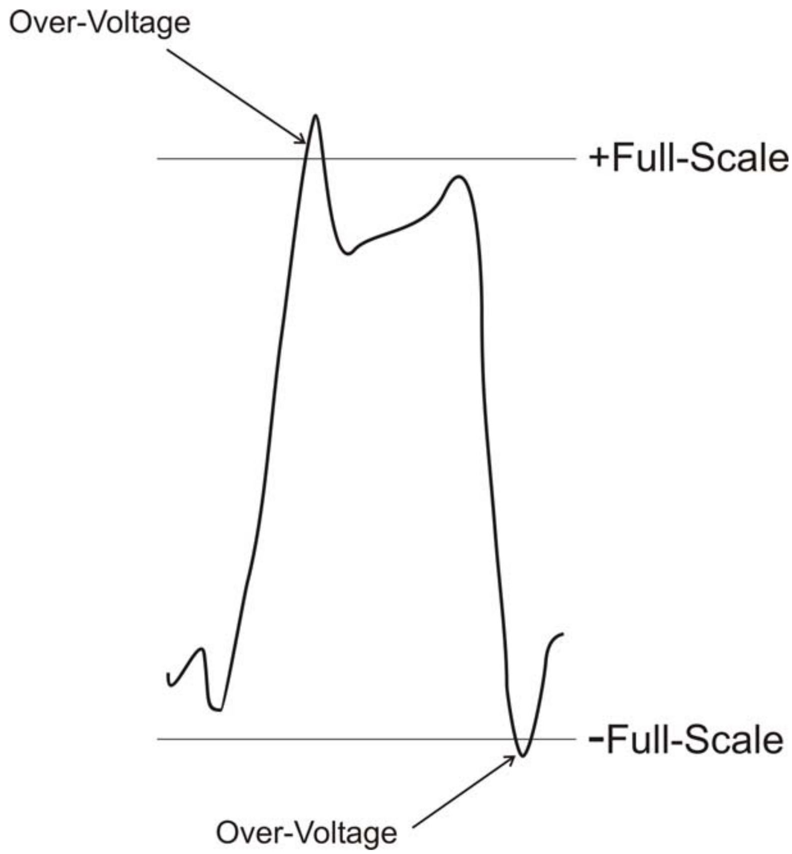


Figure 13: Invalid Voltage Measurements

RMS Measurements

The ZT412 will always return the measured value for an RMS measurement, even if a signal is clipped (over-voltage).

Cycle and Width Measurements

Cycle measurements (such as *Measure Period Query* or *Measure Frequency Query*) require at least three mid-point crossings in the capture window. Width measurements (such as *Measure Positive Width Query* or *Measure Positive Duty Cycle Query*) use at least two mid-point crossings in the capture window. If the required number of mid-point crossings is not found, the ZT412 will return the invalid measurement code.

Edge Measurements

Rising and falling edge measurements (such as *Measure Rise Time Query*) require at least as many edges present in the capture window as the user-requested edge number. An edge crossing is defined as a voltage crossing of the middle reference level. If the requested edge is greater than the number of edges in the capture window (3 or greater for a rising edge measurement on the waveform in Figure 14, the ZT410 will return the invalid measurement code. Also, an invalid measurement code will be returned if, although the requested edge is found, a complete edge is not captured and is required to make the measurement.

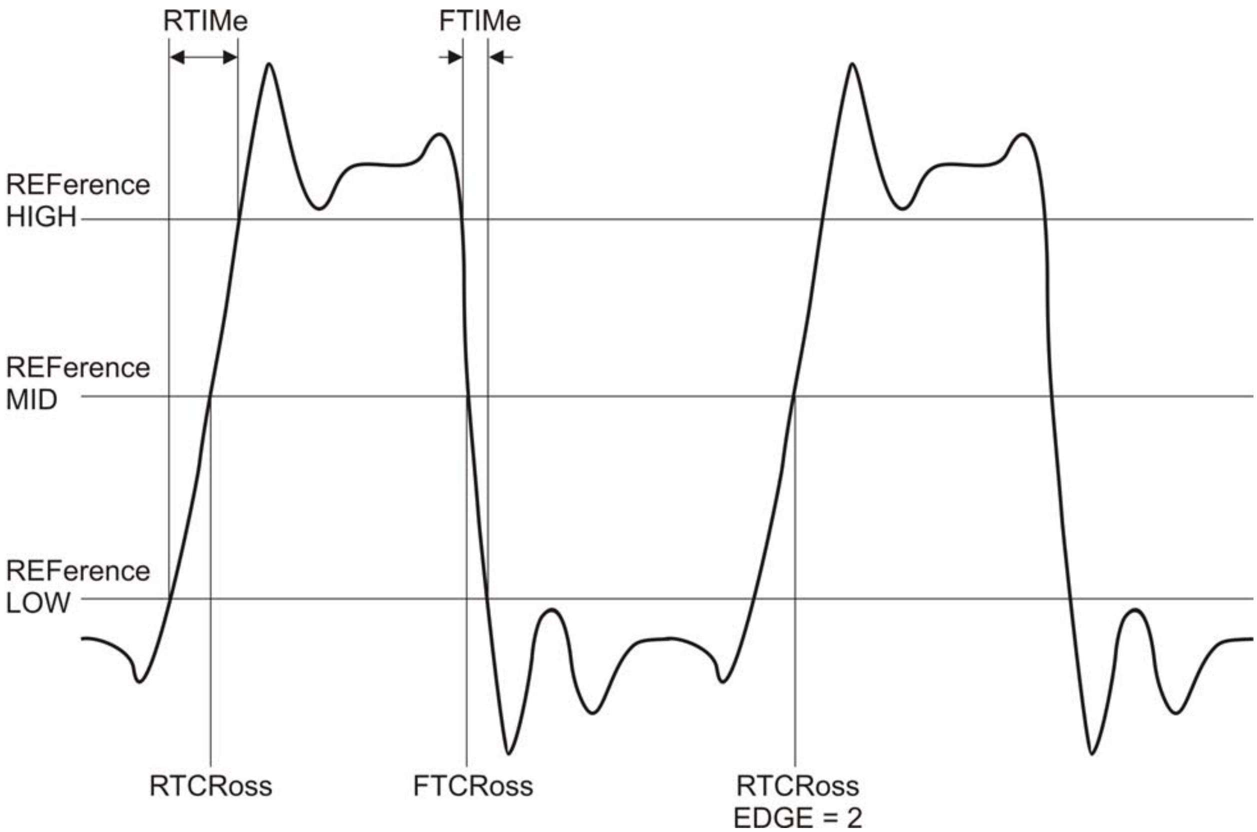


Figure 14: Invalid Edge Measurements

<u>Frequency-Domain Measurements</u>

The ZT412 will return an invalid measurement code if the input sinusoidal fundamental cannot be resolved from the noise level. The invalid measurement code will also be returned if a frequency-domain measurement is attempted upon a non-frequency domain waveform, as identified by the waveform preamble header.

Measurement Method

The following measurement methods are available on the ZT412: Entire Waveform, Gated by Time, and Gated by Points.

<u>Entire Waveform</u>

Entire Waveform measurement performs measurements on the entire captured waveform.

<u>Gated by Time</u>

Gated by Time measurement performs measurements on a waveform where the user defines a start time and stop time. "0" represents the start of the waveform.

Gated by Points

Gated by Points measurement performs measurements on a waveform where the user defines a start point and stop point. "0" represents the start of the waveform.

Measurement Reference Levels

Measurement reference levels are high, low, and mid range values that are set to take rise time, fall time, fall crossing time, and rise crossing time measurements. Reference levels are configured in relative terms of the percentage of the waveform acquired, or in absolute voltage levels. By default, the low value is 10 percent of the waveform, the mid level is 50 percent, and the high value is 90 percent.

Reference levels are user-defined using the *Measure Reference Command* and *Measure Reference Method Command*. Levels are set by percent or volts as follows:

- Low reference selects the threshold for detection of the input signal low state.
- Middle reference selects the threshold for detection of the input signal middle level.
- High reference selects the threshold for detection of the input signal high state.
- Relative reference values range from 0.0 (0 percent) to 1.0 (100 percent).
- Absolute reference values are expressed in volts.

Measurement Edge

The measurement edge is the waveform edge used in edge-related measurements (See *Measure Edge Command*). Select a falling or rising edge using a 16-bit unsigned integer value. A measurement edge is used in the following measurement types: rise time, rise crossing time, rise overshoot, rise preshoot, fall time, fall crossing time, fall preshoot, and fall overshoot.

Waveform and Reference

Read Waveform

The ZT412 uses a shared VXI memory to report acquisition results to the VXI host processor. Because the shared memory is smaller than most waveform records, a typical waveform download will require that the waveform be read in blocks multiple times and the data be appended together to form the final output. A waveform consists of signed 16-bit values.

Read Waveform Preamble

The preamble provides information necessary to convert the 16-bit integer waveform samples to and from time and voltage values. Preamble information is divided into three blocks: header information, time axis information, and voltage information as described below:

Header Information

- Waveform Type:
  - o 0 = Invalid waveform
  - o 1 = Normal voltage-time waveform
  - o 2 = Averaged voltage-time waveform
  - o 3 = Envelope waveform
  - o 4 = Equivalent Time waveform
  - o 16 = Frequency Domain Waveform (FFT)

- Count: The number of acquisitions processed to produce an output average, envelope, or equivalent time waveform.

- Size: The number of samples in the sample record.

Time Axis Information

- Increment: The time interval between samples
- Offset: Time between the trigger event and the first sample in the sample record

Note: For FFT waveforms, the X-axis displays the frequency in Hertz.

Voltage Axis Information

- Increment: The voltage interval of 1 Least Significant Bit (LSB).
- Offset: The voltage offset represented by the sample “0” code

To recreate a waveform from the preamble and sample record, use the following relationships:

- Sample Time = time offset + (sample number * time increment)
- Sample Voltage = voltage offset + (sample code value * voltage increment)

Reference Waveform

The ZT412 can save and download up to 4 reference waveforms. The reference waveforms, REF1–4, are stored in non-volatile Flash memory and are maintained when the unit is powered off. These waveforms are limited to record sizes of 32 kSamples.

- Store: copied from input, calculate and reference sources.
- Load: loaded as codes from VXI host (load preamble when loading waveform).

Utilities and Status Reporting

Auto Scale

The ZT412 can automatically adjust input signal settings. The following are the adjustable parameters for all enabled input channels:

- Input Range

- Offset
- Points (Automatically sets the number of points to 1,000)
- Sample Rate (Selected for the signal with the largest amplitude)
- Trigger Source (Selected for the signal with the largest amplitude)
- Trigger Level (Selected for the signal with the largest amplitude)

Reset and Device Clear

Use the *Reset Command* to perform a hard reset of the ZT412. This stops all acquisition and configures the unit to its default state. See *Appendix 2, Default Reset Conditions,* for a listing of all ZT412 default conditions.

Save and Recall States

The ZT412 can save and recall up to 31 instrument configuration states. These states record the input settings, horizontal sweep settings, trigger settings, and capture settings. The current instrument state can be saved and recalled later. All states are stored in non-volatile Flash memory and are maintained when the unit is powered off. The *Reset Command*, *Save Instrument State Command*, and *Recall Instrument State Command* control the instrument state configuration.

Error

There is a capability to see any and all system errors. The *System Error All Query* returns all 32 entries in the error log and clears the error log. Multiple errors are stored sequentially in the error log with the oldest error first. A zero value is returned for all non-error entries when there are less than 32 errors stored in the error log. The *System Error Count Query* returns the number of errors in the error log. The *System Error Query* returns and clears the first entry in the error log. See *Appendix 3, System Error Codes*, for a list of error codes.

Status

The status register structure provides a common way to perform status reporting according to the IEEE 488.2 specification. This status register structure allows the user to examine the conditions of the following subsystems on the ZT412: Voltage, Frequency, Calibration, Self-Test, Questionable, Operation, and Standard Event.

Each status data register set contains a condition register, an event register, and an event enable register. The summary output of a status data register set may be used to propagate the status summary to the next status level, and ultimately to the Status Byte. Figure 17 shows the complete status register structure for the ZT412. Each individual status data register set contains the following registers with the following functionality:

<u>Condition</u>

A condition register provides the current device condition or state. The condition register reflects the TRUE or FALSE states in its condition bits, may range in length from 1 to 16 bits, and may contain unused bits. Unused bits will return a zero (0) value when read.

Note: Reading a condition register does not change its contents.

Event

An event register captures changes in the associated condition register. Each event bit in an event register corresponds to a condition bit in a condition register. Event registers range in length from 1 to 16 bits and may contain unused bits. Unused bits will return a zero (0) when the register is read.

An event becomes TRUE when the associated device condition transitions to a TRUE state. The event register guarantees that the application cannot miss a condition that is removed before the condition register can be read. An event register bit will be set TRUE when an associated event occurs. These bits, once set, cannot be cleared even if they do not reflect the current status of a related condition, until the event register is read by the application. Also, the ZT412 provides a command to clear all event registers.

Note: Event bits are cleared when read.

Enable

An enable register selects which event bits in the corresponding event register will cause a TRUE summary output when an event occurs. The summary output enabled by the event enable register is used to propagate the status summary to the next status level. Each event bit in the event register has a corresponding enable bit in the event enable register. When an event enable bit is TRUE, the corresponding event will propagate to the status summary output. Any unused bits in the event enable register correspond with unused bits in the event register. The value of unused bits is zero (0) when the event enable register is read and is ignored when written to by commands.

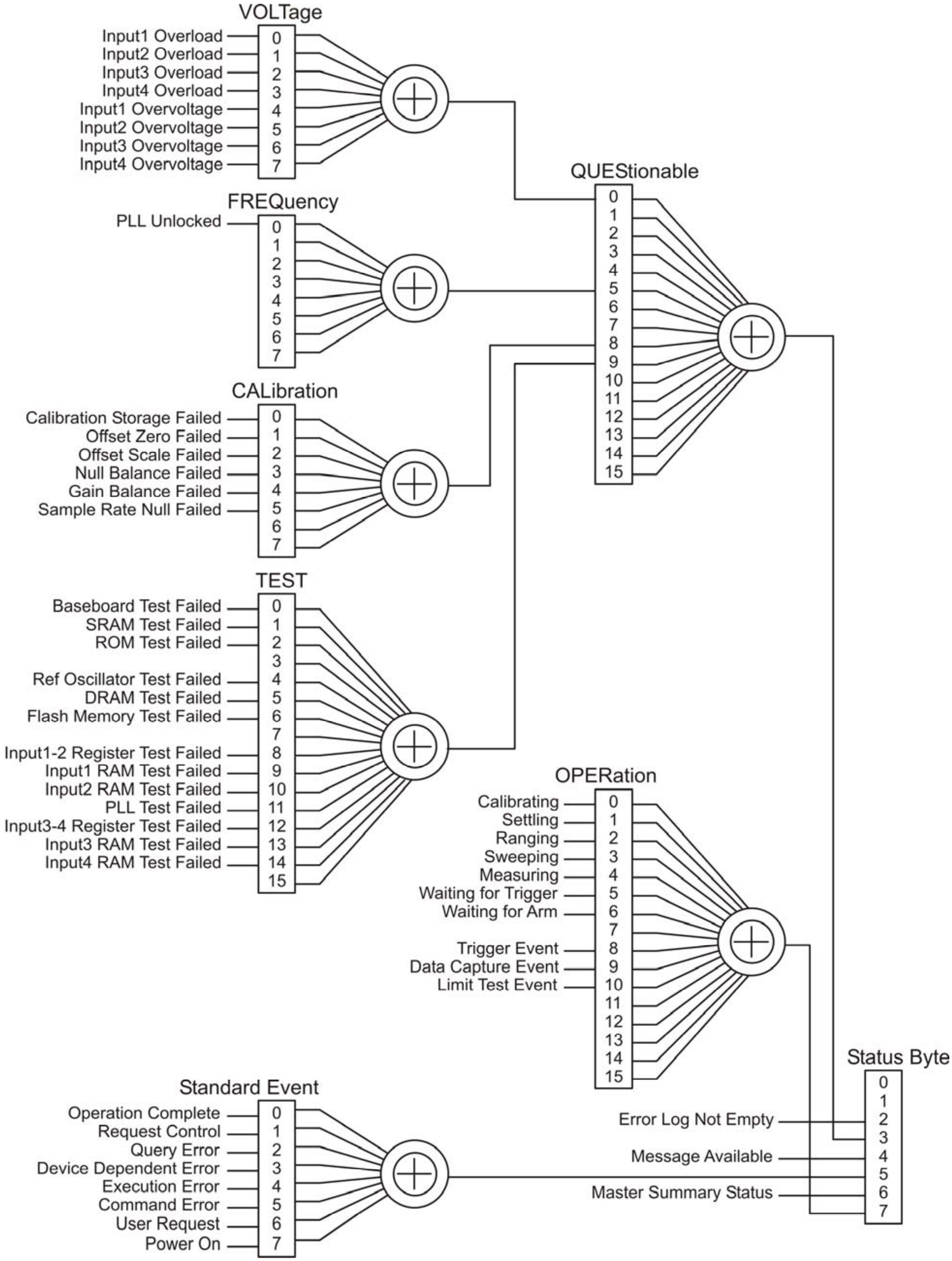


Figure 15: Status Register

Status Commands and Queries

The device status of the ZT412 can be viewed at any time using any of the following commands or queries:

Note: See the appropriate command and query syntax in Chapter 4.

Command or Query	Response
Clear Status Command	Clears all event status registers and the error log.
Event Status Enable Command	Sets the state of the event status enable register. The event status enable register allows the enabled standard events to affect the event summary status bit within the status byte.
Event Status Enable Query	Returns the state of the event status enable register. The event status enable register allows the enabled standard events to affect the event summary status bit within the status byte.
Event Status Query	Returns the status bits for the standard event status register.
Service Request Enable Command	Sets the state of the service request enable register. The service request enable register allows the enabled status byte events to affect the service request summary bits with the status byte.
Service Request Enable Query	Returns the state of the service request enable register. The service request enable register allows the enabled status byte events to affect the service request summary bits with the status byte.
Status Byte Query	Returns the Status Byte code.
Status Calibration Condition Query	Returns the current condition of the questionable calibration status register.
Status Calibration Enable Command	Allows the user to enable or disable the bits in the questionable calibration status register. The parameter is a bit mask which enables the corresponding questionable calibration status register bits.
Status Calibration Enable Query	Returns the bit mask of the questionable calibration status enable register.
Status Calibration Event Query	Returns the latched event state for the questionable calibration status register.
Status Frequency Condition Query	Returns the current condition of the questionable frequency status register.

Command or Query	Response
Status Frequency Enable Command	Allows the user to enable or disable the bits in the questionable frequency status register. The parameter is a bit mask which enables the corresponding questionable frequency status register bits.
Status Frequency Enable Query	Returns the bit mask of the questionable frequency status enable register.
Status Frequency Event Query	Returns the latched event state for the questionable frequency status register.
Status Operation Condition Query	Returns the present condition of the operation status register.
Status Operation Enable Command	Sets the contents of the operation status enable register. The parameter is a bit mask which enables the corresponding operation status register bits.
Status Operation Enable Query	Returns the bit mask of the operation status enable register.
Status Operation Event Query	Returns the latched event state for the operation status register.
Status Preset Command	Sets the enable register to all 1s. For the mandatory status data structures, it sets the enable register to 0s. Also, it sets the error/event queue enabling to report only errors.
Status Questionable Condition Query	Returns the present condition for the questionable status register.
Status Questionable Enable Command	Allows the user to enable or disable the bits in the questionable status register. The parameter is a bit mask which enables the corresponding questionable status register bits.
Status Questionable Enable Query	Returns the bit mask of the questionable status register. That is, it returns a bit mask that indicates which questionable status register bits are enabled.
Status Questionable Event Query	Returns the latched event state for the Questionable Status Register.
Status Test Condition Query	Returns the present condition of the questionable test status register.
Status Test Enable Command	Allows users to Enable or Disable bits in the questionable test status register. The parameter is a bit mask which enables the corresponding questionable test status register bits.
Status Test Enable Query	Returns the bit mask of the Questionable Test Status

Command or Query	Response
	Enable Register. That is, it returns a bit mask that indicates which Questionable Test Status Register bits are enabled.
Status Test Event Query	Returns the latched event state for the questionable test status register.
Status Voltage Condition Query	Returns the current condition of the questionable voltage status register.
Status Voltage Enable Command	Sets the contents of the voltage status enable register. The parameter is a bit mask which enables the corresponding questionable voltage status register bits.
Status Voltage Enable Query	Returns the bit mask of the questionable voltage enable register.
Status Voltage Event Query	Returns the latched event state for the questionable voltage status register.

Table 7: System Status

Self Test

The ZT412 can initiate an instrument self test and return any test error results as a 16-bit code (See *Test Query*). The self test is initiated on instrument power up and returns:

Hex Number Code	Error Type
0001_{16}	Baseboard Test Failed Bit
0002_{16}	SRAM Test Failed Bit
0004_{16}	ROM Test Failed Bit
0008_{16}	Unused Bits
0010_{16}	Reference Oscillator Test Failed Bit
0020_{16}	DRAM Test Failed
0040_{16}	Flash Memory Test Failed Bit
0080_{16}	Unused Bits
0100_{16}	Input 1–2 Register Test Failed Bit
0200_{16}	Input 1 RAM Test Failed Bit
0400_{16}	Input 2 RAM Test Failed Bit
0800_{16}	PLL Test Failed
1000_{16}	Input 3–4 Register Test Failed Bit
2000_{16}	Input 3 RAM Test Failed Bit
4000_{16}	Input 4 RAM Test Failed Bit
8000_{16}	Unused Bits

Table 8: Self-Test Errors

Calibrate

The ZT412 can perform an automatic, internal self-calibration upon command. The internal calibration determines the zero DC offset, the DC offset adjust scale factor, and the ADC balance for all input range settings for both input channels. Note that the external cables should be removed or 0.0 VDC should be applied to both input channels before commanding a self-calibration. The internal calibration process can take several minutes to complete. Upon completion of the self-calibration process, the ZT412 will respond indicating the status of the calibration. A zero (0) value is returned if the calibration is completed successfully; a one (1) value is returned if the calibration failed; otherwise, a two (2) value is returned to indicate corruption of the calibration. Calibration errors are also reported through the status-reporting system. All self-calibration data is stored in non-volatile memory and shall be used as the default, power-on calibration data. If desired, the original factory default calibration data can be restored upon command.

Identification and Version

Use the *ID Query* to return the ZT412 instrument identification including manufacturer, model number, serial number, and firmware version. The results are returned as a block of ASCII string data up to 44 characters in length.

Example: ZTEC,ZT412VXI-50,S/N 100,Version 1.00

VXI Interface

Interface Description

The ZT412 is a message-based VXIbus module that supports both VXIbus Instrument and VXIbus 488.2 Instrument protocols. It is compliant with the instrument specifications outlined in the VXI-1 Revision 1.4 and IEEE Standard 488.2-1992 specifications. In accordance with these specifications, the ZT412 supports a number of levels of communication protocols including low-level VXIbus word-serial commands, IEEE 488.2 common commands. This section describes the low-level VXIbus interface of the ZT412.

VXIbus Interface

The ZT412 is an A16 message-based slave module with A32 register-based memory. The instrument, as an A16 message-based slave, receives commands using the VXIbus word-serial protocol. In order to support the word-serial protocol, the ZT412 provides a set of A16 accessible registers as shown in Table 9. Each of the registers is defined by detailed bit descriptions given in Tables 9–17. The ZT412 also provides A32 register-based memory that is directly address-mapped in the VXIbus 32-bit address space. The A32 address space is used to transfer waveform data to and from the unit. Table 10 shows the ZT412 A32 address map.

Offset	Function	Type
00_{16}	ID Register	Read-Only
02_{16}	Device Type Register	Read/Write
04_{16}	Status/Control Register	Read/Write
06_{16}	Offset Register	Read/Write
08_{16}	Protocol Register	Read-Only
$0A_{16}$	Response Register	Read-Only
$0C_{16}$	Unused	
$0E_{16}$	Data Low Register	Read/Write
10_{16}–$3E_{16}$	Unused	

Table 9: VXIbus A16 Address Space

Offset	Function	Type	Size
0000000_{16}–$3FFFFFE_{16}$	Waveform Data Memory (SDRAM)	Read/Write	64M

Table 10: VXIbus A32 Address Space

ID Register

The ID Register is a read-only register located at address offset 00_{16}. Reading the ID Register returns the ZT412 Device Class, Address Space, and Manufacturer's ID. The Device Class for the ZT412 is Message (10_2). The Address Space for the ZT412 is A16/A32 (01_2). The Manufacturer's ID for the ZT412 is that of ZTEC Instruments Inc.: 3712 ($0E80_{16}$). Writing to the ID Register has no effect on it. Table 11 defines the function of each bit in the read-only ID Register.

Bit #	Function	Type	Value	Meaning
15–14	Device Class	Read	10_2	Message-Based
13–12	Address Space	Read	01_2	A16/A32 Memory
11–0	Manufacturer's ID	Read	$E80_{16}$	ZTEC Instruments

Table 11: ID Register Bit Descriptions (Read-Only)

Device Type Register

The Device Type Register is a read-only register located at address offset 02_{16}. Reading the Device Type Register returns the ZT412 Required Memory and Model Code. The Required Memory for the ZT412 is 64 Mbytes for A32 address space. The Model Code for the ZT412 is 412_{10} ($19C_{16}$). Writing to the Device Type Register has no effect on it. Table 12 defines the function of each bit in the read-only Device Type Register.

Bit #	Function	Type	Value	Meaning
15–12	Required Memory	Read	5_{16}	64 MBytes
11–0	Model Code	Read	$19C_{16}$	ZT412VXI

Table 12: Device Type Register Bit Descriptions (Read-Only)

Status/Control Register

The Status/Control Register is a read/write register located at address offset 04_{16}. Writing to the Status/Control Register changes the Control Register. Table 13 defines the function of each bit in the Control Register. Reading the Status/Control Register returns the contents of the Status Register. Table 14 defines the function of each bit in the read-only Status Register.

Bit #	Name	Function	Type
15	A32 Enable	0: Disable A32 Memory 1: Enable A32 Memory	Write
14–2	Unused		
1	SFInh	0: Enable Sysfail* Driver 1: Inhibit Sysfail* Driver	Write
0	SReset	0: Enable Unit 1: Soft Reset Unit	Write

Table 13: Control Register Bit Descriptions (Write-Only)

Bit #	Name	Function	Type
15	A32 Active	0: A32 Memory Inactive 1: A32 Memory Active	Read
14	Mod ID*	0: Mod ID Line Driven 1: Mod ID Line Not Driven	Read
13–4	Unused		
3	Ready	0: Unit In Configure State 1: Unit In Normal Operation State	Read
2	Passed	0: Unit Failed Self-Test 1: Unit Passed Self-Test	Read
1	SFInh	0: Sysfail* Driver Enabled 1: Sysfail* Driver Inhibited	Read
0	SReset	0: Unit Enabled 1: Unit Software Reset	Read

Table 14: Status Register Bit Descriptions (Read-Only)

Offset Register

The Offset Register is read/write register located at address offset 06_{16}. Writing or reading the Offset Register changes or returns the contents of the Offset Register. The Offset Register defines the base address for the ZT412 A32 register-based memory. Only the 5 most-significant bits of the Offset Register are used. Bit 15 is the most-significant A32 address select bit, corresponding to A31 of the address selection. Table 15 defines the function of each bit in the Offset Register.

Bit #	Name	Function	Type
15–11	Offset Address	Address Select for A32 space	Read/Write
10–0	Unused		

Table 15: Offset Register Bit Descriptions (Read/Write)

Protocol Register

The Protocol Register is read-only register located at address offset 08_{16}. Reading the Protocol Register indicates the additional communication protocols that are supported by the ZT412. The ZT412 supports interrupter capability only. Writing to the Protocol Register has no effect on it. Table 16 defines the function of each bit in the Protocol Register.

Bit #	Name	Function	Type
15	CMDR*	1: ZT412VXI has Servant-Only capability	Read-Only
14	Signal Register*	1: ZT412VXI does not have a Signal Register	Read-Only
13	Master*	1: ZT412VXI does not have Master capability	Read-Only
12	Interrupter	1: ZT412VXI has Interrupter capability	Read-Only
11	FHS*	1: ZT412VXI does not support Fast Handshake	Read-Only
10	Shared Memory*	1: ZT412VXI does not support Shared Memory	Read-Only
9–0	unused	1: Unused	Read-Only

Table 16: Protocol Register Bit Descriptions (Read-Only)

Response Register

The Response Register is read-only register located at address offset $0A_{16}$. A read of the Response Register returns the status of the communication registers on the ZT412. Writing to the Response Register has no effect on it. Table 17 defines the read-only functions for each bit in the Response Register.

Bit #	Name	Function	Type
15	Unused	0: Unused	Read-Only
14	Unused	1: Unused	Read-Only
13	DOR	1: Data Out ready for Byte Request	Read-Only
12	DIR	1: Data In ready for Byte Available	Read-Only
11	ERR*	0: Error in Word Serial Protocol	Read-Only
10	Read Ready	1: Ready for VXI Read Operation	Read-Only
9	Write Ready	1: Ready for VXI Write Operation	Read-Only
8–0	Unused	1: Unused	Read-Only

Table 17: Response Register Bit Descriptions (Read-Only)

Data Low Register

The Data Low Register is read/write register located at address offset $0E_{16}$. Writing or reading the Data Low Register provides the mechanism to transmit word-serial commands between the VXIbus host processor and the ZT412. Writing to the Data Low register causes the ZT412 to perform some action. Responses to these actions can be read back from the Data Low Register.

Bit #	Name	Function	Type
15–0	Data Low	Word Serial Message To/From ZT412VXI	Read-Write

Table 18: Data Low Register Bit Descriptions (Read-Write)

A32 Address Space

The ZT412 contains up to 64 Mbytes (32 MSamples) of register-based memory in the VXIbus A32 address space. This memory is used to store and transfer blocks of waveform data to the VXIbus host processor. In order to maximize data transfer rates, the waveform data is available through direct register-based data transfers. Each waveform data sample requires two bytes to store the 16-bit data. The SDRAM memory is available as VXIbus A32 address space.

<u>Writing to the ZT412</u>

Step-by-Step

To write reference waveforms to the ZT412:

1. Load data into the A32 space.

2. Issue TRAC:LOAD:REF command.

3. Issue *OPC? Query, and then wait for a "1" to be returned indicating the load is complete.

<u>Reading from the ZT412</u>

Step-by-Step

To read input, calculate or reference waveforms from the ZT412:

1. Issue TRAC:LOAD:INP?, TRAC:LOAD:CALC? or TRAC:LOAD:REF? query.

2. Wait for a "1" to be returned indicating operation complete.

3. Read data from A32 space.

Low-Level VXIbus Commands

The ZT412 is a message-based VXIbus instrument supporting low-level VXIbus commands. These commands are sent to the ZT412 by *reads of* and *writes to* its Data Low Register using the VXIbus word-serial protocol. Each command is defined with a unique 16-bit value that is written to the Data Low Register. These low-level commands are used by the VXIbus processor at its lowest level of data transfer protocol (transparent to most users). Most users need not concern themselves with these commands, which are listed in Table 19. More information on the low-level VXIbus commands and the word-serial protocol can be found in the VXIbus specification.

Command	Function
Byte Available	Sends a byte of data to the ZT412.
Byte Request	Requests a byte of data from the ZT412.
Abort Normal Operation	Causes ZT412 to cease all operations immediately and enter its configuration state.
Begin Normal Operation	Notifies ZT412 that it can begin normal operations and enter its normal-operation state.
End Normal Operation	Causes ZT412 to cease all operations in an orderly fashion and enter its configuration state.
Clear	Clears the VXIbus interface and any pending operations on the ZT412.
Asynchronous Mode Control	Directs the path of events and responses on the ZT412.
Control Event	Selectively enables the generation of events by the ZT412.
Read STB	Requests the reporting of the Status Byte from the ZT412.
Read Protocol	Requests the reporting of protocols supported by the ZT412 (EG, I, I4).
Read Protocol Error	Requests the reporting of the current error state of the ZT412 and resets all asserted errors.
Assign Interrupter Line	Assigns a particular backplane IRQn* line to the ZT412 for asserting interrupts.
Read Interrupter Line	Requests the reporting of the current IRQn* line assigned to the ZT412.
Read Interrupters	Requests the reporting of the number of interrupters within the ZT412 (One)

Table 19: Low-Level VXIbus Commands

Command Reference

This chapter describes IEEE 488.2 Common (*) commands and Standard Commands for Programmable Instruments (SCPI) applicable to the ZT412.

Common Command Format

The IEEE 488.2 standard defines the Common Commands that perform functions like reset, self-test, status byte query, etc. Common commands are four or five characters in length, always begin with the asterisk character (*), and may include one or more parameters. The command keyword is separated from the first parameter by a space character. Some examples of Common Commands are shown below:

- *RST
- *CLS
- *STB?

SCPI Command Format

Standard Commands for Programmable Instruments (SCPI) perform functions like setting parameters, performing measurements, querying instrument states, and retrieving data. A subsystem command structure is a hierarchical structure that usually consists of a top level (or root) command, one or more lower-level subcommands, and their parameters. The following example shows part of a typical subsystem:

```
[SENSe:]
    INPut<number>
        :COUPling AC | DC
            :IMPedance?
```

[SENSe:] is the root command, :INPut is the second-level subcommand with <number> as a parameter, and :COUPling and :IMPedance? are third-level commands/queries with AC | DC as a parameter.

Command Separator

A colon (:) always separates one command from the next lower level command as shown below:

[SENSe:]INPut<n>:IMPedance?

Colons separate the root command from the second-level command ([SENSe:]INPut), and the second-level from the third-level query (INPut<n>:IMPedance?).

Abbreviated Commands

The command syntax shows most commands as a mix of upper and lower case letters. The upper case letters indicate the abbreviated spelling for the command. For shorter program lines, send only the abbreviated form. For better program readability, you may send the entire command. The instrument will only accept either the abbreviated form or the entire command.

For example, if the command syntax shows *IMPedance?*, then *IMP?* and *IMPedance?* are both acceptable forms. Other forms of *IMPedance?*, such as *IM?*, will generate an error. You may use upper or lower case letters. Therefore, *IMPEDANCE?* and *IMPeDaNcE?* are acceptable.

Implied Commands

Implied commands are those which appear in square brackets ([ ]) in the command syntax. (Note that the brackets are not part of the command and are not sent to the instrument.) If a root level and second-level command are sent, but not a third-level implied command, the instrument assumes use of an implied command. Examine the portion of the [SENSe:] subsystem shown below:

```
[SENSe:]
    INPut<n>
        :COUPling AC | DC
        :IMPedance?
```

The first-level command [SENSe:] is an implied command. To query the instrument's input 1 impedance selection, send either of the following command statements:

[SENS:]INP1:IMP? *or* **INP1:IMP?**

Parameters

The following table contains explanations and examples of parameter types.

Parameter Type	Explanations and Examples
Mask	Bit mask (<mask>) where every bit represents a condition or event.
Numeric	Accepts all commonly used decimal representations of numbers including optional signs, decimal points, and scientific notation. 123 or 1.23E2; -123 or -1.23E2; .123, 1.23E-1, or 1.23000E-01. Accepts all commonly used suffixes with decimal representations of numbers, including optional sgns and decimal points. .123S or 123MS; 1234OHM or 1.234KOHM. Voltage = "UV" for E-6, "MV" for E-3, "V"for E0, "KV" for E3 Percent = "PCT" Ohms = "OHM", "KOHM" for E3, "MOHM" for E6. Frequency = "HZ" for E0, "KHZ" for E3, "MHZ" for E6, "GHZ" for E9. Time = "PS" for E-12, "NS" for E-9, "US" for E-6, "MS" for E-3, "S" for E0. Phase = "DEG" for (π/180) degrees or "RAD" for radians Special cases include MIN and MAX. MIN (selects minimum value available), and MAX (selects maximum value available). Numeric values are: 1. Integer Numbers • Unsigned 16-bit (U16)—range of 0 to 65535 • Signed 16-bit (S16)—range of -32768 to 32767 • Unsigned 32-bit (U32)—range of 0 to 4,294,967,295 • Signed 32-bit (S32)—range of -2,147,483,648 to 2,147,483,647 2. Standard 32-bit Floating Point (Float) **Note:** All command parameters represented as floating point numbers sent to the ZT412VXI <u>must</u> have 12 digits or less, including before and after the decimal point. Examples: 123456.789012 or 0.12345678901
Discrete	Selects from a finite number of values. These parameters use mnemonics to represent each valid setting. An example is the [SENSe:]INPut<n> COUPling <mode> command, where <mode> can be AC or DC. Another example is a single binary condition that is either true or false. 1 or ON; 0 or OFF

Table 20: Parameter Types

Optional Parameters

Parameters shown within square brackets ([ ]) are optional parameters. (Note that the brackets are not part of the command and are not sent to the instrument.) If you do not specify a value for an optional parameter, the instrument chooses a default value.

Parameters Out of Range

An out of range parameter is automatically adjusted to the closest acceptable value. For example, if INP1:ATT 0.1 is entered, the value is set to 0.9 (lowest available setting).

Linking Commands

Linking IEEE 488.2 Common Commands with SCPI Commands

Use a semicolon between the commands.

For example: ***CLS;*RST;AUT**

Linking Multiple SCPI Commands

Use both a semicolon and a colon between the commands.

For example: **INP1:COUP AC;:SYST:ERR?**

SCPI also allows several commands within the same subsystem to be linked with a semicolon.
For example: **INP1:COUP AC;:INP1:RANG 10** *or* **INP1:COUP AC;RANG 10**

IEEE 488.2 Common Commands

The following is an alphabetic list of IEEE 488.2 Common Commands.

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calibrate Query
*CAL?</td><td>Initiates and returns the results of the unit self-calibration process. The internal calibration determines the zero DC offset, the DC offset adjust scale factor, and the ADC balance for all input range settings for all input channels. The internal calibration process can take several minutes to complete. The instrument is reset upon completion of the calibration process.

Note: The input channels <u>must</u> be disconnected or be driven with 0.0 VDC before starting the calibration.

The timeout value should be set to infinite before starting the calibration, and reset to the default value when completed.

Note: Do <u>not</u> interrupt the instrument during calibration or the calibration tables could be corrupted.

Command Syntax
None

Query Syntax
*CAL? → <result>

Parameters:
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><result></td><td>U16</td><td>0 Pass
1 Fail (Did not converge)
2 Corrupt</td></tr></table></td></tr>
<tr><td>Clear Status Command
*CLS</td><td>The clear status command clears all event registers, the request for OPC flag, and all status queues (except the response queue).

Command Syntax
*CLS

Query Syntax
None

Parameters:
None</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Event Status Enable Command
*ESE

Event Status Enable Query
*ESE?</td><td>Sets and returns the state of the event status enable register. The event status enable register allows the enabled standard events to affect the event summary status bit within the status byte.

Command Syntax
*ESE <enable>

Query Syntax
*ESE? → <enable>

Parameters

<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><enable></td><td>U16</td><td>0 to 255
Bit 0: Operation Complete
Bit 1: Request Control
Bit 2: Query Error
Bit 3: Device Dependent Error
Bit 4: Execution Error
Bit 5: Command Error
Bit 6: User Request
Bit 7: Power on</td></tr></table></td></tr>
<tr><td>Event Status Register Query
*ESR?</td><td>The event status register query returns the state of the event status register. The event status register provides the standard event status information.

Command Syntax
None

Query Syntax
*ESR? → <state>

Parameters

<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><state></td><td>U16</td><td>0 to 255
Bit 0: Operation Complete
Bit 1: Request Control
Bit 2: Query Error
Bit 3: Device Dependent Error
Bit 4: Execution Error
Bit 5: Command Error
Bit 6: User Request
Bit 7: Power on</td></tr></table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Identification Query
*IDN?</td><td>The identification query returns the instrument identification information. The response contains four fields separated by commas in the form:

“Manufacturer, model number, serial number, firmware revision level.”

Command Syntax
None

Response Syntax
*IDN? → <id_string>

Parameters

<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><id_string></td><td>String</td><td>See above</td></tr></table></td></tr>
<tr><td>Operation Complete Command
*OPC

Operation Complete Query
*OPC?</td><td>The command sets the request for the operation complete flag when all pending operations have completed. When all operations have completed, the operation complete bit in the event status register will be set.

The query returns a 0 to indicate that all pending operations have <u>not</u> completed and a 1 to indicate that all pending operations have completed.

Command Syntax
*OPC

Query Syntax
*OPC? → <state>

Parameters:

<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><state></td><td>U16</td><td>0 All operations are not complete
1 All operations complete</td></tr></table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Recall Instrument State Command
*RCL</td><td>Recalls the selected saved instrument state from non-volatile memory.

Command Syntax
*RCL <number>

Query Syntax
None

Parameters:
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><number></td><td>U16</td><td>1 to 31</td></tr></table></td></tr>
<tr><td>Reset Command
*RST</td><td>Performs a hardware reset function that returns the instrument to the initial default condition. Status registers are <u>not</u> cleared.

Command Syntax
*RST

Query Syntax
None

Parameters:
None</td></tr>
<tr><td>Save Instrument State Command
*SAV</td><td>Stores the current state of the instrument to the selected storage index in non-volatile memory.

Command Syntax
*SAV <number>

Query Syntax
None

Parameters:
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><number></td><td>U16</td><td>1 to 31</td></tr></table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Service Request Enable Command
*SRE

Service Request Enable Query
*SRE?</td><td>Selects and returns the enabled bits for the Status Byte. The parameter is a bit mask which enables the corresponding status byte bits.

Command Syntax
*SRE <enable>

Query Syntax
*SRE? → <enable>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><enable></td><td>U16</td><td>0 to 65535
Bit 0: Unused
Bit 1: Unused
Bit 2: Error Log Not Empty
Bit 3: Questionable Summary
Bit 4: Message Available
Bit 5: Standard Event Summary
Bit 6: Master Summary Status
Bit 7: Operation Summary</td></tr>
</table></td></tr>
<tr><td>Status Byte Query
*STB?</td><td>Returns the Status Byte.

Command Syntax
None

Query Syntax
*STB? → <mask>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><mask></td><td>U16</td><td>0 to 65535
Bit 0: Unused
Bit 1: Unused
Bit 2: Error Log Not Empty
Bit 3: Questionable Summary
Bit 4: Message Available
Bit 5: Standard Event Summary
Bit 6: Master Summary Status
Bit 7: Operation Summary</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger Immediate Command
*TRG</td><td>Causes an immediate trigger event for any selected trigger source. If enabled, the trigger outputs on the VXIbus backplane will also toggle when a trigger immediate command is issued.

Command Syntax
*TRG

Query Syntax
None

Parameters:
None</td></tr>
<tr><td>Test Query
*TST?</td><td>Initiates an instrument self test and returns the test results as a 16-bit code. The self test is initiated on instrument power up.

Command Syntax
None

Query Syntax
*TST? → <code>

Parameters:
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><code></td><td>U16</td><td>0 to 65535
Bit 0: Baseboard Test Failed
Bit 1: SRAM Test Failed
Bit 2: ROM Test Failed
Bit 3: Unused
Bit 4: Ref Oscillator Test Failed
Bit 5: DRAM Test Failed
Bit 6: Flash Memory Test Failed
Bit 7: Unused
Bit 8: Input 1–2 Register Test Failed
Bit 9: Input1 RAM Test Failed
Bit 10: Input2 RAM Test Failed
Bit 11: PLL Test Failed
Bit 12: Input 3–4 Register Test Failed
Bit 13: Input3 RAM Test Failed
Bit 14: Input4 RAM Test Failed
Bit 15: Unused</td></tr></table></td></tr>
</table>

Name	Description
Wait to Continue Command *WAI	Allows the user to force the interface to wait until operations are complete before resuming. **Command Syntax** *WAI **Query Syntax** None **Parameters:** None

SCPI Commands and Queries

The SCPI Commands are presented in an alphabetic list below. Each SCPI command parameter table includes parameter name, parameter type, and range of values. The parameter type follows the definition and information given in Table 20.

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Abort Command</td><td>Terminates waveform capture. When an abort is received the unit will end any on-going capture activity and return to its idle state. The waveform being captured will be dropped but any previous captured waveforms will be available. The unit start state can be queried from the Status Register.

Command Syntax
ABORt

Query Syntax
None

Parameters
None</td></tr>
<tr><td>Arm Command

Arm Query</td><td>Arms or disarms the unit through software when manual arm source selected. The unit will begin trigger detection when armed. When disarmed, the unit ignores triggers. The Arm Query returns the arm condition.

Command Syntax
ARM[IMMediate] <state>

Query Syntax
ARM? → <state>

Parameters

<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><state></td><td>Discrete</td><td>ON or 1 (Arm)
OFF or 0 (Disarm)</td></tr></table></td></tr>
</table>

Name	Description
Arm Polarity Command **Arm Polarity Query**	**Description** Sets or queries the active state of the selected source. If an arm source is selected and the state of the selected source matches the ARM POLARITY state, the unit will arm. The following considerations apply when setting the arm polarity: • *POSitive* state defines the active state as the selected source in its high state • *NEGative* state defines the active state as the selected source in its low state **Command Syntax** ARM:POLarity <polarity> **Query Syntax** ARM:POLarity? → <polarity> **Parameters**

Name	Type	Range
<polarity>	Discrete	NEGative (negative polarity) POSitive (positive polarity)

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Arm Source Command

Arm Source Query</td><td>Sets or queries the Arm Source setting that will be used to arm the unit. For example, if the Arm Source Command is set to ARM, the front panel ARM IN signal will be used to arm the unit. If an immediate output is desired regardless of trigger, Arm Source Command can be set to IMMediate.

Command Syntax
ARM:SOURce <source>

Query Syntax
ARM:SOURce?

Parameters

<table>
<tr><th>Name</th><th>Type</th><th colspan="2">Range</th></tr>
<tr><td><source></td><td>Discrete</td><td>ARM</td><td>Front panel ARM IN signal</td></tr>
<tr><td></td><td></td><td>TTLTrg<n></td><td>VXIbus TTL trigger line, where <n> may be 0, 1, 2, 3, 4, 5, 6, or 7</td></tr>
<tr><td></td><td></td><td>ECLTrg<n></td><td>VXIbus ECL trigger line, where <n> may be 0 or 1</td></tr>
<tr><td></td><td></td><td>MANual</td><td>Manual arm</td></tr>
<tr><td></td><td></td><td>IMMediate</td><td>Bypass arm detection</td></tr>
<tr><td></td><td></td><td>EXTernal<n></td><td>Front panel TRG IN signal</td></tr>
</table>
</td></tr>
<tr><td>Auto Scale Command</td><td>Commands the instrument to auto scale on enabled channels only. Auto scale changes the range, offset, impedance, sample rate, trigger source, and trigger level based on the input signal. It also sets the number of points to 1000.

Command Syntax
[SENSe]:AUToscale

Query Syntax
None

Parameters
None</td></tr>
</table>

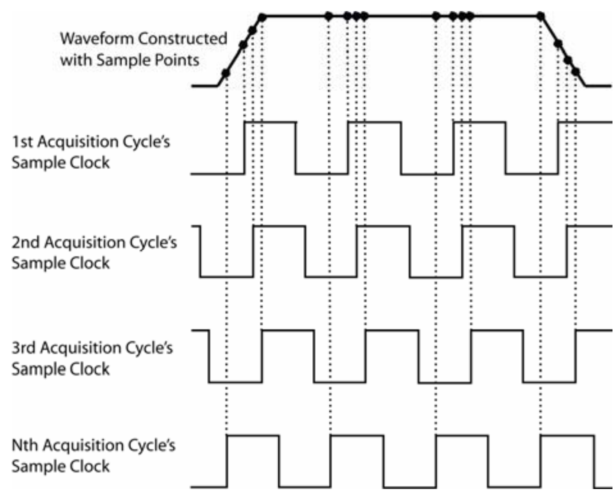
<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Average Count Command

Average Count Query</td><td>Sets or queries the acquisition count for repetitive acquisition modes:

• In Scalar (Average) mode, this specifies the number of waveforms to be averaged before the acquisition is complete.

• In Envelope mode, this specifies the number of waveforms for which to capture minimum and maximum values before the acquisition is complete.

• In Equivalent Time mode, a picture of a repetitive waveform is constructed by capturing a little bit of information from each repetition. Because the points appear randomly along the waveform, it is important to note that an entire waveform may <u>not</u> be constructed unless there are sufficient repetitions. Unfilled points will be constructed using a zero-order hold and are flagged with a “1” in the LSB of the 16-bit waveform code. Also, the number of points per point can be set to increase the resolution of the waveform. (See the Average Equivalent Time Points Command).



Command Syntax
[SENSe]:AVERage:COUNt <count>

Query Syntax
[SENSe]:AVERage:COUNt? → <count>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><count></td><td>U16</td><td>2 to 65535
MINimum (2)
MAXimum (65535)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Average Envelope View Command

Average Envelope View Query</td><td>Sets or queries the active envelope view, that is, controls which envelope view to set active. The default view is MINimum.

Command Syntax
[SENSe]:AVERage:ENVelope:VIEW <view>

Query Syntax
[SENSe]:AVERage:ENVelope:VIEW? → <view>

Parameters:
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><view></td><td>Discrete</td><td>MINimum
MAXimum</td></tr></table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Average Equivalent Time Points Command

Average Equivalent Time Points Query</td><td>Sets or queries the number of user-defined points-per-point for equivalent time sampling of a waveform. When a DSO uses equivalent time sampling, it can acquire any signal up to the analog bandwidth of the scope regardless of the sample rate. In this mode, the scope gathers the necessary number of samples across several triggers. The following considerations apply when using the Average Equivalent Time Points Command and Average Equivalent Time Points Query:

The waveform is constantly sampled and digitized.
The sampling rate is determined by the instrument clock and not the trigger repetition rate.
The input signal must be repetitive to generate the multiple triggers needed for equivalent-time sampling
The points-per-point value is user-defined at 2 to 100 points per waveform point.
Since each calculation channel is limited to a maximum waveform size of 32K, the waveform size <u>must</u> be less than 32k / points-per-point value.
Very precise time interval measurements can be made on very high bandwidth waveforms.
The trigger source must be set to an enabled input channel.

Command Syntax
[SENSe]:AVERage:ETIMe:POINts <points>

Query Syntax
[SENSe]:AVERage:ETIMe:POINts? → <points>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><points></td><td>U16</td><td>2 to 100
MINimum (2)
MAXimum (100)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Average State Command

Average State Query</td><td>Sets or queries the waveform averaging state. The following considerations apply when using the Average State Command and Average State Query:

When averaging is enabled, only the final output waveform is retained; the raw, un-averaged data is not available.
When averaging is disabled, (Normal acquisition mode), waveforms from the digitizers are passed through without modification.
The number of waveforms averaged and the average operation mode can be selected using the Average Count Command and Average Type Command respectively.
The average enable setting affects all active input channels.

Command Syntax
[SENSe]:AVERage[:STATe] <state>

Query Syntax
[SENSe]:AVERage[:STATe]? → <state>

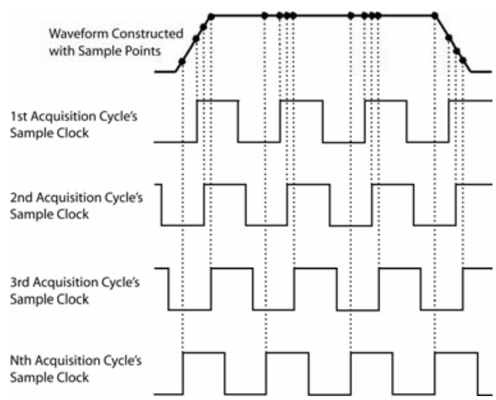
Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><state></td><td>Discrete</td><td>ON or 1 Enable Averaging
OFF or 0 Disable Averaging</td></tr>
</table>
</td></tr>
</table>

Name	Description
Average Type Command **Average Type Query**	Sets or queries the type of acquisition that is to take place.

- In Scalar mode, multiple captured waveforms are averaged together, providing higher resolution and less noise. In Scalar mode, only one channel per channels 1-2 and one channel per channels 3-4 may be enabled.

- In Envelope mode, the minimum and maximum waveform points from multiple acquisitions are combined to form a waveform (an envelope) that shows min/max changes over time. In Envelope mode, only one channel per channels 1-2 and one channel per channels 3-4 may be enabled

- In Equivalent Time mode, a picture of a repetitive waveform is constructed by capturing a little bit of information from each repetition. Because the points appear randomly along the waveform, it is important to note that an entire waveform may not be constructed unless there are sufficient repetitions. Unfilled points will be constructed using a zero-order hold and are flagged with a “1” in the LSB of the 16-bit waveform code. Also, the number of points per point can be set using the *Average Equivalent Time Points Command* to increase the resolution of the waveform.



Command Syntax
[SENSe]:AVERage:TYPE <type>

Query Syntax
[SENSe]:AVERage:TYPE? → <type>

Parameters:

Name	Type	Range	
<type>	Discrete	SCALar ENVelope ETIMe	Average Envelope Equivalent Time

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Absolute Value Command</td><td>Sets Calculate Channel to determine the absolute value of a waveform and place the result in its output, point by point.

Command Syntax
CALCulate<n>:AVALue <source>

Query Syntax
None

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Add Command</td><td>
Sets Calculate Channel to add two waveforms (source1 and source2) and place the result in its output.

Command Syntax
CALCulate<n>:ADD <source1>,<source2>

Query Syntax
None

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><source1></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><source2></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Copy Command</td><td>Sets Calculate Channel to copy a source waveform into its output.

Command Syntax
CALCulate<n>:COPY <source>

Query Syntax
None

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Data Format Command

Calculate Data Format Query</td><td>Sets or queries the data format for FFT waveforms (frequency transform calculate data). The default data format is linear, and causes waveforms to be represented in linear codes that can be converted to Volts using the calculate channel preamble. Logarithmic data format causes post-processing of the linear magnitude data into logarithmic codes. Log data codes can be converted to dBV using the calculate channel preamble. The following considerations apply:

Log data applies to FFT waveforms only.
When converting to log, zeros and negatives are increased to the lowest positive voltage code (-32767)

Command Syntax
CALCulate<n>:FORMat <format>

Query Syntax
CALCulate<n>:FORMat? → <format>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><format></td><td>Discrete</td><td>LINear
LOGarithmic</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Derivative Command</td><td>Sets Calculate Channel <n> to calculate the derivative of a source waveform and place the result into its output.

Command Syntax
CALCulate<n>:DERivative <source>

Query Syntax
None

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table>
</td></tr>
</table>

Name	Description
Calculate Function Query	Queries the current Calculate Channel function. Although two sources are always returned, the second source may be ignored for functions that only require one source. **Command Syntax** None **Query Syntax** CALCulate<n>:FUNCtion? → <function>, <source1>,<source2> **Parameters:**

Name	Type	Range
<n>	U16	1 Calculation Channel 1 2 Calculation Channel 2
<function>	Discrete	ADD Add AVALue Absolute Value COPY Copy DERivative Derivative INTegral Integral INVert Invert MULTiply Multiply SUBTract Subtract LIMit Limit FTRanform Frequency Transform TTRransform Time Transform
<source1>	Discrete	INPut<n> Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1. REFerence<n> Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1. CALCulate<n> Calculate channels, where <n> may be 1 or 2. Example: CALC1
<source2>	Discrete	INPut<n> Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1. REFerence<n> Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1. CALCulate<n> Calculate channels, where <n> may be 1 or 2. Example: CALC1

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Immediate Command</td><td>Forces the instrument to immediately perform calculations on the calculation channel.

Command Syntax
CALCulate<n>:IMMediate

Query Syntax
None

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
</table>
</td></tr>
<tr><td>Calculate Integral Command</td><td>Sets Calculate Channel to calculate the integral of a source waveform and place the result into its output.

Command Syntax
CALCulate<n>:INTegral <source>

Query Syntax
None

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Invert Command</td><td>Sets Calculate Channel <n> to invert the sign of a source waveform and place the result into its output.

Command Syntax
CALCulate<n>:INVert <source>

Query Syntax
None

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table>
</td></tr>
<tr><td>Calculate Limit Test Clear Command</td><td>Clears the statistics from a limit or mask test for the specified calculation channel.

Command Syntax
CALCulate<n>:LIMit:CLEar

Query Syntax
None

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Limit Test Command</td><td>Configures the instrument to perform a limit test.

Command Syntax
CALCulate<n>:LIMit <source>

Query Syntax
None

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table></td></tr>
<tr><td>Calculate Limit Test Continuous Command

Calculate Limit Test Continuous Query</td><td>Sets or returns if the calculation channel is doing a continuous limit test. Continuous ON runs the limit test until aborted. Continuous OFF stops the limit test upon the first failure.

Command Syntax
CALCulate<n>:LIMit:CONTinuous <state>

Query Syntax
CALCulate<n>:LIMit:CONTinuous? → <state>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><state></td><td>Discrete</td><td>ON or 1 Limit Test Continuous ON
OFF or 0 Limit Test Continuous OFF</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Limit Test Fail Query</td><td>Returns whether the limit test has failed. A “0” indicates no failures and a “1” indicates a failed limit test.

Command Syntax
None

Query Syntax
CALCulate<n>:LIMit:FAIL? → <fail_num>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><fail_num></td><td>Discrete</td><td>0 No Failures
1 Failed Limit Test</td></tr>
</table></td></tr>
<tr><td>Calculate Limit Test Lower Command

Calculate Limit Test Lower Query</td><td>Sets or returns the lower limit for a limit test.

Command Syntax
CALCulate<n>:LIMit:LOWer <value>

Query Syntax
CALCulate<n>:LIMit:LOWer? → <value>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><value></td><td>Float</td><td>Variable</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Limit Test Measurement Command

Calculate Limit Test Measurement Query</td><td>Set or queries the measurement to use for the limit test.

Command Syntax
CALCulate<n>:LIMit:MEASure <meas>

Query Syntax
CALCulate<n>:LIMit:MEASure? → <meas>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td><td></td></tr>
<tr><td><n></td><td>U16</td><td>1
2</td><td>Calculation Channel 1
Calculation Channel 2</td></tr>
<tr><td><meas></td><td>Discrete</td><td>AC
AMPLitude
AVERage
CAVerage
CFRequency
CPERiod
CRMS
DC
ENOB
FOVershoot
FPReshoot
FTCRoss
FTIMe
FREQuency
HIGH
LOW
MAXimum
MINimum
MID
NDUTycycle
NWIDth
PAC
PDC
PDUTycycle
PWIDth
PERiod
PHASe
PTPeak
ROVershoot
RPReshoot
RTCRoss
RTIMe
SFDR
SNDR
SNR
THD
TMAXimum
TMINimum
MASK</td><td>AC RMS
Amplitude
Average
Cycle Average
Cycle Frequency
Cycle Period
Cycle RMS
DC RMS
Effective Number of Bits
Fall Overshoot
Fall Preshoot
Fall Crossing Time
Fall Time
Frequency
High
Low
Maximum
Minimum
Middle
Negative Duty
Negative Width
Precision AC RMS
Precision DC RMS
Positive Duty
Positive Width
Period
Phase
Peak-to-peak
Rise Overshoot
Rise Preshoot
Rise Crossing Time
Rise Time
Spurious-Free Dynamic Range
S/N & Distortion Ratio
Signal-to-Noise Ratio
Total Harmonic Distortion
Time of Maximum
Time of Minimum
Mask</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Limit Test Report Query</td><td>
Returns all of the limit test reports for a calculation channel. The values returned include the number of tests performed, the number of test failures encountered, the minimum measurement result, the maximum measurement result, the average measurement result, and the most recent measurement performed.

Command Syntax
None

Query Syntax
CALCulate<n>:LIMit:REPort? → <count>,<fail_num>,<min>,<max> <average>,<last_val>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><count></td><td>U16</td><td>0 to 65535</td></tr>
<tr><td><fail_num></td><td>U16</td><td>0 to 65535</td></tr>
<tr><td><min></td><td>Float</td><td>Variable</td></tr>
<tr><td><max></td><td>Float</td><td>Variable</td></tr>
<tr><td><average></td><td>Float</td><td>Variable</td></tr>
<tr><td><last_val></td><td>Float</td><td>Variable</td></tr>
</table>
</td></tr>
<tr><td>Calculate Limit Test Upper Command

Calculate Limit Test Upper Query</td><td>
Sets or returns the upper limit for a limit test.

Command Syntax
CALCulate<n>:LIMit:UPPer <value>

Query Syntax
CALCulate<n>:LIMit:UPPer? → <value>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><value></td><td>Float</td><td>Variable</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Mask Test Lower Command

Calculate Mask Test Lower Query</td><td>Sets or returns the source to use for the lower reference for a mask test.

Note: Do <u>not</u> perform a mask test where the calculation source and destination use the same calculation channel.

Command Syntax
CALCulate<n>:LIMit:MASK:LOWer <source>

Query Syntax
CALCulate<n>:LIMit:MASK:LOWer?

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>CALC1 Calculation Channel 1
CALC2 Calculation Channel 2</td></tr>
<tr><td><source></td><td>Discrete</td><td>REFerence<n>
REF1 Reference Channel 1
REF2 Reference Channel 2
REF3 Reference Channel 3
REF4 Reference Channel 4

Note: Reference channels are only mask sources for Mask tests.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Mask Test Upper Command

Calculate Mask Test Upper Query</td><td>Sets or returns the source to use for the upper reference for a mask test.

Note: Do <u>not</u> perform a mask test where the calculation source and destination use the same calculation channel.

Command Syntax
CALCulate<n>:LIMit:MASK:UPPer <source>

Query Syntax
CALCulate<n>:LIMit:MASK:UPPer?

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>CALC1 Calculation Channel 1
CALC2 Calculation Channel 2</td></tr>
<tr><td><source></td><td>Discrete</td><td>REFerence<n>
REF1 Reference Channel 1
REF2 Reference Channel 2
REF3 Reference Channel 3
REF4 Reference Channel 4

Note: Reference channels are only mask sources for Mask tests.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Multiply Command</td><td>Sets the Calculation Channel to multiply two waveforms and place the result in its output.

Command Syntax
CALCulate<n>:MULTiply <source1>,<source2>

Query Syntax
None

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><source1></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><source2></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate State Command

Calculate State Query</td><td>Sets or queries the Calculate Channel processing enable. Enabled channels are processed upon every capture cycle. Disabled channels may be processed after a waveform capture using the Calculate Immediate Command.

When a Calculate Channel is initially configured, the unit selects a nominal voltage range and offset for the selected calculate operation. This nominal voltage range and offset may not be optimum for the applied signals and can be modified using the Calculate Voltage Range Command and the Calculate Voltage Offset Command.

Command Syntax
CALCulate<n>[:STATe] <state>

Query Syntax
CALCulate<n>[:STATe]? → <state>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><state></td><td>Discrete</td><td>ON or 1 Active
OFF or 0 Inactive</td></tr>
</table>
</td></tr>
</table>

Name	Description
Calculate Subtract Command	Sets the Calculation Channel to subtract two waveforms and place the result in its output. The result is generated by source1 – source2.

Command Syntax
CALCulate<n>:SUBTract <source1>,<source2>

Query Syntax
None

Parameters:

Name	Type	Range
<n>	U16	1 Calculation Channel 1 2 Calculation Channel 2
<source1>	Discrete	INPut<n> Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1. REFerence<n> Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1. CALCulate<n> Calculate channels, where <n> may be 1 or 2. Example: CALC1
<source2>	Discrete	INPut<n> Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1. REFerence<n> Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1. CALCulate<n> Calculate channels, where <n> may be 1 or 2. Example: CALC1

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Transform Frequency Command</td><td>
Sets the instrument to calculate an FFT.

Command Syntax
CALCulate<n>:TRANsform:FREQuency <source>

Query Syntax
None

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table>
</td></tr>
<tr><td>Calculate Transform Frequency Window Command

Calculate Transform Frequency Window Query</td><td>
Sets or queries the type of Transform Frequency window to use when calculating an FFT.

Command Syntax
CALCulate<n>:TRANsform:FREQuency:WINDow <window>

Query Syntax
CALCulate<n>:TRANsform:FREQuency:WINDow? → <window>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><window></td><td>Discrete</td><td>RECTangular Rectangular
HAMMing Hamming
HANNing Hanning
BLACkman Blackman</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Transform Time Command</td><td>Sets the instrument to perform a time transform on a waveform. A time transform performs a second order IIR low pass filter operation on the data.

Command Syntax
CALCulate<n>:TRANsform:TIME <source>

Query Syntax
None

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table>
</td></tr>
<tr><td>Calculate Transform Time Points Command

Calculate Transform Time Points Query</td><td>Sets or queries the number of filter length points used to calculate a Time Transform.

Command Syntax
CALCulate<n>:TRANsform:TIME:POINts <points>

Query Syntax
CALCulate<n>:TRANsform:TIME:POINts? → <points>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><points></td><td>U16</td><td>2 to 40 points
MINimum 2
MAXimum 40</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calculate Voltage Offset Command

Calculate Voltage Offset Query</td><td>Sets or queries the DC voltage offset for the Calculate Channel that is represented at the vertical center for the selected channel.

When a Calculate Channel is initially configured, the unit selects a nominal voltage range and offset for the selected calculate operation. This nominal voltage range and offset may not be optimum for the applied signals and can be modified using the Calculate Voltage Range Command and the Calculate Voltage Offset Command.

Command Syntax
CALCulate<n>:OFFSet <volts>

Query Syntax
CALCulate<n>:OFFSet? → <volts>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><volts></td><td>Float</td><td>Variable</td></tr>
</table></td></tr>
<tr><td>Calculate Voltage Range Command

Calculate Voltage Range Query</td><td>Sets or queries the expected range for the Calculate Channel.

When a Calculate Channel is initially configured, the unit selects a nominal voltage range and offset for the selected calculate operation. This nominal voltage range and offset may not be optimum for the applied signals and can be modified using the Calculate Voltage Range Command and the Calculate Voltage Offset Command.

Command Syntax
CALCulate<n>:RANGe <range>

Query Syntax
CALCulate<n>:RANGe? → <volts>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><volts></td><td>Float</td><td>Variable</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Calibration Restore Command</td><td>Restores the factory default calibration data. This will reset all self-calibration data resulting from the Calibrate Query that is used to automatically calibrate the zero DC offset, the DC offset adjust scale factor, and the ADC balance.

Command Syntax
CALibration:RESTore

Query Syntax
None

Parameters:
None</td></tr>
<tr><td>Calibration Skew Command

Calibration Skew Query</td><td>Sets or queries the channel-to-channel timing skew from input channels 1-and-2 to input channels 3-and-4 (1&2 to 3&4). The following considerations apply when using the calibration skew command:
Channel-to-channel skew between Inputs 1 and 2 are fixed in hardware and not adjustable. Channel-to-channel skew between Inputs 3 and 4 are fixed in hardware and not adjustable.
Channel-to-channel skew between the two pairs of channels (1&2 to 3&4) is adjustable to enable corrections for cabling or circuit delay mismatches.
The magnitude of the skew adjustment must be less than one ADC sample interval.
Positive skew values add delay to Inputs 1&2 relative to Inputs 3&4. Negative skew values add delay to Inputs 3&4 relative to Inputs 1&2.
Command Syntax
CALibration:SKEW <seconds>

Query Syntax
CALibration:SKEW? → <seconds>

Parameters:
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><seconds></td><td>Float</td><td>-300 ps to +300 ps
MINimum -300 ps
MAXimum 300 ps

Resolution: 10 ps</td></tr></table></td></tr>
</table>

Name	Description

Clock Frequency Command

Clock Frequency Query

Sets or queries the sample clock frequency in Hertz. The maximum internal sample clock frequency varies depending upon the product option as shown in the following table.

Product Option	2 Channel Maximum Sample Rate	4 Channel Maximum Sample Rate
ZT412-2X	500 MS/s	250 MS/s
ZT412-5X	400 MS/s	200 MS/s

The following considerations apply when setting the clock frequency:

- A frequency must be entered when using an external sample clock in order to setup timing parameters.
- Whenever a frequency changes for an external sample clock, the new frequency must be entered after the external clock has settled at the new frequency.

Command Syntax

[SENSe]:SWEep:CLOCk:FREQuency <hertz>

Query Syntax

[SENSe]:SWEep:CLOCk:FREQuency? → <hertz>

Parameters

Name	Type	Range
<hertz>	Float	Internal Sample Clock: 10 kS/s to 4-Channel Maximum Sample Rate in 1, 2.5, 4 and 5 steps 500 MS/s, 2 channel interleaved (ZT412-2X) 400 MS/s, 2 channel interleaved (ZT412-5X) External Sample Clock: 40 MS/s to 2-Channel Maximum Sample Rate 2 channels enabled: sample at external frequency 4 channels enabled: sample at half external frequency

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Clock Source Command

Clock Source Query</td><td>Sets or queries the selected sample clock source. Internal generator or external input sample clock sources are supported.

Note: An external clock source <u>must</u> be present before setting to external clock.

Command Syntax
[SENSe]:SWEep:CLOCk:SOURce <source>

Query Syntax
[SENSe]:SWEep:CLOCk:SOURce? → <source>

Parameters
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><source></td><td>Discrete</td><td>INTernal Internal Sample Clock
EXTernal External Sample Clock</td></tr></table></td></tr>
<tr><td>Format Byte Command

Format Byte Query</td><td>Sets of returns the current byte order setting. Normal byte order is MSB first. Swapped byte order is LSB first.

Command Syntax
FORMat:BORDer <order>

Query Syntax
FORMat:BORDer? → <order>

Parameters:
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><order></td><td>Discrete</td><td>SWAPped Swapped byte order (LSB first)
NORMal Normal byte order (MSB first)</td></tr></table></td></tr>
<tr><td>Initiate Command

Initiate Query</td><td>Initiates the instrument. While initiated, the instrument is enabled to acquire waveforms and perform calculations and measurements.

Command Syntax
INITiate[:IMMediate]

Query Syntax
INITiate?

Parameters:
None</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Initiate Continuous Command

Initiate Continuous Query</td><td>Sets or returns the instrument initiate continuous state. This is usually only used for limit and mask tests.

Command Syntax
INITiate:CONTinuous <state>

Query Syntax
INITiate:CONTinuous? → <state>

Parameters:
<table>
<tr><th>Name</th><th>Type</th><th>Range</th></tr>
<tr><td><state></td><td>Discrete</td><td>ON or 1 Initiate continuous ON
OFF or 0 Initiate continuous OFF</td></tr>
</table>
</td></tr>
<tr><td>Input Attenuation Command

Input Attenuation Query</td><td>Sets or queries the external attenuation for an input signal. The attenuation feature allows the user to set voltage levels using non-attenuated values.

Command Syntax
[SENSe]:INPut<n>:ATTenuation <atten>

Query Syntax
[SENSe]:INPut<n>:ATTenuation? → <atten>

Parameters:
<table>
<tr><th>Name</th><th>Type</th><th>Range</th></tr>
<tr><td><n></td><td>U16</td><td>1 Input Channel 1
2 Input Channel 2
3 Input Channel 3
4 Input Channel 4</td></tr>
<tr><td><atten></td><td>Float</td><td>0.9 to 1000.0
MINimum 0.9
MAXimum 1000.0</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr>
<td>Input Coupling Command

Input Coupling Query</td>
<td>
Sets or queries the input signal coupling. The signal coupling feature allows the user to selectively pass or block the DC component of an input signal. The following considerations apply when using the input coupling command:

DC coupling passes all frequencies equally
AC coupling blocks low frequencies.
AC coupling and input impedance setting interact to set the low frequency cutoff frequency. AC and high input impedance attenuates frequencies below 10 Hz. AC coupling and low input impedance attenuates frequencies below 200 kHz.
When switching to AC coupling, ensure that the signal has settled before capturing waveform data. With high input impedance, the 10 Hz cutoff requires more than 0.7 seconds to reject a DC signal and to settle within 0.1% of 0.0 VDC.

Command Syntax
[SENSe]:INPut<n>:COUPling <coupling>

Query Syntax
[SENSe]:INPut<n>:COUPling? → <coupling>

Parameters:

<table>
<tr><th>Name</th><th>Type</th><th>Range</th></tr>
<tr><td><n></td><td>U16</td><td>1 Input Channel 1
2 Input Channel 2
3 Input Channel 3
4 Input Channel 4</td></tr>
<tr><td><coupling></td><td>Discrete</td><td>AC AC coupling
DC DC coupling</td></tr>
</table>
</td>
</tr>
</table>

Name	Description

Input Impedance Command

Input Impedance Query

Sets or queries the input impedance in ohms.

Note: When setting the input impedance, set a temporary level first to preclude setting an incompatible impedance range. Set the range to 10.0V (which is valid for both low and high impedances) before changing the value to the new setting.

Command Syntax
[SENSe]:INPut<n>:IMPedance <ohms>

Query Syntax
SENSe]:INPut<n>:IMPedance? → <ohms>

Parameters:

Name	Type	Range
<n>	U16	1 Input Channel 1 2 Input Channel 2 3 Input Channel 3 4 Input Channel 4
<ohms>	Float	50 or 1e6 MINimum 50 ohms (Low impedance) MAXimum 1e6 ohms (High impedance)

Input State Command

Input State Query

Enables or disables an input channel capture.

Command Syntax
[SENSe]:INPut<n>[:STATe] <state>

Query Syntax
[SENSe]:INPut<n>[:STATe]? → <state>

Parameters:

Name	Type	Range
<n>	U16	1 Input Channel 1 2 Input Channel 2 3 Input Channel 3 4 Input Channel 4
<state>	Discrete	ON or 1 Input channel capture ON OFF or 0 Input channel capture OFF

Name	Description

Input Voltage Offset Command

Input Voltage Offset Query

Sets or queries the specified input channel voltage offset. The limits upon the input voltage offset setting are dependent upon the input voltage range.

Command Syntax

[SENSe]:VOLTage<n>:RANGe:OFFSet <volts>

Query Syntax

[SENSe]:VOLTage<n>:RANGe:OFFSet? → <volts>

Parameters:

Name	Type	Range
<n>	U16	1 Input Channel 1 2 Input Channel 2 3 Input Channel 3 4 Input Channel 4
<volts>	Float	Offset in Volts, See table MINimum MAXimum

Impedance	Range	Offset
1 MΩ	50 Vpp	0V
	25 Vpp	±12.5V
	10 Vpp	±5V
	5 Vpp	±5V
	2.5 Vpp	±5V
	1.25 Vpp	±5V
	0.5 Vpp	±5V
	0.25 Vpp	±5V
50Ω	10 Vpp	0V
	5 Vpp	±2.5V
	2 Vpp	±1V
	1 Vpp	±1V
	0.5 Vpp	±1V
	0.25 Vpp	±1V
	0.1 Vpp	±1V
	0.05 Vpp	±1V

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr>
<td>Input Voltage Protection State Command

Input Voltage Protection State Query</td>
<td>Sets or queries the specified input channel voltage protection state.

Command Syntax
[SENSe]:VOLTage<n>:PROTection[:STATe] <state>

Query Syntax
[SENSe]:VOLTage<n>:PROTection[:STATe]? → <state>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Input Channel 1
2 Input Channel 2
3 Input Channel 3
4 Input Channel 4</td></tr>
<tr><td><state></td><td>Discrete</td><td>ON or 1 Input channel voltage protection ON
OFF or 0 Input channel voltage protection OFF</td></tr>
</table>
</td>
</tr>
</table>

Name	Description
Input Voltage Range Command **Input Voltage Range Query**	Sets or queries the specified input channel voltage range. The limits upon the input voltage offset setting are dependent upon the input voltage range. **Command Syntax** [SENSe]:VOLTage<n>:RANGe:PTPeak <volts> **Query Syntax** [SENSe]:VOLTage<n>:RANGe:PTPeak? → <volts> **Parameters:**

Name	Type	Range
<n>	U16	1 Input Channel 1 2 Input Channel 2 3 Input Channel 3 4 Input Channel 4
<volts>	Float	Range in Volts, See table MINimum MAXimum

Impedance	Range	Offset
1 MΩ	50 Vpp	0V
	25 Vpp	±12.5V
	10 Vpp	±5V
	5 Vpp	±5V
	2.5 Vpp	±5V
	1.25 Vpp	±5V
	0.5 Vpp	±5V
	0.25 Vpp	±5V
50Ω	10 Vpp	0V
	5 Vpp	±2.5V
	2 Vpp	±1V
	1 Vpp	±1V
	0.5 Vpp	±1V
	0.25 Vpp	±1V
	0.1 Vpp	±1V
	0.05 Vpp	±1V

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure AC High-Precision Query</td><td>
Queries the high-precision AC RMS level of the selected waveform source. This measurement increases the measurement accuracy of waveform records having more than 8-bit resolution, such as averaged waveforms. The added precision requires approximately 10X processing time. The added precision is most noticeable when there is a non-zero input offset adjustment.

Root-mean-square (RMS) refers to the most common mathematical method of defining the effective voltage or current of an AC wave. This method subtracts the AC voltage average before computing the RMS value.

For a true sine wave, the rms value is 0.707 times the peak value, or 0.354 times the peak-to-peak value.

Command Syntax
None

Query Syntax
MEASure:VOLTage:AC? <source> → <volts>

Parameters
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure AC RMS Query</td><td>
Queries the AC RMS level of the selected waveform source.
Root-mean-square (rms) refers to the most common mathematical method of defining the effective voltage or current of an AC wave. This method subtracts the AC voltage average before computing the RMS value.
For a sine wave, the rms value is 0.707 times the <u>peak</u> value, or 0.354 times the <u>peak-to-peak</u> value.
Command Syntax
None
Query Syntax
MEASure:VOLTage:AC? <source> → <volts>
Parameters
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Amplitude Query</td><td>
Queries the amplitude of the selected waveform source. The amplitude measurement assumes a bi-level signal with distinct high and low levels and is defined as (waveform high level – waveform low level). Use the Measure Peak-to-Peak Voltage Query to detect signal amplitude without assuming a bi-level signal.
Command Syntax
None
Query Syntax
MEASure:VOLTage:AMPLitude? <source> → <volts>
Parameters
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

Name	Description
Measure Average Voltage Query	Queries the average level of the selected waveform source. The average is defined as the sum of all the sample values in the waveform record divided by the number of samples. **Command Syntax** None **Query Syntax** MEASure:VOLTage:AVERage? <source> → <volts> **Parameters**

Name	Type	Range
<source>	Discrete	INPut<n> Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1. REFerence<n> Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1. CALCulate<n> Calculate channels, where <n> may be 1 or 2. Example: CALC1
<volts>	Float	Voltage in Volts

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Cycle Average Query</td><td>Queries the average level of one cycle of the selected waveform source. This is always the first cycle seen.

Command Syntax
None

Query Syntax
MEASure:VOLTage:CAVerage? <source> → <volts>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Cycle Frequency Query</td><td>Queries the frequency for one cycle of the waveform. This is always the first cycle seen.

Command Syntax
None

Query Syntax
MEASure:VOLTage:CFRequency? <source> → <frequency>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><frequency></td><td>Float</td><td>Frequency in Hertz</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Cycle Period Query</td><td>Queries the period for one cycle of the waveform. This is always the first cycle seen.

Command Syntax
None

Query Syntax
MEASure:VOLTage:CPERiod? <source> → <seconds>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><seconds></td><td>Float</td><td>Time in seconds</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Cycle RMS Query</td><td>Queries the voltage RMS for one cycle of the waveform, measured from mid-point to mid-point. This is always the first cycle seen.

Command Syntax
None

Query Syntax
MEASure:VOLTage:CRMS? <source> → <volts>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure DC High-Precision RMS Query</td><td>Queries the high-precision DC RMS level of the selected waveform source. This measurement increases the measurement accuracy of waveform records having more than 8-bit resolution, such as averaged waveforms. The added precision requires approximately 10X processing time. The added precision is most noticeable when there is a non-zero input offset adjustment.

Command Syntax
None

Query Syntax
MEASure:VOLTage:DC? <source> → <volts>

Parameters
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure DC RMS Query</td><td>Queries the DC RMS level of the selected waveform source.

Command Syntax
None

Query Syntax
MEASure:VOLTage:DC? <source> → <volts>

Parameters
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table></td></tr>
<tr><td>Measure Edge Command

Measure Edge Query</td><td>Sets or queries the waveform edge used in edge-related measurements. A falling or rising edge is selected using a 16-bit unsigned integer value and is used in the following measurement types: rise time, rise crossing time, rise overshoot, rise preshoot, fall time, fall crossing time, fall preshoot, and fall overshoot.

Command Syntax
MEASure:EDGE <edge_num>

Query Syntax
MEASure:EDGE? → <edge_num>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><edge_num></td><td>U16</td><td>1 to 65535
MINimum 1
MAXimum 65535</td></tr>
</table></td></tr>
</table>

Name	Description
Measure Effective Number of Bits Query	Performs an effective number of bits measurement upon a frequency-domain waveform such as a FFT calculate channel.

Effective Number of Bits (ENOB) provides a measure of the input signal dynamic range as if the signal were converted with an ideal analog-to-digital converter (ADC). ENOB provides the number of bits of an ideal ADC that would result in quantization noise equivalent to the sum of all input signal noise and distortion sources. ENOB is directly related to SINAD by the following equation:

$$ENOB = (SINAD - 1.763) / 6.02$$

This measurement is expressed in bits and is a positive value.

An invalid measurement code will be returned if the input sinusoidal fundamental cannot be resolved from the noise level. An invalid measurement code will also be returned if this measurement is attempted upon a non-frequency domain waveform, as identified by the waveform preamble header.

Due to the quantization level of the fixed point processing algorithm for this measurement, the three lowest-value codes in a frequency domain waveform (-32768, -32767, -32766) are not counted as signal, noise, or harmonics while performing the measurement.

Returned Format: The measurement is returned as a positive numeric value representing the measured effective number of bits.

Command Syntax
None

Query Syntax
MEASure:VOLTage:ENOB? <source> → <bits>

Parameters:

Name	Type	Range
<source>	Discrete	INPut<n> Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1. REFerence<n> Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1. CALCulate<n> Calculate channels, where <n> may be 1 or 2. Example: CALC1
<bits>	Float	Number of bits

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Fall Crossing Time Query</td><td>
Queries time of the falling edge of a waveform crossing the middle reference threshold measured from the start of the waveform. The edge number is selectable.

Command Syntax
None

Query Syntax
MEASure:VOLTage:FTCRoss? <source> → <seconds>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><seconds></td><td>Float</td><td>Time in seconds</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Fall Overshoot Query</td><td>
Queries the voltage overshoot of the selected waveform. The following considerations apply when using the Measure Fall Overshoot Time Query:

Overshoot is defined as the amount of voltage past the low level of a bi-level signal that a signal travels as it transitions from its high state to its low state.
Overshoot is calculated as the (signal minimum voltage – low level voltage)/signal amplitude.
The value returns as a percent in a decimal. For example, a 10 percent overshoot will be returned as 0.1.

Command Syntax
None
Query Syntax
MEASure:VOLTage:FOVershoot? <source> → <percent>
Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><percent></td><td>Float</td><td>0.0 (0 percent) to 1.0 (100 percent)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Fall Preshoot Query</td><td>Queries the voltage preshoot of the selected waveform. The following considerations apply when using the measure preshoot query:

• Preshoot is defined as the amount of voltage past the high level of a bi-level signal that a signal travels as it transitions from its high state to its low state.

• Preshoot is calculated as the signal maximum voltage – high level voltage/signal amplitude.

• The value returns as a percent in a decimal. For example, a 10 percent preshoot will be returned as 0.1.

Command Syntax
None

Query Syntax
MEASure:VOLTage:FPReshoot? <source> → <percent>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><percent></td><td>Float</td><td>0.0 (0 percent) to 1.0 (100 percent)</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Fall Time Query</td><td>
Queries the fall time of the selected waveform. Fall time is the time it takes the falling edge of a pulse to go from the upper threshold (high reference) to the lower threshold (low reference).

fall time = low cross time – high cross time

Returned Format: The measurement is returned as a numeric value representing measured fall time in seconds.

Command Syntax
None

Query Syntax
MEASure:VOLTage:FTIMe? <source> → <seconds>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><seconds></td><td>Float</td><td>Time in seconds</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Frequency Query</td><td>
Queries the frequency of the selected waveform source. All cycles in the entire capture window are used.

Command Syntax
None

Query Syntax
MEASure:VOLTage:FREQuency? <source> → <frequency>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><frequency></td><td>Float</td><td>Frequency in Hertz</td></tr>
</table>
</td></tr>
<tr><td>Measure Gate Points Command

Measure Gate Points Query</td><td>
Sets or queries the measurement of the selected waveform source using gate start and stop points.

Command Syntax
MEASure:GATE:POINts <start_points>,<stop_points>

Query Syntax
MEASure:GATE:POINts? → <start_points>,<stop_points>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><start_points></td><td>U32</td><td>0 to Maximum waveform size
MINimum 0
MAXimum Maximum waveform size</td></tr>
<tr><td><stop_points></td><td>U32</td><td>0 to Maximum waveform size
MINimum 0
MAXimum Maximum waveform size</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Gate Time Command

Measure Gate Time Query</td><td>Sets or queries the measurement of the selected waveform source using gate start and stop times relative to the trigger time.

Command Syntax
MEASure:GATE[:TIME] <gate_start>,<gate_stop>

Query Syntax
MEASure:GATE[:TIME]? → <gate_start>,<gate_stop>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><gate_start></td><td>Float</td><td>0 to Maximum waveform size
MINimum 0
MAXimum Maximum waveform size</td></tr>
<tr><td><gate_stop></td><td>Float</td><td>0 to Maximum waveform size
MINimum 0
MAXimum Maximum waveform size</td></tr>
</table></td></tr>
<tr><td>Measure High Voltage Query</td><td>Queries the high voltage level of the selected waveform source. The high voltage level measurement assumes a bi-level signal with distinct high and low levels and is defined as the waveform upper level. Use the Measure Maximum Voltage Query to detect the most positive waveform voltage level without assuming a bi-level signal.

Command Syntax
None

Query Syntax
MEASure:VOLTage:HIGH? <source> → <volts>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Low Voltage Query</td><td>Queries the low voltage level of the selected waveform source. The low voltage level measurement assumes a bi-level signal with distinct high and low levels and is defined as the waveform lower level. Use the Measure Minimum Voltage Query to detect the most negative waveform voltage level without assuming a bi-level signal.

Command Syntax
None

Query Syntax
MEASure:VOLTage:LOW? <source> → <volts>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Maximum Voltage Query</td><td>Queries the most positive voltage of the selected waveform source. The maximum level measurement is defined as the waveform most positive voltage level. Use the Measure High Voltage Query to detect the upper level in a bi-level signal such as a square wave.

Command Syntax
None

Query Syntax
MEASure:VOLTage:MAXimum? <source> → <volts>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

Name	Description
Measure Method Command **Measure Method Query**	Sets or queries the method to use for measurements. The following method types are available on the ZT412: <u>Entire Waveform</u> Entire Waveform measurement is used to perform measurements upon the entire captured waveform. <u>Gated by Time</u> Gated by Time measurement is used to perform measurements upon a waveform, where the user defines a start time and stop time. “0” represents the start of the waveform. <u>Gated by Points</u> Gated by Points measurement is used to perform measurements on a waveform, where the user defines a start point and stop point. “0” represents the start of the waveform. **Command Syntax** MEASure:METHod <method> **Query Syntax** MEASure:METHod? → <method> **Parameters:**

Name	Type	Range	
<method>	Discrete	ALL GATE	Entire waveform Gated

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Mid Voltage Query</td><td>
Queries the mid level voltage of the selected waveform source. The mid level voltage measurement assumes a bi-level signal with distinct high and low voltage levels and is defined as the waveform mid level.

Mid level = (High Level + Low Level)/2

Use the Measure Maximum Voltage Query to detect the most positive waveform voltage level, and the Measure Minimum Voltage Query to detect the most negative voltage, without assuming a bi-level signal.

Command Syntax
None

Query Syntax
MEASure:VOLTage:MID? <source> → <volts>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Minimum Voltage Query</td><td>
Queries the most negative voltage of the selected waveform source. The minimum voltage level measurement is defined as the most negative voltage level of the waveform. Use the Measure Low Voltage Query to detect the lower level in a bi-level signal such as a square wave.

Command Syntax
None

Query Syntax
MEASure:VOLTage:MINimum? <source> → <volts>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Negative Duty Cycle Query</td><td>
Queries the percent of a cycle the selected waveform is below the mid voltage value. The threshold is defined as the mid voltage level, or midway between high and low levels.

Command Syntax
None

Query Syntax
MEASure:VOLTage:NDUTycycle? <source> → <percent>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><percent></td><td>Float</td><td>0.0 (0 percent) to 1.0 (100 percent)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Negative Width Query</td><td>Queries the time that the selected waveform is below the mid voltage value. The threshold is defined as the mid voltage level, or midway between high and low levels.

Command Syntax
None

Query Syntax
MEASure:VOLTage:NWIDth? <source> → <seconds>

Parameters:

<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr><tr><td><seconds></td><td>Float</td><td>Time in seconds</td></tr></table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Peak-To-Peak Voltage Query</td><td>The Measure Peak-To-Peak Voltage Query performs a peak-to-peak voltage measurement on the specified source, and then returns the measurement results to the output buffer. The method the instrument uses to determine peak-to-peak voltage is to measure the high and low voltages, and then calculate peak-to-peak voltage as follows:

peak-to-peak voltage = high voltage – low voltage

Command Syntax
None

Query Syntax
MEASure:VOLTage:PTPeak? <source> → <volts>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><volts></td><td>Float</td><td>Voltage in Volts</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Period Query</td><td>
The Measure Period Query performs a period measurement on the specified source, and then returns the measurement results to the output buffer. The period of the signal (1/frequency) is measured using all cycles in the entire capture window.

Command Syntax
None

Query Syntax
MEASure:VOLTage:PERiod? <source> → <seconds>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><seconds></td><td>Float</td><td>Time in seconds</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Phase Query</td><td>The Measure Phase Query performs a phase measurement on the specified source. This is a measurement of the phase of a periodic signal at the start of the waveform in radians.

Command Syntax
None

Query Syntax
MEASure:VOLTage:PHASe? <source> → <phase>

Parameters:

<table>
<tr><th>Name</th><th>Type</th><th>Range</th></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><phase></td><td>Float</td><td>0 to 2*PI (6.283185307)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Positive Duty Cycle Query</td><td>Queries the percent of a cycle of the selected waveform that is above the mid voltage value. The threshold is defined as the mid voltage level, or midway between high and low levels.

Command Syntax
None

Query Syntax
MEASure:VOLTage:PDUTycycle? <source> → <percent>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><percent></td><td>Float</td><td>0.0 (0 percent) to 1.0 (100 percent)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Positive Width Query</td><td>Queries the time that the selected waveform is above the mid voltage value. The threshold is defined as the mid voltage level, or midway between high and low levels.

Command Syntax
None

Query Syntax
MEASure:VOLTage:PWIDth? <source> → <seconds>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><seconds></td><td>Float</td><td>Time in seconds</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Reference Command

Measure Reference Query</td><td>
Sets or queries the high, low, and middle reference levels that are used to take rise time, fall time, fall crossing time, and rise crossing time measurements. Reference levels are configured in relative terms of the percentage of the waveform acquired, or in absolute voltage levels (see the Measure Reference Method Command). By default, the low value is 10% of the waveform, the mid level is 50%, and the high value is 90%.
Reference levels can be set by either percentage or voltage as follows:

Low reference selects the threshold for detection of the input signal low state.
Middle reference selects the threshold for detection of the input signal middle level.
High reference selects the threshold for detection of the input signal high state.
The allowed relative reference values range from 0.0 (0 percent) to 1.0 (100 percent).
Absolute reference values are expressed in voltages.

Command Syntax
MEASure:REFerence <low_value>,<mid_value>,<high_value>
Query Syntax
MEASure:REFerence? → <low_value>,<mid_value>,<high_value>
Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><low_value></td><td>Float</td><td>Variable</td></tr>
<tr><td><mid_value></td><td>Float</td><td>Variable</td></tr>
<tr><td><high_value></td><td>Float</td><td>Variable</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Reference Method Command

Measure Reference Method Query</td><td>Sets or queries the reference method used in waveform voltage analysis. Reference methods are in absolute voltages or relative percentages.

Command Syntax
MEASure:REFerence:METHod <reference_method>

Query Syntax
MEASure:REFerence:METHod? → <reference_method>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td><td></td></tr>
<tr><td><reference_method></td><td>Discrete</td><td>ABSolute
RELative</td><td>Absolute voltages
Relative percentages</td></tr>
</table>
</td></tr>
<tr><td>Measure Rise Crossing Time Query</td><td>Queries time of the rising edge of a waveform crossing the middle reference threshold measured from the start of the waveform. The edge number is selectable.

Command Syntax
None

Query Syntax
MEASure:VOLTage:RTCRoss? <source> → <seconds>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><seconds></td><td>Float</td><td>Time in seconds</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Rise Overshoot Query</td><td>
Queries the voltage overshoot of the selected waveform. The following considerations apply when using the measure fall overshoot query:

Overshoot is defined as the amount of voltage past the high level of a bi-level signal that a signal travels as it transitions from its low state to its high state.
Overshoot is calculated as the signal (maximum voltage – high level voltage)/signal amplitude.
The value returns as a percent in a decimal. For example, a 10 percent overshoot will be returned as 0.1.

Command Syntax
None

Query Syntax
MEASure:VOLTage:ROVershoot? <source> → <percent>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><percent></td><td>Float</td><td>0.0 (0 percent) to 1.0 (100 percent)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Rise Preshoot Query</td><td>Queries the voltage preshoot of the selected waveform. The following considerations apply when using the measure preshoot query:

• Preshoot is defined as the amount of voltage past the low level of a bi-level signal that a signal travels as it transitions from its low state to its high state.

• Preshoot is calculated as the signal (minimum voltage – low level voltage)/signal amplitude.

• The value returns as a percent in a decimal. For example, a 10 percent overshoot will be returned as 0.1.

Command Syntax
None

Query Syntax
MEASure:VOLTage:RPReshoot? <source> → <percent>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><percent></td><td>Float</td><td>0.0 (0 percent) to 1.0 (100 percent)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Rise Time Query</td><td>
Performs a rise time measurement one time on the signal present, and then sends the measurement results to the output buffer.

Rise time is the time it takes the rising edge of a pulse to go from the lower threshold (low reference) to the upper threshold (high reference).

Rise time = high cross time – low cross time

Returned Format: The measurement is returned as a numeric value representing measured rise time (in seconds).

Command Syntax
None

Query Syntax
MEASure:VOLTage:RTIMe? <source> → <seconds>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><seconds></td><td>Float</td><td>Time in seconds</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Signal-to-Noise Ratio Query</td><td>
Performs a signal-to-noise ratio measurement upon a frequency-domain waveform such as a FFT calculate channel.
Signal-to-Noise Ratio (SNR) is the ratio of the RMS amplitude of the input signal fundamental to the RMS amplitude of the sum of all non-harmonic noise sources. The input signal is assumed to be a perfect single-frequency sinusoidal signal. All signal components other than the input signal fundamental are considered to be harmonic distortion or noise. SNR does NOT include the first nine (second through tenth-order) harmonics as noise. This measurement is expressed in decibels relative to carrier (dBc).
An invalid measurement code will be returned if the input sinusoidal fundamental cannot be resolved from the noise level. An invalid measurement code will also be returned if this measurement is attempted upon a non-frequency domain waveform, as identified by the waveform preamble header.
Due to the quantization level of the fixed point processing algorithm for this measurement, the three lowest-value codes in a frequency domain waveform (-32768, -32767, -32766) are not counted as signal, noise, or harmonics while performing the measurement.
Returned Format: The measurement is returned as a positive numeric value representing the measured signal-to-noise ratio in decibels relative to carrier (dBc).
Command Syntax
None
Query Syntax
MEASure:VOLTage:SNR? <source> → <dBc>
Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><dBc></td><td>Float</td><td>Ratio in dBc</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Signal-to-Noise and Distortion Ratio Query</td><td>
Performs a signal-to-noise and distortion ratio measurement upon a frequency-domain waveform such as a FFT calculate channel.
Signal-to-Noise and Distortion Ratio (SINAD) is the ratio of the RMS amplitude of the input signal fundamental to the RMS amplitude of the sum of all noise and distortion sources. The input signal is assumed to be a perfect single-frequency sinusoidal signal. All signal components other than the input signal fundamental are considered to be harmonic distortion or noise. SINAD is equivalent to the RMS sum of SNR and THD. This measurement is expressed in decibels relative to carrier (dBc).
An invalid measurement code will be returned if the input sinusoidal fundamental cannot be resolved from the noise level. An invalid measurement code will also be returned if this measurement is attempted upon a non-frequency domain waveform, as identified by the waveform preamble header.
Due to the quantization level of the fixed point processing algorithm for this measurement, the three lowest-value codes in a frequency domain waveform (-32768, -32767, -32766) are not counted as signal, noise, or harmonics while performing the measurement.
Returned Format: The measurement is returned as a positive numeric value representing the measured signal-to-noise and distortion ratio in decibels relative to carrier (dBc).
Command Syntax
None
Query Syntax
MEASure:VOLTage:SNDR? <source> → <dBc>
Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><dBc></td><td>Float</td><td>Ratio in dBc</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Spurious-Free Dynamic Range Query</td><td>
Performs a spurious-free dynamic range measurement upon a frequency-domain waveform such as a FFT calculate channel.
Spurious-Free Dynamic Range (SFDR) is the ratio of the RMS amplitude of the input signal fundamental to the RMS amplitude of the largest spurious signal. The spurious signal can be either a harmonic or non-harmonic of the input signal fundamental. The input signal is assumed to be a perfect single-frequency sinusoidal signal. All signal components other than the input signal fundamental are considered to be spurious signals. This measurement is expressed in decibels relative to carrier (dBc).
An invalid measurement code will be returned if the input sinusoidal fundamental cannot be resolved from the noise level. An invalid measurement code will also be returned if this measurement is attempted upon a non-frequency domain waveform, as identified by the waveform preamble header.
Due to the quantization level of the fixed point processing algorithm for this measurement, the three lowest-value codes in a frequency domain waveform (-32768, -32767, -32766) are not counted as signal, noise, or harmonics while performing the measurement.
Returned Format: The measurement is returned as a positive numeric value representing the measured spurious-free dynamic range in decibels relative to carrier (dBc).
Command Syntax
None
Query Syntax
MEASure:VOLTage:SFDR? <source> → <dBc>
Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><dBc></td><td>Float</td><td>Ratio in dBc</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Time/Frequency of Maximum Voltage Query</td><td>Returns the time in seconds at which the first maximum voltage occurred on the acquired waveform. Zero seconds corresponds to the first point in the waveform. If the measurement is being performed upon a calculate FFT waveform, the result is the frequency of the maximum magnitude in Hertz.

Command Syntax
None

Query Syntax
MEASure:VOLTage:TMAXimum? <source> → <seconds>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><seconds></td><td>Float</td><td>Time in seconds</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Time/Frequency of Minimum Voltage Query</td><td>Returns the time at which the first minimum voltage occurred on the acquired waveform. Zero seconds corresponds to the first point in the waveform. If the measurement is being performed upon a calculate FFT waveform, the result is the frequency of the minimum magnitude in Hertz.

Command Syntax
None

Query Syntax
MEASure:VOLTage:TMINimum? <source> → <seconds>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><seconds></td><td>Float</td><td>Time in seconds</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Measure Total Harmonic Distortion Query</td><td>
Performs a total harmonic distortion measurement upon a frequency-domain waveform such as a FFT calculate channel.
Total Harmonic Distortion (THD) is the ratio of the RMS amplitude of the sum of the first nine (second through tenth-order) harmonics to the RMS amplitude of the input signal fundamental. The input signal is assumed to be a perfect single-frequency sinusoidal signal. All signal components other than the input signal fundamental are considered to be harmonic distortion or noise. This measurement is expressed in decibels relative to carrier (dBc).
An invalid measurement code will be returned if the input sinusoidal fundamental cannot be resolved from the noise level. An invalid measurement code will also be returned if this measurement is attempted upon a non-frequency domain waveform, as identified by the waveform preamble header.
Due to the quantization level of the fixed point processing algorithm for this measurement, the three lowest-value codes in a frequency domain waveform (-32768, -32767, -32766) are not counted as signal, noise, or harmonics while performing the measurement.
Returned Format: The measurement is returned as a negative numeric value representing the measured total harmonic distortion in decibels relative to carrier (dBc).
Command Syntax
None
Query Syntax
MEASure:VOLTage:THD? <source> → <dBc>
Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
<tr><td><dBc></td><td>Float</td><td>Distortion in dBc</td></tr>
</table>
</td></tr>
</table>

Name	Description
Output ECL Trigger Polarity Command **Output ECL Trigger Polarity Query**	Sets or queries the unit VXIbus ECLTRGn output driver polarity. The following considerations apply:

- When positive output polarity is selected, the VXIbus output driver will force logic 1 onto the ECL output when the signal source is active; i.e. for a POSitive polarity setting and an ARM source signal, the ECL trigger line will be logic 1 when the unit is armed.
- Output driver polarity does not affect VXIbus ECLTRG line sensing used by other unit functions.
- Each output line polarity is selected individually
- ECLTRG lines can be sourced and sensed simultaneously.

Command Syntax
OUTPut:ECLTrg<n>:POLarity <polarity>

Query Syntax
OUTPut:ECLTrg<n>:POLarity?

Parameters

Name	Type	Range	
<n>	U16	0 1	ECLTRG0 ECLTRG1
<polarity>	Discrete	POSitive NEGative	Positive polarity Negative polarity

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Output ECL Trigger Source Command

Output ECL Trigger Source Query</td><td>Sets or queries the unit VXIbus ECLTRGn output driver source. The following considerations apply:

• The output driver may be enabled or disabled and the polarity selected; refer to the OUTPut:ECLTrg<n>:POLarity and OUTPut:ECLTrg<n>:SOURce commands.

• Each output line source is selected individually

• ECLTRG lines can be sourced and sensed simultaneously.

Command Syntax
OUTPut:ECLTrg<n>:SOURce <source>

Query Syntax
OUTPut:ECLTrg<n>:SOURce?

Parameters

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>0 ECLTRG0
1 ECLTRG1</td></tr>
<tr><td><source></td><td>Discrete</td><td>TRIGger Trigger event
CONStant Constant event
OPC Operation complete event
ARM Arm event</td></tr>
</table>
</td></tr>
</table>

Name	Description
Output ECL Trigger State Command **Output ECL Trigger State Query**	Sets or queries the unit VXIbus ECLTRGn output driver state. The following considerations apply: • The output source and polarity are selectable. • Each output line driver state is selected individually. • ECLTRG lines can be sourced and sensed simultaneously. • When OFF, the output remains in an inactive state. **Command Syntax** OUTPut:ECLTrg<n>[:STATe] <state> **Query Syntax** OUTPut:ECLTrg<n>[:STATe]? **Parameters**

Name	Type	Range
<n>	U16	0 ECLTRG0 1 ECLTRG1
<state>	Discrete	ON or 1 Enable OFF or 0 Disable

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Output Reference Source Command

Output Reference Source Query</td><td>Sets or queries the unit multi-function front-panel reference output signal source, REF OUT. The reference output can be selected from the following sources:

• a precision +8V voltage reference
• the selected 10 MHz TTL reference oscillator output
• a 500 Hz TTL probe compensation output
• the TTL Trigger event
• a 10 ns TTL pulse at 1 ms repetition interval
• the TTL Arm event

Command Syntax
OUTPut:REFerence:SOURce <source>

Query Syntax
OUTPut:REFerence:SOURce?

Parameters

<table>
<tr><td>Name</td><td>Type</td><td colspan="2">Range</td></tr>
<tr><td><source></td><td>Discrete</td><td>VOLTage
REFerence
COMPensate
TRIGger
PULSe
ARM</td><td>Voltage
Reference
Compensate
Trigger
Pulse
Arm</td></tr>
</table></td></tr>
<tr><td>Output Reference State Command

Output Reference State Query</td><td>Sets or queries the unit output reference state. When disabled, the reference output sources a TTL low signal level.

Command Syntax
OUTPut:REFerence[:STATe] <state>

Query Syntax
OUTPut:REFerence[:STATe]?

Parameters

<table>
<tr><td>Name</td><td>Type</td><td colspan="2">Range</td></tr>
<tr><td><state></td><td>Discrete</td><td>ON or 1
OFF or 0</td><td>Enable
Disable</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Output TTL Trigger Polarity Command

Output TTL Trigger Polarity Query</td><td>Sets or queries the unit VXIbus TTLTRGn output driver polarity. The following considerations apply:

When positive output polarity is selected, the output driver will force logic 0 onto the TTL output when the signal source is active, i.e. for a positive polarity setting and an Arm source signal, the TTL trigger line will be logic 0 when the unit is armed.
Output driver polarity does not affect VXIbus TTLTRG line sensing used by other unit functions.
Each output line polarity is selected individually
TTLTRG lines can be sourced and sensed simultaneously.

Command Syntax
OUTPut:TTLTrg<n>:POLarity <polarity>

Query Syntax
OUTPut:TTLTrg<n>:POLarity? → <polarity>

Parameters
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td>TTLTrg<n></td><td>U16</td><td>VXIbus TTL trigger line, where <n> may be 0, 1, 2, 3, 4, 5, 6, or 7</td></tr>
<tr><td><polarity></td><td>Discrete</td><td>NEGative Negative polarity
POSitive Positive polarity</td></tr>
</table>
</td></tr>
</table>

Name	Description
Output TTL Trigger Source Command **Output TTL Trigger Source Query**	Sets or queries the unit VXIbus TTLTRGn output driver source. The following considerations apply: • The output driver may be enabled or disabled and the polarity selected. • Each output line source is selected individually • TTLTRG lines can be sourced and sensed simultaneously. **Command Syntax** OUTPut:TTLTrg<n>:SOURce <source> **Query Syntax** OUTPut:TTLTrg<n>:SOURce? → <source> **Parameters**

Name	Type	Range
TTLTrg<n>	U16	VXIbus TTL trigger line, where <n> may be 0, 1, 2, 3, 4, 5, 6, or 7
<source>	Discrete	TRIGger Trigger event CONStant Constant event OPC Operation complete event ARM Arm event

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Output TTL Trigger State Command

Output TTL Trigger State Query</td><td>Sets or queries the unit VXIbus TTLTRGn output driver state. The following considerations apply:
The output source and polarity are selectable.
Each output line driver state is selected individually
TTLTRG lines can be sourced and sensed simultaneously.
Command Syntax
OUTPut:TTLTrg<n>[:STATe] <state>

Query Syntax
OUTPut:TTLTrg<n>[:STATe]? → <state>

Parameters
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td>TTLTrg<n></td><td>U16</td><td>VXIbus TTL trigger line, where <n> may be 0, 1, 2, 3, 4, 5, 6, or 7</td></tr><tr><td><state></td><td>Discrete</td><td>ON or 1 Active state
OFF or 0 Inactive state</td></tr></table></td></tr>
<tr><td>Reference Oscillator Source Command

Reference Oscillator Source Query</td><td>Sets or queries the source for the 10 MHz reference clock that provides the instrument time base.

Command Syntax
[SENSe]:ROSCillator:SOURce <source>

Query Syntax
[SENSe]:ROSCillator:SOURce? → <source>

Parameters:
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><source></td><td>Discrete</td><td>INTernal Local reference selected
VXI VXI reference selected</td></tr></table></td></tr>
</table>

Name	Description
Status Calibration Condition Query	Queries the contents of the Questionable Calibration Condition Register, represented by <condition> in the parameters below. The Questionable Calibration Condition Register identifies current questionable results from all internally-generated calibration conditions. The following considerations apply when using the Status Calibration Condition Query:

- The Questionable Calibration Condition Register identifies current conditions. Use the Status Questionable Calibration Event Query to identify a history of which calibration conditions have failed since the last event status check.
- The Status Questionable Calibration Condition Query does not clear the Questionable Calibration Condition Register.

Command Syntax
None

Query Syntax
STATus:QUEStionable:CALibration:CONDition? → <condition>

Parameters:

Name	Type	Range
<condition>	U16	0 to 65535 Bit 0 Calibration storage failed Bit 1 Zero DC Offset cal failed Bit 2 DC Offset calibration failed Bit 3 ADC Null balance cal failed Bit 4 ADC Gain balance cal failed Bit 5 Sample rate null cal failed Bits 6–15 Unused

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Status Calibration Enable Command

Status Calibration Enable Query</td><td>Sets or queries the contents of the Questionable Calibration Enable Register. The Questionable Calibration Enable Register enables the reporting of questionable calibration events to the Questionable Summary Register. The following considerations apply when using the Status Calibration Enable Command/Query:

The Questionable Calibration Enable Register is a bit mask that allows selected questionable calibration events to be reported to the Questionable Summary Register.
Only low to high (inactive to active) Questionable Calibration Enable Register bit transitions are reported.
Questionable calibration events are reported in bit 8 of the Questionable Summary Register.
The Status Preset Command sets all 15 LSBs to one (1) which enables all event reporting.

Command Syntax
STATus:QUEStionable:CALibration:ENABle <enable>

Query Syntax
STATus:QUEStionable:CALibration:ENABle? → <enable>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><enable></td><td>U16</td><td>0 to 65535
Bit 0 Calibration storage failed
Bit 1 Zero DC Offset cal failed
Bit 2 DC Offset calibration failed
Bit 3 ADC Null balance cal failed
Bit 4 ADC Gain balance cal failed
Bit 5 Sample rate null cal failed
Bits 6–15 Unused</td></tr>
</table>
</td></tr>
</table>

Name	Description
Status Calibration Event Query	Queries the Questionable Calibration Event Register. The Questionable calibration Event Register identifies calibration processes that have completed with questionable results. The following considerations apply when using the Status Calibration Event Query: • The Questionable Calibration Event Register records the history of the questionable calibration process results generated since the previous Questionable Calibration Event Query. • The Status Questionable Event Query clears the Questionable Frequency Event Register after returning the current register contents. • Questionable calibration events are reported in bit 8 of the Questionable Summary Register. • In order to identify questionable results from a particular process, the Questionable Calibration Event Register must be cleared by reading it before the processes are run. **Command Syntax** None **Query Syntax** STATus:QUEStionable:CALibration [:EVENt]? → <event> **Parameters:**

Name	Type	Range	
<event>	U16	0 to 65535	
		Bit 0	Calibration storage failed
		Bit 1	Zero DC Offset cal failed
		Bit 2	DC Offset calibration failed
		Bit 3	ADC Null balance cal failed
		Bit 4	ADC Gain balance cal failed
		Bit 5	Sample rate null cal failed
		Bits 6–15	Unused

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Status Frequency Condition Query</td><td>
Queries the contents of the Questionable Frequency Condition Register, represented by <condition> in the parameters below. The Questionable Frequency Condition Register identifies current questionable results from all internally-generated clock frequency conditions. The following considerations apply when using the Status Frequency Condition Query:

The Questionable Frequency Condition Register identifies current conditions. Use the Status Questionable Frequency Event Query to identify a history of which frequency conditions have failed since the last event status check.
The Status Frequency Condition Query does <u>not</u> clear the Questionable Frequency Condition Register.

Command Syntax
None
Query Syntax
STATus:QUEStionable:FREQuency:CONDition? → <condition>
Parameters
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><condition></td><td>U16</td><td>0 to 65535
Bit 0 PLL Unlocked
Bits 1–15 Unused</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Status Frequency Enable Command

Status Frequency Enable Query</td><td>Sets or queries the contents of the Questionable Frequency Enable Register. The Questionable Frequency Enable Register enables the reporting of questionable frequency events to the Questionable Summary Register. The following considerations apply when using the Status Frequency Enable Command/Query:

• The Questionable Frequency Enable Register is a bit mask that allows selected questionable frequency events to be reported to the Questionable Summary Register.

• Only low to high (inactive to active) Questionable Frequency Enable Register bit transitions are reported.

• Questionable frequency events are reported in bit 5 of the Questionable Summary Register.

• The Status Preset Command sets all 15 LSBs to one (1), which enables all event reporting.

Command Syntax
STATus:QUEStionable:FREQuency:ENABle <enable>

Query Syntax
STATus:QUEStionable:FREQuency:ENABle? → <enable>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><enable></td><td>U16</td><td>0 to 65535
Bit 0 PLL Unlocked
Bits 1–15 Unused</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Status Frequency Event Query</td><td>
Queries the Questionable Frequency Event Register. The Questionable Frequency Event Register identifies frequency processes that have completed with questionable results. The following considerations apply when using the Status Frequency Event Query:

• The Questionable Frequency Event Register records the history of the questionable frequency process results generated since the previous questionable frequency event query.

• The Status Questionable Event Query clears the Questionable Frequency Event Register after returning the current register contents.

• Questionable frequency events are reported in bit 5 of the Questionable Summary Register.

• In order to identify questionable results from a particular process, the Questionable Frequency Event Register <u>must</u> be cleared by reading it before the processes are run.

Command Syntax
None

Query Syntax
STATus:QUEStionable:FREQuency[:EVENt]? → <event>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><event></td><td>U16</td><td>0 to 65535
Bit 0 PLL Unlocked
Bit 0 PLL Unlocked
Bits 1–15 Unused</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Status Operation Condition Query</td><td>
Queries the contents of the Operation Status Condition Register. The Operation Status Condition Register identifies currently running processes, such as waveform acquisition. The following considerations apply when using the Status Operation Condition Query:

The Operation Status Condition Register identifies current running processes. Use the Status Operation Event Query to identify a history of which processes have run since the last operation event status check.
The Status Operation Condition Query does <u>not</u> clear the Operation Status Condition Register

Command Syntax
None

Query Syntax
STATus:OPERation:CONDition? → <condition>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><condition></td><td>U16</td><td>0 to 65535
Bit 0 Calibrating
Bit 1 Settling
Bit 2 Ranging
Bit 3 Sweeping
Bit 4 Measuring
Bit 5 Waiting for trigger
Bit 6 Waiting for arm
Bit 7 Unused
Bit 8 Trigger event
Bit 9 Data capture event
Bit 10 Limit test event
Bits 11–15 Unused</td></tr>
</table>
</td></tr>
</table>

Name	Description
Status Operation Enable Command **Status Operation Enable Query**	Sets or queries the contents of the Operation Status Enable Register. The Operation Status Enable Register enables the reporting of operation events to the Status Byte. The following considerations apply when using the Status Operation Enable Command/Query:

- The Operation Status Enable Register is a bit mask that allows selected operation status events to be reported to the Status Byte.
- Only low to high (inactive to active) Operation Status Event Register bit transitions are reported.
- Operation status events report in bit 7 of the Status Byte.
- The Status Preset Command sets all register bits to zero (0), which disables all operation event reporting.

Command Syntax
STATus:OPERation:ENABle <enable>

Query Syntax
STATus:OPERation:ENABle? → <enable>

Parameters:

Name	Type	Range	
<enable>	U16	0 to 65535	
		Bit 0	Calibrating
		Bit 1	Settling
		Bit 2	Ranging
		Bit 3	Sweeping
		Bit 4	Measuring
		Bit 5	Waiting for trigger
		Bit 6	Waiting for arm
		Bit 7	Unused
		Bit 8	Trigger event
		Bit 9	Data capture event
		Bit 10	Limit test event
		Bits 11–15	Unused

Name	Description
Status Operation Event Query	Queries the contents of the Operation Event Status Register. The Operation Event Status Register identifies unit processes that have been run, such as waveform acquisition. The following considerations apply when using the Status Operation Event Query: • The Operation Event Status Register records the history of the processes that have been run since the previous operation event status query. Use the Status Operation Condition Query to identify currently running processes • The Status Operation Event Query clears the Operation Status Event Register after returning the current register contents • Operation Status Event Register bits may be summarized in the Status Byte. • In order to identify which processes have run between two times, ex. acquisition start and later status check, the Operation Event Status Register must be cleared by reading it before the processes are run. **Command Syntax** None **Query Syntax** STATus:OPERation[:EVENt]? → <event> **Parameters:**

Name	Type	Range	
<event>	U16	0 to 65535	
		Bit 0	Calibrating
		Bit 1	Settling
		Bit 2	Ranging
		Bit 3	Sweeping
		Bit 4	Measuring
		Bit 5	Waiting for trigger
		Bit 6	Waiting for arm
		Bit 7	Unused
		Bit 8	Trigger event
		Bit 9	Data capture event
		Bit 10	Limit test event
		Bits 11–15	Unused

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Status Preset Command</td><td>Sets the status reporting event enable data structures to a known state. The condition and event register contents are not affected. All device-dependent status registers which cascade events into the Questionable Status and the Operation Status Registers are enabled by setting those device-dependent event enable registers to $7FFF_{16}$ (the 15 LSBs set). The IEEE-488.2 mandatory status data structures are disabled by setting the Questionable Status and Operation Status event enable registers to 0000_{16}. The Status Byte and Standard Event Status Registers as defined by IEEE 488.2 are not affected.

Command Syntax
STATus:PRESet

Query Syntax
None

Parameters:
None</td></tr>
<tr><td>Status Questionable Condition Query</td><td>Queries the contents of the Questionable Status Condition Register. The Questionable Status Condition Register identifies current questionable results from running processes, such as self-test. The following considerations apply when using the questionable condition status query:

• The Questionable Status Condition Register identifies current questionable results from running processes. Use the Status Questionable Event Query to identify which questionable results generated since the last questionable event status check.

• The Status Questionable Condition Query does <u>not</u> clear the Questionable Status Condition Register

Command Syntax
None

Query Syntax
STATus:QUEStionable:CONDition? → <condition>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><condition></td><td>U16</td><td>0 to 65535
Bit 0 Voltage bit
Bits 1–4 Unused
Bit 5 Frequency bit
Bits 6–7 Unused
Bit 8 Calibration bit
Bit 9 Test bit
Bits 10–15 Unused</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr>
<td>Status Questionable Enable Command

Status Questionable Enable Query</td>
<td>Sets or queries the contents of the Questionable Status Enable Register. The Questionable Status Enable Register enables the reporting of questionable events to the Status Byte. The following considerations apply when using the Status Questionable Enable Command/Query:

The Questionable Status Enable Register is a bit mask that allows selected questionable status events to be reported to the Status Byte.
Only low to high (inactive to active) Questionable Status Event Register bit transitions are reported.
Questionable status events are reported in bit 3 of the Status Byte. Refer to the Status Byte query.
Status Preset Command sets all register bits to zero (0) which disables all questionable event reporting.

Command Syntax
STATus:QUEStionable:ENABle <enable>

Query Syntax
STATus:QUEStionable:ENABle? → <enable>

Parameters:
<table>
<tr><th>Name</th><th>Type</th><th>Range</th></tr>
<tr><td><enable></td><td>U16</td><td>0 to 65535
Bit 0 Voltage bit
Bits 1–4 Unused
Bit 5 Frequency bit
Bits 6–7 Unused
Bit 8 Calibration bit
Bit 9 Test bit
Bits 10–15 Unused</td></tr>
</table>
</td>
</tr>
</table>

Name	Description
Status Questionable Event Query	Queries the Questionable Status Event Register. The Questionable Event Status Register identifies unit processes that have completed with questionable results, such as self test errors. The following considerations apply when using the Status Questionable Event Query:

- The Questionable Status Event Register records the history of the questionable process results generated since the previous questionable event status query.
- The Status Questionable Event query clears the Questionable Status Event Register after returning the current register contents.
- Questionable Status Event Register bits may be summarized in the Status Byte.
- In order to identify questionable results from a particular process, the Questionable Status Event Register must be cleared by reading it before the processes are run.

Command Syntax
None

Query Syntax
STATus:QUEStionable[:EVENt]? → <event>

Parameters:

Name	Type	Range
<event>	U16	0 to 65535 Bit 0 Voltage bit Bits 1–4 Unused Bit 5 Frequency bit Bits 6–7 Unused Bit 8 Calibration bit Bit 9 Test bit Bits 10–15 Unused

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Status Test Condition Query</td><td>
Queries the contents of the Questionable Test Condition Register. The Questionable Test Condition Register identifies the test results of memory (RAM, DRAM, and Flash) tests, along with register and PLL tests. The following considerations apply when using the Status Test Condition Query:

The Questionable Test Condition Register identifies current tests. Use the Status Questionable Test Condition Query to identify a history of which tests have failed since the last test status check.
Questionable test events report in bit 9 of the Questionable Summary Register.
The Status Questionable Test Condition Query does not clear the Questionable Test Condition Register.

Command Syntax
None
Query Syntax
STATus:QUEStionable:TEST:CONDition? → <condition>
Parameters:
<table>
<tr><td>Name</td><td>Type</td><td colspan="2">Range</td></tr>
<tr><td><condition></td><td>U16</td><td colspan="2">0 to 65535</td></tr>
<tr><td></td><td></td><td>Bit 0</td><td>Baseboard test failed</td></tr>
<tr><td></td><td></td><td>Bit 1</td><td>SRAM test failed</td></tr>
<tr><td></td><td></td><td>Bit 2</td><td>ROM test failed</td></tr>
<tr><td></td><td></td><td>Bit 3</td><td>Unused</td></tr>
<tr><td></td><td></td><td>Bit 4</td><td>Ref oscillator test failed</td></tr>
<tr><td></td><td></td><td>Bit 5</td><td>DRAM test failed</td></tr>
<tr><td></td><td></td><td>Bit 6</td><td>Flash memory test failed</td></tr>
<tr><td></td><td></td><td>Bit 7</td><td>Unused</td></tr>
<tr><td></td><td></td><td>Bit 8</td><td>Input1–2 register test failed</td></tr>
<tr><td></td><td></td><td>Bit 9</td><td>Input1 RAM test failed</td></tr>
<tr><td></td><td></td><td>Bit 10</td><td>Input2 RAM test failed</td></tr>
<tr><td></td><td></td><td>Bit 11</td><td>PLL test failed</td></tr>
<tr><td></td><td></td><td>Bit 12</td><td>Input3–4 register test failed</td></tr>
<tr><td></td><td></td><td>Bit 13</td><td>Input3 RAM test failed</td></tr>
<tr><td></td><td></td><td>Bit 14</td><td>Input4 RAM test failed</td></tr>
<tr><td></td><td></td><td>Bit 15</td><td>Unused</td></tr>
</table>
</td></tr>
</table>

Name	Description
Status Test Enable Command **Status Test Enable Query**	Sets or queries the contents of the Questionable Test Status Register. The Questionable Test Status Register enables the reporting of questionable events to the Status Byte. The following considerations apply when using the Status Test Enable Command/Query: • The Questionable Test Status Register is a bit mask that allows selected questionable self-test events to be reported to the Status Byte. • Only low to high (inactive to active) Questionable Test Status Register bit transitions are reported. • Questionable test status events report in bit 9 of the Questionable Summary Register. • The Status Preset Command sets the 15 LSB enable register bits to “1”, which enables all test event reporting. **Command Syntax** STATus:QUEStionable:TEST:ENABle <enable> **Query Syntax** STATus:QUEStionable:TEST:ENABle? → <enable> **Parameters:**

Name	Type	Range	
<enable>	U16	0 to 65535	
		Bit 0	Baseboard test failed
		Bit 1	SRAM test failed
		Bit 2	ROM test failed
		Bit 3	Unused
		Bit 4	Ref oscillator test failed
		Bit 5	DRAM test failed
		Bit 6	Flash memory test failed
		Bit 7	Unused
		Bit 8	Input1–2 register test failed
		Bit 9	Input1 RAM test failed
		Bit 10	Input2 RAM test failed
		Bit 11	PLL test failed
		Bit 12	Input3–4 register test failed
		Bit 13	Input3 RAM test failed
		Bit 14	Input4 RAM test failed
		Bit 15	Unused

Name	Description
Status Test Event Query	Queries the Questionable Test Status Register. The Questionable Test Status Register identifies unit self tests that have completed with questionable results, such as self test errors. The following considerations apply when using the Status Test Event Query: • The Questionable Test Status Register records the history of the questionable process results generated since the previous questionable event status query. • The Status Questionable Event Query clears the Questionable Status Event Register after returning the current register contents. • Questionable Test Status Event Register reports in bit 9 of the Questionable Summary Register. • In order to identify questionable results from a particular process, the Questionable Status Event Register must be cleared by reading it before the processes are run. **Command Syntax** None **Query Syntax** STATus:QUEStionable:TEST[:EVENt]? → <event> **Parameters:**

Name	Type	Range	
<event>	U16	0 to 65535	
		Bit 0	Baseboard test failed
		Bit 1	SRAM test failed
		Bit 2	ROM test failed
		Bit 3	Unused
		Bit 4	Ref oscillator test failed
		Bit 5	DRAM test failed
		Bit 6	Flash memory test failed
		Bit 7	Unused
		Bit 8	Input1–2 register test failed
		Bit 9	Input1 RAM test failed
		Bit 10	Input2 RAM test failed
		Bit 11	PLL test failed
		Bit 12	Input3–4 register test failed
		Bit 13	Input3 RAM test failed
		Bit 14	Input4 RAM test failed
		Bit 15	Unused

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Status Voltage Condition Query</td><td>Queries the contents of the Questionable Voltage Condition Register. The Questionable Voltage Condition Register identifies the voltage overages for the input channels. The following considerations apply when using the Status Voltage Condition Query:

• The Questionable Voltage Condition Register identifies voltage overloads. Use the Status Questionable Voltage Condition Query to identify a history of which voltages have had overloads since the last voltage status check.

• Questionable voltage events report in bit 0 of the Questionable Summary Register.

• The Status Questionable Voltage Condition Query does <u>not</u> clear the Questionable Voltage Condition Register.

Command Syntax
None

Query Syntax
STATus:QUEStionable:VOLTage:CONDition? → <condition>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><condition></td><td>U16</td><td>0 to 65535
Bit 0 Input1 overload
Bit 1 Input2 overload
Bit 2 Input3 overload
Bit 3 Input4 overload
Bit 4 Input1 overvoltage
Bit 5 Input2 overvoltage
Bit 6 Input3 overvoltage
Bit 7 Input4 overvoltage
Bits 8–15 Unused</td></tr>
</table>
</td></tr>
</table>

Name	Description
Status Voltage Enable Command **Status Voltage Enable Query**	Sets or queries the contents of the Questionable Voltage Status Register. The Questionable Voltage Status Register enables the reporting of questionable events to the Status Byte. The following considerations apply when using the Status Voltage Enable Command/Query: • The Questionable Voltage Status Register is a bit mask that allows selected questionable voltage overload and over voltage events to be reported to the Status Byte. • Only low to high (inactive to active) Questionable Voltage Status Register bit transitions are reported. • Questionable voltage status events report in bit 0 of the Questionable Summary Register. • The Status Preset Command sets the 15 LSB enable register bits to “1”, which enables all voltage overload and over voltage reporting. **Command Syntax** STATus:QUEStionable:VOLTage:ENABle <enable> **Query Syntax** STATus:QUEStionable:VOLTage:ENABle? → <enable> **Parameters:**

Name	Type	Range
<enable>	U16	0 to 65535 Bit 0 Input1 overload Bit 1 Input2 overload Bit 2 Input3 overload Bit 3 Input4 overload Bit 4 Input1 overvoltage Bit 5 Input2 overvoltage Bit 6 Input3 overvoltage Bit 7 Input4 overvoltage Bits 8–15 Unused

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Status Voltage Event Query</td><td>
Queries the Questionable Voltage Status Register. The Questionable Voltage Status Register identifies unit voltage overloads and over voltages that have completed with questionable results. The following considerations apply when using the Status Voltage Event Query:

The Questionable Voltage Status Register records the history of the questionable process results generated since the previous questionable event status query.
The Status Questionable Event Query clears the Questionable Status Event Register after returning the current register contents.
Questionable Voltage Status Event Register reports in bit 0 of the Questionable Summary Register.
In order to identify questionable results from a particular process, the Questionable Status Event Register <u>must</u> be cleared by reading it before the processes are run.

Command Syntax
None
Query Syntax
STATus:QUEStionable:VOLTage [:EVENt]? → <event>
Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><event></td><td>U16</td><td>0 to 65535
Bit 0 Input1 overload
Bit 1 Input2 overload
Bit 2 Input3 overload
Bit 3 Input4 overload
Bit 4 Input1 overvoltage
Bit 5 Input2 overvoltage
Bit 6 Input3 overvoltage
Bit 7 Input4 overvoltage
Bits 8–15 Unused</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Sweep Mode Command

Sweep Mode Query</td><td>Sets or queries the trigger mode to enable automatic triggering in absence of a trigger event.

Command Syntax
[SENSe]:SWEep:MODE <trigger_mode>

Query Syntax
[SENSe]:SWEep:MODE? → <trigger_mode>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td><td></td></tr>
<tr><td><trigger_mode></td><td>Discrete</td><td>AUTO
NORMal</td><td>Automatic triggering
Normal triggering</td></tr>
</table></td></tr>
</table>

Name	Description
Sweep Offset Reference Command **Sweep Offset Reference Query**	Sets or queries the waveform record offset reference location. The following considerations apply: • The waveform offset reference can be considered a “handle” on the record. The waveform offset reference location is the location of the handle relative to the start of the waveform record. • The offset reference is used by the *Sweep Offset Time Command* to move the record relative to the trigger event. • An offset reference value of 0.0 places the handle at the first point of the record; a value of 0.5 selects the mid point; and a value of 1.0 selects the last point. • When the offset time is set to 0.0, a reference location of 0.0 will place the trigger event at the waveform record start, a reference location of 1.0 will place the trigger event at the waveform record end. • All captured waveforms use the same offset reference. **Command Syntax** [SENSe]:SWEep:OREFerence:LOCation <percent> **Query Syntax** [SENSe]:SWEep:OREFerence:LOCation? → <percent> **Parameters:**

Name	Type	Range
<percent>	Float	0.0 (0 percent) to 1.0 (100 percent) MINimum 0.0 (start of waveform) MAXimum 1.0 (end of waveform)

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Sweep Offset Time Command

Sweep Offset Time Query</td><td>Sets or queries the time between the sweep offset reference and the trigger event. The following considerations apply when using the sweep offset time command:

The offset time is the time between the trigger event and the offset reference, refer to the Sweep Offset Reference Command for more information on the offset reference.
Offset times move the offset reference after the trigger event.
The time of the last sample taken may be calculated as:

Last Sample Time = offset time + sweep time * (1 – offset reference location)
All channels use the same sweep offset time

Command Syntax
[SENSe]:SWEep:OFFSet:TIME <seconds>

Query Syntax
[SENSe]:SWEep:OFFSet:TIME? → <seconds>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><seconds></td><td>Float</td><td>0 to 655 seconds
Resolution: See table below</td></tr>
</table>
Resolution:
<table>
<tr><td>Resolution</td><td>Time in Seconds</td></tr>
<tr><td>10 ns</td><td>0 to 655.36 µs</td></tr>
<tr><td>100 ns</td><td>655.36 µs to 6.5536 ms</td></tr>
<tr><td>1 µs</td><td>6.5536 ms to 65.536 ms</td></tr>
<tr><td>10 µs</td><td>65.536 ms to 655.36 ms</td></tr>
<tr><td>100 µs</td><td>655.36 ms to 6.5536s</td></tr>
<tr><td>1 ms</td><td>6.5536s to 65.536s</td></tr>
<tr><td>10 ms</td><td>65.536s to 655 s</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Sweep Points Command

Sweep Points Query</td><td>Sets or queries the number of samples in a waveform record. The range of points varies with the size of the installed digitizer memory. The following considerations apply when using the sweep points command:
The minimum record length is 100 data points.
The maximum record length is the size of the digitizer memory. The maximum record size is further limited by the number of active channels.
When in 4 channel mode, each channel may use up to half of the full digitizer memory size. When using two-channel interleaved mode, each of two channels may use up to the full digitizer memory size.
Note: The capture circuitry requires a small number of samples in the digitizer memory for housekeeping. At maximum waveform sizes, bad samples may be returned at the beginning of the waveform record.
Use the Sweep Rate Query to read the current sample rate in samples per second.
All active channels share the same record length setting.
Command Syntax
[SENSe]:SWEep:POINts <points>

Query Syntax
[SENSe]:SWEep:POINts? → <points>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><points></td><td>U32</td><td>100 to N, where N is the maximum memory size
MINimum 100
MAXimum Maximum memory size</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Sweep Time Query</td><td>Queries the time span or duration of the waveform acquisition gate. The following considerations apply when using the sweep time query:

• The minimum sweep time is 50 ns; the maximum sweep time is N/10 KS/s, where N is the maximum memory.

• All channels share the same sweep time span.

Command Syntax
None

Query Syntax
[SENSe]:SWEep:TIME? → <seconds>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><seconds></td><td>Float</td><td>50 ns to N/10 KS/s, where N is the maximum memory
MINimum 50 ns
MAXimum N/10 KS/s, where N is the maximum memory</td></tr>
</table>
</td></tr>
<tr><td>System Delay Bypass Command

System Delay Bypass Query</td><td>Sets or queries the system delay bypass condition. The system delay causes wait states on the instrument to allow specific hardware changes to settle when control commands are issued. This ensures that the instrument hardware is at the proper state before returning from the control command. When bypassed, the wait state is disabled and the hardware may not have settled to its new condition when the instrument returns from the control command. This command is not recommended under normal operating conditions. The delay bypass state is always reset (delay enabled) upon a Reset Command.

Command Syntax
SYSTem:DELay:BYPass <state>

Query Syntax
SYSTem:DELay:BYPass? → <state>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><state ></td><td>Discrete</td><td>ON or 1 Active State (delay bypassed)
OFF or 0 Inactive state (delay enabled)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>System Error All Query</td><td>Returns all 32 entries in the error log and clears the error log. Multiple errors are stored sequentially in the error log with the oldest error first. A zero value is returned for all non-error entries when there are less than 32 errors stored in the error log.

Command Syntax
None

Query Syntax
SYSTem:ERRor:ALL? → <error_number>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><error_number></td><td>S16</td><td>0 to –32768
0 No error

Note: See Appendix 3, Error Table, for a description of errors.</td></tr>
</table></td></tr>
<tr><td>System Error Count Query</td><td>Returns the number of errors in the error log.

Command Syntax
None

Query Syntax
SYSTem:ERRor:COUNt? → <error_count>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><error_count></td><td>U16</td><td>0 to 32</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>System Error Query</td><td>Returns and clears the first entry in the error log. Multiple errors are stored sequentially in the error log with the oldest error first. A zero value is returned if there are no errors in the log.

Note: This command is not recommended for new applications. Use System Error All Query instead.

Command Syntax
None

Query Syntax
SYSTem:ERRor[:NEXT]? → <error>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><error></td><td>S16</td><td>0 to –32768
0 No error

Note: See Appendix 3, Error Table, for a description of errors.</td></tr>
</table>
</td></tr>
<tr><td>System Memory Query</td><td>Returns the total available waveform memory for use by the four digitizer channels.

Command Syntax
None

Query Syntax
SYSTem:MEMory? → <size>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><size></td><td>U32</td><td>Memory Options
4 MSamples Total
64 MSamples Total

Note: Total samples must be equal to the installed memory option.</td></tr>
</table>
</td></tr>
</table>

Name	Description
System Version Query	Returns the SCPI version to which the instrument complies. **Command Syntax** None **Query Syntax** SYSTem:VERSion? → <version> **Parameters:**

Name	Type	Range
<version>	Float	Variable

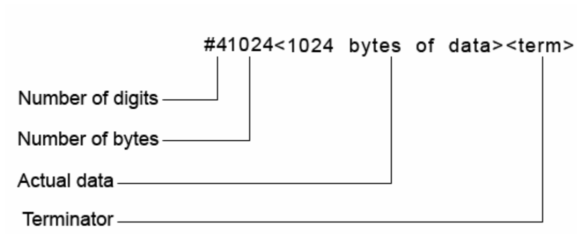
Name	Description

Trace Data Command

Trace Data Query

Reads or writes waveform data to the instrument using the word-serial interface. This command is used to download waveform data from Input Channels 1–4 (INPut<n>) or Calculate Channels 1-2 (CALCulate<n>), and to load or download waveform data to/from Reference Channels 1-4 (REFerence<n>).

Data Interchange Format data (DIF_data) is definite-length block response data. DIF_data allows any type of device-dependent data to be transmitted over the system interface as a series of 8-bit binary data types. It is especially useful for sending large quantifies of data or 8-bit extended ASCII codes. DIF_data uses a specific syntax, a pound sign (#) followed by a non-zero digit representing the number of digits in the decimal integer. After the non-zero digit is the decimal integer that states the number of 8-bit data bytes sent. This is followed by the actual data. The following is an example for writing 1024 bytes of data:



Command Syntax
TRACe[:DATA] <destination>, DIF_data

Query Syntax
TRACe[:DATA]? <source> → DIF_data

Parameters:

Name	Type	Range
<destination>	Discrete	REFerence<n> Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.
<source>	Discrete	INPut<n> Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1. REFerence<n> Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1. CALCulate<n> Calculate channels, where <n> may be 1 or 2. Example: CALC1
DIF_data	S16	Binary Data

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trace Copy Reference Command</td><td>Stores the reference waveform in nonvolatile memory.

Command Syntax
TRACe:COPY:REFerence<n> <source>

Query Syntax
None

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Reference Channel 1
2 Reference Channel 2
3 Reference Channel 3
4 Reference Channel 4</td></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n>
Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1.

REFerence<n>
Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1.

CALCulate<n>
Calculate channels, where <n> may be 1 or 2. Example: CALC1</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trace Load Calculate Query</td><td>Queries calculate channel waveform data using the VXI A32 space. The data starting at the offset location in the waveform (wave_offset) with size (length) is loaded into the A32 memory address (mem_addr). The instrument returns a “1” in response to the query when it has moved all of the data into the A32 memory.

Command Syntax
None

Query Syntax
TRACe:LOAD:CALCulate<n>? <wave_offset>, <mem_addr>, <length> → “1”

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Calculation Channel 1
2 Calculation Channel 2</td></tr>
<tr><td><wave_offset></td><td>U32</td><td>0 to 32768 (maximum calculate waveform size)</td></tr>
<tr><td><mem_addr></td><td>U32</td><td>0 to 67108864 (maximum A32 memory size)</td></tr>
<tr><td><length></td><td>U32</td><td>0 to 32768 (maximum calculate waveform size)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trace Load Input Query</td><td>
Queries input channel waveform data using the VXI A32 space. The data starting at the offset location in the waveform (wave_offset) with size (length) is loaded into the A32 memory address (mem_addr). The instrument returns a “1” in response to the query when it has moved all of the data into the A32 memory.

Command Syntax
None

Query Syntax
TRACe:LOAD:INPut<n>? <wave_offset>, <mem_addr>, <length> → “1”

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Input Channel 1
2 Input Channel 2
3 Input Channel 3
4 Input Channel 4</td></tr>
<tr><td><wave_offset></td><td>U32</td><td>0 to 33554432 (maximum input waveform size)</td></tr>
<tr><td><mem_addr></td><td>U32</td><td>0 to 67108864 (maximum A32 memory size)</td></tr>
<tr><td><length></td><td>U32</td><td>0 to 33554432 (maximum input waveform size)</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trace Load Reference Command

Trace Load Reference Query</td><td>Read or write the reference channel waveform data using the VXI A32 space. When querying (downloading) data from the instrument, the data starting at the offset location in the waveform (wave_offset) with size (length) is loaded into the A32 memory address (mem_addr). The instrument returns a “1” in response to the query when it has moved all of the data into the A32 memory. When writing (loading) data to the instrument, the data <u>must</u> be loaded into the A32 space before issuing this command. The data starting at the offset location in the waveform (wave_offset) with size (length) must be loaded into the A32 memory address (mem_addr).

Note: Use Trace Preamble Command/Query to read/write the appropriate time and voltage conversion factors.

Command Syntax
TRACe:LOAD:REFerence<n> <wave_offset>, <mem_addr>, <length>

Query Syntax
TRACe:LOAD:REFerence<n>? <wave_offset>, <mem_addr>, <length> → “1”

Parameters:

<table>
<tr><th>Name</th><th>Type</th><th>Range</th></tr>
<tr><td><n></td><td>U16</td><td>1 Reference Channel 1
2 Reference Channel 2
3 Reference Channel 3
4 Reference Channel 4</td></tr>
<tr><td><wave_offset></td><td>U32</td><td>0 to 32768 (maximum reference waveform size)</td></tr>
<tr><td><mem_addr></td><td>U32</td><td>0 to 67108864 (maximum A32 memory size)</td></tr>
<tr><td><length></td><td>U32</td><td>0 to 32768 (maximum reference waveform size)</td></tr>
</table></td></tr>
<tr><td>Trace Preamble Command

Trace Preamble Query</td><td>Reads or writes the waveform trace preamble to the unit. The Trace Preamble is an ordered syntax of values. The values are shown below.

• Source: Selects the source to read the waveform preamble from.

• Type: Returns the type of acquisition used.

• Points: Returns the number of points in a waveform.

• Count: Returns the acquired waveform count used to create the selected average or envelope waveform. In Normal acquisition the Acquisition Count is always 1.

• Time Interval: Returns the time interval between points.</td></tr>
</table>

Name	Description

- **Time Offset:** Returns the time in seconds of the first data point relative to the trigger.
- **Voltage Interval:** Returns the voltage resolution.
- **Voltage Offset:** Returns the zero-voltage reference or DC offset voltage for the specified waveform.

Command Syntax
TRACe:PREamble <source> <type>, <points>, <count>, <time_interval>, <time_offset>, <voltage_interval>, <voltage_offset>

Query Syntax
TRACe:PREamble? <source> <type>, <points>, <count>, <time_interval>, <time_offset>, <voltage_interval>, <voltage_offset> → "1"

Parameters:

Name	Type	Range
<source>	Discrete	INPut<n> Input channels, where <n> may be 1, 2, 3, or 4. Example: INP1. REFerence<n> Reference channels, where <n> may be 1, 2, 3, or 4. Example: REF1. CALCulate<n> Calculate channels, where <n> may be 1 or 2. Example: CALC1
<type>	U16	0 = Invalid waveform 1 = Normal voltage-time waveform 2 = Averaged voltage-time waveform 3 = Envelope waveform 4 = Equivalent Time waveform 16 = Frequency Domain waveform
<points>	U32	0 to maximum memory
<count>	U16	1 to 2048
<time_interval>	Float	Time in seconds
<time_offset>	Float	Time in seconds
<voltage_interval>	Float	Voltage in Volts
<voltage_offset>	Float	Voltage in Volts

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr>
<td>Trigger A Event Count Command

Trigger A Event Count Query</td>
<td>Sets or queries the number of events that Trigger A must count before it enables other waveform capture functions. The following considerations apply when using the Trigger A Event Count Command:

The event count affects the capture cycle as follows:

Arm
Count trigger events
When the event count is reached, Trigger A will:

Enable Trigger B detection
Enable the sweep timer

End the capture cycle and begin post-capture processing

The event counter uses the output of the trigger qualifier as an event source (Refer to the Trigger A Pattern Command, Trigger A Pulse High Limit Command, Trigger A Pulse Low Limit Command, Trigger A Slope Command, Trigger A Source Command, and the Trigger A Type Command for information on configuring trigger qualifier parameters).

Command Syntax
TRIGger[:A]:ECOunt <count>

Query Syntax
TRIGger[:A]:ECOunt? → <count>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><count></td><td>U16</td><td>1 to 65535
MINimum 1
MAXimum 65535</td></tr>
</table>
</td>
</tr>
</table>

Name	Description
Trigger A Pulse High Limit Command **Trigger A Pulse High Limit Query**	Sets or queries the upper pulse width limit. The instrument triggers when the pulse width is greater than or less than the upper limit value. For example, to trigger when the pulse width is greater than 50 ns, set the upper limit to 50 ns. The instrument triggers when the pulse width is greater than 50 ns. **Note:** Pulse high limit and pulse low limit use the same resolution for *pulse width in* and *pulse width out* when both limits are used. **Command Syntax** TRIGger[:A]:PULSe:HLIMit <seconds> **Query Syntax** TRIGger[:A]:PULSe:HLIMit? → <seconds> **Parameters:**

Name	Type	Range
<seconds>	Float	20 ns to 655 seconds Resolution: see table below

Resolution:

Resolution	Time in Seconds
10 ns	20 ns to 655.36 μs
100 ns	655.36 μs to 6.5536 ms
1 μs	6.5536 ms to 65.536 ms
10 μs	65.536 ms to 655.36 ms
100 μs	655.36 ms to 6.5536s
1 ms	6.5536s to 65.536s
10 ms	65.536s to 655 s

Name	Description

Trigger A Pulse Low Limit Command

Trigger A Pulse Low Limit Query

Sets or queries the lower pulse width limit. The instrument triggers when the pulse width is less than the lower limit value.

Note: Pulse high limit and pulse low limit use the same resolution for *pulse width in* and *pulse width out* when both limits are used.

Command Syntax
TRIGger[:A]:PULSe:LLIMit <seconds>

Query Syntax
TRIGger[:A]:PULSe:LLIMit? → <seconds>

Parameters:

Name	Type	Range
<seconds>	Float	10 ns to 655 seconds Resolution: see table below

Resolution:

Resolution	Time in Seconds
10 ns	10 ns to 655.36 µs
100 ns	655.36 µs to 6.5536 ms
1 µs	6.5536 ms to 65.536 ms
10 µs	65.536 ms to 655.36 ms
100 µs	655.36 ms to 6.5536s
1 ms	6.5536s to 65.536s
10 ms	65.536s to 655 s

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger A Slope Command

Trigger A Slope Query</td><td>Sets or queries the active edge of the selected trigger.

Command Syntax
TRIGger[:A]:SLOPe <slope>

Query Syntax
TRIGger[:A]:SLOPe? → <slope>

Parameters:

<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><slope></td><td>Discrete</td><td>POSitive Rising Edge
NEGative Falling Edge</td></tr></table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger A Source Command

Trigger A Source Query</td><td>Sets or queries the Trigger A signal source. The following considerations apply:

• All sources are assumed to be low-to-high transitioning signals when active. If a source becomes active on a high-to-low transition, use the Trigger A Slope Command to select negative slope.

• Trigger A detection must complete before Trigger B detector or sweep offset timer are enabled.

Command Syntax
TRIGger[:A]:SOURce <source>

Query Syntax
TRIGger[:A]:SOURce? → <source>

Parameters:

<table>
<tr><th>Name</th><th>Type</th><th colspan="2">Range</th></tr>
<tr><td><source></td><td>Discrete</td><td>INPut<n></td><td>Input channels, where <n> may be 1, 2, 3, or 4.</td></tr>
<tr><td></td><td></td><td>EXTernal</td><td>External trigger source</td></tr>
<tr><td></td><td></td><td>TTLTrg<n></td><td>VXIbus TTL trigger line, where <n> may be 0, 1, 2, 3, 4, 5, 6, or 7</td></tr>
<tr><td></td><td></td><td>ECLTrg<n></td><td>VXIbus ECL trigger line, where <n> may be 0 or 1</td></tr>
<tr><td></td><td></td><td>ARM</td><td>Front panel ARM IN signal</td></tr>
<tr><td></td><td></td><td>MANual</td><td>Manual trigger</td></tr>
<tr><td></td><td></td><td>PATTern</td><td>Pattern trigger</td></tr>
</table>
</td></tr>
</table>

Name	Description
Trigger A Type Command **Trigger A Type Query**	Sets or queries the Trigger A detection type. The following considerations apply when using the Trigger A type: • Six trigger types are accepted: edge, pulse inside limits, pulse outside limits, pulse less than limit, pulse greater than limit, and video. • Edge triggering looks for a rising or falling edge from the selected signal source. • Pulse inside limits looks for a pulse width greater than a low-time limit and less than or equal to a high-time limit. • Pulse outside limits looks for a pulse width less than a low time limit or greater than a high time limit. • Pulse less than looks for a pulse width less than a low time limit. • Pulse greater than looks for a pulse width greater than a high time limit. • Video triggering looks for a specific field or line from the selected source. • Trigger A detection must be complete before the Trigger B detector or sweep offset timer are enabled **Command Syntax** TRIGger[:A]:TYPE <type> **Query Syntax** TRIGger[:A]:TYPE? → <type> **Parameters:**

Name	Type	Range	
<type>	Discrete	EDGE	Edge trigger
		PIN	Pulse inside limits
		POUT	Pulse outside limits
		PLTHan	Pulse less than
		PGTHan	Pulse greater than
		VIDeo	Video trigger

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger A Video Field Command

Trigger A Video Field Query</td><td>Sets or queries which video field to select a line to trigger on.

Command Syntax
TRIGger[:A]:VIDeo:FIELd <field>

Query Syntax
TRIGger[:A]:VIDeo:FIELd? → <field>

Parameters:
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><field></td><td>U16</td><td>1 or 2</td></tr></table></td></tr>
<tr><td>Trigger A Video Line Command

Trigger A Video Line Query</td><td>Sets or queries which video line to trigger on.

Note: Range depends on video standard and field.

Command Syntax
TRIGger[:A]:VIDeo:LINE <line>

Query Syntax
TRIGger[:A]:VIDeo:LINE? → <line>

Parameters:
<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><line></td><td>U16</td><td><u>NTSC</u>
Field 1: 1 to 263
MINimum 1
MAXimum 263

Field 2: 1 to 262
MINimum 1
MAXimum 262

<u>PAL/SECAM</u>
Field 1: 1 to 313
MINimum 1
MAXimum 313

Field 2: 314 to 625
MINimum 314
MAXimum 625</td></tr></table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger A Video Standard Command

Trigger A Video Standard Query</td><td>Sets or queries the video standard.

Command Syntax
TRIGger[:A]:VIDeo:STANdard <video_standard>

Query Syntax
TRIGger[:A]:VIDeo:STANdard? → <video_standard>

Parameters:

<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><video_standard></td><td>Discrete</td><td>PAL PAL standard
NTSC NTSC standard
SECam SECAM standard</td></tr></table></td></tr>
<tr><td>Trigger B Slope Command

Trigger B Slope Query</td><td>Sets or queries the active edge of the selected trigger.

Command Syntax
TRIGger:B:SLOPe <slope>

Query Syntax
TRIGger:B:SLOPe? → <slope>

Parameters:

<table><tr><td>Name</td><td>Type</td><td>Range</td></tr><tr><td><slope></td><td>Discrete</td><td>POSitive Rising Edge
NEGative Falling Edge</td></tr></table></td></tr>
</table>

Name	Description

Trigger B Source Command

Trigger B Source Query

Sets or queries the Trigger B signal source. The following considerations apply when using setting the Trigger B source:

- Trigger B slope is selectable using edge triggering.
- If a source becomes active on a high-to-low transition, use the Trigger B Slope Command.
- Trigger A detection must be complete before the Trigger B detector or sweep offset timer are enabled.

Command Syntax
TRIGger:B:SOURce <source>

Query Syntax
TRIGger:B:SOURce? → <source>

Parameters:

Name	Type	Range	
<source>	Discrete	INPut<n> may	Input channels, where <n> be 1, 2, 3, or 4.
		EXTernal	External trigger source
		TTLTrg<n>	VXIbus TTL trigger line, where <n> may be 0, 1, 2, 3, 4, 5, 6, or 7
		ARM	Front panel ARM IN signal
		MANual	Manual trigger

Trigger B State Command

Trigger B State Query

Sets or queries the status state of Trigger B.

Command Syntax
TRIGger:B[:STATe] <state>

Query Syntax
TRIGger:B[:STATe]? → <state>

Parameters:

Name	Type	Range	
<state>	Discrete	ON or 1 OFF or 0	Enable Disable

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger Hold Off Command

Trigger Hold Off Query</td><td>Sets or queries the duration (in seconds) to hold off or ignore all other triggers before recognizing the next trigger event.

Command Syntax
TRIGger:HOLDoff <seconds>

Query Syntax
TRIGger:HOLDoff? → <seconds>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><seconds></td><td>Float</td><td>0 to 655 seconds
MINimum 0
MAXimum 655

Resolution: see table below</td></tr>
</table>

Resolution:

<table>
<tr><td>Resolution</td><td>Time in Seconds</td></tr>
<tr><td>10 ns</td><td>0 to 655.36 μs</td></tr>
<tr><td>100 ns</td><td>655.36 μs to 6.5536 ms</td></tr>
<tr><td>1 μs</td><td>6.5536 ms to 65.536 ms</td></tr>
<tr><td>10 μs</td><td>65.536 ms to 655.36 ms</td></tr>
<tr><td>100 μs</td><td>655.36 ms to 6.5536s</td></tr>
<tr><td>1 ms</td><td>6.5536s to 65.536s</td></tr>
<tr><td>10 ms</td><td>65.536s to 655 s</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger External Level Command

Trigger External Level Query</td><td>Sets or queries the external trigger input level.

Command Syntax
TRIGger:EXTernal:LEVel <volts>

Query Syntax
TRIGgerEXTernal:LEVel? → <volts>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><volts></td><td>Float</td><td>±1 volts
MINimum –1 volts
MAXimum +1 volts

Resolution: 0.5 mV</td></tr>
</table></td></tr>
<tr><td>Trigger Input Level Command

Trigger Input Level Query</td><td>Sets or queries an analog input channel trigger level. The following considerations apply when setting the input trigger level:

• The trigger level has the same range as the input range.

• Each input channel has an independent trigger level.

• The logic level out of the threshold detector is reported to the Trigger A and B multiplexers for selection as the input trigger sources.

Command Syntax
TRIGger:INPut<n>:LEVel <volts>

Query Syntax
TRIGger:INPut<n>:LEVel? → <volts>

Parameters:
<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><n></td><td>U16</td><td>1 Input Channel 1
2 Input Channel 2
3 Input Channel 3
4 Input Channel 4</td></tr>
<tr><td><volts></td><td>Float</td><td>Input Offset $\pm \frac{\text{Input Range}}{2}$</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger Pattern Mask Command

Trigger Pattern Mask Query</td><td>Sets or queries which sources to use in the pattern.

Command Syntax
TRIGger:PATTern:MASK <pattern_mask >

Query Syntax
TRIGger:PATTern:MASK? → <pattern_mask>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><pattern_mask></td><td>U16</td><td>0 to 255
0 Do <u>not</u> use in pattern trigger
1 Use in pattern trigger

Source Order (MSB–LSB):

• Bits 15–8 are ignored
• Bit 7—External Arm
• Bit 6—VXI ECLTRG1
• Bit 5—VXI ECLTRG0
• Bit 4—External Trigger
• Bit 3—Channel 4
• Bit 2—Channel 3
• Bit 1—Channel 2
• Bit 0—Channel 1</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger Pattern Truth Command

Trigger Pattern Truth Query</td><td>Sets or queries the state of each source for the pattern trigger to occur.

Command Syntax
TRIGger:PATTern:TRUTh <pattern_truth>

Query Syntax
TRIGger:PATTern:TRUTh? → <pattern_truth>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><pattern_truth></td><td>U16</td><td>0 to 255
0 Low
1 High

Source Order (MSB–LSB):

• Bits 15–8 are ignored
• Bit 7—External Arm
• Bit 6—VXI ECLTRG1
• Bit 5—VXI ECLTRG0
• Bit 4—External Trigger
• Bit 3—Channel 4
• Bit 2—Channel 3
• Bit 1—Channel 2
• Bit 0—Channel 1</td></tr>
</table>
</td></tr>
<tr><td>Trigger Timestamp Query</td><td>Returns the trigger timestamp of the most recent trigger event in fractional seconds with a 1 second period.

Command Syntax
None

Query Syntax
TRIGger:TIMestamp? → <seconds>

Parameters:

<table>
<tr><td>Name</td><td>Type</td><td>Range</td></tr>
<tr><td><seconds></td><td>Float</td><td>0 to 1 second
Resolution: 100 ns</td></tr>
</table>
</td></tr>
</table>

Specifications

Analog Input

Channels — Quantity 4

Bandwidth — DC to 250 MHz typical, 200 MHz minimum (50 Ω)
DC to 125 MHz typical, 100 MHz minimum (1 MΩ)

Slew Rate — 4000 V/µs (50 Ω)
800 V/µs (1 MΩ)

Maximum Input (50 Ω) — ±5 V (DC + peak AC)
Input load protection @ ±6 VDC

Maximum Input (1 MΩ) — ±25 V [DC + peak AC (<10 MHz)]
Peak AC, de-rated 20 dB/decade above 10 MHz

Full Scale Input Range & Offset Adjust

Impedance	Range	Offset
1 MΩ	50 Vpp	0V
	25 Vpp	±12.5V
	10 Vpp	±5V
	5 Vpp	±5V
	2.5 Vpp	±5V
	1.25 Vpp	±5V
	0.5 Vpp	±5V
	0.25 Vpp	±5V
50Ω	10 Vpp	0V
	5 Vpp	±2.5V
	2 Vpp	±1V
	1 Vpp	±1V
	0.5 Vpp	±1V
	0.25 Vpp	±1V
	0.1 Vpp	±1V
	0.05 Vpp	±1V

DC Gain Accuracy — < ±0.25% full scale range (50 Ω)
< ±0.25% full scale range (1 MΩ)

Zero DC Offset	< ± (0.25% full scale range + 1 mV) @ +25 °C (50 Ω) < ±(0.25% full scale range + 5 mV) @ +25 °C (1 MΩ)
Zero DC Offset Drift	< ± 0.05% maximum offset adjust/°C
Offset Adjust Accuracy	< ± 1%
Impedance	1 MΩ || 12 pF or 50 Ω
Impedance Accuracy	± 1%
Input VSWR (50 Ω)	≤ 1.3:1, DC to 50Ω Bandwidth
Input Bias (50 Ω)	≤ ±25 μA (50 Ω) ≤ ±1 nA (1 MΩ)
Coupling	DC or AC
AC Coupling	200 kHz high pass (50Ω) 10 Hz high pass (1 MΩ)
Probe Attenuation	0.9 to 1000:1
RMS Noise	≤ (0.1% of range + 200 μV) (50 Ω) ≤ (0.1% of range + 1.25 mV) (1M Ω)
Connectors	BNC

Analog-to-Digital Converter

Sample Rate

10 kS/s to 200 MS/s in 1, 2.5, 4, and 5 steps
250 MS/s (ZT412-20 only)
400 MS/s, 1 channel interleaved
500 MS/s, 1 channel interleaved (ZT412-2X only)

Resolution &
Maximum Sample Rate

Product Option	ADC Resolution	1 Channel Maximum Sample Rate	2 Channel Maximum Sample Rate
ZT412-2X	14-bit	500 MS/s	250 MS/s
ZT412-5X	16-bit	400 MS/s	200 MS/s

Acquisition Time Range

Product Option	Minimum Acquisition Time	Maximum Acquisition Time
ZT412-2X	200 ns	3,355 seconds
ZT412-5X	250 ns	3,355 seconds

Channel-to-Channel Skew	≤ 100 ps difference with channels at same input settings
Skew Adjust	-300 ps to +300 ps in 10 ps steps (Channels: 1-and-2 to 3-and-4)

Dynamic Range
10.7 MHz (Typical)

50Ω **Input Range (Vpp)**	**Signal-to Noise Ratio (SNR)**	**Total Harmonic Distortion (THD)**	**Signal-to-Noise + Distortion (SINAD)**
1.0 to 10.0	74.5 dBc	70.2 dBc	68.8 dBc
0.5	70.5 dBc	70.2 dBc	67.3 dBc
0.25	64.5 dBc	70.5 dBc	63.5 dBc
0.1	56.5 dBc	70.9 dBc	56.3 dBc
0.05	50.5 dBc	70.9 dBc	50.5 dBc

60.1 MHz (Typical)

50Ω **Input Range (Vpp)**	**Signal-to Noise Ratio (SNR)**	**Total Harmonic Distortion (THD)**	**Signal-to-Noise + Distortion (SINAD)**
1.0 to 10.0	74.5 dBc	50.1 dBc	50.1 dBc
0.5	70.5 dBc	50.1 dBc	50.0 dBc
0.25	64.5 dBc	50.5 dBc	50.3 dBc
0.1	56.5 dBc	50.8 dBc	49.8 dBc
0.05	50.5 dBc	50.8 dBc	47.6 dBc

Note: Dynamic range for interleaved sample rates is degraded by 4 dB for input channels 2 & 4.

Waveform Memory

Total Memory
Up to 8M samples/channel
Up to 16M samples/channel (2 channels interleaved)

Memory Options
2M samples total (ZT412-X0)
32M samples total (ZT412-X1)

Acquisition Modes

Types
Normal, Average, Envelope, and Equivalent-Time

Channels
Normal & Equivalent-Time: Quantity 4 inputs simultaneous
Average & Envelope: Quantity 2 inputs simultaneous

Waveform Size
100 samples to total memory (Normal)
100 samples to 32k samples (Average, Envelope, Equivalent-Time)

Waveform Count
2 to 65535 waveforms

Averaging
16-bit waveform averaging resolution

Envelope
Minimum and Maximum Envelope

Equivalent-Time
High sample rate waveform reconstruction

Equivalent-Time Points
2 to 100 equivalent-time points per real-time point
2 to 100 times equivalent-time sample rate

Trigger

Trigger Source	Channels 1 to 4, External Trigger, ECLTRG0-1, TTLTRG0–7*, External Arm, Pattern, Software
Trigger Slope/Polarity	Positive or Negative
Trigger Position	0% to 100% of waveform time + trigger delay ±1 sample interval position accuracy
Post-Trigger Delay	0 to 655 seconds
Pre-Trigger Delay	0 to waveform time
Trigger Holdoff	Programmable delay after trigger before recognizing next trigger event
Holdoff Range	0 to 655 seconds
Trigger B	Second edge trigger event qualifier
Pattern Trigger	Pattern match true or false
Pattern Sources	Channels 1 to 4, External Trigger, External Arm, ECLTRG0-1
Event Trigger	Event Counter: 1 to 65535 trigger events
Trigger Modes	Edge, Pulse Width, Video
Edge Trigger Mode	Rising or Falling Edge
Pulse Width Trigger Mode	Triggers on pulse width greater than, less than, or between limits
Pulse Width Type	< limit1, > limit1, < limit1 & > limit2
Pulse Width Range	20 ns to 655 seconds
Pulse Width Resolution	10 ns
Video Trigger Mode	PAL (50 Hz), NTSC (60 Hz), SECAM (50 Hz) Standard, Field, Line selectable
Ch 1–4 Trigger Level	(offset – full scale/2) to (offset + full scale/2)
Ch 1–4 Trigger Sensitivity	5% of full scale (DC to 75 MHz)
Ch 1–4 Trigger Bandwidth	≥ 200 MHz
Ch 1–4 Trigger Hysteresis	5% (overdrive required)
Ch 1–4 Level Resolution	0.025% of full scale
Ch 1–4 Level Accuracy	±(2% full scale + 5 mV + offset accuracy)
Trigger Timestamp	100 ns resolution, 1 second rollover

External Trigger Input

Maximum Input	±5V, no damage
Threshold Input	±1V
Threshold Accuracy	±20 mV
Threshold Resolution	0.5 mV
Input Impedance	50 Ω ±2%
Connector	BNC

Trigger Outputs

Functionality	Event Output Signals
Outputs	TTLTRG0–7*, ECLTRG0–1
Source	Trigger Event, Arm Event, OPC, Constant

Reference Output

Functionality	reference voltage, ground, trigger event, arm event, 10 MHz clock, 500 Hz probe compensation, 10 ns pulse @ 1ms rate
Reference Voltage Output	+8 V ± 1% into 10 kΩ load
Logic Output (all other types)	TTL Compatible
Connector	BNC

Arm

Functionality	Arm to qualify trigger event
Source	External Trigger, External Arm, ECLTRG0-1, TTLTRG0–7*, Software
Polarity	Positive or Negative

External Arm Input

Maximum Input	0V to 5V, no damage
Nominal Level	TTL Compatible
Input Impedance	1 kΩ ±2%, pull-up to +5V
Connector	BNC

External Sampling Clock Input

Function	External Sampling Clock bypasses Phase Locked Loop, All ADC channels synchronized to external clock
Clock Rates	40 MHz to maximum external clock frequency 1 channel enabled: sample at external frequency 2 channels enabled: sample at half external frequency

Product Option	Maximum Ext Clock Frequency	1 Channel Maximum Sample Rate	2 Channel Maximum Sample Rate
ZT412-2X	500 MHz	500 MS/s	250 MS/s
ZT412-5X	400 MHz	400 MS/s	200 MS/s

Maximum Input	±5V, no damage
Input Signal Level	500 mVpp to 1 Vpp, sine or square wave
Input Impedance	AC coupled, 50Ω ±2%
Connector	BNC

10 MHz Time Base Reference

Clock Source	Internal TCXO, VXI Backplane CLK10
Internal TCXO	± 2.5 ppm accuracy

Data Processing

Auto Scale	Automatic adjust to input signals: Input Range, Offset, Sample Rate, Trigger Source, and Trigger Level
Self-Calibration	Automatic internal calibration: Input DC Offset Zero, Input DC Offset Adjust Gain, ADC leveling

Measurements

Measurements	Min, Max, Low, High, Mid, Average, Amplitude, Peak-to-Peak, DC RMS, AC RMS, +Width, –Width, Period, Frequency, +Duty, –Duty, Phase, Rise Time, Rise Overshoot, Rise Preshoot, Rise Crossing Time, Fall Time, Fall Overshoot, Fall Preshoot, Fall Crossing Time, Time of Maximum, Time of Minimum, Cycle Average, Cycle RMS, Cycle Frequency, Cycle Period, AC High-Precision, DC High-Precision, SNR, THD, SINAD, ENOB, SFDR
Measurement Methods	Entire Waveform, Gated by Time, Gated by Points

Measurement Levels	Low, Mid, High reference levels for edge measurements set in absolute voltages or relative percentages

Measurement Accuracy

Delta DC Voltage	± (DC gain accuracy)
Absolute DC Voltage	± [(DC gain accuracy)+(offset accuracy)]
Time	± (time resolution)
Frequency	± [1/(time resolution)]

Note: time resolution = one sample interval or one ETS sample interval (for ETS acquisition)

Reference Waveforms

Reference Channels	Quantity 4
Reference Storage	Non-volatile memory storage
Reference Size	32k maximum waveform size

Calculations

Calculate Channels	Quantity 2
Calculate Size	32k maximum waveform size
Calculate Functions	Add, Subtract, Multiply, Copy, Invert, Integral, Derivative, Absolute Value, Limit Test, Mask Test, Frequency Transform, Time Transform
Limit Test	Measurement Limit Range Testing or Waveform Mask Testing
Limit Test Reports	Measurement maximum, minimum, average, current value, pass/fail counts
Frequency Transform	FFT Magnitude
FFT Windowing	Rectangular, Hamming, Hanning, Blackman
Time Transform	Infinite Impulse Response (IIR) filtering
IIR Filter Count	2 to 40 data points

Instrument Setup Storage

Reset	Non-volatile storage of default instrument setup configuration
Save & Recall	Non-volatile storage of 31 instrument setup configurations

Data Interface

VXIbus Connection	Standard P1 and P2 interface
Command Interface	A16 SCPI message-based
Interrupt Operation	Programmable interrupter, Level 1–7
Data Interface	64MB A32 register-based
Manufacturer ID	3712 ($E80_{16}$)
Model Code	412 ($19C_{16}$)

VXIbus P2 Trigger & Clock Pin Usage

Pin A1	ECLTRG0	(ECL level bidirectional)
Pin A3	ECLTRG1	(ECL level bidirectional)
Pin A23	TTLTRG0*	(TTL level bidirectional)
Pin A24	TTLTRG2*	(TTL level bidirectional)
Pin A26	TTLTRG4*	(TTL level bidirectional)
Pin A27	TTLTRG6*	(TTL level bidirectional)
Pin C1	CLK10+	(ECL level input)
Pin C2	CLK10–	(ECL level input)
Pin C23	TTLTRG1*	(TTL level bidirectional)
Pin C24	TTLTRG3*	(TTL level bidirectional)
Pin C26	TTLTRG5*	(TTL level bidirectional)
Pin C27	TTLTRG7*	(TTL level bidirectional)

LED Indicators

READY	Unit has passed power-up self-diagnostics. Toggles when unit has an error pending in error queue.
VXI	VXI access occurring or VXIbus MODID asserted
TRG	Flashes when trigger event occurs
BUSY	Unit is busy with one of the following operations: auto-scale, self-calibration, self-test, data capture, download or storage

DC Power

Total Cooling &
Power Consumption

Product Option	Typical Cooling & Power	Maximum Cooling & Power
ZT412-20	55.4 W	68.5 W
ZT412-21	69.8 W	82.9 W
ZT412-50	57.0 W	70.1 W
ZT412-51	71.4 W	84.5 W

Power Supplies

Product Option	Voltage	Typical Current	Maximum Current
ZT412-20	+5V +12V +24V –2V –5.2V –12V –24V	9.44A 0.45A 0.0A 0.05A 0.51A 0.0A 0.0A	11.46A 0.52A 0.0A 0.08A 0.92A 0.0A 0.0A
ZT412-21	+5V +12V +24V –2V –5.2V –12V –24V	12.32A 0.45A 0.0A 0.05A 0.51A 0.0A 0.0A	14.34A 0.52A 0.0A 0.08A 0.92A 0.0A 0.0A
ZT412-50	+5V +12V +24V –2V –5.2V –12V –24V	9.76A 0.45A 0.0A 0.05A 0.51A 0.0A 0.0A	11.78A 0.52A 0.0A 0.08A 0.92A 0.0A 0.0A
ZT412-51	+5V +12V +24V –2V –5.2V –12V –24V	12.64A 0.45A 0.0A 0.05A 0.51A 0.0A 0.0A	14.66A 0.52A 0.0A 0.08A 0.92A 0.0A 0.0A

Physical

Physical size Single-Wide C-size VXIbus

Weight 3.5 lbs. or 1.59 kg

Temperature Range

Operating	0 °C to +40 °C Ambient
Storage	-40 °C to +75 °C
Calibration Range	+20 °C to +30 °C Ambient, after a 20 minute warm-up period, to meet all calibration specification accuracies.

Relative Humidity

Operating or Storage	10 to 90%, non-condensing, up to +40 °C

Altitude

Operating	Up to 2,000 m
Storage	Up to 15,000 m

Default Reset Conditions

Parameter	Default
Acquisition Mode	Normal
Arm Polarity	Positive
Arm Source	Immediate
Averaging Waveform Count	8
Averaging Type	Average
Calculation Channel Enable	False
Calculation Operation	Add Input1 and Input 2
ECL Trigger Outputs	Disabled, Positive Polarity, Trigger Event Source
Envelope View	Minimum
Equivalent-Time Points	10
Initiate Continuous	OFF
Input Channel Enable	True
Input Coupling	DC
Input Impedance	1 MΩ
Input Probe Attenuation	1.0
Input Voltage Offset	0.0V
Input Voltage Range	10.0Vpp
Limit Test Continuous	On
Limit Test Lower Limit	0.0
Limit Test Measurement	AC RMS

Parameter	Default
Limit Test Upper Limit	0.0
Mask Test Lower Source	Reference 1
Mask Test Upper Source	Reference 1
Measurement Edge	1
Measurement Gate Start Points	0
Measurement Gate Start Time	0.0 seconds
Measurement Gate Stop Points	1000
Measurement Gate Stop Time	10 μs
Measurement High Reference Level	0.9 (90 percent)
Measurement Low Reference Level	0.1 (10 percent)
Measurement Method	All
Measurement Mid Reference Level	0.5 (50 percent)
Measurement Reference Method	Relative Percentages
Pulse Width Trigger High	2 μs
Pulse Width Trigger Low	1 μs
Reference Output	Disabled, Compensation Source (500 Hz)
Sample Clock Source	Internal
Sample Rate	100 MHz
Sweep Mode (Auto trigger)	Normal
Sweep Offset Reference	0.5 (50 percent)
Sweep Offset Time	0.0 seconds
Sweep Points	1000
Time Base Reference	Internal
Transform Frequency Window	Rectangular
Transform Time Points	40
Trigger B Polarity	Positive

Parameter	Default
Trigger B Source	Channel 2
Trigger B State	Bypass
Trigger Event Count	1
Trigger Hold Off	0.0 seconds
Trigger Level Channel 1	0.0V
Trigger Level Channel 2	0.0V
Trigger Level Channel 3	0.0V
Trigger Level Channel 4	0.0V
Trigger Level External	0.0V
Trigger Pattern Mask	0
Trigger Pattern Truth	0
Trigger Polarity	Positive
Trigger Source	Channel 1
Trigger Type	Edge
TTL Trigger Outputs	Disabled, Positive Polarity, Trigger Event Source
Video Trigger Field	1
Video Trigger Line	1
Video Trigger Standard	NTSC
Waveform Byte Order	Normal (MSB First)

Error Codes

The ZT412 maintains an error queue containing codes for faults conditions encountered during unit operation. These codes are listed in the table below along with a brief description of the code meaning. The error log may be read by using the SYSTEM ERROR QUERY.

Code	Error Summary	Description
-100	Command error	A generic syntax error (only used when a more specific error does not apply)
-101	Invalid character	A syntactic element contains a character which is invalid for that type
-102	Syntax error	An unrecognized command or data type was encountered
-103	Invalid separator	The parser was expecting a separator and encountered an illegal character
-104	Data type error	The parser recognized a data element different than the one allowed
-105	Get not allowed	
-108	Parameter not allowed	More parameters were received than expected
-109	Missing parameter	Fewer parameters were received than expected
-110	Command header error	A generic error was detected in the command
-111	Header separator error	A character which was not a legal separator was encountered while parsing the command
-112	Mnemonic too long	The command contains too many characters
-113	Undefined header	The command is correct, but undefined for the specific instrument
-114	Header suffix out-of-range	The suffix number makes the command invalid
-118	Query not allowed	
-120	Numeric data error	A generic numeric syntax error (only used when a more specific error does not apply)

Code	Error Summary	Description
-121	Invalid char in number	An invalid character for the data type was encountered
-123	Exponent too large	The magnitude of the exponent was larger than 32000
-124	Too many digits	The mantissa of a decimal numeric data element contained more than 255 digits excluding zero
-128	Numeric data not allowed	A legal numeric data element was received, but the instrument does not accept one in this position in the command
-130	Suffix error	General command suffix error (only used when a more specific error does not apply)
-131	Invalid suffix	The command suffix is invalid for this instrument
-134	Suffix too long	The command suffix is too long
-138	Suffix not allowed	A suffix was encountered after a numeric element which does not allow suffixes
-140	Character data error	General character data element error (only used when a more specific error does not apply)
-141	Invalid character data	Either a invalid character in the parameter or the character data is not valid for this command
-144	Character data too long	The character parameter contains too many characters
-148	Character data not allowed	The character data is legal but not supported by this instrument
-150	String data error	General data string error (only used when a more specific error does not apply)
-151	Invalid string data	An invalid string command parameter
-158	String data not allowed	A string element was in the wrong place for this instrument command
-160	Block data error	General block data error (only used when a more specific error does not apply)
-161	Invalid block data	An invalid block data element was received for this instrument command

Code	Error Summary	Description
-168	Block data not allowed	Block data element not allowed by this instrument command at this parameter
-170	Expression error	General expression error (only used when a more specific error does not apply)
-171	Invalid expression	Invalid expression data element like unmatched parentheses or illegal character
-178	Expression data not allowed	A legal expression was encountered but is not allowed by this instrument in this command
-180	Macro error	General macro error (only used when a more specific error does not apply)
-181	Invalid outside macro	Indicates that a macro parameter placeholder was encountered outside a macro definition
-183	Invalid inside macro	Syntactically invalid message unit sequence
-184	Macro parameter error	The command inside the macro definition had the wrong number or type of parameters
-200	Execution error	General execution error (only used when a more specific error does not apply)
-201	Invalid while in local	Indicates that the command is not executable while the device is in local control
-202	Settings lost due to RTL	The settings were lost when the instrument was returned to local control
-203	Command protected	Indicates that a legal password protected program command or query could not be executed because the command was disabled
-210	Trigger Error	General trigger error
-211	Not ready for trigger	Indicates that a trigger was received by the instrument but was ignored because of timing considerations
-212	Not ready for arm	Indicates that a ARM was received by the instrument but was ignored
-213	Already initiated	Indicates that a measurement request was ignored because another measurement was already in progress
-214	Not ready for trigger	The trigger is deadlocked because a measurement result was requested before the instrument measurement was triggered

Code	Error Summary	Description
-220	Parameter error	General program parameter error (only used when a more specific error does not apply)
-221	Settings conflict	Indicates that a legal command was received by the instrument but could not be executed because of the current state of the instrument
-222	Data out of range	Indicates that a valid parameter was received but could not be executed because the parameter is out of range for the instrument
-223	Too much data	The command contained more data than the instrument memory could support
-224	Illegal parameter value	A value outside the list of possible values was received
-225	Out of memory	The instrument contains insufficient memory to perform the requested operation
-226	Lists not the same length	The lists do not have equal lengths
-230	Data corrupt or stale	New reading started but not completed resulting in invalid data
-231	Questionable data	Indicates that there is a problem with the instrument measurement accuracy
-232	Data has invalid format	The command tried to execute using an inappropriate data format or structure
-233	Incompatible version	Indicates that a file version or instrument version is not appropriate for this command
-240	Hardware error	Indicates that a general error occurred because there was a hardware problem in the instrument (only used when a more specific error does not apply)
-241	Hardware missing	Indicates that a command could not be executed because a hardware option is not present
-250	Mass storage error	General mass storage error (only used when a more specific error does not apply)
-251	Missing mass storage	The command could not be executed because an optional mass storage device was not present
-252	Missing media	The command could not be executed because of a missing media (disk) from a storage device

Code	Error Summary	Description
-253	Corrupt media	Indicates that the requested media is corrupt (bad or unformatted disk)
-254	Media full	Indicates that the requested media is full
-255	Directory full	Indicates that the requested media directory is full
-256	File name not found	Indicates that the command or query could not be executed because the requested file could not be found
-257	File name error	Indicates that the command or query could not be executed because the requested file was in error
-258	Media protected	Indicates that the requested media is protected
-260	Expression execution failed	General command expression error (only used when a more specific error does not apply)
-261	Math expression execution failed	Indicates that a command tried to perform an illegal math operation
-270	Macro execution error	General macro error (only used when a more specific error does not apply)
-271	Macro syntax error	The command could not be executed because there is an error within the syntax of the macro
-272	Macro execution error	The command could not be executed because there is an error within the macro definition
-273	Illegal macro label	The macro label is not valid for this instrument
-274	Macro parameter error	The macro definition improperly uses a macro parameter placeholder
-275	Macro definition too long	The string or block content of a macro was too long for the instrument
-276	Macro recursion error	The macro program data sequence could not be executed because the instrument found it to be recursive
-277	Macro redefinition not allowed	The command could not be executed because the macro label was already defined
-278	Macro header not found	Could not execute the macro because the macro was not previously defined

Code	Error Summary	Description
-280	Program error	General downloaded program error (only used when a more specific error does not apply)
-281	Can not create program	Indicates that the attempt to create a downloaded program was unsuccessful generally due to lack of memory
-282	Illegal program name	The command referenced a nonexistent program or attempted to redefine an existing program
-283	Illegal variable name	An attempt was made to reference a nonexistent program variable
-284	Program currently running	An attempt was made to redefine or delete an existing program while it is running
-285	Program syntax error	Indicates that a syntax error appears in a downloaded program
-286	Program runtime error	A runtime error exists in a downloaded program
-290	Memory usage error	Indicates that the user request has directly or indirectly caused an error related to memory
-291	Out of memory	The instrument memory is full
-292	Reference name does not exist	The reference name does not exist
-293	Reference name already exists	The reference name already exists
-294	Incompatible Type	Indicates that the type or structure of a memory item is inadequate
-300	Device specific error	General instrument error (only used when a more specific error does not apply)
-310	System error	Indicates that an instrument system error has occurred
-311	Memory error	Indicates a physical fault in the instruments memory, such as a parity fault
-312	PUD memory lost	Indicates that the protected user data in the instrument has been lost
-313	Calibration memory corrupted	Indicates that the instruments nonvolatile calibration memory has been lost or corrupted
-314	Configuration memory corrupted	Indicates that the instruments nonvolatile memory that was saved has been lost or corrupted

Code	Error Summary	Description
-315	Manufacturing info corrupted	Indicates that the instruments nonvolatile configuration memory has been lost or corrupted
-320	Storage Fault	Indicates that the firmware detected a fault when using data storage. Generally this error does not indicate a hardware error
-321	Out of memory for an internal operation	An internal operation needed more memory than was available
-330	Self test failed	The internal self test failed. This self test is either run on power up or by command
-340	Calibration failed	The instrument internal calibration failed
-350	Queue overflow	This code indicates that there is no room in the queue and an error occurred but was not recorded
-360	Communications error	General instrument communications error (only used when a more specific error does not apply)
-361	Parity error in program message	The serial port parity bit was not correct when data was received
-362	Framing error in program message	A serial port stop bit was not detected when data was received
-363	Input buffer overrun	The input buffer on a serial port overflowed with data caused by improper or nonexistent spacing
-400	Query error	General query error (only used when a more specific error does not apply)
-410	Query interrupt error	Indicates that a command was received before the query was fully executed
-420	Query un-terminated error	An incomplete query command was received
-430	Query deadlock error	The instrument is locked due to a incomplete query command
-440	Query un-terminated after indefinite response	Indicates that a query was received in the same command after a query requesting an indefinite response was executed
-500	Power on	The instrument has detected an off to on transition in its power system

Code	Error Summary	Description
-600	User request	The instrument has detected the activation of a user request for local control
-700	Request control	The instrument requested to become the active controller-in-charge
-800	Operation complete	The instrument has completed all selected pending operations
-1001	PLL unlocked	The instrument clock in not locked to the PLL
-1002	Boot Failed	Firmware boot failure detected
-1003	Wave Invalid	Unable to create a desired waveform due to invalid parameter set

Commands Index

The following are alphabetic lists of the commands for the ZT412.

IEEE-488 Common Commands

Name	Command Syntax
Calibrate Query	*CAL?
Clear Status Command	*CLS
Event Status Enable Command/Query	*ESE
Event Status Register Query	*ESR?
Identification Query	*IDN?
Operation Complete Command/Query	*OPC
Recall Instrument State Command	*RCL
Reset Command	*RST
Save Instrument State Command	*SAV
Service Request Enable Command/Query	*SRE
Status Byte Query	*STB?
Trigger Immediate Command	*TRG
Test Query	*TST?
Wait to Continue Command	*WAI

SCPI Instrument Specific Commands

Name	Command Syntax
Abort Command	ABOR
Arm Command/Query	ARM[:IMMediate]
Arm Polarity Command/Query	ARM:POLarity
Arm Source Command/Query	ARM:SOURce
Auto Scale Command	[SENSe]:AUToscale

Name	Command Syntax
Average Count Command/Query	[SENSe]:AVERage:COUNt
Average Envelope View Command/Query	[SENSe]:AVERage:ENVelope:VIEW
Average Equivalent Time Points Command/Query	[SENSe]:AVERage:ETIMe:POINts
Average State Command/Query	[SENSe]:AVERage[:STATe]
Average Type Command/Query	[SENSe]:AVERage:TYPE
Calculate Absolute Value Command	CALCulate<n>:AVALue
Calculate Add Command	CALCulate<n>:ADD
Calculate Copy Command	CALCulate<n>:COPY
Calculate Derivative Command	CALCulate<n>:DERivative
Calculate Function Query	CALCulate<n>:FUNCtion
Calculate Immediate Command	CALCulate<n>:IMMediate
Calculate Integral Command	CALCulate<n>:INTegral
Calculate Invert Command	CALCulate<n>:INVert
Calculate Limit Test Clear Command	CALCulate<n>:LIMit:CLEar
Calculate Limit Test Command	CALCulate<n>:LIMit
Calculate Limit Test Continuous Command/Query	CALCulate<n>:LIMit:CONTinuous
Calculate Limit Test Fail Query	CALCulate<n>:LIMit:FAIL
Calculate Limit Test Lower Command/Query	CALCulate<n>:LIMit:LOWer
Calculate Limit Test Measurement Command/Query	CALCulate<n>:LIMit:MEASure
Calculate Limit Test Report Query	CALCulate<n>:LIMit:REPort
Calculate Limit Test Upper Command/Query	CALCulate<n>:LIMit:UPPer
Calculate Mask Test Lower Command/Query	CALCulate<n>:LIMit:MASK:LOWer
Calculate Mask Test Upper Command/Query	CALCulate<n>:LIMit:MASK:UPPer
Calculate Multiply Command	CALCulate<n>:MULTiply
Calculate State Command/Query	CALCulate<n>[:STATe]
Calculate Subtract Command	CALCulate<n>:SUBTract
Calculate Transform Frequency Command	CALCulate<n>:TRANsform:FREQuency
Calculate Transform Frequency Window Command/Query	CALCulate<n>:TRANsform:FREQuency:WINDow
Calculate Transform Time Command	CALCulate<n>:TRANsform:TIME

Name	Command Syntax
Calculate Transform Time Points Command/Query	CALCulate<n>:TRANsform:TIME:POINts
Calculate Voltage Offset Command/Query	CALCulate<n>:OFFSet
Calculate Voltage Range Command/Query	CALCulate<n>:RANGe
Calibration Restore Command	CALibration:RESTore
Calibration Skew Command/Query	CALibration:SKEW
Clock Frequency Command/Query	[SENSe]:SWEep:CLOCk:FREQuency
Clock Source Command/Query	[SENSe]:SWEep:CLOCk:SOURce
Format Byte Command/Query	FORMat:BORDer
Initiate Command/Query	INITiate[:IMMediate]
Initiate Continuous Command/Query	INITiate:CONTinuous
Input Attenuation Command/Query	[SENSe]:INPut<n>:ATTenuation
Input Coupling Command/Query	[SENSe]:INPut<n>:COUPling
Input Filter State Command/Query	[SENSe]:INPut<n>:FILTer[:LPASs][:STATe]
Input Impedance Command/Query	SENSe]:INPut<n>:IMPedance
Input State Command/Query	[SENSe]:INPut<n>[:STATe]
Input Voltage Offset Command/Query	[SENSe]:VOLTage<n>:RANGe:OFFSet
Input Voltage Protection State Command/Query	[SENSe]:VOLTage<n>:PROTection[:STATe]
Input Voltage Range Command/Query	[SENSe]:VOLTage<n>:RANGe:PTPeak
Measure AC RMS Query	MEASure:VOLTage:AC
Measure Amplitude Query	MEASure:VOLTage:AMPLitude
Measure Average Voltage Query	MEASure:VOLTage:AVERage
Measure Cycle Average Query	MEASure:VOLTage:CAVerage
Measure Cycle Frequency Query	MEASure:VOLTage:CFRequency
Measure Cycle Period Query	MEASure:VOLTage:CPERiod
Measure Cycle RMS Query	MEASure:VOLTage:CRMS
Measure DC RMS Query	MEASure:VOLTage:DC
Measure Edge Command/Query	MEASure:EDGE
Measure Effective Number of Bits Query	MEASure:VOLTage:ENOB
Measure Fall Crossing Time Query	MEASure:VOLTage:FTCRoss
Measure Fall Overshoot Time Query	MEASure:VOLTage:FOVershoot
Measure Fall Preshoot Time Query	MEASure:VOLTage:FPReshoot
Measure Fall Time Query	MEASure:VOLTage:FTIMe

Name	Command Syntax
Measure Frequency Query	MEASure:VOLTage:FREQuency
Measure Gate Points Command/Query	MEASure:GATE:POINts
Measure Gate Time Command/Query	MEASure:GATE[:TIME]
Measure High Voltage Query	MEASure:VOLTage:HIGH
Measure Low Voltage Query	MEASure:VOLTage:LOW
Measure Maximum Voltage Query	MEASure:VOLTage:MAXimum
Measure Method Command/Query	MEASure:METHod
Measure Mid Voltage Query	MEASure:VOLTage:MID
Measure Minimum Voltage Query	MEASure:VOLTage:MINimum
Measure Negative Duty Cycle Query	MEASure:VOLTage:NDUTycycle
Measure Negative Width Query	MEASure:VOLTage:NWIDth
Measure Peak-To-Peak Voltage Query	MEASure:VOLTage:PTPeak
Measure Period Query	MEASure:VOLTage:PERiod
Measure Phase Query	MEASure:VOLTage:PHASe
Measure Positive Duty Cycle Query	MEASure:VOLTage:PDUTycycle
Measure Positive Width Query	MEASure:VOLTage:PWIDth
Measure Precision AC Query	MEASure:VOLTage:PAC
Measure Precision DC Query	MEASure:VOLTage:PDC
Measure Reference Command/Query	MEASure:REFerence
Measure Reference Method Command/Query	MEASure:REFerence:METHod
Measure Rise Crossing Time Query	MEASure:VOLTage:RTCRoss
Measure Rise Overshoot Query	MEASure:VOLTage:ROVershoot
Measure Rise Preshoot Query	MEASure:VOLTage:RPReshoot
Measure Rise Time Query	MEASure:VOLTage:RTIMe
Measure Signal-to-Noise Ratio Query	MEASure:VOLTage:SNR
Measure Signal-to-Noise & Distortion Ratio Query	MEASure:VOLTage:SNDR
Measure Spurious-Free Dynamic Range Query	MEASure:VOLTage:SFDR
Measure Time of Maximum Voltage Query	MEASure:VOLTage:TMAXimum
Measure Time of Minimum Voltage Query	MEASure:VOLTage:TMINimum
Measure Total Harmonic Distortion Query	MEASure:VOLTage:THD
Output ECL Trigger Polarity Command/Query	OUTPut:ECLTrg<n>:POLarity

Name	Command Syntax
Output ECL Trigger Source Command/Query	OUTPut:ECLTrg<n>:SOURce
Output ECL Trigger State Command/Query	OUTPut:ECLTrg<n>[:STATe]
Output TTL Trigger Polarity Command/Query	OUTPut:TTLTrg<n>:POLarity
Output TTL Trigger Source Command/Query	OUTPut:TTLTrg<n>:SOURce
Output TTL Trigger State Command/Query	OUTPut:TTLTrg<n>[:STATe]
Output Reference Source Command/Query	OUTPut:REFerence:SOURce
Output Reference State Command/Query	OUTPut:REFerence[:STATe]
Reference Oscillator Source Command/Query	[SENSe]:ROSCillator:SOURce
Status Calibration Condition Query	STATus:QUEStionable:CALibration:CONDition
Status Calibration Enable Command/Query	STATus:QUEStionable:CALibration:ENABle
Status Calibration Event Query	STATus:QUEStionable:CALibration [:EVENt]
Status Frequency Condition Query	STATus:QUEStionable:FREQuency:CONDition
Status Frequency Enable Command/Query	STATus:QUEStionable:FREQuency:ENABle
Status Frequency Event Query	STATus:QUEStionable:FREQuency[:EVENt]
Status Operation Condition Query	STATus:OPERation:CONDition
Status Operation Enable Command/Query	STATus:OPERation:ENABle
Status Operation Event Query	STATus:OPERation[:EVENt]
Status Preset Command	STATus:PRESet
Status Questionable Condition Query	STATus:QUEStionable:CONDition
Status Questionable Enable Command/Query	STATus:QUEStionable:ENABle
Status Questionable Event Query	STATus:QUEStionable[:EVENt]
Status Test Condition Query	STATus:QUEStionable:TEST:CONDition
Status Test Enable Command/Query	STATus:QUEStionable:TEST:ENABle
Status Test Event Query	STATus:QUEStionable:TEST[:EVENt]
Status Voltage Condition Query	STATus:QUEStionable:VOLTage:CONDition
Status Voltage Enable Command/Query	STATus:QUEStionable:VOLTage:ENABle
Status Voltage Event Query	STATus:QUEStionable:VOLTage[:EVENt]
Sweep Mode Command/Query	[SENSe]:SWEep:MODE

Name	Command Syntax
Sweep Offset Reference Command/Query	[SENSe]:SWEep:OREFerence:LOCation
Sweep Offset Time Command/Query	[SENSe]:SWEep:OFFSet:TIME
Sweep Points Command/Query	[SENSe]:SWEep:POINts
Sweep Time Query	[SENSe]:SWEep:TIME
System Error All Query	SYSTem:ERRor:ALL
System Error Count Query	SYSTem:ERRor:COUNt
System Error Query	SYSTem:ERRor[:NEXT]
System Memory Query	SYSTem:MEMory
System Version Query	SYSTem:VERSion
Trace Copy Reference Command	TRACe:COPY:REFerence<n>
Trace Data Command/Query	TRACe[:DATA]
Trace Load Calculate Query	TRACe:LOAD:CALCulate<n>
Trace Load Input Query	TRACe:LOAD:INPut<n>
Trace Load Reference Command/Query	TRACe:LOAD:REFerence<n>
Trace Preamble Command/Query	TRACe:PREamble
Trigger A Event Count Command/Query	TRIGger[:A]:ECOunt
Trigger A Pulse High Limit Command/Query	TRIGger[:A]:PULSe:HLIMit
Trigger A Pulse Low Limit Command/Query	TRIGger[:A]:PULSe:LLIMit
Trigger A Slope Command/Query	TRIGger[:A]:SLOPe
Trigger A Source Command/Query	TRIGger[:A]:SOURce
Trigger A Type Command/Query	TRIGger[:A]:TYPE
Trigger A Video Field Command/Query	TRIGger[:A]:VIDeo:FIELd
Trigger A Video Line Command/Query	TRIGger[:A]:VIDeo:LINE
Trigger A Video Standard Command/Query	TRIGger[:A]:VIDeo:STANdard
Trigger B Slope Command/Query	TRIGger:B:SLOPe
Trigger B Source Command/Query	TRIGger:B:SOURce
Trigger B State Command/Query	TRIGger:B[:STATe]
Trigger External Level Command/Query	TRIGger:EXTernal:LEVel
Trigger Hold Off Command/Query	TRIGger:HOLDoff
Trigger Input Level Command/Query	TRIGger:INPut<n>:LEVel
Trigger Pattern Mask Command/Query	TRIGger:PATTern:MASK
Trigger Pattern Truth Command/Query	TRIGger:PATTern:TRUTh

Name	Command Syntax
Trigger Timestamp Query	TRIGger:TIMestamp

ZTEC Instruments