14 Bit and 16 Bit
PXIbus and PCIbus
Digital Storage Oscilloscope

Models ZT410PXI and ZT410PCI

User's Manual: 0004-000052
Revision 1

April 24, 2006

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

Copyright 2006 by ZTEC Instruments

Printed in the United States of America.

All rights reserved under copyright laws of the United States and other countries.

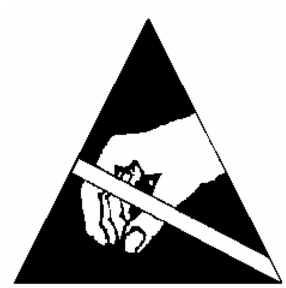
All technical data and computer software documentation contained herein is proprietary and confidential to ZTEC Instruments, Inc. or its licensor. The reproduction and/or transmission of this publication in whole or in part by any means, electronic or mechanical, is prohibited without the prior written consent of ZTEC Instruments, Inc.

ZTEC Instruments, Inc. and the ZTEC logo are registered trademarks of ZTEC Instruments.

ZTEC Instruments has attempted throughout this publication to distinguish proprietary trademarks from descriptive terms by following the capitalization style used by the manufacturer. Product names listed are trademarks of their respective manufacturers. Company names listed are trademarks or trade names of their respective companies.

LabVIEW™	National Instruments Corporation
Windows™	Microsoft Corporation

The material in this manual is for informational purposes only and is subject to change without notice. ZTEC Instruments, Inc. assumes no responsibility for any error or for consequential damages that may result from the use or misinterpretation of any of the procedures in this publication.



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 must be conductive and reliably connected to an earth ground with a safety resistance of approximately 250 kilo Ohms.

- The surface must NOT be metal. A resistance of 30–300 kilo Ohms 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 must 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 kilo Ohms to 1 mega Ohm 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

Table of Contents

List of Figures

List of Tables

Introduction

Description

The Model ZT410PXI (See Figure 1) and Model ZT410PCI (See Figure 2) are 14-bit or 16-bit Digital Storage Oscilloscopes. Each is built around a high speed, deep memory digitizer and embedded TMS320VC5409 Digital Signal Processor (DSP). The ZT410PXI is a single-wide PXIbus module (3U); the ZT410PCI is a single-slot PCIbus card (short). Together with the host processor and software, each provides a simple and 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 ZT410PXI and ZT410PCI digitize signals in user-selectable record sizes. Analog signal processing allows selectable one or two channel operation, input impedance, AC or DC coupling, and input signal range and offset. Waveform records are transferred from the digitizer into DSP memory, which can then be accessed from the PXIbus or PCIbus. 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 PXIbus or PCIbus host processor.

Product Options, Part Numbers & Naming Convention

Part Number	Maximum Sample Rate	Sampling Resolution	Memory	Manual Naming
ZT410PXI-20	500 MS/s	14-bit	1 MSample	ZT410-2X
ZT410PXI-21	500 MS/s	14-bit	16 MSample	ZT410-2X
ZT410PXI-50	400 MS/s	16-bit	1 MSample	ZT410-5X
ZT410PXI-51	400 MS/s	16-bit	16 MSample	ZT410-5X

Table 1: ZT410PXI and ZT410PCI Product Options

Table 1 shows the product options for the ZT410PXI and ZT410PCI that define maximum sample rate, sampling resolution and memory size. Note that the PCI products are not shown within Table 1, but simply replace PXI with PCI within the part number. 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 ZT410PXI and ZT410PCI product line shall hereafter be designated as ZT410. 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 ZT410PXI and ZT410PCI

cPCI/PXI Interoperability for the ZT410PXI

The ZT410PXI module is compliant with the Peripheral Component Interconnect (PCI) Version 2.2 standard. This PCI Bus system features 33 MHz speed, 32-bit addressing, and 16-bit data. The ZT410PXI supports universal voltage requirements of +3.3 volts and +5 volts.

The ZT410PXI may be used in both CompactPCI (cPCI) and PCI eXtensions for Instrumentation (PXI) applications. The PXI standard is a derivative of cPCI and maintains a high level of interoperability; as a result, PXI cards may safely be used in cPCI mainframes for most applications. Compatibility issues arise because the PXI standard reserves several normally undedicated pins on the cPCI P2 connector for a selection of trigger, clock, and inter-card communication functions. Before installing the ZT410PXI into a cPCI mainframe that uses the P2 connector, the user must ensure that pin use conflicts will not occur.

PCI Interoperability for the ZT410PCI

The ZT410PCI module is compliant with the Peripheral Component Interconnect (PCI) Version 2.2 standard. This PCI Bus system features 33 MHz speed, 32-bit addressing, and 16-bit data. The Universal PCI connector is compatible with universal +3.3V and +5V connections.

PCI Timing Expansion Connector

The PCI Timing Expansion Connector enables the synchronization of the trigger, arm and clock timebase for multiple PCI modules from ZTEC Instruments. When using this configuration, one board acts as a timing and arm source; all of the other boards are receivers. The Trigger0-7 inputs for the PCI module mating connector are active low, with internal pull-up resistors. The outputs are open-collector, and consequently can be driven by any module. In most cases, the one source will drive all others in the system, but a wired-or configuration can also be used.

When installing multiple PCI modules, use the ribbon cable (FFSD-10-D-6.00-01-N-D4) to connect to the PCI Timing Expansion Connectors on the top of the modules (See Figure 2). The ribbon cable can be matched in length to achieve a zero clock skew between modules. Table 2 shows the pin out for the mating connectors to the PCI Timing Expansion Connector.

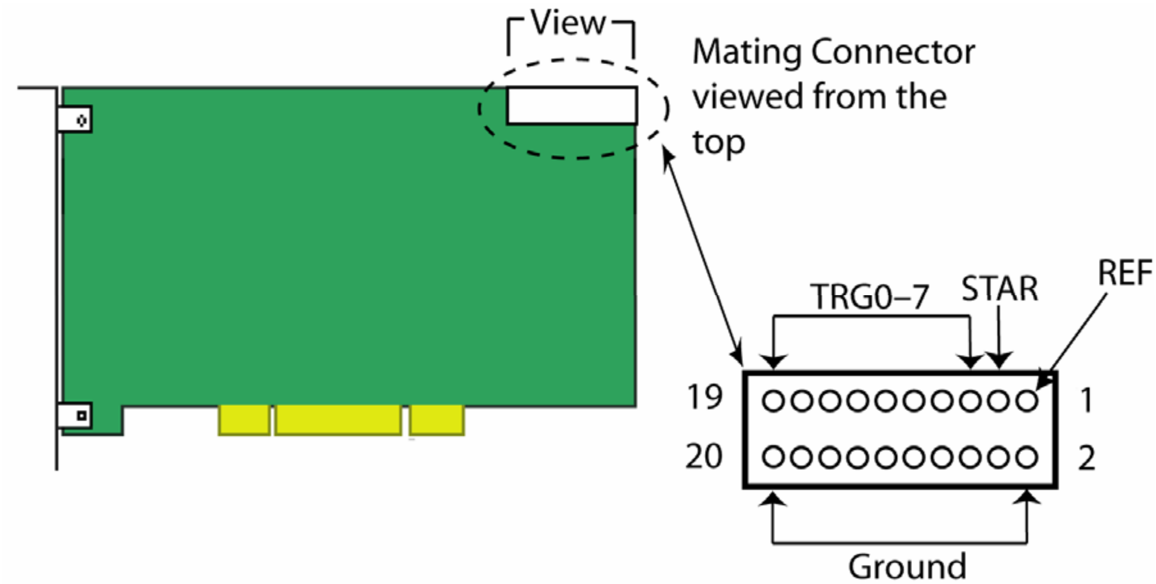


Figure 2: ZT410PCI Timing Expansion Connector

Pin	Function	Pin	Function
1	REF	**2**	GND
3	STAR	**4**	GND
5	TRG7	**6**	GND
7	TRG6	**8**	GND
9	TRG5	**10**	GND
11	TRG4	**12**	GND
13	TRG3	**14**	GND
15	TRG2	**16**	GND
17	TRG1	**18**	GND
19	TRG0	**20**	GND

Table 2: ZT410PCI Timing Expansion Connector Pin Out

Front Panel

A photo of the ZT410PXI front panel is shown in Figure 3 and a photo of the ZT410PCI front panel is shown in Figure 3. On the front panel, there are two SMB coaxial connectors and two BNC connectors. Table 3 lists the front panel connector functions.

Figure 3: Photo of the ZT410PXI Front Panel

Figure 4: Photo of the ZT410PCI Front Panel

Label	Description
Channel 1	Channel 1 input signal (BNC Connector)
TRG I/O	Trigger Input/Output (SMB Connector)
EXT CLK	External Clock Input (SMB Connector)
Channel 2	Channel 2 Input Signal (BNC Connector)

Table 3: Front Panel Connectors

Accessory Kits

ZTEC Instruments, Inc. offers the following kit solutions for the ZT410:

ZTPXIKIT-00

This kit consists of four each of the following cables:

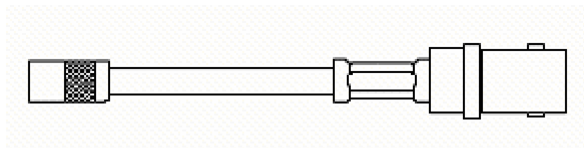


Figure 5: Three-Inch Cable, SMB to Female BNC

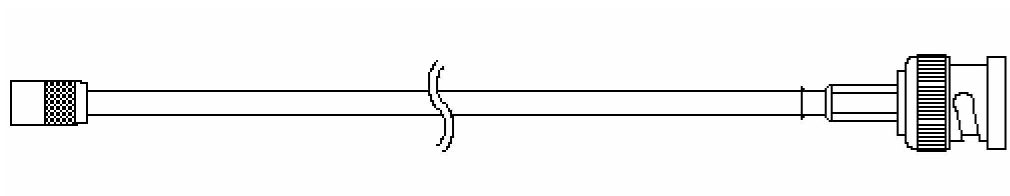


Figure 6: One-Meter Cable, SMB to Male BNC

Probes

ZTEC Instruments, Inc. offers the probe accessories for the ZT410. Consult the website for a list of compatible probe accessories.

Functionality and Operation

Functional Block Diagram

The functional blocks of the ZT410 are shown in Figure 7. 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 ZT410, 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.

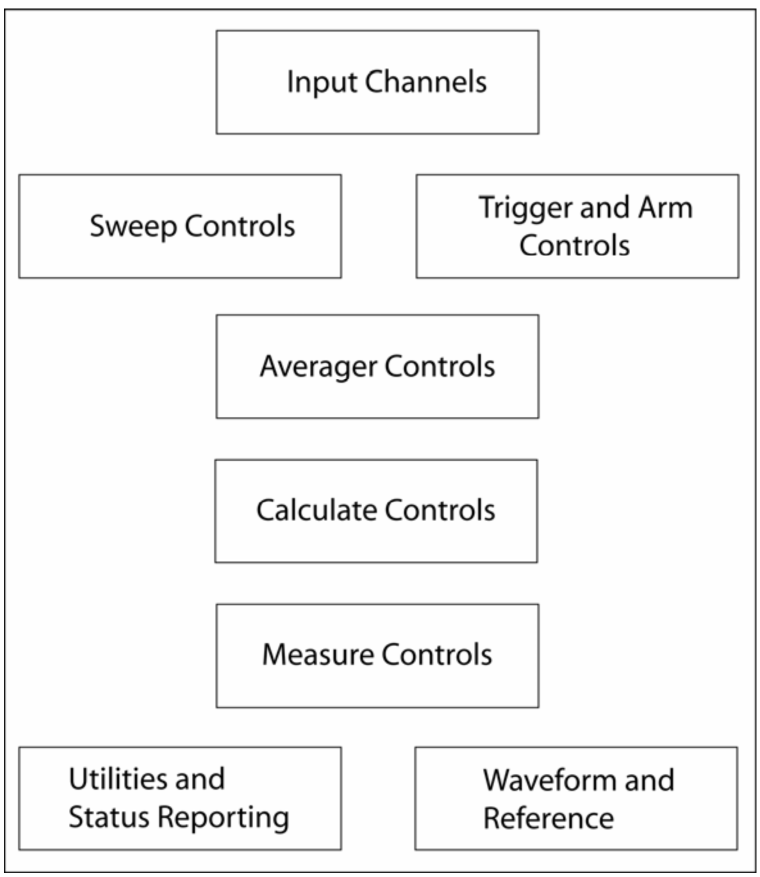


Figure 7: ZT410 Block Diagram

Data Flow

The data flow of the ZT410 is shown in Figure 8. Raw data is input through the two 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 ZT410 has four types of available acquisition: normal, average, envelope, and equivalent-time (See *Averager Controls*). New waveforms can now be mathematically created in the ZT410. 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 ZT410 is capable of providing measurements using the following methods: Entire Waveform, Gated by Time, and Gated by Points. The ZT410 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.

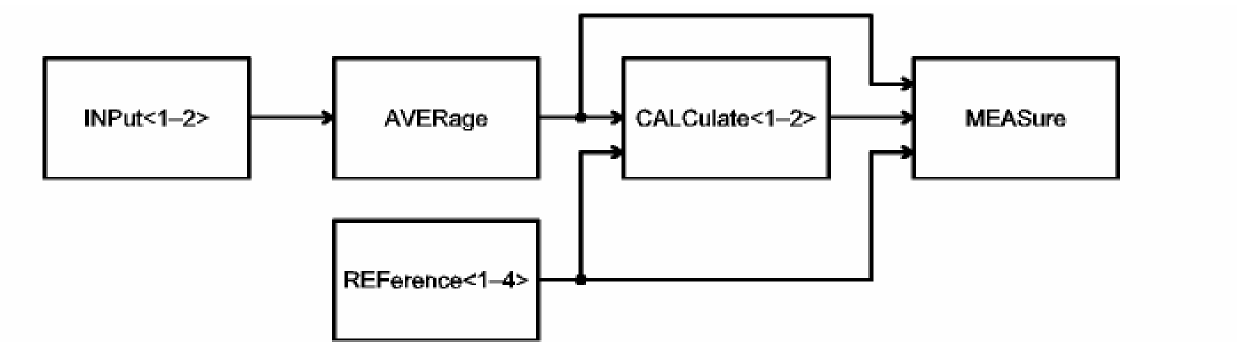


Figure 8: Data Flow

Input Channels

Product Option	1 Channel Maximum Sample Rate	2 Channel Maximum Sample Rate	Sampling Resolution
ZT410-2X	500 MS/s	250 MS/s	14-bit
ZT410-5X	400 MS/s	200 MS/s	16-bit

Table 4: Sample Rates and ADC Resolution of Product Options

Input Channel Enable

The ZT410 allows capture of two input signals at rates from 10 kS/s up to one-half the maximum sample rate. One interleaved channel can be sampled at the maximum sample rate. The maximum sample rate is 500 MS/s for the ZT410-2X and 400 MS/s for the ZT410-5X.

Input Signal Conditioning

The ZT410 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 high pass filter to be inserted before the input amplifiers to limit the input signal frequency: 200 kHz for low impedance (50 Ω) and 10 Hz for high impedance (1 MΩ). Selecting DC 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 not apply signals having a peak value over ±25 VDC when using the high impedance (1MΩ) setting as unit damage may result.

Do not 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 5: 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 ZT410 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 steps of 1, 2.5, 4 or 5. Two input signals can be sampled simultaneously at rates from 10 kS/s up to one-half the maximum sample rate. One interleaved channel can be sampled at the maximum sample rate. The maximum sample rate is 500 MS/s for the ZT410-2X and 400 MS/s for the ZT410-5X. 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 ZT410 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 PXIbus backplane CLK10 or the PCI Timing Expansion Connector 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 ZT410 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 one channel is enabled, the sampling occurs at the applied external frequency. When both 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 setup 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 the *Sweep Time Query*. Record lengths can range from 100 Samples up to 8 MSamples per channel, or up to 16 MSamples/channel (using 1 channel interleaved). Memory options for the ZT410 are 1 MSamples and 16 MSamples total.

The ZT410 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 ZT410 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.

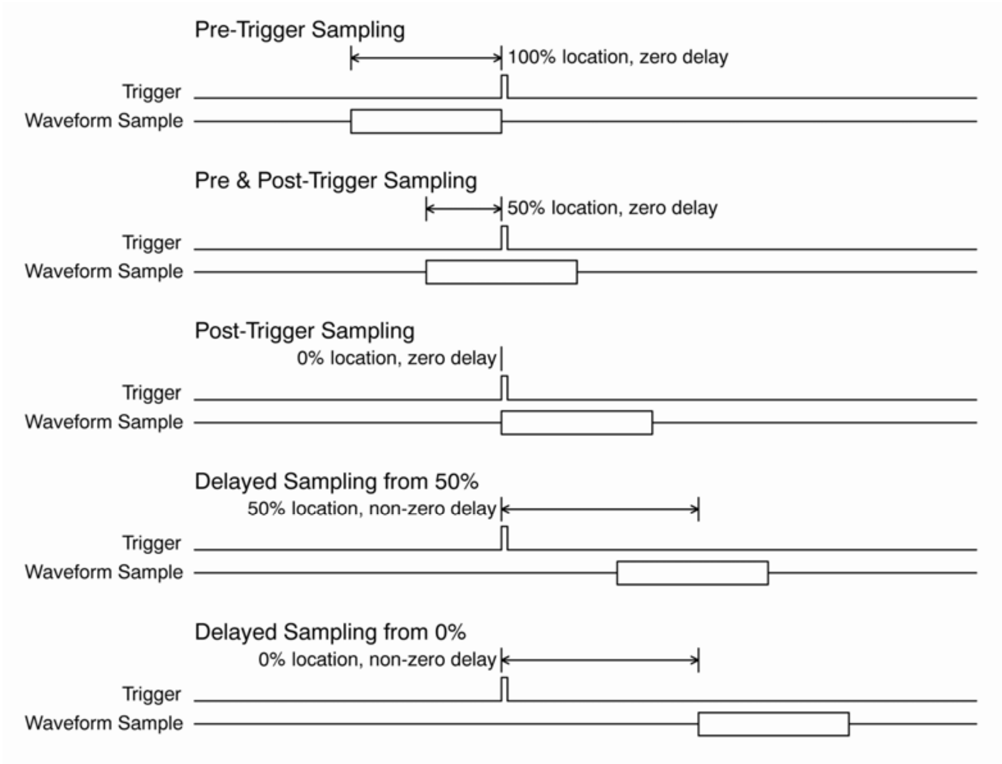


Figure 9: ZT410 Sweep Reference Scenarios

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 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 ZT410 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 ZT410, 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 ZT410: 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 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 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 10). 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*).

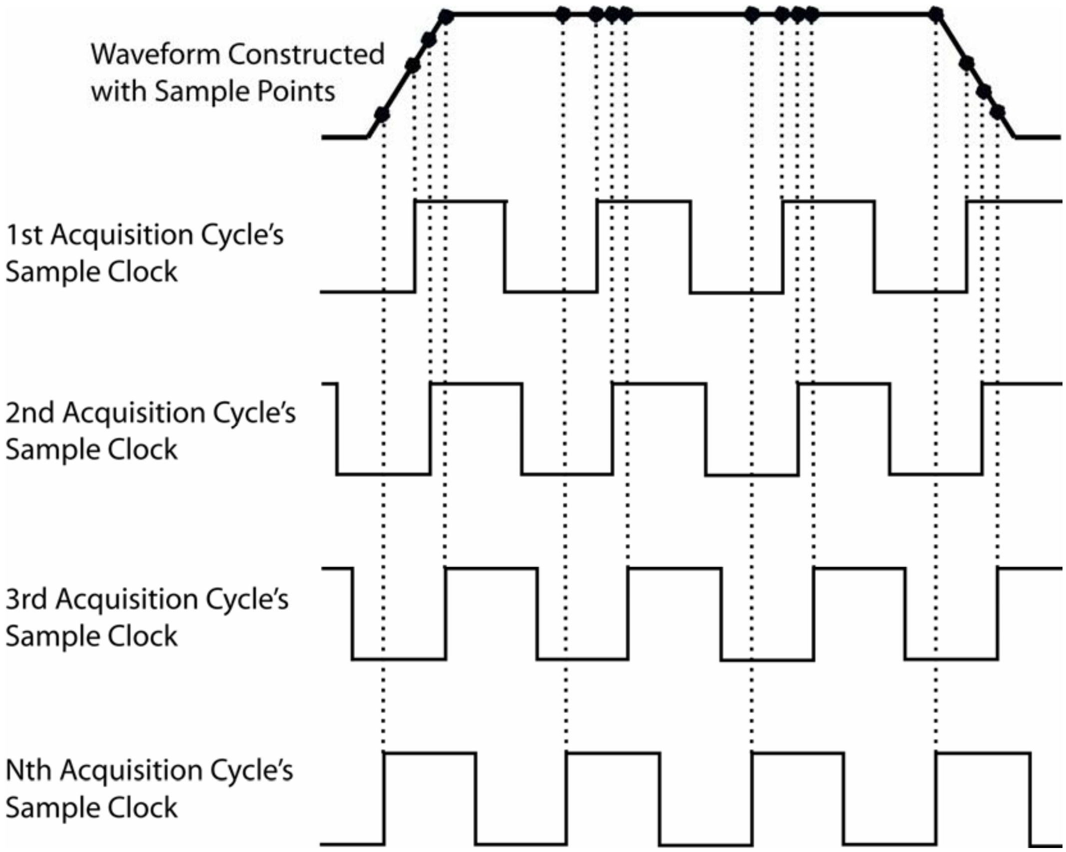


Figure 10: Equivalent Time Acquisition

Trigger and Arm Controls

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

Trigger Initiate Model

The ZT410 uses an arm-trigger model to control data acquisition. All acquisition cycles are started using the *Initiate Command*. Upon receiving an “initiate”, the ZT410 will sequence into the “wait for arm” state. When the arm source goes active or if the arm source is set to immediate, the ZT410 will sequence into the “wait for trigger” state. When a trigger event is detected, the ZT410 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 ZT410 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.

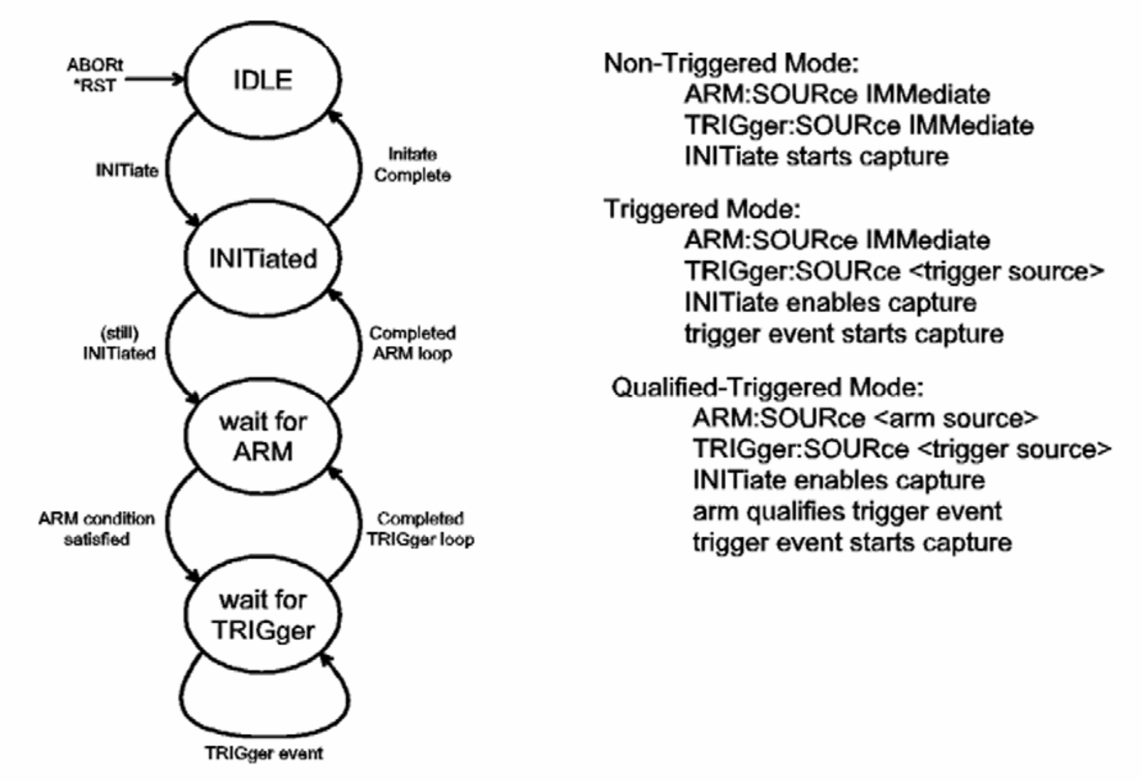


Figure 11: Trigger Initiate Model

Trigger Processing

The ZT410 accepts triggers from the following sources:

- Channels 1 and 2 (BNC)
- External Trigger (SMB)
- Star Trigger (PXI Backplane or PCI Timing Expansion Connector)
- TTL Trigger 0–7 (PXI Backplane or PCI Timing Expansion Connector)
- Pattern
- Software

Figure 12 shows a diagram of the ZT410 trigger processing.

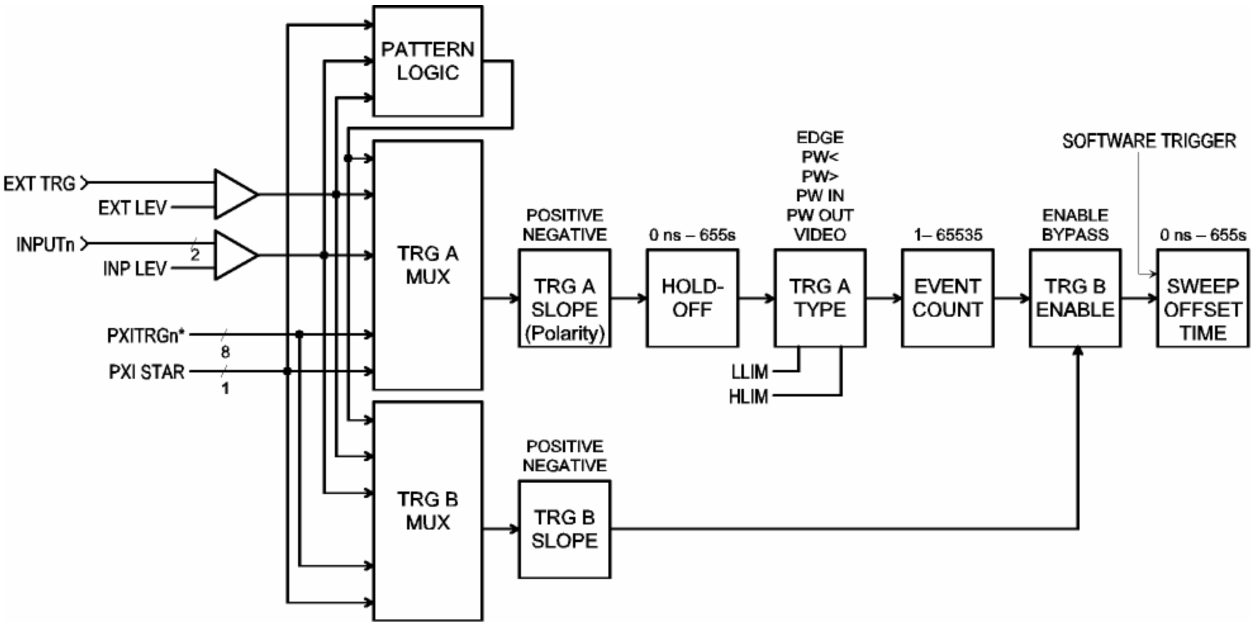


Figure 12: Trigger Processing

Trigger Types

Several types of triggers are used with the ZT410, 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 irregardless 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 2, the External Trigger, and the Star Trigger (PXI Backplane or PCI Timing Expansion Connector). 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, Star is LOW, and the External Trigger is HIGH.

Video Trigger

A video trigger occurs when the ZT410 finds valid video signal synchronization. The ZT410 includes a mode for triggering on NTSC (60 Hz), PAL (50 Hz), and SECAM (50 Hz) format video signals, 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 ZT410 allows a range of 1 to 65535 trigger events. 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 are enabled.

Arm

Each trigger must be qualified by an associated arm state condition. The arm polarity can be either positive or negative. The arm is sourced from the following:

- External Trigger (SMB)
- Star Trigger (PXI Backplane or PCI Timing Expansion Connector)
- TTL Trigger 0–7 (PXI Backplane or PCI Timing Expansion Connector)
- 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

The ZT410 can drive signal outputs on any combination of the front panel External Trigger and the eight TTL trigger outputs (PXI backplane or PCI Timing Expansion Connector). Each output can be independently configured with unique source and enable controls.

The front panel External Trigger output can be selected from the following sources:

- Trigger event
- Arm event
- OPC event that occurs when all ZT410 operations are complete using the *Operation Complete Command*.
- Constant level
- 10 MHz timebase reference clock
- 500 Hz clock
- 10 ns pulse at 1 ms repetition interval

The eight TTL trigger outputs can be selected from the following sources:

- Trigger event
- Arm event
- OPC event that occurs when all ZT410 operations are complete using the *Operation Complete Command*.
- Constant level

Calculate Controls

The ZT410 can create new waveforms mathematically. The unit has two calculate channels, each capable of a 32K maximum waveform size. Sources include the following:

- 2 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 upon 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 channel from the other channel.

Multiply

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

Copy

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

Invert

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

Integral

Use the *Calculate Integral Command* to calculate the integral of a 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 from any given function. The derivative equation is:

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

Absolute Value

Absolute value establishes an absolute value of the calculation source. All negative values are converted to positive (See the *Calculate Absolute Value Command*).

Limit Test

This conducts a limit test on a waveform (See Chapter 3 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 Most Recent Measurement Performed.

- 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 3 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 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, 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 6: FFT Sample Range and Size

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

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-Harris	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 7: 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 ZT410 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.

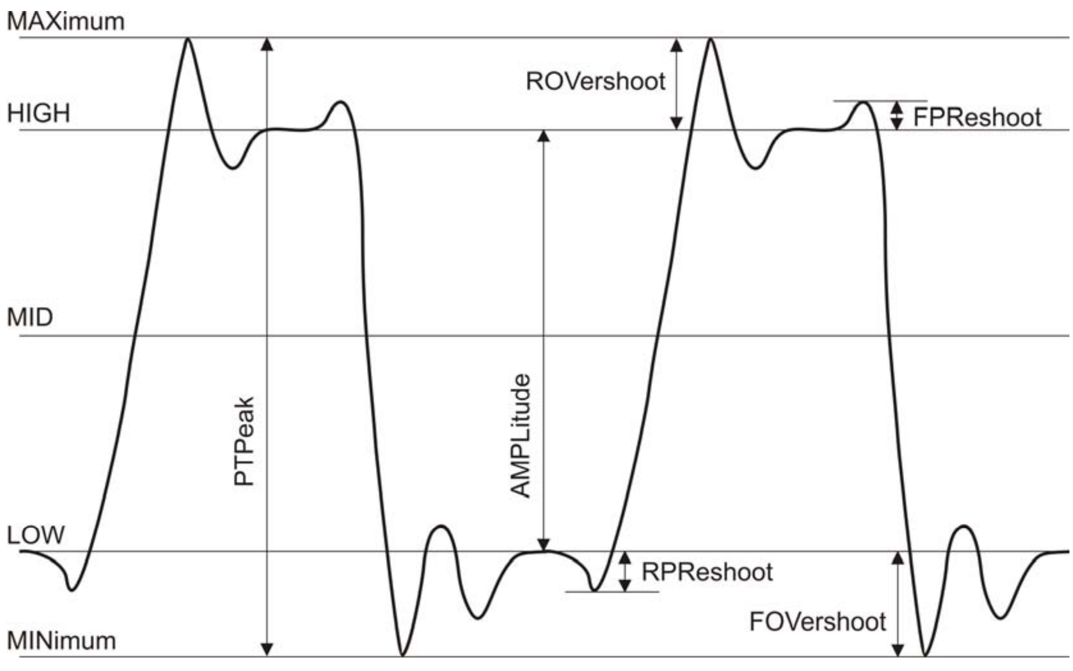


Figure 13: Vertical-Axis Measurement Terminology

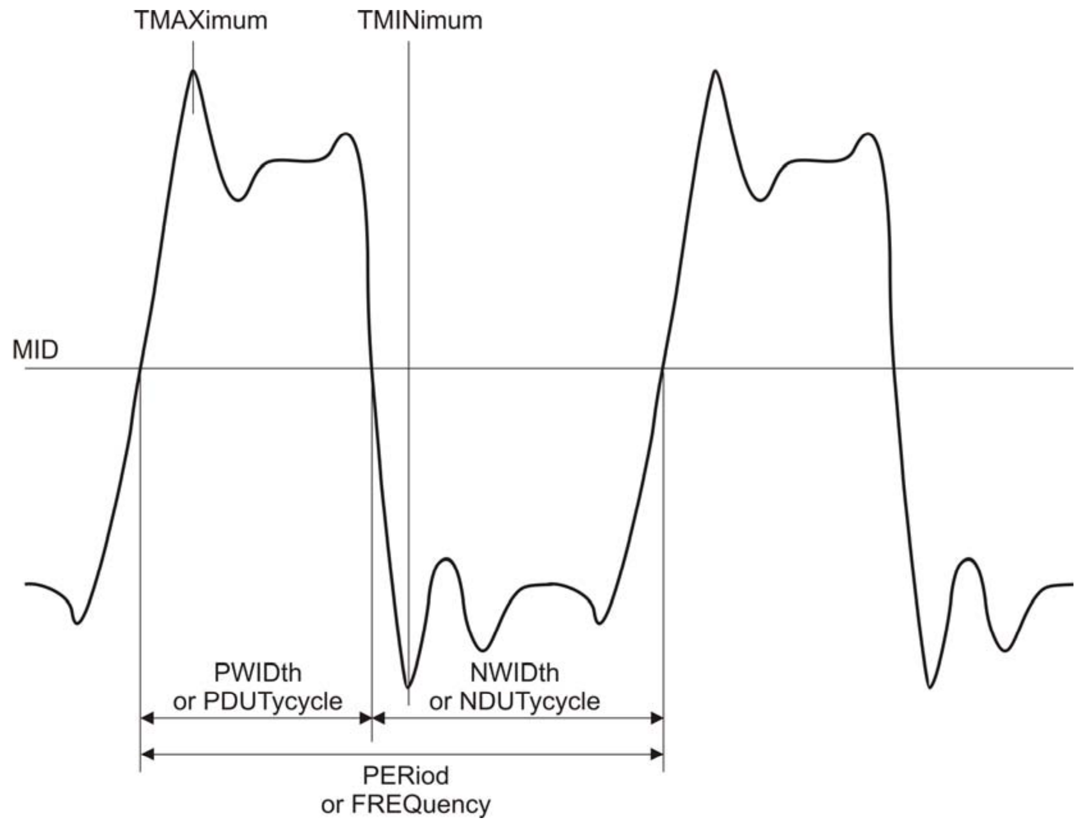


Figure 14: Horizontal-Axis Measurement Terminology

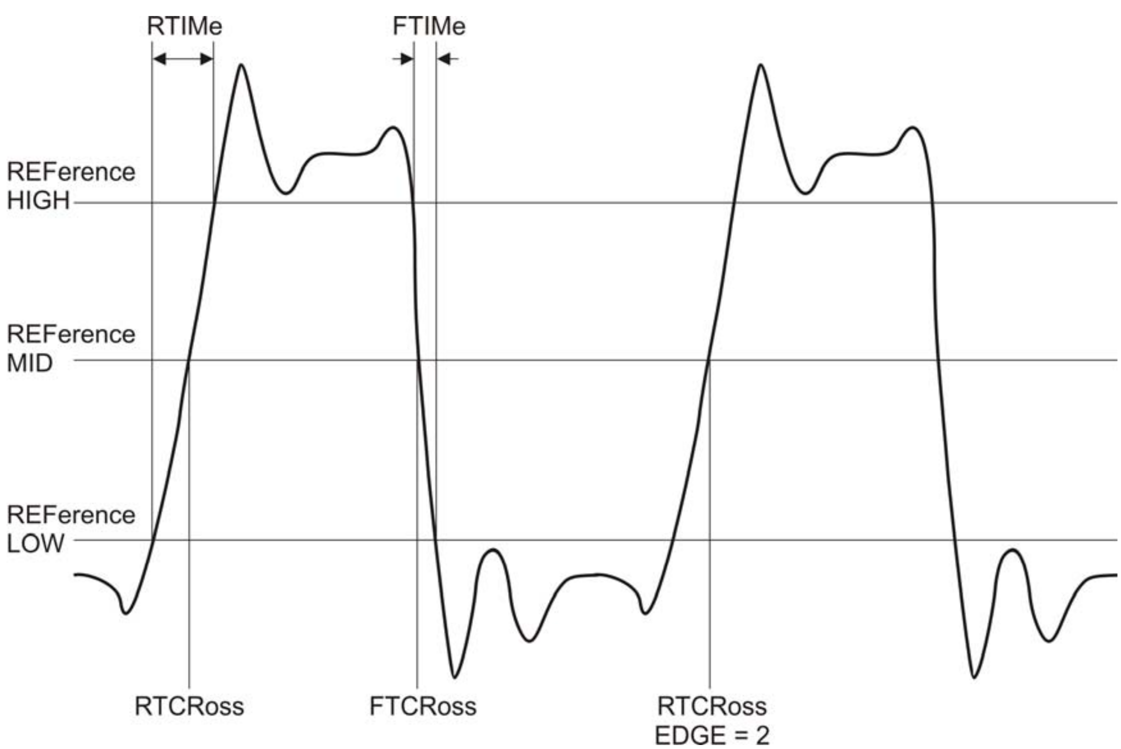


Figure 15: Edge Measurement Terminology

Frequency-Domain Measurements

Figure 16 shows 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:

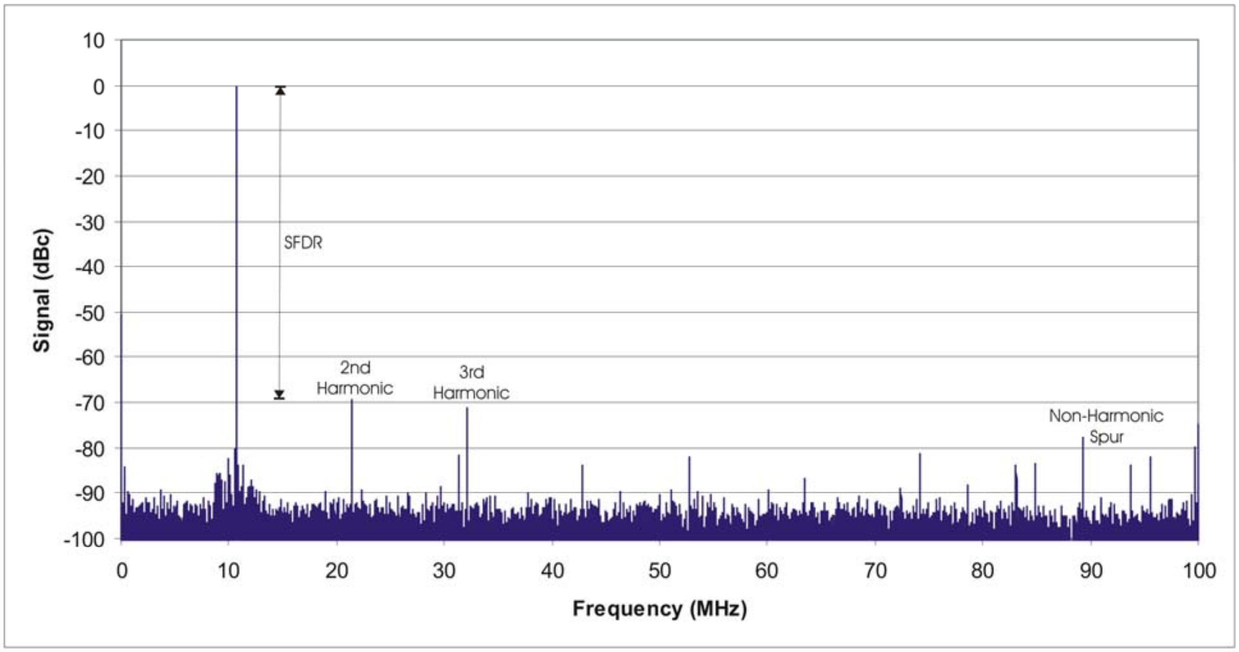


Figure 16: Frequency-Domain Measurement Terminology

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 ZT410 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 17).

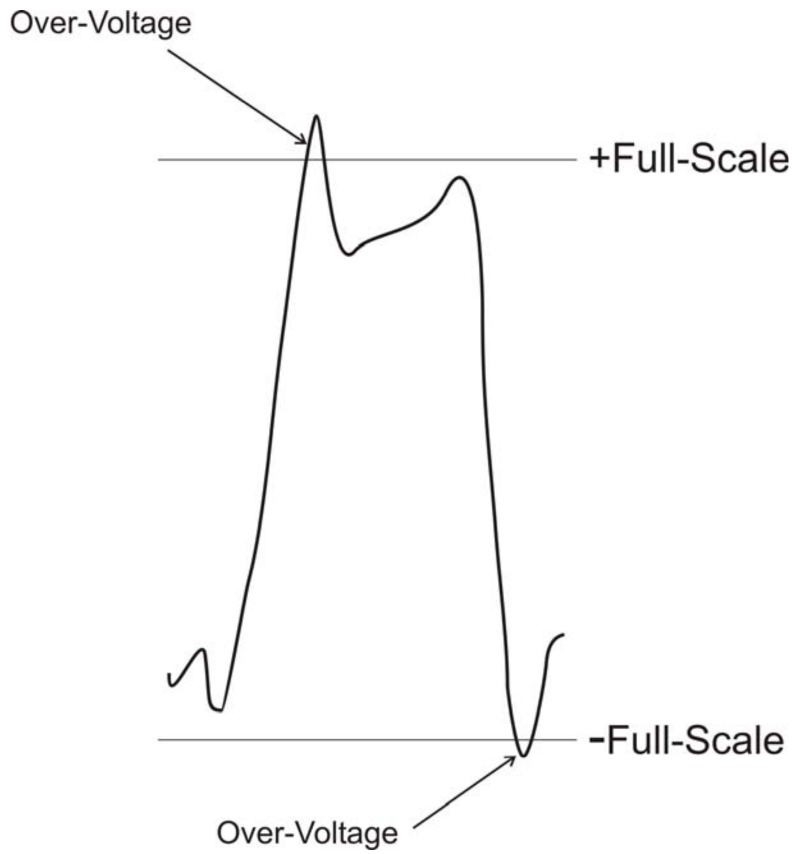


Figure 17: Invalid Voltage Measurements

<u>RMS Measurements</u>

The ZT410 will always return the measured value for an RMS measurement, even if a signal is clipped (over-voltage).

<u>Cycle and Width Measurements</u>

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 ZT410 will return the invalid measurement code.

<u>Rising and Falling Edge Measurements</u>

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 18), 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.

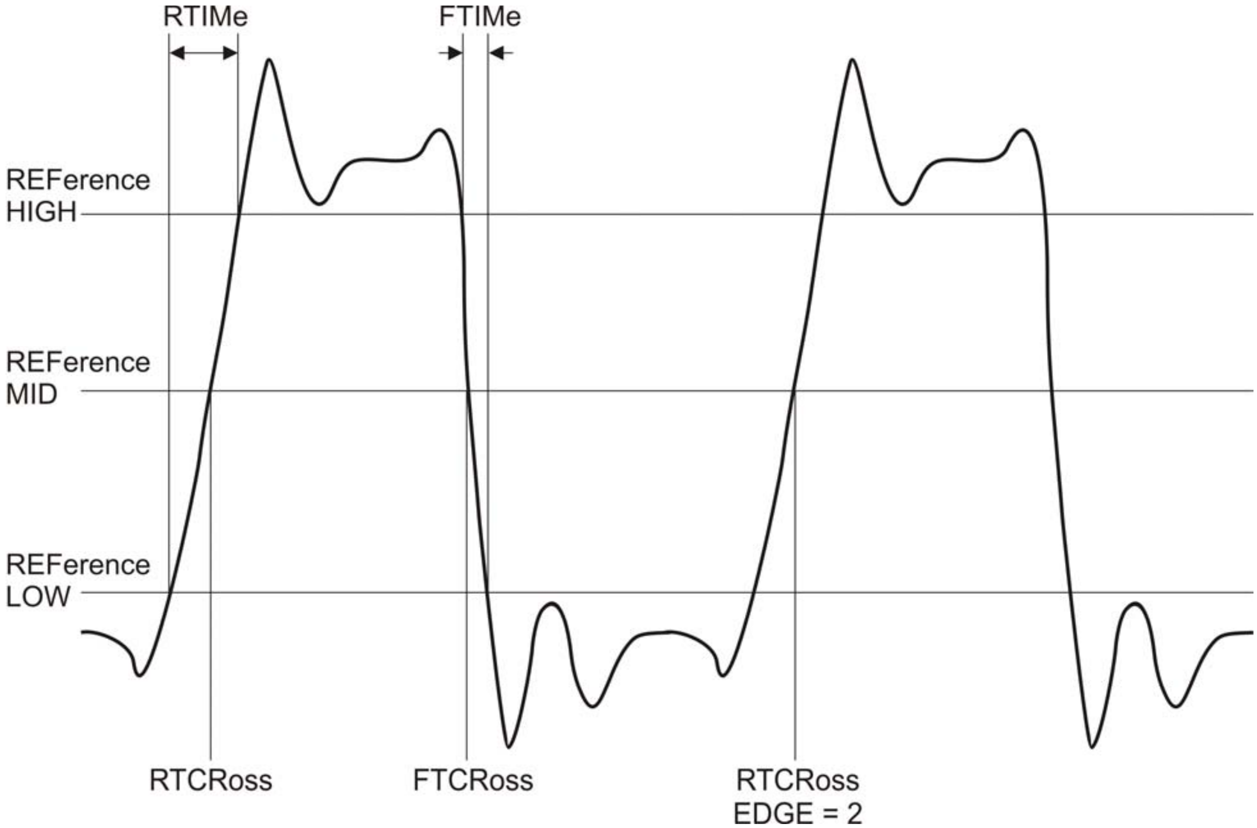


Figure 18: Invalid Edge Measurements

Frequency-Domain Measurements

The ZT410 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 are the measurement methods available on the ZT410: Entire Waveform, Gated by Time, and Gated by Points.

Entire Waveform

Entire Waveform measurement is used to perform measurements upon the entire captured waveform.

Gated by Time

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.

Gated by Points

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.

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 can be set either by 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.

Measurement Edge

The measurement edge is the waveform edge used in edge-related measurements (See *Measure Edge Command*). 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.

Waveform and Reference

Read Waveform

The ZT410 uses a shared PXI memory to report acquisition results to the PXI host processor. Because the shared PXI 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:
  - 0 = Invalid waveform
  - 1 = Normal voltage-time waveform
  - 2 = Averaged voltage-time waveform
  - 3 = Envelope waveform
  - 4 = Equivalent Time waveform
  - 16 = Frequency Domain Waveform (FFT)

- Count: The number of waveforms processed to produce an output, N average count for averaged or enveloped waveforms, or equivalent time waveforms.

- 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 ZT410 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 32 kSamples record sizes.

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

Utilities and Status Reporting

Auto Scale

The ZT410 can automatically adjust settings to the input signal. 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 (Available if signals are present on both enabled input channels)

Reset and Device Clear

Use the *Reset Command* to perform a hard reset of the ZT410. This stops all acquisition and configures the unit to its default state. See *Appendix 2, Default Reset Conditions,* for a listing of all ZT410 default conditions.

Save and Recall States

The ZT410 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 (0) 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 ZT410: 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 19 shows the complete status register structure for the ZT410. 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 ZT410 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.

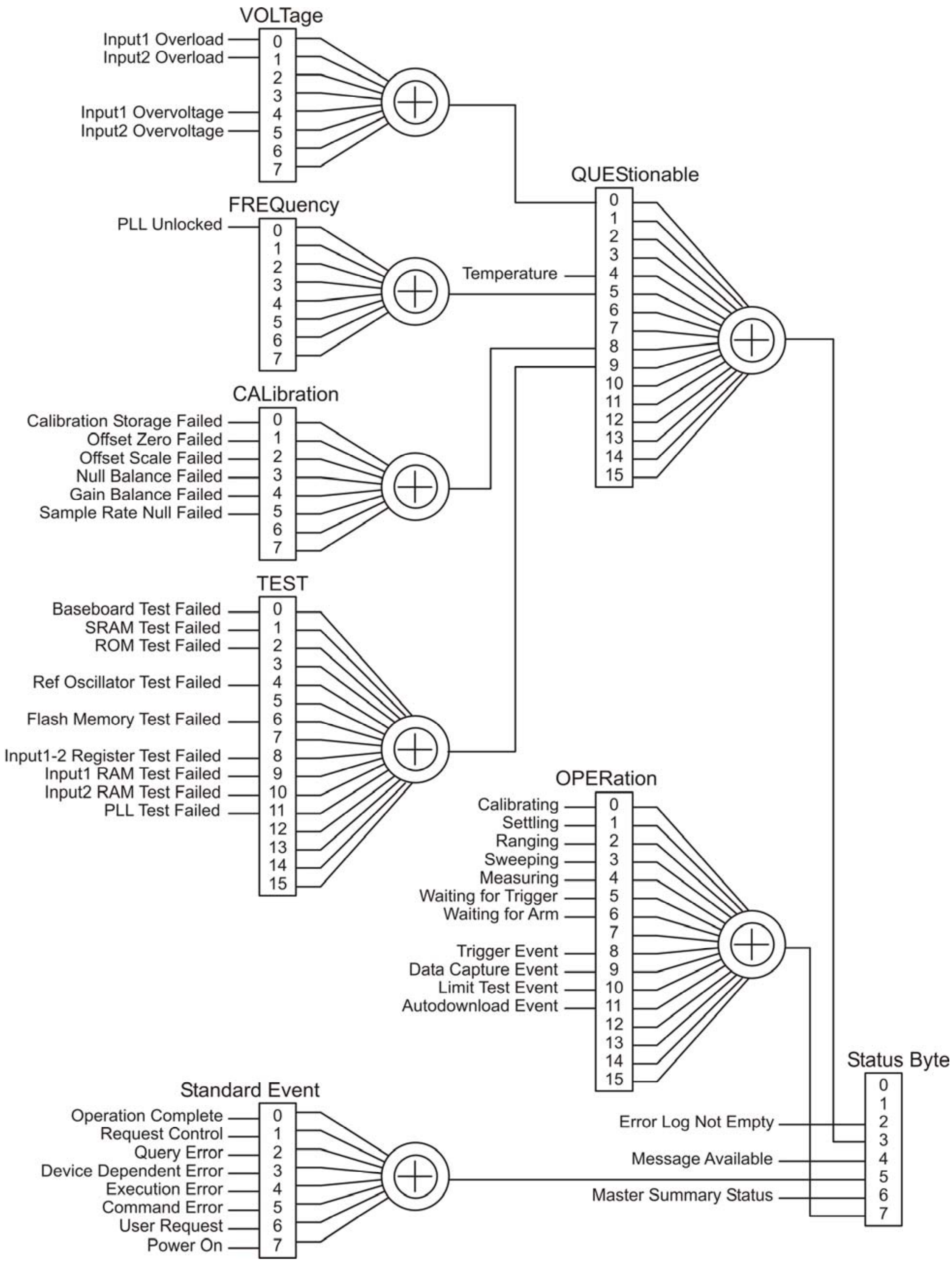


Figure 19: Status Registers

Status Commands and Queries

The device status of the ZT410 can be viewed at any time using any of the following commands or queries:

Note: See the appropriate command or query syntax in Chapter 3.

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.
Status Frequency Enable Command	Allows the user to enable or disable the bits in the

Command or Query	Response
	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 1's. For the mandatory status data structures, it sets the enable register to 0's. 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 8: System Status

Self Test

The ZT410 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}	Unused Bits
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
$F000_{16}$	Unused Bits

Table 9: Self-Test Errors

Calibrate

The ZT410 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 ZT410 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 ZT410 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,ZT410PXI-50,S/N 100,Version 1.00

Temperature Monitoring

Use the *System Temperature Query* to return the temperature value of the instrument in degrees Celsius. Temperature has a high threshold of 65 degrees Celsius. If the instrument temperature exceeds the high threshold, the instrument will immediately power down and must be reset to return to a normal operating state.

Command Interface

Introduction

The ZTEC Instruments, Inc. ZT410 uses a non-queued, one command packet in / one response packet out, control technique that is encapsulated in the device driver software. Burst data transfers may proceed in parallel with command/response activity.

Both models use the Texas Instruments PCI2040 PCI-to-DSP Bridge to handle PCIbus traffic. The PCI2040 responds to configuration space queries by the PCI bus and presents a simple four register interface for commands and data transfers. PCI Base Address Register 1 points to this four register interface. Table 10 shows the command and data transfer interface.

Address		
AD31–AD15	**AD14–AD00**	**Description**
Set by config	0000_{16}	Control Register
Set by config	0800_{16}	Data Register (auto-increment)
Set by config	1000_{16}	Address Register
Set by config	1800_{16}	Data Register (no increment)

Table 10: Low Level PCI Interface

The control register is set by the initialization function to a value of 0101_{16} and should be left unchanged. Host access to shared memory is accomplished by writing to the Address Register to set the target address and then reading/writing one of the data registers. Any access to address 0800_{16} causes the contents of the Address Register to be incremented. This facility is useful for transferring command parameters or data blocks.

Commands to and responses from the ZT410 are exchanged using shared memory located within the DSP. Table 11 describes the memory map.

Address Register	Name	Description
4000_{16}	HOST_SIGNAL	Host computer command activity signal
4001_{16}	HOST_COMMAND	Host computer command code
4002_{16}–4041_{16}	HOST_PARAMETER	Parameters required by host computer command
4042_{16}	UNIT_STATUS	Status flags
4043_{16}	UNIT_SIGNAL	Response activity signal
4044_{16}	UNIT_RESPONSE	Response code
4045_{16}–4084_{16}	UNIT_PARAMETER	Parameters reported with response

Table 11: Command/Response Memory Map

In order to exchange commands and responses, the host computer and ZT410 must execute the following steps:

1. The host computer must write a 0 to the UNIT_SIGNAL register.
2. The host computer must set up the HOST_COMMAND and HOST_PARAMETERS.
3. The host computer must write a 0002_{16} to the HOST_SIGNAL register and begin monitoring the UNIT_SIGNAL register for a non-zero value.
4. The ZT410 monitors the HOST_SIGNAL register and execute the HOST_COMMAND when a non-zero HOST_SIGNAL is detected.
5. Upon completion of command processing the ZT410 copies the command code into the UNIT_RESPONSE register, set up the UNIT_PARAMETER registers, and write a 0 to the HOST_SIGNAL register.
6. Finally, the ZT410 updates the UNIT_SIGNAL register. This code, as shown in Table 12, defines the command processing result.

Response Code	Meaning
0001_{16}	CMD_DONE, command successfully completed
0002_{16}	CMD_UNK_CMD, unknown command
0004_{16}	CMD_INV_DATA, command parameter error
0008_{16}	CMD_REJECT, command rejected, incompatible with current state
0010_{16}	CMD_FAIL, requested operation failed

Table 12: ZT410 Response Signals Codes

Data Interface

Waveform records are sent to the host computer using the shared memory located on the baseboard DSP chip. This memory always begins at 4085_{16} and has a maximum length of 800_{16} (2048_{10}).

Parameters

Command and response parameters are sent and returned as 16-bit integers. 32-bit integers and 32-bit floating point numbers must be broken into 16-bit integers to be loaded as parameters. The ZT410 processes these 32-bit integers as two 16-bit words as follows:

MSW (16 bits)	LSW (16 bits)

32-bits

Note: MSW equals the Most Significant Word and LSW equals the Least Significant Word

Example:

Host Parameter Array

4002_{16}	MSW (16 bits)
4003_{16}	LSW (16 bits)

MSW, LSW, and Long Value are determined using the following C-language equations:

```
LSW = LONG_VALUE && 0xFFFF;
MSW = LONG_VALUE >> 16;
LONG VALUE = (MSW << 16) + LSW;
```

For a 32-bit floating integer, use the following C-language equations to determine Long_value, and Float_value:

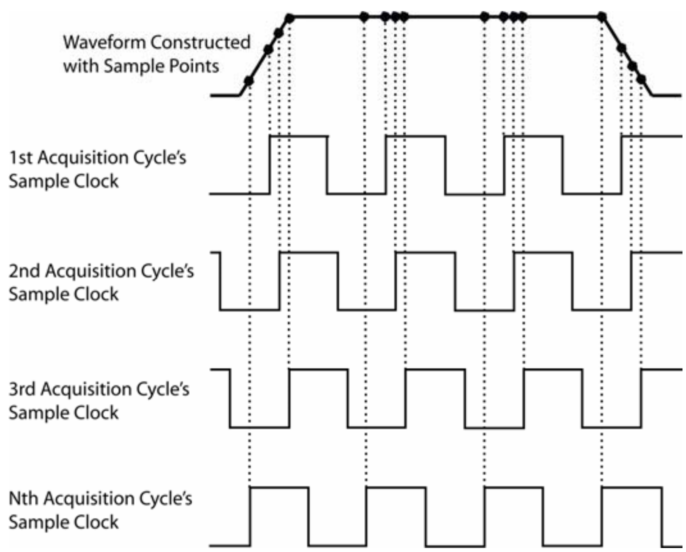
```
LONG_VALUE = *(LONG*) &FLOAT_VALUE;
MSW = (LONG_VALUE) && 0xFFFF;
LSW = (LONG_VALUE) >> 16;
LONG VALUE = (MSW << 16) + LSW;
FLOAT_VALUE = *(FLOAT*) &LONG_VALUE;
```

Commands and Queries

Name	Description
Abort Command 0305_{16}	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** 0305_{16} **Query Syntax** None **Parameters** None
Arm Command 0306_{16} **Arm Query** 8306_{16}	The *Arm Command* 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** 0306_{16} <state> **Query Syntax** 8306_{16} → <state> **Parameters** <state> Type: 16-bit unsigned integer 0000_{16} Disarmed state 0001_{16} Armed state

Name	Description
Arm Polarity Command 0308_{16} **Arm Polarity Query** 8308_{16}	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** 0308_{16} <polarity> **Query Syntax** 8308_{16} → <polarity> **Parameters** <polarity> Type: 16-bit unsigned integer 0000_{16} Negative Polarity 0001_{16} Positive Polarity

Name	Description
Arm Source Command 0307_{16} **Arm Source Query** 8307_{16}	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 0009_{16}, the Star Trigger will be used to arm the unit. If an immediate output is desired regardless of trigger, *Arm Source Command* can be set to $000E_{16}$. **Command Syntax** 0307_{16} <source> **Query Syntax** 8307_{16} → <source> **Parameters** <source> Type: 16-bit unsigned integer 0000_{16} Software 0001_{16} TTL Trigger0 0002_{16} TTL Trigger1 0003_{16} TTL Trigger2 0004_{16} TTL Trigger3 0005_{16} TTL Trigger4 0006_{16} TTL Trigger5 0007_{16} TTL Trigger6 0008_{16} TTL Trigger7 0009_{16} Star Trigger $000A_{16}$ Channel 1 $000B_{16}$ Channel 2 $000C_{16}$ External Trigger $000D_{16}$ Pattern Trigger $000E_{16}$ Arm Immediate
Auto Scale Command 0400_{16}	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** 0400_{16} **Query Syntax** None **Parameters** None

Name	Description
Average Count Command 0422_{16} **Average Count Query** 8422_{16}	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** 0422_{16} <count> **Query Syntax** 8422_{16} → <count> **Parameters:** <count> Type: 16-bit unsigned integer Range: 2 to 65535

Name	Description
Average Envelope View Command 0423_{16} **Average Envelope View Query** 8423_{16}	Sets or queries the active envelope view; that is, which envelope view to set active. The default view is Minimum. **Command Syntax** 0423_{16} <view> **Query Syntax** 8423_{16} → <view> **Parameters:** <view> Type: 16-bit unsigned integer 0000_{16} Minimum 0001_{16} Maximum

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Average Equivalent Time Points Command
0424_{16}

Average Equivalent Time Points Query
8424_{16}</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 (Input Channel 1 or Input Channel 2).
Command Syntax
0424_{16} <points>

Query Syntax
8424_{16} → <points>

Parameters:
<points>
Type: 16-bit unsigned integer

Range: 2 to 100</td></tr>
</table>

Name	Description
Average State Command 0420_{16} **Average State Query** 8420_{16}	Sets or queries the waveform averaging state. The following considerations apply: • When the average is enabled, only the final output waveform is retained, the raw, un-averaged data is not available. • When the average 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** 0420_{16} <state> **Query Syntax** 8420_{16} → <state> **Parameters:** <state> Type: 16-bit unsigned integer 0000_{16} Disable averaging 0001_{16} Enable averaging

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>

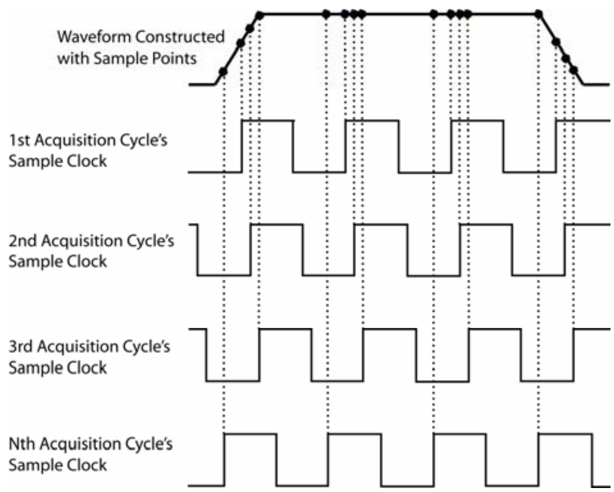
Average Type Command
0421_{16}

Average Type Query
8421_{16}

</td><td>

Sets or queries the type of averaging that is to take place.

- In Scalar mode, multiple captured waveforms are averaged together, providing higher resolution and less noise.
- 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 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 using the *Average Equivalent Time Points Command* to increase the resolution of the waveform.



Non-captured points have an initial start value of $XXX1_{16}$. Captured points have an initial value of $XXX0_{16}$.

Command Syntax
0421_{16} <type>

Query Syntax
8421_{16} → <type>

Parameters:
<type>
Type: 16-bit unsigned integer

0001_{16} Scalar (Average)
0002_{16} Envelope
0003_{16} Equivalent Time

</td></tr>
</table>

Name	Description
Calculate Absolute Value Command 0811$_{16}$	Sets the calculate channel <calc_channel> to determine the absolute value of a waveform and place the result in its output, point by point. **Command Syntax** 0811$_{16}$ <calc_channel> <source> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002$_{16}$ Calculation Channel 1 0003$_{16}$ Calculation Channel 2 <source> Type: 16-bit unsigned integer 0000$_{16}$ Input Channel 1 0001$_{16}$ Input Channel 2 0002$_{16}$ Calculation Channel 1 0003$_{16}$ Calculation Channel 2 0004$_{16}$ Reference Channel 1 0005$_{16}$ Reference Channel 2 0006$_{16}$ Reference Channel 3 0007$_{16}$ Reference Channel 4

Name	Description
Calculate Add Command 0810_{16}	Sets the calculate channel <calc_channel> to add two waveforms and place the result in its output. **Command Syntax** 0810_{16} <calc_channel> <source1> <source2> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <source1> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <source2> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4

Name	Description
Calculate Copy Command 0812_{16}	Sets the calculate channel <calc_channel> to copy a source waveform into its output. **Command Syntax** 0812_{16} <calc_channel> <source> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4

Name	Description
Calculate Data Format Command 0805_{16} **Calculate Data Format Query** 8805_{16}	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** 0805_{16} <format> **Query Syntax** 8805_{16} → <format> **Parameters:** <format> Type: 16-bit unsigned integer 0000_{16} Linear format 0001_{16} Logarithmic format

Name	Description
Calculate Derivative Command 0813_{16}	Sets the calculate channel <calc_channel> to calculate the derivative of a source waveform. The derivative is calculated using a discrete-time difference equation. **Command Syntax** 0813_{16} <calc_channel> <source> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4

Name	Description
Calculate Function Query 8803_{16}	Queries the current calculate channel <calc_channel> function. **Command Syntax** None **Query Syntax** 8803_{16} <calc_channel> → <function> <source1> <source2> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <function> Type: 16-bit unsigned integer 0000_{16} Add 0001_{16} Absolute Value 0002_{16} Copy 0003_{16} Derivative 0004_{16} Integral 0005_{16} Invert 0006_{16} Multiply 0007_{16} Subtract 0008_{16} Limit 0009_{16} FFT $000A_{16}$ Time Transform <source1> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <source2> **Note:** Queries the source2. This is always returned by a query, but has no meaning when using only one source. Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4

Name	Description
Calculate Immediate Command 0804_{16}	Forces the instrument to immediately perform calculations on the calculation channel. **Command Syntax** 0804_{16} <calc_channel> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2
Calculate Integral Command 0814_{16}	Sets the calculate channel <calc_channel> to calculate the integral of a source waveform and place the result into its output. **Command Syntax** 0814_{16} <calc_channel> <source> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4

Name	Description
Calculate Invert Command 0815_{16}	Sets the calculate channel <calc_channel> to invert the sign of a source waveform and place the result into its output. **Command Syntax** 0815_{16} <calc_channel> <source> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4
Calculate Limit Test Clear Command 0824_{16}	Clears the statistics from a limit or mask test for the specified channel. **Command Syntax** 0824_{16} <calc_channel> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2

Name	Description
Calculate Limit Test Command 0820_{16}	Configures the instrument to perform a limit test. **Command Syntax** 0820_{16} <calc_channel> <source> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4
Calculate Limit Test Continuous Command 0825_{16} **Calculate Limit Test Continuous Query** 8825_{16}	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** 0825_{16} <calc_channel> <state> **Query Syntax** 8825_{16} <calc_channel> $\rightarrow$ <state> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <state> Type: 16-bit unsigned integer 0000_{16} Off 0001_{16} On

Name	Description
Calculate Limit Test Fail Query 8822_{16}	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** 8822_{16} <calc_channel> → <fail_num> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <fail_num> Type: 16-bit unsigned integer Range: 0 or 1
Calculate Limit Test Lower Command 0827_{16} **Calculate Limit Test Lower Query** 8827_{16}	Sets or returns the lower limit for a limit test. **Command Syntax** 0827_{16} <calc_channel> <lower_limit> **Query Syntax** 8827_{16} <calc_channel> → <lower_limit> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <lower_limit> Type: 32-bit floating point Range: Variable
Calculate Limit Test Measurement Command 0821_{16} **Calculate Limit Test Measurement Query** 8821_{16}	Set or queries the measurement to use for the limit test. **Command Syntax** 0821_{16} <calc_channel> <measurement> **Query Syntax** 8821_{16} <calc_channel> → <measurement> **Parameters:** <calc_channel> Type: 16-bit unsigned integer

Name	Description
	0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <measurement> Type: 16-bit unsigned integer 0000_{16} AC RMS 0001_{16} Amplitude 0002_{16} Average 0003_{16} DC RMS 0004_{16} Fall Overshoot 0005_{16} Fall Preshoot 0006_{16} Fall Crossing Time 0007_{16} Fall Time 0008_{16} Frequency 0009_{16} High $000A_{16}$ Low $000B_{16}$ Maximum $000C_{16}$ Minimum $000D_{16}$ Middle $000E_{16}$ Negative Duty $000F_{16}$ Negative Width 0010_{16} Positive Duty 0011_{16} Positive Width 0012_{16} Period 0013_{16} Phase 0014_{16} Peak-To-Peak 0015_{16} Rise Overshoot 0016_{16} Rise Preshoot 0017_{16} Rise Crossing Time 0018_{16} Rise Time 0019_{16} Time of Maximum $001A_{16}$ Time of Minimum $001B_{16}$ Cycle Average $001C_{16}$ Cycle Frequency $001D_{16}$ Cycle Period $001E_{16}$ Cycle RMS $001F_{16}$ AC High-Precision 0020_{16} DC High-Precision 0021_{16} Signal-to-Noise Ratio 0022_{16} Total Harmonic Distortion 0023_{16} Signal-to-Noise and Distortion 0024_{16} Effective Number of Bits 0025_{16} Spurious-Free Dynamic Range $00FF_{16}$ Mask

Name	Description
Calculate Limit Test Report Query 8823_{16}	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** 8823_{16} <calc_channel> → <count> <fail_num> <min> <max> <average> <last_val> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <count> Type: 16-bit unsigned integer Range: 0 to 65535 <fail> Type: 16-bit unsigned integer Range: 0 to 65535 <min> Type: 32-bit Floating Point <max> Type: 32-bit Floating Point <average> Type: 32-bit Floating Point <last_val> Type: 32-bit Floating Point

Name	Description
Calculate Limit Test Upper Command 0826_{16} **Calculate Limit Test Upper Query** 8826_{16}	Sets or returns the upper limit for a limit test. **Command Syntax** 0826_{16} <calc_channel> <upper_limit> **Query Syntax** 8826_{16} <calc_channel> → <upper_limit> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <upper_limit> Type: 32-bit floating point Range: Variable
Calculate Mask Test Lower Command 0829_{16} **Calculate Mask Test Lower Query** 8829_{16}	Sets or returns the source to use for the lower reference for a mask test. **Note:** Do not perform a mask test where the calculation source and destination use the same calculation channel. **Command Syntax** 0829_{16} <calc_channel> <lower_reference> **Query Syntax** 8829_{16} <calc_channel> → <lower_reference> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <lower_reference> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4

Name	Description
Calculate Mask Test Upper Command 0828_{16} **Calculate Mask Test Upper Query** 8828_{16}	Sets or returns the source to use for the upper reference for a mask test. **Note:** Do not perform a mask test where the calculation source and destination use the same calculation channel. **Command Syntax** 0828_{16} <calc_channel> <upper_reference> **Query Syntax** 8828_{16} <calc_channel> $\rightarrow$ <upper_reference> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <upper_reference> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4

Name	Description
Calculate Multiply Command 0816_{16}	Sets the Calculation Channel to multiply two waveforms and place the result in its output. **Command Syntax** 0816_{16} <calc_channel> <source1> <source2> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <source1> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <source2> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4

Name	Description
Calculate State Command 0800_{16} **Calculate State Query** 8800_{16}	Sets or queries the Calculate Channel <calc_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*. **Command Syntax** 0800_{16} <calc_channel> <state> **Query Syntax** 8800_{16} <calc_channel> → <state> **Parameters:** <calc_channel> This is where the result of the calculation is stored. Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <state> Type: 16-bit unsigned integer 0000_{16} Inactive 0001_{16} Active

Name	Description
Calculate Subtract Command 0817_{16}	Sets the Calculation Channel to subtract two waveforms and place the result in its output. The result is generated by source1 – source2. **Command Syntax** 0817_{16} <calc_channel> <source1> <source2> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <source1> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <source2> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4

Name	Description
Calculate Transform Frequency Command 0830_{16}	Sets the instrument to calculate an FFT. **Command Syntax** 0830_{16} <calc_channel> <source> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4
Calculate Transform Frequency Window Command 0831_{16} **Calculate Transform Frequency Window Query** 8831_{16}	Sets or queries the type of Transform Frequency window to use when calculating an FFT. **Command Syntax** 0831_{16} <calc_channel> <window> **Query Syntax** 8831_{16} <calc_channel> → <window> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <window> Type: 16-bit unsigned integer 0000_{16} Rectangular 0001_{16} Hamming 0002_{16} Hanning 0003_{16} Blackman-Harris

Name	Description
Calculate Transform Time Command 0840_{16}	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** 0840_{16} <calc_channel> <source> <state> **Query Syntax** None **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <state> Type: 16-bit unsigned integer 0000_{16} Off 0001_{16} On

Name	Description
Calculate Transform Time Points Command 0841_{16} **Calculate Transform Time Points Query** 8841_{16}	Sets or queries the number of filter length points used to calculate a Time Transform. **Command Syntax** 0841_{16} <calc_channel> <points> **Query Syntax** 8841_{16} <calc_channel> → <points> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <points> Type: 16-bit unsigned integer Range: 2 to 40 points
Calculate Voltage Offset Command 0802_{16} **Calculate Voltage Offset Query** 8802_{16}	Sets or queries the DC voltage offset for the calculate channel (calc_channel). When a calculate channel is configured, the unit derives a DC offset voltage that is represented at the vertical center for the selected channel. **Command Syntax** 0802_{16} <calc_channel> <offset_volts> **Query Syntax** 8802_{16} <calc_channel> → <offset_volts> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <offset_volts> Type: 32-bit floating point

Name	Description
Calculate Voltage Range Command 0801_{16} **Calculate Voltage Range Query** 8801_{16}	Sets or queries the expected voltage range for the calculate channel (calc_channel). **Command Syntax** 0801_{16} <calc_channel> <range> **Query Syntax** 8801_{16} <calc_channel> $\rightarrow$ <range> **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <range> Type: 32-bit floating point
Calibrate Query 8600_{16}	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 both input channels. The internal calibration process can take several minutes to complete. The instrument is reset upon completion of the calibration process. **Note:** The two 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** 8600_{16} $\rightarrow$ <result> **Parameters:** <result> Type: 16-bit signed integer 0 = Pass 1 = Fail (Did not converge) 2 = Corrupt

Name	Description
Calibrate Restore Factory Defaults Command 0610_{16}	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** 0610_{16} **Query Syntax** None **Parameters:** None
Clear Status Command $0A00_{16}$	The clear status command clears all event registers, the request for OPC flag, and all status queues (except the response queue). **Command Syntax** $0A00_{16}$ **Query Syntax** None **Parameters** None

Name	Description

Clock Frequency Command
0405_{16}

Clock Frequency Query
8405_{16}

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	1 Channel Maximum Sample Rate	2 Channel Maximum Sample Rate
ZT410-2X	500 MS/s	250 MS/s
ZT410-5X	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.
- When using an external sample clock, the number of enabled channels affects the sample rate. When one channel is enabled, the sampling occurs at the applied external frequency. When both channels are enabled, the sampling occurs at one-half the applied external frequency. Set the channel enable configuration before selecting external sample clock.

Command Syntax
0405_{16} <frequency>

Query Syntax
8405_{16} $\rightarrow$ <frequency>

Parameters
<frequency>
Type: 32-bit floating point

Range: Internal Clock Source
10 kS/s to 2-Channel Maximum Sample Rate
in 1, 2.5, 4 and 5 steps
500 MS/s, 1 channel interleaved (ZT410-2X)
400 MS/s, 1 channel interleaved (ZT410-5X)

Range: External Clock Source
40 MS/s to 1-Channel Maximum Sample Rate
1 channel enabled: sample at external frequency
2 channels enabled: sample at half external frequency

Name	Description
Clock Source Command 0404_{16} **Clock Source Query** 8404_{16}	Sets or queries the selected sample clock source. Internal generator or external input sample clock sources are supported. The following considerations apply when using an external sample clock: • An external clock source must be present before setting to external clock. • 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. • When using an external sample clock, the number of enabled channels affects the sample rate. When one channel is enabled, the sampling occurs at the applied external frequency. When both channels are enabled, the sampling occurs at one-half the applied external frequency. Set the channel enable configuration before selecting external sample clock. **Command Syntax** 0404_{16} <source> **Query Syntax** 8404_{16} → <source> **Parameters** <source> Type: 16-bit unsigned integer 0000_{16} Internal sample clock 0001_{16} External sample clock

Name	Description
Event Status Enable Command $0A03_{16}$ **Event Status Enable Query** $8A03_{16}$	The command allows users to Enable or Disable bits in the Standard Event Status Register. The query returns the bit mask of the standard event status register. That is, it returns a bit mask which indicates those standard event status register bits that are enabled. The command parameter is a bit mask which enables the corresponding standard event status register bits. **Command Syntax** $0A03_{16}$ <enable> **Query Syntax** $8A03_{16}$ → <enable> **Parameters:** <enable> Type: 16-bit unsigned integer Range: 0 to 255 0001_{16} Operation Complete Bit 0002_{16} Request Control Bit 0004_{16} Query Error Bit 0008_{16} Device Dependent Error Bit 0010_{16} Execution Error Bit 0020_{16} Command Error Bit 0040_{16} User Request Bit 0080_{16} Power On Bit
Event Status Query $8A02_{16}$	Returns the status bits for the Standard Event Status Register. **Command Syntax** None **Query Syntax** $8A02_{16}$ → <status> **Parameters** <status> Type: 16-bit unsigned integer Range: 0 to 255 0001_{16} Operation Complete Bit 0002_{16} Request Control Bit 0004_{16} Query Error Bit 0008_{16} Device Dependent Error Bit 0010_{16} Execution Error Bit 0020_{16} Command Error Bit 0040_{16} User Request Bit 0080_{16} Power On Bit

Name	Description
External Trigger Polarity Command $020F_{16}$ **External Trigger Polarity Query** $820F_{16}$	Sets the polarity of the external trigger output when the output is enabled. When positive output polarity is selected, the output driver forces a low 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 external trigger output will be a low TTL logic level when the unit is armed. **Command Syntax** $020F_{16}$ <polarity> **Query Syntax** $820F_{16}$ → <polarity> **Parameters** <polarity> Type: 16-bit unsigned integer 0000_{16} Negative polarity 0001_{16} Positive polarity
External Trigger Source Command $020E_{16}$ **External Trigger Source Query** $820E_{16}$	Selects the source for the external trigger output when enabled. **Command Syntax** $020E_{16}$ <source> **Query Syntax** $820E_{16}$ → <source> **Parameters** <source> Type: 16-bit unsigned integer 0000_{16} Arm Event 0001_{16} Trigger Event 0002_{16} Constant State 0003_{16} Operation Complete 0004_{16} 10 MHz Timebase Reference Clock 0005_{16} 500 Hz Clock 0006_{16} 10 ns Pulse at 1 ms Repetition Interval

Name	Description
External Trigger State Command $020D_{16}$ **External Trigger State Query** $820D_{16}$	Allows the user to enable or disable the external trigger output. **Command Syntax** $020D_{16}$ <state> **Query Syntax** $820D_{16}$ → <state> **Parameters** <state> Type: 16-bit unsigned integer 0000_{16} Inactive state 0001_{16} Active state
Format Byte Command 0010_{16} **Format Byte Query** 8010_{16}	Sets and returns the current byte order setting. Normal byte order is MSB first, Swapped is LSB first. **Command Syntax** 0010_{16} <order> **Query Syntax** 8010_{16} → <order> **Parameters:** <order> Type: 16-bit unsigned integer 0000_{16} Normal byte order (MSB first) 0001_{16} Swapped byte order (LSB first)
ID Query 8103_{16}	Returns the instrument identification including manufacturer, model number, serial number and firmware version as a block of ASCII string data up to 44 characters in length. The string will be of this form: "ZTEC,ZT410PXI-nn,S/N nnn,Version n.nn". **Command Syntax** None **Query Syntax** 8103_{16} → <char_n> **Parameters** <char_n> Type: character string Example: Returns a string of characters in the form: "ZTEC,ZT410PXI-00,S/N 100,Version 1.00"

Name	Description
Initiate Command 0304_{16} **Initiate Query** 8304_{16}	Initiates the instrument. While initiated, the instrument is enabled to acquire waveforms and perform calculations and measurements. **Command Syntax** 0304_{16} <state> **Query Syntax** 8304_{16} → <state> **Parameters:** <state> Type: 16-bit unsigned integer 0000_{16} Inactive 0001_{16} Active
Initiate Continuous Command 0309_{16} **Initiate Continuous Query** 8309_{16}	Sets or returns the instrument initiate continuous state. This is usually only used for limit and mask tests. **Command Syntax** 0309_{16} <state> **Query Syntax** 8309_{16} → <state> **Parameters:** <state> Type: 16-bit unsigned integer 0000_{16} Initiate Continuous OFF 0001_{16} Initiate Continuous ON

Name	Description
Input Attenuation Command 0411_{16} **Input Attenuation Query** 8411_{16}	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** 0411_{16} <source> <attenuation> **Query Syntax** 8411_{16} <source> → <attenuation> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 <attenuation> Type: 32-bit floating point Range: 0.9 to 1000.0

Name	Description
Input Coupling Command 0412_{16} **Input Coupling Query** 8412_{16}	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** 0412_{16} <input> <coupling> **Query Syntax** 8412_{16} <input> → <coupling> **Parameters:** <input> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 <coupling> Type: 16-bit unsigned integer 0000_{16} AC 0001_{16} DC

Name	Description
Input Impedance Command 0413_{16} **Input Impedance Query** 8413_{16}	Sets or queries the input impedance in ohms. **Note:** When setting the input impedance, set a temporary level 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** 0413_{16} <input> <ohms> **Query Syntax** 8413_{16} <input> → <ohms> **Parameters:** <input> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 <ohms> Type: 32-bit floating point 50 Low Impedance 1e6 High Impedance
Input State Command 0410_{16} **Input State Query** 8410_{16}	Enables or disables an input channel capture. **Command Syntax** 0410_{16} <input> <input_state> **Query Syntax** 8410_{16} <input> → <input_state> **Parameters:** <input> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 <input_state> Type: 16-bit unsigned integer 0000_{16} Off 0001_{16} On

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr>
<td>Input Voltage Offset Command
040C$_{16}$

Input Voltage Offset Query
840C$_{16}$</td>
<td>Sets or queries the specified input channel voltage offset.

Note: The limits upon the input voltage offset setting are dependent upon the input voltage range.

Command Syntax
040C$_{16}$ <input> <volts>

Query Syntax
840C$_{16}$ <input> → <volts>

Parameters:
<input>
Type: 16-bit unsigned integer

0000$_{16}$ Input Channel 1
0001$_{16}$ Input Channel 2

<volts>
Type: 32-bit floating point

<table>
<tr><th>Impedance</th><th>Range</th><th>Offset</th></tr>
<tr><td rowspan="8">1 MΩ</td><td>50 Vpp</td><td>0V</td></tr>
<tr><td>25 Vpp</td><td>±12.5V</td></tr>
<tr><td>10 Vpp</td><td>±5V</td></tr>
<tr><td>5 Vpp</td><td>±5V</td></tr>
<tr><td>2.5 Vpp</td><td>±5V</td></tr>
<tr><td>1.25 Vpp</td><td>±5V</td></tr>
<tr><td>0.5 Vpp</td><td>±5V</td></tr>
<tr><td>0.25 Vpp</td><td>±5V</td></tr>
<tr><td rowspan="8">50Ω</td><td>10 Vpp</td><td>0V</td></tr>
<tr><td>5 Vpp</td><td>±2.5V</td></tr>
<tr><td>2 Vpp</td><td>±1V</td></tr>
<tr><td>1 Vpp</td><td>±1V</td></tr>
<tr><td>0.5 Vpp</td><td>±1V</td></tr>
<tr><td>0.25 Vpp</td><td>±1V</td></tr>
<tr><td>0.1 Vpp</td><td>±1V</td></tr>
<tr><td>0.05 Vpp</td><td>±1V</td></tr>
</table>
</td>
</tr>
</table>

Name	Description
Input Voltage Protection State Command $040B_{16}$ **Input Voltage Protection State Query** $840B_{16}$	Sets or queries the specified input channel voltage protection state. **Command Syntax** $040B_{16}$ <input> <state> **Query Syntax** $840B_{16}$ <input> → <state> **Parameters:** <input> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 <state> Type: 16-bit unsigned integer 0000_{16} Off 0001_{16} On

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>

Input Voltage Range Command
0407_{16}

Input Voltage Range Query
8407_{16}

</td><td>

Sets or queries the specified input channel voltage range.

Note: The limits upon the input voltage offset setting are dependent upon the input voltage range. When setting the input voltage range, set a temporary offset level of 0.0V to preclude setting an incompatible range and offset.

Command Syntax
0407_{16} <input> <volts>

Query Syntax
8407_{16} <input> → <volts>

Parameters:
<input>
Type: 16-bit unsigned integer

0000_{16} Input Channel 1
0001_{16} Input Channel 2

<volts>
Type: 32-bit floating point

<table>
<tr><th>Impedance</th><th>Range</th><th>Offset</th></tr>
<tr><td>1 MΩ</td><td>50 Vpp</td><td>0V</td></tr>
<tr><td></td><td>25 Vpp</td><td>±12.5V</td></tr>
<tr><td></td><td>10 Vpp</td><td>±5V</td></tr>
<tr><td></td><td>5 Vpp</td><td>±5V</td></tr>
<tr><td></td><td>2.5 Vpp</td><td>±5V</td></tr>
<tr><td></td><td>1.25 Vpp</td><td>±5V</td></tr>
<tr><td></td><td>0.5 Vpp</td><td>±5V</td></tr>
<tr><td></td><td>0.25 Vpp</td><td>±5V</td></tr>
<tr><td>50Ω</td><td>10 Vpp</td><td>0V</td></tr>
<tr><td></td><td>5 Vpp</td><td>±2.5V</td></tr>
<tr><td></td><td>2 Vpp</td><td>±1V</td></tr>
<tr><td></td><td>1 Vpp</td><td>±1V</td></tr>
<tr><td></td><td>0.5 Vpp</td><td>±1V</td></tr>
<tr><td></td><td>0.25 Vpp</td><td>±1V</td></tr>
<tr><td></td><td>0.1 Vpp</td><td>±1V</td></tr>
<tr><td></td><td>0.05 Vpp</td><td>±1V</td></tr>
</table>

</td></tr>
</table>

Name	Description
Measure AC High-Precision Query $872F_{16}$	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 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** $872F_{16}$ <source> → <ac_volts> **Parameters** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <ac_volts> Type: 32-bit floating point

Name	Description
Measure AC RMS Query 8710_{16}	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 peak value, or 0.354 times the peak-to-peak value. **Command Syntax** None **Query Syntax** 8710_{16} <source> → <ac_volts> **Parameters** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <ac_volts> Type: 32-bit floating point

Name	Description
Measure Amplitude Query 8711_{16}	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** 8711_{16} <source> → <amplitude_volts> **Parameters** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <amplitude_volts> Type: 32-bit floating point

Name	Description
Measure Average Voltage Query 8712_{16}	Queries the average voltage level of the selected waveform source. The average voltage is defined as the sum of all the sample voltage values in the waveform record divided by the number of samples. **Command Syntax** None **Query Syntax** 8712_{16} <source> → <average_volts> **Parameters** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <average_volts> Type: 32-bit floating point
Measure Cycle Average Query $872C_{16}$	Queries the average level of one cycle of the selected waveform source. This is always the first cycle seen. **Command Syntax** None **Query Syntax** $872C_{16}$ <source> → <average> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <average> Type: 32-bit floating point

Name	Description
Measure Cycle Frequency Query $872D_{16}$	Queries the frequency for one cycle of the waveform. This is always the first cycle seen. **Command Syntax** None **Query Syntax** $872D_{16}$ <source> → <frequency> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <frequency> Type: 32-bit floating point
Measure Cycle Period Query $872E_{16}$	Queries the period for one cycle of the waveform. This is always the first cycle seen. **Command Syntax** None **Query Syntax** $872E_{16}$ <source> → <period> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <period> Type: 32-bit floating point

Name	Description
Measure Cycle RMS Query $872B_{16}$	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** $872B_{16}$ <source> → <volts_rms> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <volts_rms> Type: 32-bit floating point

Name	Description
Measure DC High-Precision Query 8730_{16}	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** 8730_{16} <source> → <dc_volts> **Parameters** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <dc_volts> Type: 32-bit floating point
Measure DC RMS Query 8713_{16}	Queries the DC RMS level of the selected waveform source. **Command Syntax** None **Query Syntax** 8713_{16} <source> → <dc_volts> **Parameters** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <dc_volts> Type: 32-bit floating point

Name	Description
Measure Edge Command 0700_{16} **Measure Edge Query** 8700_{16}	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** 0700_{16} <edge> **Query Syntax** 8700_{16} $\rightarrow$ <edge> **Parameters:** <edge> Type: 16-bit unsigned integer Range: 1 to 65535

Name	Description
Measure Effective Number of Bits Query 8734_{16}	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** 8734_{16} <source> → <bits> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <bits> Type: 32-bit floating point

Name	Description
Measure Fall Crossing Time Query 8716_{16}	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** 8716_{16} <source> → <fall_crossing_time> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <fall_crossing_time> Type: 32-bit floating point

Name	**Description**
Measure Fall Overshoot Time Query 8714_{16}	Queries the voltage overshoot of the selected waveform. The following considerations apply when using the *Measure Fall Overshoot Query*: • Fall 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. • Fall overshoot is calculated as the (signal minimum voltage – low level voltage)/signal amplitude. • The value returns as a percent in a decimal format. For example, a 10 percent fall overshoot will be returned as 0.1. **Command Syntax** None **Query Syntax** 8714_{16} <source> → <overshoot> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <overshoot> Type: 32-bit floating point Range: 0.0 (0 percent) to 1.0 (100 percent)

Name	Description
Measure Fall Preshoot Time Query 8715_{16}	Queries the voltage preshoot of the selected waveform. The following considerations apply when using the *Measure Fall Preshoot Query*: • Fall 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. • Fall preshoot is calculated as the (signal maximum voltage – high level voltage)/signal amplitude. • The value returns as a percent in a decimal format. For example, a 10 percent fall preshoot will be returned as 0.1. **Command Syntax** None **Query Syntax** 8715_{16} <source> → <preshoot> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <preshoot> Type: 32-bit floating point Range: 0.0 (0 percent) to 1.0 (100 percent)

Name	Description
Measure Fall Time Query 8717_{16}	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 the measured fall time in seconds. **Command Syntax** None **Query Syntax** 8717_{16} <source> → <fall_time> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <fall_time> Type: 32-bit floating point

Name	**Description**
Measure Frequency Query 8718_{16}	Queries the frequency of the selected waveform source. All cycles in the entire capture window are used. **Command Syntax** None **Query Syntax** 8718_{16} <source> → <frequency> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <frequency> Type: 32-bit floating point

Name	Description
Measure Gate Points Command 0704_{16} **Measure Gate Points Query** 8704_{16}	Sets or queries the measurement of the selected waveform source using gate start and stop points. **Command Syntax** 0704_{16} <source> <gate_start> <gate_stop> **Query Syntax** 8704_{16} <source> → <gate_start> <gate_stop> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <gate_start> Type: 32-bit unsigned integer Range: 0 to maximum waveform size <gate_stop> Type: 32-bit unsigned integer Range: 0 to maximum waveform size

Name	Description
Measure Gate Time Command 0703_{16} **Measure Gate Time Query** 8703_{16}	Sets or queries the measurement of the selected waveform source using gate start and stop times relative to the trigger time. **Command Syntax** 0703_{16} <source> <start> <stop> **Query Syntax** 8703_{16} <source> → <start> <stop> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <start> Type: 32-bit Floating Point <stop> Type: 32-bit Floating Point

Name	Description
Measure High Voltage Query 8719_{16}	Queries the high level of the selected waveform source. The high 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** 8719_{16} <source> → <voltage_high> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <voltage_high> Type: 32-bit floating point

Name	Description
Measure Low Voltage Query $871A_{16}$	Queries the low level of the selected waveform source. The low 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** $871A_{16}$ <source> → <voltage_low> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <voltage_low> Type: 32-bit floating point

Name	Description
Measure Maximum Voltage Query 871B$_{16}$	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** 871B$_{16}$ <source> → <voltage_max> **Parameters:** <source> Type: 16-bit unsigned integer 0000$_{16}$ Input Channel 1 0001$_{16}$ Input Channel 2 0002$_{16}$ Calculation Channel 1 0003$_{16}$ Calculation Channel 2 0004$_{16}$ Reference Channel 1 0005$_{16}$ Reference Channel 2 0006$_{16}$ Reference Channel 3 0007$_{16}$ Reference Channel 4 <voltage_max> Type: 32-bit floating point

Name	Description
Measure Method Command 0701_{16} **Measure Method Query** 8701_{16}	Sets or queries the method to use for measurements. The following method types are available on the ZT410: Entire Waveform Entire Waveform measurement is used to perform measurements upon the entire captured waveform. Gated by Time 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. Gated by Points 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** 0701_{16} <method> **Query Syntax** 8701_{16} → <method> **Parameters:** <method> Type: 16-bit unsigned integer 0000_{16} Entire Waveform 0001_{16} Gated

Name	Description
Measure Mid Voltage Query 871D$_{16}$	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. $$\textbf{Mid Level} = \frac{\textbf{High Level + Low Level}}{\textbf{2}}$$ 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** 871D$_{16}$ <source> → <voltage_mid> **Parameters:** <source> Type: 16-bit unsigned integer 0000$_{16}$ Input Channel 1 0001$_{16}$ Input Channel 2 0002$_{16}$ Calculation Channel 1 0003$_{16}$ Calculation Channel 2 0004$_{16}$ Reference Channel 1 0005$_{16}$ Reference Channel 2 0006$_{16}$ Reference Channel 3 0007$_{16}$ Reference Channel 4 <voltage_mid> Type: 32-bit floating point

Name	Description
Measure Minimum Voltage Query $871C_{16}$	Queries the most negative voltage of the selected waveform source. The minimum level measurement is defined as the most negative voltage level of the waveform. Use the *Measure Low Voltage Query* to detect the low level in a bi-level signal such as a square wave. **Command Syntax** None **Query Syntax** $871C_{16}$ <source> → <voltage_min> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <voltage_min> Type: 32-bit floating point
Measure Negative Duty Cycle Query $871E_{16}$	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** $871E_{16}$ <source> → <negative_duty> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <negative_duty> Type: 32-bit floating point

Name	Description
Measure Negative Width Query $871F_{16}$	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** $871F_{16}$ <source> → <negative_width> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <negative_width> Type: 32-bit floating point

Name	Description
Measure Peak-To-Peak Voltage Query 8724_{16}	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** 8724_{16} <source> → <voltage_ptp> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <voltage_ptp> Type: 32-bit floating point

Name	Description
Measure Period Query 8722_{16}	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** 8722_{16} <source> → <period> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <period> Type: 32-bit floating point

Name	Description
Measure Phase Query 8723_{16}	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** 8723_{16} <source> → <phase> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <phase> Type: 32-bit floating point Range: 0 to 2 PI (6.283185307)

Name	Description
Measure Positive Duty Cycle Query 8720_{16}	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** $871E_{16}$ <source> → <positive_duty> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <positive_duty> Type: 32-bit floating point
Measure Positive Width Query 8721_{16}	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** 8721_{16} <source> → <positive_width> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <positive_width> Type: 32-bit floating point

Name	Description
Measure Reference Command 0708_{16} **Measure Reference Query** 8708_{16}	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 either by 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** 0708_{16} <low_reference> <mid_reference> <high_reference> **Query Syntax** 8708_{16} → <low_reference> <mid_reference> <high_reference> **Parameters:** <low_reference> Type: 32-bit floating point <mid_reference> Type: 32-bit floating point <high_reference> Type: 32-bit floating point

Name	Description
Measure Reference Method Command 0709_{16} **Measure Reference Method Query** 8709_{16}	Sets or queries the reference method used in waveform voltage analysis. Reference methods are in absolute voltages or relative percentages. **Command Syntax** 0709_{16} <reference_method> **Query Syntax** 8709_{16} → <reference_method> **Parameters:** <reference_method> Type: 16-bit unsigned integer 0000_{16} Voltages 0001_{16} Percentages
Measure Rise Crossing Time Query 8727_{16}	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** 8727_{16} <source> → <rise_crossing_time> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <rise_crossing_time> Type: 32-bit floating point

Name	Description
Measure Rise Overshoot Query 8725_{16}	Queries the voltage overshoot of the selected waveform. The following considerations apply when using the *Measure Rise Overshoot Query*: • Rise 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. • Rise overshoot is calculated as the signal (maximum voltage – high level voltage)/signal amplitude. • The value returns as a percent in a decimal format. For example, a 10 percent overshoot will be returned as 0.1. **Command Syntax** None **Query Syntax** 8725_{16} <source> → <overshoot> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <overshoot> Type: 32-bit floating point Range: 0.0 (0 percent) to 1.0 (100 percent)

Name	Description
Measure Rise Preshoot Query 8726_{16}	Queries the voltage preshoot of the selected waveform. The following considerations apply when using the *Measure Rise Preshoot Query*: • Rise 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. • Rise preshoot is calculated as the signal (minimum voltage – low level voltage)/signal amplitude. • The value returns as a percent in a decimal format. For example, a 10 percent overshoot will be returned as 0.1. **Command Syntax** None **Query Syntax** 8726_{16} <source> → <preshoot> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <preshoot> Type: 32-bit floating point Range: 0.0 (0 percent) to 1.0 (100 percent)

Name	Description
Measure Rise Time Query 8728_{16}	Performs a rise time measurement one time on the signal present, and then sends the measurement results to the output buffer. I 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 the measured rise time in seconds. **Command Syntax** None **Query Syntax** 8728_{16} <source> → <rise_time> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <rise_time> Type: 32-bit floating point

Name	Description
Measure Signal-to-Noise Ratio Query 8731_{16}	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** 8731_{16} <source> → <dBc> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <dBc> Type: 32-bit floating point

Name	Description
Measure Signal-to-Noise and Distortion Ratio Query 8733_{16}	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** 8733_{16} <source> → <dBc> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <dBc> Type: 32-bit floating point

Name	Description
Measure Spurious-Free Dynamic Range Query 8735_{16}	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** 8735_{16} <source> $\rightarrow$ <dBc> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <dBc> Type: 32-bit floating point

Name	Description
Measure Time/Frequency of Maximum Voltage Query 8729_{16}	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** 8729_{16} <source> → <max_voltage_time> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <max_voltage_time> Type: 32-bit floating point

Name	Description
Measure Time/Frequency of Minimum Voltage Query $872A_{16}$	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** $872A_{16}$ <source> → <min_voltage_time> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <min_voltage_time> Type: 32-bit floating point

Name	Description
Measure Total Harmonic Distortion Query 8732_{16}	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** 8732_{16} <source> → <dBc> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <dBc> Type: 32-bit floating point

Name	Description
Operation Complete Command 0004_{16} **Operation Complete Query** 8004_{16}	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 not completed and a 1 to indicate that all pending operations have completed. **Command Syntax** 0004_{16} **Query Syntax** 8004_{16} → <state> **Parameters:** <state> Type: 16-bit unsigned integer 0000_{16} All operations are not complete 0001_{16} All operations complete

Name	Description
Output Trigger Polarity Command 020A$_{16}$ **Output Trigger Polarity Query** 820A$_{16}$	Sets or queries the output driver polarity for any of the eight TTL trigger outputs (PXI Backplane or PCI Timing Expansion Connector). The following considerations apply when using the *Output Trigger Polarity Command* and *Output Trigger Polarity Query*: • When positive output polarity is selected, the output driver forces a 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 a logic 0 when the unit is armed. • Output driver polarity does not affect TTL trigger line sensing used by other unit functions. • Each output line polarity is selected individually • TTL trigger lines can be sourced and sensed simultaneously. **Command Syntax** 020A$_{16}$ <output_trigger> <polarity> **Query Syntax** 820A$_{16}$ <output_trigger> → <polarity> **Parameters** <output_trigger> Type: 16-bit unsigned integer 0000$_{16}$ TTL trigger 0 0001$_{16}$ TTL trigger 1 0002$_{16}$ TTL trigger 2 0003$_{16}$ TTL trigger 3 0004$_{16}$ TTL trigger 4 0005$_{16}$ TTL trigger 5 0006$_{16}$ TTL trigger 6 0007$_{16}$ TTL trigger 7 <polarity> Type: 16-bit unsigned integer 0000$_{16}$ Negative polarity 0001$_{16}$ Positive polarity

Name	Description
Output Trigger Source Command 0209_{16} **Output Trigger Source Query** 8209_{16}	Sets or queries the output driver source for any of the eight TTL trigger outputs (PXI Backplane or PCI Timing Expansion Connector). The following considerations apply when using the *Output Trigger Source Command* and *Output Trigger Source Query*: • The output driver may be enabled or disabled and the polarity selected. • Each output line source is selected individually • TTL trigger lines can be sourced and sensed simultaneously. **Command Syntax** 0209_{16} <output_trigger> <source> **Query Syntax** 8209_{16} <output_trigger> → <source> **Parameters** <output_trigger> Type: 16-bit unsigned integer 0000_{16} TTL trigger 0 0001_{16} TTL trigger 1 0002_{16} TTL trigger 2 0003_{16} TTL trigger 3 0004_{16} TTL trigger 4 0005_{16} TTL trigger 5 0006_{16} TTL trigger 6 0007_{16} TTL trigger 7 <source> Type: 16-bit unsigned integer 0000_{16} Arm Event 0001_{16} Trigger Event 0002_{16} Constant State 0003_{16} Operation Complete

Name	Description
Output Trigger State Command 0208_{16} **Output Trigger State Query** 8208_{16}	Sets or queries the output driver state for any of the eight TTL trigger outputs (PXI Backplane or PCI Timing Expansion Connector). The following considerations apply when using the *Output Trigger State Command* and *Output Trigger State Query*: • The output source and polarity are selectable. • Each output line driver state is selected individually • TTL trigger lines can be sourced and sensed simultaneously. **Command Syntax** 0208_{16} <output_trigger> <state> **Query Syntax** 8208_{16} <output_trigger> → <state> **Parameters** <output_trigger> Type: 16-bit unsigned integer 0000_{16} TTL trigger 0 0001_{16} TTL trigger 1 0002_{16} TTL trigger 2 0003_{16} TTL trigger 3 0004_{16} TTL trigger 4 0005_{16} TTL trigger 5 0006_{16} TTL trigger 6 0007_{16} TTL trigger 7 <state> Type: 16-bit unsigned integer 0000_{16} Inactive state 0001_{16} Active state
Recall Instrument State Command 0007_{16}	Recalls the selected saved instrument state from non-volatile memory. **Command Syntax** 0007_{16} <number> **Query Syntax** None **Parameters:** <number> Type: 16-bit unsigned integer Range: 1 to 31

Name	Description
Reference Oscillator Source Command 0200_{16} **Reference Oscillator Source Query** 8200_{16}	Sets and queries the source for the 10 MHz reference clock that provides the instrument time base. External reference source is the PXI CLK10 reference or the reference Input on the PCI Timing Expansion connector. **Command Syntax** 0200_{16} <source> **Query Syntax** 8200_{16} $\rightarrow$ <source> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Local reference selected 0004_{16} External reference selected
Reset Command 0000_{16}	Performs a hardware reset function that returns the instrument to the initial default condition. Status registers are <u>not</u> cleared. **Command Syntax** 0000_{16} **Query Syntax** None **Parameters:** None
Save Instrument State Command 0006_{16}	Stores the current state of the instrument to the selected storage index in non-volatile memory. **Command Syntax** 0006_{16} <number> **Query Syntax** None **Parameters:** <number> Type: 16-bit unsigned integer Range: 1 to 31

Name	Description
Service Request Enable Command $0A04_{16}$ **Service Request Enable Query** $8A04_{16}$	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** $0A04_{16}$ <mask> **Query Syntax** $8A04_{16}$ → <mask> **Parameters:** <mask> Type: 16-bit unsigned integer Range: 0 to 255 0001_{16} Unused Bit 0002_{16} Unused Bit 0004_{16} Error Log Not Empty Bit 0008_{16} Questionable Summary Bit 0010_{16} Message Available Bit 0020_{16} Standard Event Summary Bit 0040_{16} Master Summary Bit 0080_{16} Operation Summary Bit
Status Byte Query 8002_{16}	Returns the Status Byte. **Command Syntax** None **Query Syntax** 8002_{16} → <mask> **Parameters:** <mask> Type: 16-bit unsigned integer Range: 0 to 255 0001_{16} Unused Bit 0002_{16} Unused Bit 0004_{16} Error Log Not Empty Bit 0008_{16} Questionable Summary Bit 0010_{16} Message Available Bit 0020_{16} Standard Event Summary Bit 0040_{16} Master Summary Bit 0080_{16} Operation Summary Bit

Name	Description
Status Calibration Condition Query 8A16$_{16}$	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** 8A16$_{16}$ → <condition> **Parameters:** <condition> Type: 16-bit unsigned integer Range: 0 to 65535 0001$_{16}$ Calibration Storage Failed 0002$_{16}$ Zero DC Offset Calibration Failed 0004$_{16}$ DC Offset Adjust Scale Factor Calibration Failed 0008$_{16}$ ADC Null Balance Calibration Failed 0010$_{16}$ ADC Gain Balance Calibration Failed 0020$_{16}$ Sample Rate Null Calibration Failed FFC0$_{16}$ Unused Bits

Name	Description
Status Calibration Enable Command 0A17$_{16}$ **Status Calibration Enable Query** 8A17$_{16}$	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* and *Status Calibration Enable 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** 0A17$_{16}$ **Query Syntax** 8A17$_{16}$ → <enable> **Parameters:** <enable> Type: 16-bit unsigned integer Range: 0 to 65535 0001$_{16}$ Calibration Storage Failed 0002$_{16}$ Zero DC Offset Calibration Failed 0004$_{16}$ DC Offset Adjust Scale Factor Calibration Failed 0008$_{16}$ ADC Null Balance Calibration Failed 0010$_{16}$ ADC Gain Balance Calibration Failed 0020$_{16}$ Sample Rate Null Calibration Failed FFC0$_{16}$ Unused Bits

Name	Description
Status Calibration Event Query $8A18_{16}$	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** $8A18_{16}$ → <event> **Parameters:** <event> Type: 16-bit unsigned integer Range: 0 to 65535 0001_{16} Calibration Storage Failed 0002_{16} Zero DC Offset Calibration Failed 0004_{16} DC Offset Adjust Scale Factor Calibration Failed 0008_{16} ADC Null Balance Calibration Failed 0010_{16} ADC Gain Balance Calibration Failed 0020_{16} Sample Rate Null Calibration Failed $FFC0_{16}$ Unused Bits

Name	Description
Status Frequency Condition Query $8A13_{16}$	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 not clear the Questionable Frequency Condition Register**.** **Command Syntax** None **Query Syntax** $8A13_{16}$ → <condition> **Parameters** <condition> Type: 16-bit unsigned integer Range: 0 to 65535 0001_{16} PLL Unlocked Bit $FFFE_{16}$ Unused Bits

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>

Status Frequency Enable Command
$0A14_{16}$

Status Frequency Enable Query
$8A14_{16}$

</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* and *Status Frequency Enable 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
$0A14_{16}$

Query Syntax
$8A14_{16}$ → <enable>

Parameters:
<enable>
Type: 16-bit unsigned integer
Range: 0 to 65535

0001_{16} PLL Unlocked Bit
$FFFE_{16}$ Unused Bits

</td></tr>
</table>

Name	Description
Status Frequency Event Query 8A15$_{16}$	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 *Status 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 must be cleared by reading it before the processes are run. **Command Syntax** None **Query Syntax** 8A15$_{16}$ → <event> **Parameters:** <event> Type: 16-bit unsigned integer Range: 0 to 65535 0001$_{16}$ PLL Unlocked Bit FFFE$_{16}$ Unused Bits

Name	Description
Status Operation Condition Query $8A05_{16}$	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** $8A05_{16}$ → <condition> **Parameters:** <condition> Type: 16-bit unsigned integer Range: 0 to 65535 0001_{16} Calibrating Bit 0002_{16} Settling Bit 0004_{16} Ranging Bit 0008_{16} Sweeping Bit 0010_{16} Measuring Bit 0020_{16} Waiting for Trigger Bit 0040_{16} Waiting for Arm Bit 0080_{16} Unused Bit 0100_{16} Trigger Event Bit 0200_{16} Data Capture Event Bit 0400_{16} Limit Test Event Bit 0800_{16} Auto-download Event Bit $F000_{16}$ Unused Bits

Name	Description
Status Operation Enable Command 0A06$_{16}$ **Status Operation Enable Query** 8A06$_{16}$	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* and *Status Operation Enable 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** 0A06$_{16}$ **Query Syntax** 8A06$_{16}$ → <enable> **Parameters:** <enable> Type: 16-bit unsigned integer Range: 0 to 65535 0001$_{16}$ Calibrating Bit 0002$_{16}$ Settling Bit 0004$_{16}$ Ranging Bit 0008$_{16}$ Sweeping Bit 0010$_{16}$ Measuring Bit 0020$_{16}$ Waiting for Trigger Bit 0040$_{16}$ Waiting for Arm Bit 0080$_{16}$ Unused Bit 0100$_{16}$ Trigger Event Bit 0200$_{16}$ Data Capture Event Bit 0400$_{16}$ Limit Test Event Bit 0800$_{16}$ Auto-download Event Bit F000$_{16}$ Unused Bits

Name	Description
Status Operation Event Query 8A07$_{16}$	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** 8A07$_{16}$ → <event> **Parameters:** <event> Type: 16-bit unsigned integer Range: 0 to 65535 0001$_{16}$ Calibrating Bit 0002$_{16}$ Settling Bit 0004$_{16}$ Ranging Bit 0008$_{16}$ Sweeping Bit 0010$_{16}$ Measuring Bit 0020$_{16}$ Waiting for Trigger Bit 0040$_{16}$ Waiting for Arm Bit 0080$_{16}$ Unused Bit 0100$_{16}$ Trigger Event Bit 0200$_{16}$ Data Capture Event Bit 0400$_{16}$ Limit Test Event Bit 0800$_{16}$ Auto-download Event Bit F000$_{16}$ Unused Bits

Name	Description
Status Preset Command 0A01$_{16}$	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** 0A01$_{16}$ **Query Syntax** None **Parameters:** None
Status Questionable Condition Query 8A08$_{16}$	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 *Status Questionable Condition 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 not clear the Questionable Status Condition Register **Command Syntax** None **Query Syntax** 8A08$_{16}$ → <condition> **Parameters:** <condition> Type: 16-bit unsigned integer Range: 0 to 65535 0001$_{16}$ Voltage Bit 000E$_{16}$ Unused Bits 0010$_{16}$ Temperature Bit 0020$_{16}$ Frequency Bit 00C0$_{16}$ Unused Bits 0100$_{16}$ Calibration Bit 0200$_{16}$ Test Bit FC00$_{16}$ Unused Bits

Name	Description
Status Questionable Enable Command $0A09_{16}$ **Status Questionable Enable Query** $8A09_{16}$	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* and *Status Questionable Enable 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** $0A09_{16}$ **Query Syntax** $8A09_{16}$ → <enable> **Parameters:** <enable> Type: 16-bit unsigned integer Range: 0 to 65535 0001_{16} Voltage Bit $000E_{16}$ Unused Bits 0010_{16} Temperature Bit 0020_{16} Frequency Bit $00C0_{16}$ Unused Bits 0100_{16} Calibration Bit 0200_{16} Test Bit $FC00_{16}$ Unused Bits

Name	Description
Status Questionable Event Query $8A0A_{16}$	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** $8A0A_{16}$ → <event> **Parameters:** <event> Type: 16-bit unsigned integer Range: 0 to 65535 0001_{16} Voltage Bit $000E_{16}$ Unused Bits 0010_{16} Temperature Bit 0020_{16} Frequency Bit $00C0_{16}$ Unused Bits 0100_{16} Calibration Bit 0200_{16} Test Bit $FC00_{16}$ Unused Bits

Name	Description
Status Test Condition Query 8A10$_{16}$	Queries the contents of the Questionable Test Condition Register. The Questionable Test Condition Register identifies the test results of memory (RAM, 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 <u>not</u> clear the Questionable Test Condition Register. **Command Syntax** None **Query Syntax** 8A10$_{16}$ → <condition> **Parameters:** <condition> Type: 16-bit unsigned integer Range: 0 to 65535 0001$_{16}$ Baseboard Test Failed Bit 0002$_{16}$ SRAM Test Failed Bit 0004$_{16}$ ROM Test Failed Bit 0008$_{16}$ Unused Bits 0010$_{16}$ Ref Oscillator Test Failed Bit 0020$_{16}$ Unused Bits 0040$_{16}$ Flash Memory Test Failed Bit 0080$_{16}$ Unused Bits 0100$_{16}$ Input1–2 Register Test Failed Bit 0200$_{16}$ Input1 RAM Test Failed Bit 0400$_{16}$ Input2 RAM Test Failed Bit 0800$_{16}$ PLL Test Failed Bit F000$_{16}$ Unused Bits

Name	Description
Status Test Enable Command 0A11$_{16}$ **Status Test Enable Query** 8A11$_{16}$	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* and *Status Test Enable 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** 0A11$_{16}$ **Query Syntax** 8A11$_{16}$ → <enable> **Parameters:** <enable> Type: 16-bit unsigned integer Range: 0 to 65535 0001$_{16}$ Baseboard Test Failed Bit 0002$_{16}$ SRAM Test Failed Bit 0004$_{16}$ ROM Test Failed Bit 0008$_{16}$ Unused Bits 0010$_{16}$ Ref Oscillator Test Failed Bit 0020$_{16}$ Unused Bits 0040$_{16}$ Flash Memory Test Failed Bit 0080$_{16}$ Unused Bits 0100$_{16}$ Input1–2 Register Test Failed Bit 0200$_{16}$ Input1 RAM Test Failed Bit 0400$_{16}$ Input2 RAM Test Failed Bit 0800$_{16}$ PLL Test Failed Bit F000$_{16}$ Unused Bits

Name	Description
Status Test Event Query 8A12$_{16}$	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** 8A12$_{16}$ → <event> **Parameters:** <event> Type: 16-bit unsigned integer Range: 0 to 65535 0001$_{16}$ Baseboard Test Failed Bit 0002$_{16}$ SRAM Test Failed Bit 0004$_{16}$ ROM Test Failed Bit 0008$_{16}$ Unused Bits 0010$_{16}$ Ref Oscillator Test Failed Bit 0020$_{16}$ Unused Bits 0040$_{16}$ Flash Memory Test Failed Bit 0080$_{16}$ Unused Bit 0100$_{16}$ Input1–2 Register Test Failed Bit 0200$_{16}$ Input1 RAM Test Failed Bit 0400$_{16}$ Input2 RAM Test Failed Bit 0800$_{16}$ PLL Test Failed Bit F000$_{16}$ Unused Bits

Name	Description
Status Voltage Condition Query 8A19$_{16}$	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 not clear the Questionable Voltage Condition Register. **Command Syntax** None **Query Syntax** 8A19$_{16}$ → <condition> **Parameters:** <condition> Type: 16-bit unsigned integer Range: 0 to 65535 0001$_{16}$ Input1 Overload 0002$_{16}$ Input2 Overload 000C$_{16}$ Unused Bits 0010$_{16}$ Input1 Over voltage 0020$_{16}$ Input 2 Over voltage FFC0$_{16}$ Unused Bits

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Status Voltage Enable Command
$0A1A_{16}$

Status Voltage Enable Query
$8A1A_{16}$</td><td>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 and Status Voltage Enable 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
$0A1A_{16}$

Query Syntax
$8A1A_{16}$ → <enable>

Parameters:
<enable>
Type: 16-bit unsigned integer
Range: 0 to 65535

0001_{16} Input1 Overload
0002_{16} Input2 Overload
$000C_{16}$ Unused Bits
0010_{16} Input1 Over voltage
0020_{16} Input 2 Over voltage
$FFC0_{16}$ Unused Bits</td></tr>
</table>

Name	Description
Status Voltage Event Query $8A1B_{16}$	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 *Status Questionable Event 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 must be cleared by reading it before the processes are run. **Command Syntax** None **Query Syntax** $8A1B_{16}$ → <event> **Parameters:** <event> Type: 16-bit unsigned integer Range: 0 to 65535 0001_{16} Input1 Overload 0002_{16} Input2 Overload $000C_{16}$ Unused Bits 0010_{16} Input1 Over voltage 0020_{16} Input 2 Over voltage $FFC0_{16}$ Unused Bits

Name	Description
Sweep Mode Command 0510_{16} **Sweep Mode Query** 8510_{16}	Sets or queries the trigger mode to enable automatic triggering in absence of a trigger event. **Command Syntax** 0510_{16} <trigger_mode> **Query Syntax** 8510_{16} → <trigger_mode> **Parameters:** <trigger_mode> Type: 16-bit unsigned integer 0000_{16} Auto 0001_{16} Normal
Sweep Offset Reference Command 0514_{16} **Sweep Offset Reference Query** 8514_{16}	Sets or queries the waveform record offset reference location. The following considerations apply when using the *Sweep Offset Reference Command* and *Sweep Offset Reference Query*: • 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** 0514_{16} <location> **Query Syntax** 8514_{16} → <location> **Parameters:** <location> Type: 32-bit floating point Range: 0.0 to 1.0

Name	Description

Sweep Offset Time Command
0515_{16}

Sweep Offset Time Query
8515_{16}

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* and *Sweep Offset Time Query*:

- 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
0515_{16} <sweep_offset_time>

Query Syntax
8515_{16} → <sweep_offset_time>

Parameters:
<sweep_offset_time>
Type: 32-bit floating

Range: 0 to 655 seconds
Resolution:

Resolution	Sweep Offset Time
10 ns	0 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 6.5536 s
1 ms	6.5536 s to 65.536 s
10 ms	65.536 s to 655 s

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Sweep Points Command
0511_{16}

Sweep Points Query
8511_{16}</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 and Sweep Points Query:

• The minimum record length is 100 data points.

• The maximum record length is the size of the digitizer memory, or 4,194,304 points. The maximum record size is further limited by the number of active channels.

• When in 1 channel interleaved mode, the enabled channel may use up to the full digitizer memory size.

• When in 2 channel mode, each channel may use up to one-half the full digitizer memory size.

Note: The capture circuitry requires a small number of samples in the digitizer memory for housekeeping, thus at the maximum waveform sizes, bad samples may be returned at the beginning of the waveform record

• Use the Sweep Time Query to read the current sample rate in samples per second.

• All active channels share the same record length setting.

Command Syntax
0511_{16} <points>

Query Syntax
8511_{16} → <points>

Parameters:
<points>
Type: 32-bit unsigned integer

Range: 100 to 4,194,304.</td></tr>
</table>

Name	Description
Sweep Time Query 8512_{16}	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 500 ns, or 100 points at 200 MS/s • The maximum sweep time is 83.886 seconds, or 2,097,152 points at 25 kS/s. • All channels share the same sweep time span. **Command Syntax** None **Query Syntax** 8512_{16} → <seconds> **Parameters:** <seconds> Type: 32-bit floating point Range: 500 ns to 83.886 seconds
System Delay Bypass Command 0020_{16} **System Delay Bypass Query** 8020_{16}	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** 0020_{16} <state> **Query Syntax** 8020_{16} → <state> **Parameters** <state> Type: 16-bit unsigned integer 0000_{16} Inactive state (delay enabled) 0001_{16} Active state (delay bypassed)

Name	Description
System Error All Query 8003_{16}	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** 8003_{16} → <error_number> **Parameters:** <error_number> Type: 16-bit signed integer Range: 0 to –32768 0: No error **Note:** See *Appendix 3, Error Table*, for a description of errors.
System Error Count Query $F002_{16}$	Returns the number of errors in the error log. **Command Syntax** None **Query Syntax** $F002_{16}$ → <error_count> **Parameters:** <error_count> Type: 16-bit unsigned integer Range: 0 to 32

Name	Description
System Error Query F000$_{16}$	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** F000$_{16}$ → <error> **Parameters:** <error> Type: 16-bit signed integer Range: 0 to –32768 0: no error **Note:** See *Appendix 3, Error Table*, for a description of errors.
System Memory Query 8500$_{16}$	Returns the total available waveform memory for use by the two digitizer channels. **Command Syntax** None **Query Syntax** 8500$_{16}$ → <size> **Parameters:** <size> Type: 32-bit unsigned integer Memory 4 MSamples Total **Note:** Total samples <u>must</u> be equal to the installed memory option.

Name	Description
System Temperature Query 8008_{16}	Returns the instrument temperature in degrees Celsius. The temperature has a high threshold of 65 degrees Celsius. If the instrument temperature exceeds the high threshold, it will immediately power down. **Command Syntax** None **Query Syntax** 8008_{16} → <temperature> **Parameters:** <temperature> Type: 32-bit floating point
Test Query 8001_{16}	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** 8001_{16} → <code> **Parameters:** <code> Type: 16-bit unsigned integer Range: 0 to 65535 0001_{16} Baseboard Test Failed Bit 0002_{16} SRAM Test Failed Bit 0004_{16} ROM Test Failed Bit 0008_{16} Unused Bits 0010_{16} Ref Oscillator Test Failed Bit 0020_{16} Unused Bits 0040_{16} Flash Memory Test Failed Bit 0080_{16} Unused Bits 0100_{16} Input1–2 Register Test Failed Bit 0200_{16} Input1 RAM Test Failed Bit 0400_{16} Input2 RAM Test Failed Bit 0800_{16} PLL Test Failed Bit $F000_{16}$ Unused Bits

Name	Description
Trace Copy Reference Command 0528_{16}	Stores the reference waveform in nonvolatile memory. **Command Syntax** 0528_{16} <reference_channel> <source> **Query Syntax** None **Parameters:** <reference_channel> Type: 16-bit unsigned integer 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4
Trace Download Clear Command 0532_{16}	Clears the number of waveform capture events and skipped download events when using auto-download functionality. Auto-download causes the captured data to be automatically loaded into the data buffer upon each capture event. This functionality is useful to download data or update a waveform display as rapidly as possible. The instrument records the total number of waveform capture events and the number of skipped download events. These results can be queried with the *Trace Download Report Query*. **Command Syntax** 0532_{16} **Query Syntax** None **Parameters:** None

Name	Description
Trace Download Report Query 8531_{16}	Queries the number of waveform capture events and skipped download events when using auto-download functionality. Auto-download causes the captured data to be automatically loaded into the data buffer upon each capture event. This functionality is useful to download data or update a waveform display as rapidly as possible. The instrument records the total number of waveform capture events and the number of skipped download events. These results can be queried with this query. **Command Syntax** None **Query Syntax** 8531_{16} → <count>, <skipped> **Parameters:** <count> Type: 32-bit unsigned integer <skipped> Type: 32-bit unsigned integer

Name	Description
Trace Download State Command 0530_{16} **Trace Download State Query** 8530_{16}	Sets or queries the state of the auto-download functionality. Auto-download causes the captured data to be automatically loaded into the data buffer upon each capture event. This functionality is useful to download data or update a waveform display as rapidly as possible. When using auto-download, the following considerations apply: • Only waveforms for enabled channels are downloaded upon each capture. • The total waveform download size must be less than the buffer size. For two enabled channels, the maximum waveform size is 1024 points. For one enabled channel, the maximum waveform size is 2048 points. • Upon the download of each waveform, the auto-download event bit in the Operation Status register is set. The state of this bit is checked upon subsequent captures to ensure that the existing data has been read before over-writing. If set, the new data is not automatically downloaded to the buffer. The host should clear this bit (by querying the Operation Status event register or by issuing a Clear Status command), after reading the waveform data to indicate that the buffer is available for new data. • The instrument records the total number of waveform capture events and the number of skipped download events. These results can be queried with the *Trace Download Report Query*. **Command Syntax** 0530_{16} <state> **Query Syntax** 8530_{16} → <state> **Parameters:** <state> Type: 16-bit unsigned integer 0000_{16} Disable 0001_{16} Enable

Name	Description
Trace Load Calculate Query 8522_{16}	Returns the calculate channel waveform data. The instrument returns a “1” response to the query when it has moved all of the data into the memory. **Command Syntax** None **Query Syntax** 8522_{16} <calc_channel> <waveform_index> <buffer_address> <points> → “1” **Parameters:** <calc_channel> Type: 16-bit unsigned integer 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 <waveform_index> Type: 32-bit unsigned integer Range: 0 to 32768 <buffer_address> Type: 32-bit unsigned integer Range: 16517 (4085_{16}) <points> Type: 32-bit unsigned integer Range: 0 to 2048 (800_{16})

Name	Description
Trace Load Input Query 8520_{16}	Returns the waveform data for the input channel. The instrument returns a “1” response to the query when it has moved all of the data into the memory. **Command Syntax** None **Query Syntax** 8520_{16} <input_channel> <waveform_index> <buffer_address> <points> → “1” **Parameters:** <input_channel> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 <waveform_index> Type: 32-bit unsigned integer Range: 0 to N, where N = Maximum Memory <buffer_address> Type: 32-bit unsigned integer Range: 16517 (4085_{16}) <points> Type: 32-bit unsigned integer Range: 0 to 2048 (800_{16})

Name	Description
Trace Load Reference Command 0521_{16} **Trace Load Reference Query** 8521_{16}	Read or write the reference channel waveform data. The instrument returns a “1” response to the query when it has moved all of the data into the memory. **Note:** Use *Trace Preamble Command/Query* to read/write the appropriate time and voltage conversion factors. **Command Syntax** 0521_{16} <reference_channel> <waveform_index> <buffer_address> <points> **Query Syntax** 8521_{16} <reference_channel> <waveform_index> <buffer_address> <points> → “1” **Parameters:** <reference_channel> Type: 16-bit unsigned integer 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <waveform_index> Type: 32-bit unsigned integer Range: 0 to 32768 <buffer_address> Type: 32-bit unsigned integer Range: 16517 (4085_{16}) <points> Type: 32-bit unsigned integer Range: 0 to 2048 (800_{16})
Trace Preamble Command 0529_{16} **Trace Preamble Query** 8529_{16}	Reads or writes the waveform trace preamble to the unit. The Trace Preamble is an ordered syntax of values. The values are shown below. **Note:** Preambles may only be loaded for reference waveforms. • **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, envelope or equivalent-time waveform. In Normal acquisition the Acquisition Count is always 1. • **Time Interval:** Returns the time interval between points. • **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

Name	Description
	voltage for the specified waveform. **Command Syntax** 0529_{16} <source> <type> <points> <count> <time_interval> <time_offset> <voltage_interval> <voltage_offset> **Query Syntax** 8529_{16} <source> → <type> <points> <count> <time_interval> <time_offset> <voltage_interval> <voltage_offset> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 0002_{16} Calculation Channel 1 0003_{16} Calculation Channel 2 0004_{16} Reference Channel 1 0005_{16} Reference Channel 2 0006_{16} Reference Channel 3 0007_{16} Reference Channel 4 <type> Type: 16-bit unsigned integer 0000_{16} Invalid Waveform 0001_{16} Normal Waveform 0002_{16} Averaged Waveform 0003_{16} Envelope Waveform 0004_{16} Equivalent Time Waveform 0010_{16} Frequency Domain Waveform (FFT) <points> Type: 32-bit unsigned integer Range: 100 to Maximum memory <count> Type: 16-bit unsigned integer Range: 1 to 65535 <time_interval> Type: 32-bit floating point <time_offset> Type: 32-bit floating point <voltage_interval> Type: 32-bit floating point <voltage_offset> Type: 32-bit floating point

Name	Description

Trigger A Event Count Command
0211_{16}
 	
Trigger A Event Count Query
8211_{16}
	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:
	  o Arm
	  o Count trigger events
	  o When the event count is reached, Trigger A will:
	    ▪ Enable Trigger B detection
	    ▪ Enable the sweep timer
	  o 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 *Trigger A Type Command* for information on configuring trigger qualifier parameters).
	
	Command Syntax
	0211_{16} <count>
	
	Query Syntax
	8211_{16} → <count>
	
	Parameters:
	<count>
	Type: 16-bit unsigned integer
	Range: 1 to 65535

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>

Trigger A Pulse High Limit Command
0214_{16}

Trigger A Pulse High Limit Query
8214_{16}

</td><td>

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
0214_{16} <high_limit>

Query Syntax
8214_{16} → <high_limit>

Parameters:
<upper_limit>
Type: 32-bit floating point

Range: 0 to 655 seconds
Resolution:

<table>
<tr><th>Resolution</th><th>Pulse Width Range</th></tr>
<tr><td>10 ns</td><td>20 ns to 655.36 us</td></tr>
<tr><td>100 ns</td><td>655.36 us to 6.5536 ms</td></tr>
<tr><td>1 us</td><td>6.5536 ms to 65.536 ms</td></tr>
<tr><td>10 us</td><td>65.536 ms to 655.36 ms</td></tr>
<tr><td>100 us</td><td>655.36 ms to 6.5536 s</td></tr>
<tr><td>1 ms</td><td>6.5536 s to 65.536 s</td></tr>
<tr><td>10 ms</td><td>65.536 s to 655 s</td></tr>
</table>

</td></tr>
</table>

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger A Pulse Low Limit Command
0215_{16}

Trigger A Pulse Low Limit Query
8215_{16}</td><td>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
0215_{16} <low_limit>

Query Syntax
8215_{16} → <low_limit>

Parameters:
<lower_limit>
Type: 32-bit floating point

Range: 0 to 655 seconds
Resolution:
<table>
<tr><th>Resolution</th><th>Pulse Width Range</th></tr>
<tr><td>10 ns</td><td>10 ns to 655.36 us</td></tr>
<tr><td>100 ns</td><td>655.36 us to 6.5536 ms</td></tr>
<tr><td>1 us</td><td>6.5536 ms to 65.536 ms</td></tr>
<tr><td>10 us</td><td>65.536 ms to 655.36 ms</td></tr>
<tr><td>100 us</td><td>655.36 ms to 6.5536 s</td></tr>
<tr><td>1 ms</td><td>6.5536 s to 65.536 s</td></tr>
<tr><td>10 ms</td><td>65.536 s to 655 s</td></tr>
</table></td></tr>
<tr><td>Trigger A Slope Command
0203_{16}

Trigger A Slope Query
8203_{16}</td><td>Sets or queries the active edge of the selected trigger.

Command Syntax
0203_{16} <slope>

Query Syntax
8203_{16} → <slope>

Parameters:
<slope>
Type: 16-bit unsigned integer

0000_{16} Falling Edge
0001_{16} Rising Edge</td></tr>
</table>

Name	Description
Trigger A Source Command 0201_{16} **Trigger A Source Query** 8201_{16}	Sets or queries the Trigger A signal source. The following considerations apply when using the *Trigger A Source Command* and *Trigger A Source Query*: • 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 the Trigger B detector or sweep offset timer are enabled. **Command Syntax** 0201_{16} <source> **Query Syntax** 8201_{16} → <source> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Software 0001_{16} TTL Trigger0 0002_{16} TTL Trigger1 0003_{16} TTL Trigger2 0004_{16} TTL Trigger3 0005_{16} TTL Trigger4 0006_{16} TTL Trigger5 0007_{16} TTL Trigger6 0008_{16} TTL Trigger7 0009_{16} Star Trigger $000A_{16}$ Channel 1 Trigger $000B_{16}$ Channel 2 Trigger $000C_{16}$ External Trigger $000D_{16}$ Pattern Trigger

Name	Description
Trigger A Type Command 0210_{16} **Trigger A Type Query** 8210_{16}	Sets or queries the Trigger A detection type. The following considerations apply when using the *Trigger A Type Command* and *Trigger A Type Query*: • 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 complete before the Trigger B detector or sweep offset timer are enabled **Command Syntax** 0210_{16} <trigger_type> **Query Syntax** 8210_{16} $\rightarrow$ <trigger_type> **Parameters:** <trigger_type> Type: 16-bit unsigned integer 0000_{16} Edge 0001_{16} Pulse Inside Limits 0002_{16} Pulse Outside Limits 0003_{16} Pulse Less Than 0004_{16} Pulse Greater Than 0005_{16} Video

Name	Description
Trigger A Video Field Command 0217_{16} **Trigger A Video Field Query** 8217_{16}	Sets or queries which video field to select a line to trigger on. **Command Syntax** 0217_{16} <video_field> **Query Syntax** 8217_{16} → <video_field> **Parameters:** <video_field> Type: 16-bit unsigned integer Range: 1 or 2
Trigger A Video Line Command 0218_{16} **Trigger A Video Line Query** 8218_{16}	Sets or queries which video line to trigger on. **Note:** The line depends on the video standard and video field. **Command Syntax** 0218_{16} <video_line> **Query Syntax** 8218_{16} → <video_line> **Parameters:** <video_line> Type: 16-bit unsigned integer Range: NTSC Field 1: 1 to 263 Field 2: 1 to 262 PAL/SECAM Field 1: 1 to 313 Field 2: 314 to 625
Trigger A Video Standard Command 0216_{16} **Trigger A Video Standard Query** 8216_{16}	Sets or queries the video standard. **Command Syntax** 0216_{16} <video_standard> **Query Syntax** 8216_{16} → <video_standard> **Parameters:** <video_standard> Type: 16-bit unsigned integer 0000_{16} PAL 0001_{16} NTSC 0002_{16} SECAM

Name	Description
Trigger B Slope Command 0223_{16} **Trigger B Slope Query** 8223_{16}	Sets or queries the active edge of the selected trigger. **Command Syntax** 0223_{16} <slope> **Query Syntax** 8223_{16} $\rightarrow$ <slope> **Parameters:** <slope> Type: 16-bit unsigned integer 0000_{16} Falling Edge 0001_{16} Rising Edge
Trigger B Source Command 0221_{16} **Trigger B Source Query** 8221_{16}	Sets or queries the Trigger B signal source. The following considerations apply: • 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* to change the slope. • Trigger A detection must complete before the Trigger B detector or sweep offset timer are enabled. **Command Syntax** 0221_{16} <source> **Query Syntax** 8221_{16} $\rightarrow$ <source> **Parameters:** <source> Type: 16-bit unsigned integer 0000_{16} Software 0001_{16} TTL Trigger0 0002_{16} TTL Trigger1 0003_{16} TTL Trigger2 0004_{16} TTL Trigger3 0005_{16} TTL Trigger4 0006_{16} TTL Trigger5 0007_{16} TTL Trigger6 0008_{16} TTL Trigger7 0009_{16} Star Trigger $000A_{16}$ Channel 1 Trigger $000B_{16}$ Channel 2 Trigger $000C_{16}$ External Trigger $000D_{16}$ Pattern Trigger

<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Trigger B State Command
0220_{16}

Trigger B State Query
8220_{16}</td><td>Sets or queries the Trigger B status state.

Command Syntax
0220_{16} <state>

Query Syntax
8220_{16} $\rightarrow$ <state>

Parameters:
<state>
Type: 16-bit unsigned integer

0000_{16} Disable
0001_{16} Enable</td></tr>
<tr><td>Trigger Hold Off Command
$020B_{16}$

Trigger Hold Off Query
$820B_{16}$</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
$020B_{16}$ <seconds>

Query Syntax
$820B_{16}$ $\rightarrow$ <seconds>

Parameters:
<seconds>
Type: 32-bit floating point

Range: 0 to 655 seconds
Resolution:

<table>
<tr><th>Resolution</th><th>Hold Off Range</th></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 655s</td></tr>
</table>
</td></tr>
</table>

Name	Description
Trigger Immediate Command 0202_{16}	Causes an immediate trigger event for any selected trigger source. If enabled, the trigger outputs on the PXI backplane of PCI Timing Expansion Connector will also toggle when this command is issued. **Command Syntax** 0202_{16} **Query Syntax** None **Parameters:** None
Trigger Input Level Command 0207_{16} **Trigger Input Level Query** 8207_{16}	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. • The trigger level circuitry for input channel 1 and input channel 2 share one common trigger level DAC. Consequently, the levels for channels 1 and 2 cannot be set independently when both are used simultaneously in A-B triggering or pattern triggering. When the trigger level for one channel is set, the trigger level for the other channel is automatically updated by the ZT410. • The logic level out of the trigger threshold detector is routed to the Trigger A and B multiplexers for selection as an input trigger source. **Command Syntax** 0207_{16} <input_channel> <volts> **Query Syntax** 8207_{16} <input_channel> → <volts> **Parameters:** <input_channel> Type: 16-bit unsigned integer 0000_{16} Input Channel 1 0001_{16} Input Channel 2 <volts> Type: 32-bit floating point Range: Input Offset $\pm \frac{\text{Input Range}}{2}$

Name	Description
Trigger Pattern Mask Command 0212_{16} **Trigger Pattern Mask Query** 8212_{16}	Sets or queries which sources to use in the pattern. **Command Syntax** 0212_{16} <pattern_mask> **Query Syntax** 8212_{16} → <pattern_mask> **Parameters:** <pattern_mask> Type: 16-bit unsigned integer 0 Do not use in pattern trigger 1 Use in pattern trigger Source Order (MSB–LSB): • Bits 15–4 are ignored • Bit 3—Star Trigger • Bit 2—External Trigger • Bit 1—Channel 2 • Bit 0—Channel 1
Trigger Pattern Truth Command 0213_{16} **Trigger Pattern Truth Query** 8213_{16}	Sets or queries the state of each source for the pattern trigger to occur. **Command Syntax** 0213_{16} <pattern_truth> **Query Syntax** 8213_{16} → <pattern_truth> **Parameters:** <pattern_truth> Type: 16-bit unsigned integer 0 Low 1 High Source Order (MSB–LSB): • Bits 15–4 are ignored • Bit 3—Star Trigger • Bit 2—External Trigger • Bit 1—Channel 2 • Bit 0—Channel 1

Name	Description
Trigger Timestamp Query $820C_{16}$	Returns the trigger timestamp of the most recent trigger event in fractional seconds with a 1 second period. **Command Syntax** None **Query Syntax** $820C_{16}$ → <seconds> **Parameters:** <seconds> Type: 32-bit floating point Range: 0 to 1 second Resolution: 100 ns
Wait to Continue Command 0005_{16}	Allows the user to force the interface to wait unit operations are complete before resuming. **Command Syntax** 0005_{16} **Query Syntax** None **Parameters:** None

Specifications

Analog Input

Channels — Quantity 2

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) — ±5V (DC + peak AC)
Input load protection @ ±6 VDC

Maximum Input (1 MΩ) — ±25V [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 250 MHz
Input Bias	≤ ±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 + 150 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 (ZT410-20 only)
400 MS/s, 1 channel interleaved
500 MS/s, 1 channel interleaved (ZT410-2X only)

Resolution &
Maximum Sample Rate

Product Option	ADC Resolution	1 Channel Maximum Sample Rate	2 Channel Maximum Sample Rate
ZT410-2X	14-bit	500 MS/s	250 MS/s
ZT410-5X	16-bit	400 MS/s	200 MS/s

Acquisition Time Range

Product Option	Minimum Acquisition Time	Maximum Acquisition Time
ZT410-2X	200 ns	3,355 seconds
ZT410-5X	250 ns	3,355 seconds

Channel-to-Channel Skew — ≤ 100 ps difference with channels at same input settings

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 channel 2.

Waveform Memory

Total Memory — Up to 8M samples/channel
Up to 16M samples/channel (1 channel interleaved)

Memory Options — 1M samples total (ZT410-X0)
16M samples total (ZT410-X1)

Acquisition Modes

Types — Normal, Average, Envelope, and Equivalent-Time

Channels — Quantity 2, both 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 2, External Trigger, Pattern, Software, Star Trigger, TTL Trigger0–7 (PXI Backplane or PCI Timing Expansion Connector)
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 Hold Off	Programmable delay after trigger before recognizing next trigger event
Hold Off Range	0 to 655 seconds
Trigger B	Second edge trigger event qualifier
Pattern Trigger	Pattern match true or false
Pattern Sources	Channels 1 to 2, External Trigger, Star Trigger (PXI Backplane or PCI Timing Expansion Connector)
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, > 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 1–2 Trigger Level	(offset – full scale/2) to (offset + full scale/2)
Ch 1–2 Trigger Sensitivity	5% of full scale (DC to 75 MHz)
Ch 1–2 Trigger Bandwidth	≥ 200 MHz
Ch 1–2 Trigger Hysteresis	2.5% of full scale (overdrive required)
Ch 1–2 Level Resolution	0.025% of full scale
Ch 1–2 Level Accuracy	±(2% full scale + 5 mV + offset accuracy)
Trigger Timestamp	100 ns resolution, 1 second rollover

TTL Trigger Outputs

Functionality	Event Output Signals
Outputs	TTL Trigger0–7 (PXI Backplane or PCI Timing Expansion Connector)
Source	Trigger Event, Arm Event, OPC, Constant State

External Trigger

Functionality	Trigger Input or Output
Maximum Input	0V to 5V, no damage
Trigger Input	TTL Compatible, 10 kΩ Input Impedance
Trigger Output	TTL Compatible into 50Ω
Trigger Output Source	Trigger Event, Arm Event, OPC, Constant State, 10MHz Reference Clock, 500 Hz Clock, 10 ns Pulse at 1 ms Repetition Interval
Connector	SMB

Arm

Functionality	Arm to qualify trigger event
Source	External Trigger, Software, Star Trigger, TTL Trigger0–7 (PXI Backplane or PCI Timing Expansion Connector)
Polarity	Positive or Negative

External Arm Input

Maximum Input	0V to 5V, no damage
Nominal Level	TTL Compatible
Input Impedance	10 kΩ ±2%
Connector	SMB (Shared with Trigger I/O)

External Sampling Clock

Functionality	External Sampling Clock bypasses Phase Locked Loop
Synchronization	Both 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
ZT410-2X	500 MHz	500 MS/s	250 MS/s
ZT410-5X	400 MHz	400 MS/s	200 MS/s

Maximum Input	±5VDC, no damage
Input Signal Level	500 mVpp to 1 Vpp, sine or square wave
Input Impedance	AC coupled, 50Ω ±2%
Connector	SMB

10 MHz Time Base Reference

Clock Source	Internal TCXO, PXI Backplane or PCI Timing Expansion Connector
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 Scale Factor, ADC balance

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, and 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

PCI Bus	33 MHz, 32 bit address, 16 bit data
PCI Voltage	Universal, +3.3V or +5V
PCI Compatibility	Version 2.2
PXI Signals	PXI_TRGn input/output selectable PXI_STAR input 10 MHz reference input Left and right side buses not used
Manufacturer ID	4172 ($104C_{16}$)
Model Code	44128 ($AC60_{16}$)

PXI J2 Trigger & Clock Pin Usage

Pin A16	PXI Trigger 1	(TTL level bidirectional)
Pin A17	PXI Trigger 2	(TTL level bidirectional)
Pin A18	PXI Trigger 3	(TTL level bidirectional)
Pin B16	PXI Trigger 0	(TTL level bidirectional)
Pin B18	PXI Trigger 4	(TTL level bidirectional)
Pin C18	PXI Trigger 5	(TTL level bidirectional)
Pin D17	PXI Star Trigger	(TTL level input)
Pin E16	PXI Trigger 7	(TTL level bidirectional)
Pin E17	PXI CLK10 In	(TTL level input)
Pin E18	PXI Trigger 6	(TTL level bidirectional)

LED Indicators

RDY	Unit has passed power-up self-diagnostics. Toggles when unit has an error pending in error queue.
TRG	Flashes when trigger event occurs

Physical

Physical size	Single-Wide 3U CompactPCI/PXI Instrument (PXI) Single-Slot Short PCI Card (PCI)
Weight	1 lb.

DC Power

Cooling & Power Consumption

Product Option	Typical Cooling & Power	Maximum Cooling & Power
ZT410-20	23.1 W	27.9 W
ZT410-21	27.2 W	32.0 W
ZT410-50	23.4 W	28.3 W
ZT410-51	27.6 W	32.4 W

PCI Cooling Fan adds 0.12A to +5VDC current requirements and 0.6W to total power consumption.

Power Supplies

Product Option	Voltage	Typical Current	Maximum Current
ZT410-20	+3.3V +5V +12V –12V	3.9A 0.9A 0.2A 0.2A	3.9A 1.4A 0.25A 0.4A
ZT410-21	+3.3V +5V +12V –12V	5.4A 0.9A 0.2A 0.2A	5.4A 1.4A 0.25A 0.4A
ZT410-50	+3.3V +5V +12V –12V	3.9A 1.1A 0.2A 0.2A	3.9A 1.5A 0.25A 0.4A
ZT410-51	+3.3V +5V +12V –12V	5.3A 1.1A 0.2A 0.2A	5.3A 1.5A 0.25A 0.4A

Temperature Range

Operating — 0 °C to +40 °C Ambient

Storage — -40 °C to +75 °C

Over-Temp Protection — Automatic shutdown if internal temperature is greater than +65 °C (PCI)

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

Safety

This product is designed to meet the requirements of the following standard of safety for electrical equipment for measurement, control and laboratory use:

EN 61010-1

Electromagnetic Compatibility

CE Marking EN 61326-1:1997 with A1:1998 and A2:2001 Compliant
FCC Part 15 (Class A) Compliant

Emissions

EN 55011	Radiated Emissions, ISM Group 1, Class A, distance 10 m, emissions < 1 GHz
EN 55011	Conducted Emissions, Class A, emissions < 30 MHz

Immunity

EN 61000-4-2	Electrostatic Discharge (ESD), 4 kV by Contact, 8 kV by Air
EN 61000-4-3	RF Radiated Susceptibility, 10 V/m
EN 61000-4-4	Electrical Fast Transient Burst (EFTB), 2 kV AC Power Lines
EN 61000-4-5	Surge
EN 61000-4-6	Conducted Immunity
EN 61000-4-8	Power Frequency Magnetic Field, 30 A/m
EN 61000-4-11	Voltage Dips and Interrupts

CE Compliance

This product meets the necessary requirements of applicable European Directives for CE Marking as follows:

73/23/EEC	Low Voltage Directive (Safety)
89/336/EEC	Electromagnetic Compatibility Directive (EMC)

See Declaration of Conformity for this product for additional regulatory compliance information.

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
Envelope View	Minimum
Equivalent-Time Points	10
External Trigger Output	Disabled, Positive Polarity, Trigger Event Source
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 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
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
Trigger B Source	Channel 2

Parameter	Default
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 Pattern Mask	0
Trigger Pattern Truth	0
Trigger Polarity	Positive
Trigger Source	Channel 1
Trigger Type	Edge
TTL Trigger Output	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 ZT410 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	

Code	Error Summary	Description
-120	Numeric data error	A generic numeric syntax error (only used when a more specific error does not apply)
-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)

Code	Error Summary	Description
-161	Invalid block data	An invalid block data element was received for this instrument command
-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

Code	Error Summary	Description
-214	Not ready for trigger	The trigger is deadlocked because a measurement result was requested before the instrument measurement was triggered
-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)

Code	Error Summary	Description
-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
-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

Code	Error Summary	Description
-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
-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

Code	Error Summary	Description
-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
-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

Code	Error Summary	Description
-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
-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
-1004	Overtemp	The instrument temperature has exceeded the 60 degree Celsius high threshold.

Commands Index

The following is an alphabetic list of the commands and queries for the ZT410.

Command Code	Query Code	Command Name
0305_{16}		Abort Command
0306_{16}	8306_{16}	Arm Command/Query
0308_{16}	8308_{16}	Arm Polarity Command/Query
0307_{16}	8307_{16}	Arm Source Command/Query
0400_{16}		Auto Scale Command
0422_{16}	8422_{16}	Average Count Command/Query
0423_{16}	8423_{16}	Average Envelope View Command/Query
0420_{16}	8420_{16}	Average State Command/Query
0421_{16}	8421_{16}	Average Type Command/Query
0811_{16}		Calculate Absolute Value Command
0810_{16}		Calculate Add Command
0812_{16}		Calculate Copy Command
0805_{16}	8805_{16}	Calculate Data Format Command/Query
0813_{16}		Calculate Derivative Command
	8803_{16}	Calculate Function Query
0804_{16}		Calculate Immediate Command
0814_{16}		Calculate Integral Command
0815_{16}		Calculate Invert Command
0824_{16}		Calculate Limit Test Clear Command
0820_{16}		Calculate Limit Test Command
0825_{16}	8825_{16}	Calculate Limit Test Continuous Command/Query
	8822_{16}	Calculate Limit Test Fail Query
0827_{16}	8827_{16}	Calculate Limit Test Lower Command/Query
0821_{16}	8821_{16}	Calculate Limit Test Measurement Command/Query
	8823_{16}	Calculate Limit Test Report Query
0826_{16}	8826_{16}	Calculate Limit Test Upper Command/Query

Command Code	Query Code	Command Name
0829_{16}	8829_{16}	Calculate Mask Test Lower Command/Query
0828_{16}	8828_{16}	Calculate Mask Test Upper Command/Query
0816_{16}		Calculate Multiply Command
0800_{16}	8800_{16}	Calculate State Command/Query
0817_{16}		Calculate Subtract Command
0831_{16}	8831_{16}	Calculate Transform Freq Window Command/Query
0830_{16}		Calculate Transform Frequency Command
0802_{16}	8802_{16}	Calculate Voltage Offset Command/Query
0801_{16}	8801_{16}	Calculate Voltage Range Command/Query
	8600_{16}	Calibrate Command/Query
0610_{16}		Calibrate Restore Factory Defaults Command
$0A00_{16}$		Clear Status Command
0405_{16}	8405_{16}	Clock Frequency Command/Query
0404_{16}	8404_{16}	Clock Source Command/Query
	$F001_{16}$	Configuration Query
$0A03_{16}$	$8A03_{16}$	Event Status Enable Command/Query
	$8A02_{16}$	Event Status Query
0010_{16}	8010_{16}	Format Byte Order Command/Query
	8103_{16}	ID Query
0304_{16}	8304_{16}	Initiate Command/Query
0309_{16}	8309_{16}	Initiate Continuous Command/Query
0411_{16}	8411_{16}	Input Attenuation Command/Query
0412_{16}	8412_{16}	Input Coupling Command/Query
0416_{16}	8416_{16}	Input Filter State Command/Query
0413_{16}	8413_{16}	Input Impedance Command/Query
0415_{16}	8415_{16}	Input Smoothing Points Command/Query
0414_{16}	8414_{16}	Input Smoothing State Command/Query
0410_{16}	8410_{16}	Input State Command/Query
$040C_{16}$	$840C_{16}$	Input Voltage Offset Command/Query
$040B_{16}$	$840B_{16}$	Input Voltage Protection State Command/Query
0407_{16}	8407_{16}	Input Voltage Range Command/Query
	$872F_{16}$	Measure AC High-Precision Query
	8710_{16}	Measure AC RMS Query
	8711_{16}	Measure Amplitude Query

Command Code	Query Code	Command Name
	8712_{16}	Measure Average Voltage Query
	$872C_{16}$	Measure Cycle Average Query
	$872D_{16}$	Measure Cycle Frequency Query
	$872E_{16}$	Measure Cycle Period Query
	$872B_{16}$	Measure Cycle RMS Query
	8730_{16}	Measure DC High-Precision Query
	8713_{16}	Measure DC RMS Query
0700_{16}	8700_{16}	Measure Edge Command/Query
	8734_{16}	Measure Effective Number of Bits Query
	8716_{16}	Measure Fall Crossing Time Query
	8714_{16}	Measure Fall Overshoot Query
	8715_{16}	Measure Fall Pre-shoot Query
	8717_{16}	Measure Fall Time Query
	8718_{16}	Measure Frequency Query
0704_{16}	8704_{16}	Measure Gate Points Command/Query
0703_{16}	8703_{16}	Measure Gate Time Command/Query
	8719_{16}	Measure High Voltage Query
	$871A_{16}$	Measure Low Voltage Query
	$871B_{16}$	Measure Maximum Voltage Query
0701_{16}	8701_{16}	Measure Method Command/Query
	$871D_{16}$	Measure Mid Voltage Query
	$871C_{16}$	Measure Minimum Voltage Query
	$871E_{16}$	Measure Negative Duty Cycle Query
	$871F_{16}$	Measure Negative Width Query
	8724_{16}	Measure Peak-to-Peak Voltage Query
	8722_{16}	Measure Period Query
	8723_{16}	Measure Phase Query
	8720_{16}	Measure Positive Duty Cycle Query
	8721_{16}	Measure Positive Width Query
0708_{16}	8708_{16}	Measure Reference Command/Query
0709_{16}	8709_{16}	Measure Reference Method Command/Query
	8727_{16}	Measure Rise Crossing Time Query
	8725_{16}	Measure Rise Overshoot Query
	8726_{16}	Measure Rise Pre-shoot Query

Command Code	Query Code	Command Name
	8728_{16}	Measure Rise Time Query
	8731_{16}	Measure Signal-to-Noise Ratio Query
	8733_{16}	Measure Signal-to-Noise & Distortion Ratio Query
	8735_{16}	Measure Spurious-Free Dynamic Range Query
	8729_{16}	Measure Time/Frequency of Maximum Voltage Query
	$872A_{16}$	Measure Time/Frequency of Minimum Voltage Query
	8732_{16}	Measure Total Harmonic Distortion Query
0004_{16}	8004_{16}	Operation Complete Command/Query
$020F_{16}$	$820F_{16}$	Output External Trigger Polarity Command/Query
$020E_{16}$	$820E_{16}$	Output External Trigger Source Command/Query
$020D_{16}$	$820D_{16}$	Output External Trigger State Command/Query
$020A_{16}$	$820A_{16}$	Output Trigger Polarity Command/Query
0209_{16}	8209_{16}	Output Trigger Source Command/Query
0208_{16}	8208_{16}	Output Trigger State Command/Query
0007_{16}		Recall Instrument State Command
0200_{16}	8200_{16}	Reference Oscillator Source Command/Query
0000_{16}		Reset Command
0006_{16}		Save Instrument State Command
$0A04_{16}$	$8A04_{16}$	Service Request Enable Command/Query
	8002_{16}	Status Byte Query
	$8A16_{16}$	Status Calibration Condition Query
$0A17_{16}$	$8A17_{16}$	Status Calibration Enable Command/Query
	$8A18_{16}$	Status Calibration Event Query
	$8A13_{16}$	Status Frequency Condition Query
$0A14_{16}$	$8A14_{16}$	Status Frequency Enable Command/Query
	$8A15_{16}$	Status Frequency Event Query
	$8A05_{16}$	Status Operation Condition Query
$0A06_{16}$	$8A06_{16}$	Status Operation Enable Command/Query
	$8A07_{16}$	Status Operation Event Query
$0A01_{16}$		Status Preset Command
	$8A08_{16}$	Status Questionable Condition Query
$0A09_{16}$	$8A09_{16}$	Status Questionable Enable Command/Query
	$8A0A_{16}$	Status Questionable Event Query
	$8A10_{16}$	Status Test Condition Query

Command Code	Query Code	Command Name
$0A11_{16}$	$8A11_{16}$	Status Test Enable Command/Query
	$8A12_{16}$	Status Test Event Command/Query
	$8A19_{16}$	Status Voltage Condition Query
$0A1A_{16}$	$8A1A_{16}$	Status Voltage Enable Command/Query
	$8A1B_{16}$	Status Voltage Event Query
0510_{16}	8510_{16}	Sweep Mode Command/Query
0514_{16}	8514_{16}	Sweep Offset Reference Command/Query
0515_{16}	8515_{16}	Sweep Offset Time Command/Query
0511_{16}	8511_{16}	Sweep Points Command/Query
	8512_{16}	Sweep Time Query
0020_{16}	8020_{16}	System Delay Bypass Command/Query
	8003_{16}	System Error All Query
	$F002_{16}$	System Error Count Query
	$F000_{16}$	System Error Query
	8500_{16}	System Memory Query
	8008_{16}	System Temperature Query
	8001_{16}	Test Query
0528_{16}		Trace Copy Reference Command
0532_{16}		Trace Download Clear Command
	8531_{16}	Trace Download Report Query
0530_{16}	8530_{16}	Trace Download State Command/Query
	8522_{16}	Trace Load Calculate Query
	8520_{16}	Trace Load Input Query
0521_{16}	8521_{16}	Trace Load Reference Command/Query
0529_{16}	8529_{16}	Trace Preamble Command/Query
0211_{16}	8211_{16}	Trigger A Event Count Command/Query
0214_{16}	8214_{16}	Trigger A Pulse High Limit Command/Query
0215_{16}	8215_{16}	Trigger A Pulse Low Limit Command/Query
0203_{16}	8203_{16}	Trigger A Slope Command/Query
0201_{16}	8201_{16}	Trigger A Source Command/Query
0210_{16}	8210_{16}	Trigger A Type Command/Query
0217_{16}	8217_{16}	Trigger A Video Field Command/Query
0218_{16}	8218_{16}	Trigger A Video Line Command/Query
0216_{16}	8216_{16}	Trigger A Video Standard Command/Query

Command Code	Query Code	Command Name
0223_{16}	8223_{16}	Trigger B Slope Command/Query
0221_{16}	8221_{16}	Trigger B Source Command/Query
0220_{16}	8220_{16}	Trigger B State Command/Query
$020B_{16}$	$820B_{16}$	Trigger Hold off Command/Query
0202_{16}		Trigger Immediate Command
0207_{16}	8207_{16}	Trigger Input Level Command/Query
0212_{16}	8212_{16}	Trigger Pattern Mask Command/Query
0213_{16}	8213_{16}	Trigger Pattern Truth Command/Query
	$820C_{16}$	Trigger Timestamp Query
0005_{16}		Wait to Continue Command

ZTEC Instruments