EPICS Interface

M-Class Oscilloscopes
Models ZT4611, ZT4612,
ZT4211, ZT4212

User's Manual: 0004-000067
Revision 1b

September 15, 2008

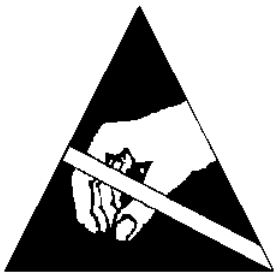
Contact

ZTEC Instruments
7715 Tiburon Street NE
Albuquerque, NM 87109

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

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

Copyright



Handling Precautions for Electronic Devices Subject to Damage by Static Electricity

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

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

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

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

Avoid placing tools or electrical parts on insulators. Do NOT use any hand tool that can generate a static charge, such as a non-conductive plunger-type solder sucker. Use a conductive strap or cable with a wrist cuff to reliably ground to the work surface. The cuff 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 kilohms to 1 megohm using a commercially-available personnel grounding device.

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

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

Revision History

Rev	Date	Section	Description
1	1-4-08	All	Initial Release
1a	4-10-08	All	Updated PV list, expanded introduction, added upload waveform information
1b	9-15-08	All	Added new PVs: gating, initiate continuous, webLXI interface

Table of Contents

Introduction

Instrument Discovery

Interfacing to the EPICS ZTEC M-Class Scope involves LAN discovery to identify the IP address of the desired instrument. The IP address can be configured or identified in one of the following ways:

1. With a DHCP server, the instrument will be dynamically assigned an IP address
2. Without a DHCP server, the instrument will use auto-IP procedures to find a non-allocated IP address.
3. To determine the instrument's IP address, consult the network manager or use any LXI compatible discovery tool.
4. After discovery, the ZTEC webLXI interface enables the network addresses to be altered, made static, and/or given an alias

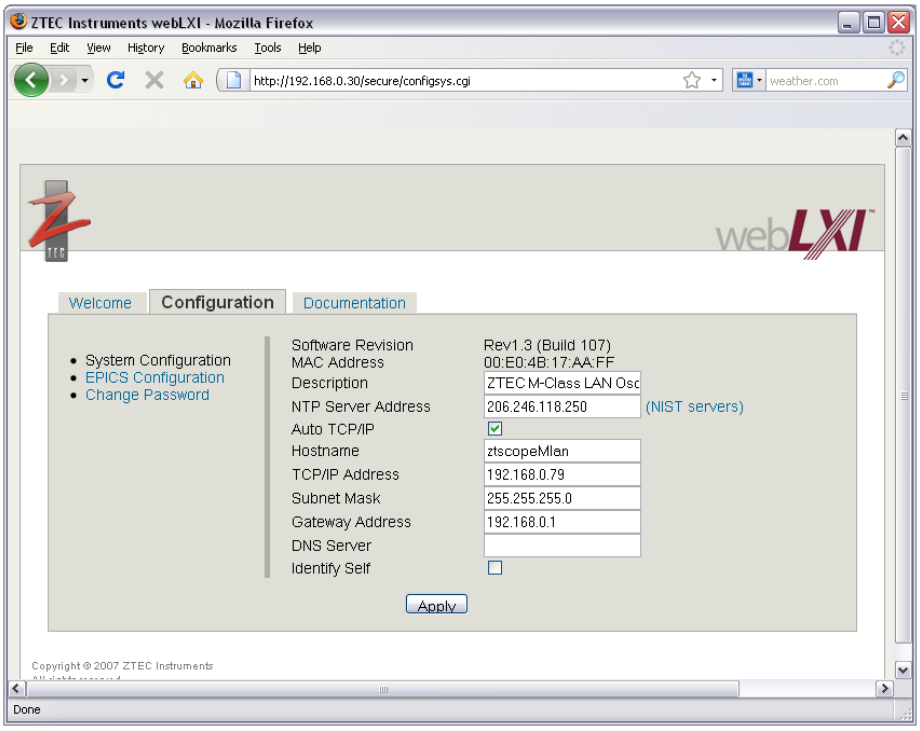
For additional help in setting up a LAN instrument for the first time, please reference the Quick Start guide that comes with the instrument or visit http://www.ztecinstruments.com/support for technical support.

Versions

The ztec EPICS interface was developed using EPICS version 3.14 and Channel Access Client version 4.11.

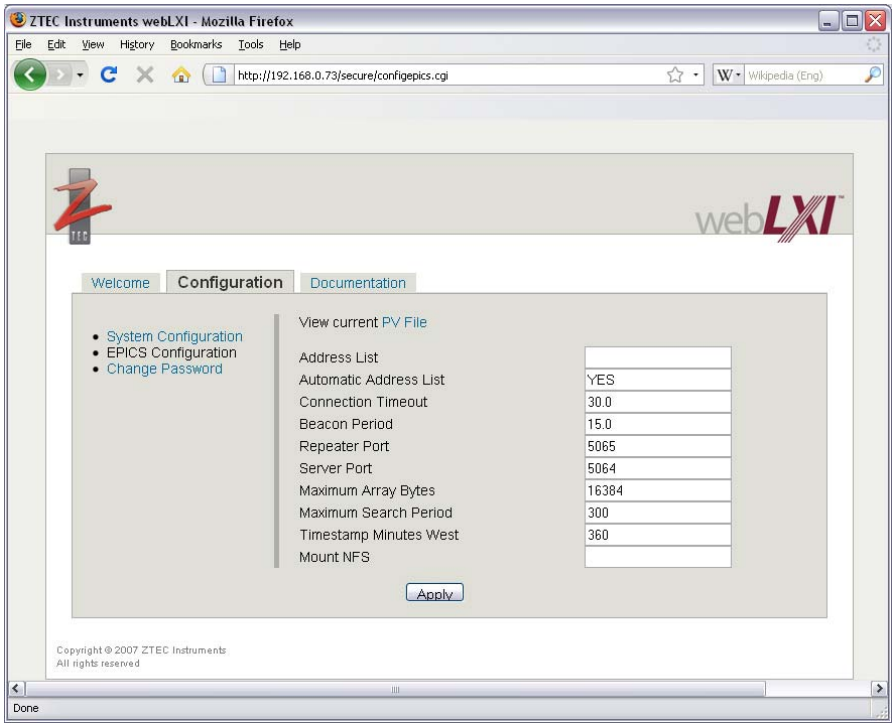
Instrument Configuration

Network and Channel Access settings may be changed using the ZTEC webLXI interface. Network configuration is done through the System Configuration page. Change the values as desired and press the Apply button.



webLXI System Configuration

Channel Access configuration is done through the EPICS Configuration page:



webLXI EPICS configuration

As with the network configuration, simply edit the values as desired and press Apply.

Database Configuration

Set/Get PVs

For all attributes that are read/write, both a *set* and a *get* PV are provided. Default names for these PVs begin with either ”set” or “get” to avoid identical names on multiple PVs. The *set* PVs allow the user to change an instrument’s state. The *get* PVs allow the user to read the value without changing the instrument’s state. *get* PVs are also useful for scanning using the .SCAN parameter, and can be chained to *set* PVs so they will notify the user of state changes. The instrument comes with a set of default fan-outs that deal with standard instrument coercion. Attributes that are only read or only write have a single PV without “get” or “set” in the default name.

Updating

ZTEC *set* PVs have the option of updating upon instrument state coercion. In default conditions when a *set* PV is changed, the user should poll the corresponding *get* PV to verify the new value. However, if setOutCoerce is set to on, *set* PVs will update their own value based on the current instrument state during processing.

Naming Conventions

The EPICS naming convention requires the ability to add a prefix to the process variables. ZTEC PVs have “ztec:” as a default prefix. The prefixes/names can be changed by editing and uploading the correct .pv file to the instrument. New database configuration files can be loaded in using the ZFind utility’s load PV Database File feature. The Installer provides a default .pv file which is installed to *installpath*/M-Class/Scope/Driver/dat. This should be used as an initial file to be modified and uploaded to the instrument if unique names are required. Please note that any name changes may affect links and fanouts; to prevent loss of functionality make sure that FLNK names are also changed. Also note that when changing the prefix that the device type is also “ztec” and should not be altered.

Changing PVs

ZTEC instruments come with a list of default PVs that exposes most instrument functionality. All PVs can be safely removed or renamed. Additional PVs may be added to the database file by following the provided template format and uploading the file as described above. Additional instrument functionality can be exposed by creating a PV associated with an instrument SCPI call. This is done by inserting the SCPI string into the SCPI section of the PV. See the instrument manual for a list of supported SCPI calls. Example PV:

```
TYPE: longout
D: ztec
NAME: ztec:setAcqCount
SCPI: "AVER:COUN"
IO: OUT
VAL: 2
HOPR: 65536
LOPR: 2
EGU: "acquisitions"
FLNK: “ztec:SweepFanout”
```

New .pv files can be loaded to the instrument using the ZTEC ZFind utility. Simply click on the desired instrument and press the EPICS icon. This will bring up an interface to load the .pv file. See the ZFind User's Guide for more information.

Since full access is allowed to the PVs, it is possible to upload a file that is non-functional. If this occurs, the instrument will disable EPICS. In order to verify that the upload was successful, attempt to access the instrument through channel access. If this fails, upload a valid PV file. PV files can be checked by using the link on the EPICS configuration webLXI page, or going to http://(your instrument's ip address)/ztscopeM.pv. PV files can be invalidated many ways, such as missing brackets, incorrect number of fields, or FLNKs to non-existent PVs.

Locking

Currently only one application may have control of the LAN device at a time. If there is an open session using another application or through a driver API then EPICS calls will be limited to read-only. This state can be observed using the UtilLockState PV. While locked out, it is safe to set values though EPICS; however, any changes will not be sent to the instrument.

Capturing

Before capturing waveforms it is best to configure the instrument as needed (see Instrument Use Example). An important PV to set is setOpInitCont; when the instrument is set to initiate continuously it will capture waveforms continuously. When the instrument is not set to initiate continuously, a single capture will occur and waveforms will be available for download. To capture, set a '1' to OpInitiate. To stop capturing, set a '1' to OpAbort.

Uploading Waveforms

Waveforms can be uploaded as an array of type DBR_DOUBLE via Channel Access using the Calc*N*Upload, Calc*N*ScaledUpload, Ref*N*Upload and Ref*N*ScaledUpload PVs (see section "Reference Process Variables"). In order to correctly store the uploaded waveform data, some waveform preamble information must be included. This is done by storing the needed values in the first four elements of the array. Note the array size transferred via Channel Access must be adjusted accordingly. Here is a simple example of a function that uses the Channel Access C function calls:

```
void upload_waveform(long *wf_buff, long type, unsigned long points,
                     double time_int, double, volt_int)
{
   int idx;
   double *upload_buff;

   upload_buff=(double *)calloc(points+4, sizeof(f64));
   upload_buff[0]=(double)type;
   upload_buff[1]=(double)points;
   upload_buff[2]=(double)time_int;
   upload_buff[3]=(double)volt_int;
   for (idx=0;idx<points;idx++)
   {
          upload_buff[idx+4]=(f64)(wf_buff[idx]);
   }
```

```
    ca_context_create(ca_disable_preemptive_callback);
    ca_create_channel("ztec:Ref1Upload",NULL,NULL,10,&chan_id);
    ca_pend_io(5.0);
    ca_array_put(DBR_DOUBLE, points+4, chan_id, (void *)upload_buff);
    ca_pend_io(5.0);
    ca_clear_channel(chan_id);
    ca_pend_io(5.0);
    free(upload_buff);
    ca_context_destroy();
}
```

This function would upload a waveform data array of *points* long integers, with waveform preamble data: waveform type = *type,* number of points = *points,* time interval = *time_int,* voltage interval = *volt_int.* For more information on waveform preamble, please refer to the M-Class Digital Storage Oscilloscope Instrument Manual

Instrument Use Example

This set up assumes a system that has the EPICS environment installed.

1. Limit your instrument communication, there are several possible options:
    a. Upload a new PV file as described above. Changing the PV prefixes will give the instrument a unique set of PVs.
    b. Set up your local channel access environment variables. If you set EPICS_CA_AUTO_ADDR_LIST to “no” and EPICS_CA_ADDR_LIST to the IP address of the instrument, this will be the only instrument seen by the local system
    c. Set up the instrument’s channel access environment variables. This will limit which systems the instrument will communicate with. The environment variables are set through webLXI as described above.The environment variables are the same whether they are on your local system or on the instrument.
2. Open a channel access command window
3. Test instrument communication by typing “caget ztec:UtilID”. If you uploaded a different PV file be sure to use the proper prefix.
4. To set up the instrument for a test, it is suggested that you use the following process:
    a. set Input PVs for all desired channels
    b. set Acquisition PVs
    c. set Horizontal PVs
    d. set Trigger PVs
    e. set Advanced Trigger PVs if required
    f. set Arm PVs for gating triggering if required
    g. set Output PVs for output signal control if required
    h. set Calculate PVs for waveform math if required
    i. set Reference PVs for waveform storage if required
    j. set Measure PVs to be performed upon acquired waveforms if required
    k. control waveform capture with Operate PVs
    l. read acquired waveforms from Waveform PVs
    m. perform miscellaneous functions with Utility PVs if required

Instrument Information

For more information on the instrument such as descriptions of functionality, specifications and default states, please refer to the M-Class Digital Storage Oscilloscope Instrument Manual.

Process Variables

Process Variable Table

The following tables list the process variables (PVs) for the oscilloscope, grouped according to instrument functionality. After each table, operational sequences for that functional group of process variables are shown.

Input Process Variables

The following table lists the PVs for the input or vertical (voltage-axis) settings of the input channels. These PVs are duplicated for each input channel. In the following table, *N* is substituted with the appropriate channel number (for example: Inp1Enable). The ZT4x11 has two channels (*N* = 1 to 2) and the ZT4x12 has four channels (*N* = 1 to 4).

Process Variable	Type	Values	Description
setInp*N*Enable getInp*N*Enable	bo bi	0 or 1	Input channel enable for selected channel: 0 = disable, 1 = enable
setInp*N*Range getInp*N*Range	ao ai	0.0125, 0.025, 0.05, 0.1, 0.25, 0.5, 1, 2, 5, 10, 20, 40, 100	Input range in Vpp for selected channel, Input range is limited to 10Vpp for 50 Ohm impedance
setInp*N*Offset getInp*N*Offset	ao ai	-50V to +50V	Input offset in Volts for selected channel, Input offset is limited according to range and impedance (see manual specifications)
setInp*N*Imped getInp*N*Imped	ao ai	50 or 1e6	Input impedance for selected channel: 50 = 50 Ohm(low), 1e6 = 1 MOhm(high)
setInp*N*Couple getInp*N*Couple	mbbo mbbi	string	Input coupling for selected channel: “AC”, “DC”
setInp*N*Filter getInp*N*Filter	bo bi	0 or 1	Input lowpass filter enable for selected channel: 0 = bypass, 1 = filter enable (20 MHz)
setInp*N*Atten getInp*N*Atten	ao ai	0.9 to 1000	Input attenuation for selected channel, typically used for probe or cable loss correction, input range and input offset are scaled accordingly, 1.0 nominal, example: use 10.0 for 10:1 probe

Horizontal Process Variables

The following table lists the PVs for the horizontal (time-axis) settings for the waveform acquisition. These PVs configure the common horizontal settings of all input channels.

Process Variable	Type	Values	Description
setHorzPoints getHorzPoints	longout longin	10 to max	Number of points in waveform: max varies for standard and extended options, max = 8M or 32M (normal acquisition, non-interleaved), max = 16M or 64M (normal acquisition, interleaved), max = 512k (other acquisition modes: average, envelope, etc.)
setHorzTime getHorzTime	ao ai	2.5e-9 to 100	Waveform sweep time (acquisition window): 2.5 ns (10 points at 4GS/s) to 100 seconds
setHorzRef getHorzRef	ao ai	0 to 100	Trigger location in acquisition window: 0% to 100%, where 0% corresponds to left-most trigger location in acquisition window
setHorzOffset getHorzOffset	ao ai	-pre-trigger time to 100	Time offset in seconds for trigger from selected trigger location offset reference: Pre-trigger: 0 to 100% of acquisition time, Post-trigger: 0 to 100 seconds
HorzRate	ai	50e3 to 200e9	Read back for sample rate setting: 50 kS/s to 2 GS/s, non-interleaved real-time, 4 GS/s, interleaved real-time, 4 GS/s to 200 GS/s, interpolated or equivalent-time
HorzInterval	ai	5e-12 to 20e-6	Readback for sample interval setting: 20 µs to 500 ps, non-interleaved real-time, 250 ps, interleaved real-time, 250 ps to 5 ps, interpolated or equivalent-time
setHorzRefOsc getHorzRefOsc	mbbo mbbi	string	Timebase 10 MHz reference oscillator: “INT” = internal, “EXT” = external input
HorzRefFreq	ai	9,999,000 to 10,001,000	Reference oscillator frequency relative to internal reference frequency upon power-on self test

Acquisition Process Variables

The following table lists the PVs for the waveform acquisition settings. These PVs configure the common acquisition settings of all input channels.

Process Variable	Type	Values	Description
setAcqType getAcqType	mbbo mbbi	string	Waveform acquisition type: “NORM” = normal “AVER” = average “ENV” = envelope “PDET” = peak detect “HRES” = high resolution “FAST” = fast acquisition “ETIM” = equivalent-time
setAcqCount getAcqCount	longout longin	2 to 65536 in powers of 2	Waveform count for multiple-capture acquisition types of average, envelope, equivalent-time, and fast acquisition
setEnvView getEnvView	bo bi	0 or 1	Envelope view select: 0 = minimum, 1 = maximum
setAverView getAverView	bo bi	0 or 1	Average view select: 0 = average, 1 = segments

Trigger Process Variables

The following table lists the PVs for the trigger settings that configure the trigger for the waveform acquisition synchronization. There are individual trigger level settings for each of the input channels, with *N* designating the appropriate channel number (for example: TrigLevInp1). The ZT4x11 has two channels (*N* = 1 to 2) and the ZT4x12 has four channels (*N* = 1 to 4).

Process Variable	Type	Values	Description
setTrigSource getTrigSource	mbbo mbbi	string	Trigger source: “TTLT0” = TTLTRG0 (external header) “TTLT1” = TTLTRG1 (external header) “TTLT2” = TTLTRG2 (external header) “TTLT3” = TTLTRG3 (external header) “TTLT4” = TTLTRG4 (external header) “TTLT5” = TTLTRG5 (external header) “TTLT6” = TTLTRG6 (external header) “TTLT7” = TTLTRG7 (external header) “INP1” = input channel 1 “INP2” = input channel 2 “INP3” = input channel 3 (4 channel only) “INP4” = input channel 4 (4 channel only) “EXT” = external input “MAN” = software only “PATT” = pattern

Process Variable	Type	Values	Description
setTrigType getTrigType	mbbo mbbi	string	Trigger type: “EDGE” = edge “IN” = pulse width in “OUT” = pulse width out “LTH” = pulse width less than “GTH” = pulse width greater than “EQU” = pulse width equal “NEQ” = pulse width not equal “GLIT” = glitch “VID” = video
setTrigMode getTrigMode	mbbo mbbi	string	Trigger mode: “AUTO” = auto-trigger “NORM” = normal trigger
setTrigSlope getTrigSlope	mbbo mbbi	string	Trigger slope or polarity: “NEG” = negative, “POS” = positive
setTrigLevInp*N* getTrigLevInp*N*	ao ai	(offset – range/2) to (offset + range/2)	Trigger level in Volts for an input channel, The trigger level must be within the vertical range and offset settings for the corresponding input channel
setTrigLevExt getTrigLevExt	ao ai	-2.0 to 2.0	Trigger level in Volts for the external input trigger
setTrigImpedExt getTrigImpedExt	ao ai	50 or 1e6	Input impedance for external input trigger: 50 = 50 Ohm, 1e6 = 1 MOhm
setTrigHoldoff getTrigHoldoff	ao ai	0 to 100	Holdoff time in seconds to wait before detecting trigger
setTrigCount getTrigCount	longout longin	1 to 65535	Trigger event count to qualify trigger

Advanced Trigger Process Variables

The following table lists the PVs for the advanced trigger settings. These PVs configure advanced trigger functions including trigger B, pulse width triggering, glitch triggering, pattern triggering, and video triggering. Consult the ztscopeM Instrument Manual for complete details on advanced triggering options.

Process Variable	Type	Values	Description
setTrigBEnable getTrigBEnable	bo bi	0 or 1	Trigger B enable: 0 = disable, 1 = enable
setTrigBSource getTrigBSource	mbbo mbbi	string	Trigger B source: “TTLT0” = TTLTRG0 (external header) “TTLT1” = TTLTRG1 (external header) “TTLT2” = TTLTRG2 (external header) “TTLT3” = TTLTRG3 (external header) “TTLT4” = TTLTRG4 (external header) “TTLT5” = TTLTRG5 (external header) “TTLT6” = TTLTRG6 (external header) “TTLT7” = TTLTRG7 (external header) “INP1” = input channel 1 “INP2” = input channel 2 “INP3” = input channel 3 (4 channel only) “INP4” = input channel 4 (4 channel only) “EXT” = external input “MAN” = software only “PATT” = pattern
setTrigBSlope getTrigBSlope	mbbo mbbi	string	Trigger slope or polarity: “NEG” = negative, “POS” = positive
setTrigBHoldoff getTrigBHoldoff	ao ai	0 to 100	Holdoff time in seconds to wait before detecting trigger B
setTrigBCount getTrigBCount	longout longin	1 to 65535	Trigger B event count to qualify trigger B
setTrigPWLower getTrigPWLower	ao ai	10e-9 to 500e-3	Lower limit for pulse width triggering time in seconds: 10 ns to 500 ms, 5 ns resolution
setTrigPWUpper getTrigPWUpper	ao ai	10e-9 to 500e-3	Upper limit for pulse width triggering time in seconds: 10 ns to 500 ms, 5 ns resolution
setTrigGlitchLimit getTrigGlitchLimit	ao ai	10e-9 to 500e-3	Limit for glitch triggering time in seconds: 10 ns to 500 ms, 5 ns resolution

Process Variable	Type	Values	Description
setTrigPattMask getTrigPattMask	longout longin	0000_{16} to $1FFF_{16}$	Pattern trigger mask for defining pattern match criteria (0 = ignore, 1 = match required), used in conjunction with set/getTrigPattTruth: Bit 0 = input channel 1 Bit 1 = input channel 2 Bit 2 = input channel 3 (4 channel only) Bit 3 = input channel 4 (4 channel only) Bit 4 = external input Bit 5 = TTLTRG0 (external header) Bit 6 = TTLTRG1 (external header) Bit 7 = TTLTRG2 (external header) Bit 8 = TTLTRG3 (external header) Bit 9 = TTLTRG4 (external header) Bit 10 = TTLTRG5 (external header) Bit 11 = TTLTRG6 (external header) Bit 12 = TTLTRG7 (external header)
setTrigPattTruth getTrigPattTruth	longout longin	0000_{16} to $1FFF_{16}$	Pattern trigger truth for defining pattern match criteria (0 = low, 1 = high), used in conjunction with set/getTrigPattMask: Bit 0 = input channel 1 Bit 1 = input channel 2 Bit 2 = input channel 3 (4 channel only) Bit 3 = input channel 4 (4 channel only) Bit 4 = external input Bit 5 = TTLTRG0 (external header) Bit 6 = TTLTRG1 (external header) Bit 7 = TTLTRG2 (external header) Bit 8 = TTLTRG3 (external header) Bit 9 = TTLTRG4 (external header) Bit 10 = TTLTRG5 (external header) Bit 11 = TTLTRG6 (external header) Bit 12 = TTLTRG7 (external header)
setTrigVidStan getTrigVidStan	mbbo mbbi	string	Video trigger standard: “PAL” = PAL, “NTSC” = NTSC, “SEC” = SECAM
setTrigVidField getTrigVidField	longout longin	1 or 2	Video trigger field: 1 = field 1, 2 = field 2
setTrigVidLine getTrigVidLine	longout longin	1 to 625	Video trigger line, depends upon TrigVidStan and TrigVidField: For NTSC: field 1, line = 1 to 263, field 2, line = 1 to 262 For PAL or SECAM; field 1, line = 1 to 313 filed 2, line = 314 to 625

Arm Process Variables

The following table lists the PVs for the arm settings that configure the arm-qualified trigger for the waveform acquisition synchronization. When using the input channels or external input as the arm source, the arm threshold level is set using the corresponding trigger PV.

Process Variable	Type	Values	Description
ArmState	bi	0 or 1	The instrument arm state.
setArmSource getArmSource	mbbo mbbi	string	Arm source: “IMM” = immediate (bypass arm) “TTLT0” = TTLTRG0 (external header) “TTLT1” = TTLTRG1 (external header) “TTLT2” = TTLTRG2 (external header) “TTLT3” = TTLTRG3 (external header) “TTLT4” = TTLTRG4 (external header) “TTLT5” = TTLTRG5 (external header) “TTLT6” = TTLTRG6 (external header) “TTLT7” = TTLTRG7 (external header) “INP1” = input channel 1 “INP2” = input channel 2 “INP3” = input channel 3 (4 channel only) “INP4” = input channel 4 (4 channel only) “EXT” = external input “MAN” = software only “PATT” = pattern
setArmPolarity getArmPolarity	mbbo mbbi	string	Trigger slope or polarity: “NEG” = negative, “POS” = positive

Output Process Variables

The following table lists the PVs for the output settings that configure the signal outputs. These outputs include the external output on the front panel and TTLTRG*N* outputs on the external trigger header. The trigger output PVs are duplicated for each TTLTRG*N* output. In the following table, *N* is substituted with the appropriate trigger number (for example: OutTrig1Enable).

Process Variable	Type	Values	Description
setOutExtEnable getOutExtEnable	bo bi	0 or 1	External output enable: 0 = disable, 1 = enable
setOutExtPolarity getOutExtPolarity	mbbo mbbi	string	Trigger slope or polarity: “NEG” = negative, “POS” = positive
setOutExtSource getOutExtSource	mbbo mbbi	string	External output source: “ARM” = arm event “TRIG” = trigger complete event “ATR” = trigger A event “BTR” = trigger B event “CAPT” = capture complete event “OPC” = operation complete event “CONS” = constant state “PULS” = programmable pulse “REF” = 10 MHz reference “CLOC” = programmable clock “LIM” = passed limit test “MSS” = master summary status event “PROB” = probe compensation output
setOutExtPulsePer getOutExtPulsePer	ao ai	26.67e-9 to 100	External output programmable pulse or clock period in seconds: 26.667 ns to 100 seconds
setOutputEventTime getOutputEventTime	ao ai	50e-9 to 163e-3	External output programmable event driven pulse width in seconds: 50 ns to 163 ms
setOutTrig*N*Enable getOutTrig*N*Enable	bo bi	0 or 1	Output enable for selected trigger: 0 = disable, 1 = enable
setOutTrig*N*Polarity getOutTrig*N*Polarity	mbbo mbbi	string	Trigger slope or polarity: “NEG” = negative, “POS” = positive
setOutTrig*N*Source getOutTrig*N*Source	mbbo mbbi	string	Output source for selected trigger: “ARM = arm event “TRIG” = trigger complete event “ATR” = trigger A event “BTR” = trigger B event “CAPT” = capture complete event “OPC” = operation complete event “CONS” = constant state “MSS” = master summary status event

Calculate Process Variables

The following table lists the PVs for the calculate channel settings that configure the waveform calculations or math. These PVs are duplicated for each calculate channel. In the following table, *N* is substituted with the appropriate channel number from 1 to 4 (for example: Calc1Enable). Range and offset variables need to be updated after a capture.

Process Variable	Type	Values	Description
setCalc*N*Enable getCalc*N*Enable	bo bi	0 or 1	Channel enable for selected calculate channel: 0 = disable, 1 = enable
setCalc*N*Range getCalc*N*Range	ao ai	-inf to +inf	Range in Vpp for selected calculate channel, Calculate range is set automatically when the calculate operation is first selected
setCalc*N*Offset getCalc*N*Offset	ao ai	-inf to +inf	Offset in Volts for selected calculate channel, Calculate offset is set automatically when the calculate operation is first selected
setCalc*N*Op getCalc*N*Op	mbbo mbbi	string	Operation for selected calculate channel: “ADD” = add “AVAL” = absolute value “COPY” = copy “DER” = derivative “INT” = integrate “INV” = invert “MULT” = multiply “SUBT” = subtract “LIM” = limit test “FTR” = FFT frequency transform “TTR” = time transform(smooth) “MEAS” = measurement statistics “HIST” = histogram “NOOP” = no operation; static channel
setCalc*N*Source1 getCalc*N*Source1	mbbo mbbi	string	Source 1 for selected calculate channel: “INP1” = input channel 1 “INP2” = input channel 2 “INP3” = input channel 3 (4 channel only) “INP4” = input channel 4 (4 channel only) “CALC1” = calculate channel 1 “CALC2” = calculate channel 2 “CALC3” = calculate channel 3 “CALC4” = calculate channel 4 “REF1” = reference channel 1 “REF2” = reference channel 2 “REF3” = reference channel 3 “REF4” = reference channel 4

Process Variable	Type	Values	Description
setCalc*N*Source2 getCalc*N*Source2	mbbo mbbi	string	Source 2 for selected calculate channel, Only required for add, subtract, and multiply calculate operations: “INP1” = input channel 1 “INP2” = input channel 2 “INP3” = input channel 3 (4 channel only) “INP4” = input channel 4 (4 channel only) “CALC1” = calculate channel 1 “CALC2” = calculate channel 2 “CALC3” = calculate channel 3 “CALC4” = calculate channel 4 “REF1” = reference channel 1 “REF2” = reference channel 2 “REF3” = reference channel 3 “REF4” = reference channel 4

Advanced Calculate Process Variables

The following table lists the PVs for the advanced calculate channel settings that configure specific calculations or math. These PVs are duplicated for each calculate channel. In the following table, *N* is substituted with the appropriate channel number from 1 to 4 (for example: Calc1FFTWin).

Process Variable	Type	Values	Description
setCalc*N*FFTWin getCalc*N*FFTWin	mbbo mbbi	string	FFT window for selected calculate channel: “RECT” = rectangular “HAMM” = hamming “HANN” = hanning “BLAC” = blackman “FLAT” = flattop
setCalc*N*FFTFormat getCalc*N*FFTFormat	mbbo mbbi	string	FFT format for selected calculate channel: “MLIN” = magnitude linear “MLOG” = magnitude logarithmic “PHAS” = phase “REAL” = real “IMAG” = imaginary
setCalc*N*FiltPoints getCalc*N*FiltPoints	longout longin	2 to 40	Digital lowpass IIR filter smoothing points for selected calculate channel
setCalc*N*LimitMeas getCalc*N*LimitMeas	stringout stringin	string	The measurement to check during a limit test. “MASK” = Mask Test “AC” = AC RMS “AMPL” = Amplitude “AVER” = Average “CAV” = Cycle Average “CFR” = Cycle Frequency “CPER” = Cycle Period “CRMS” = Cycle RMS “DC” = DC RMS

Process Variable	Type	Values	Description
			"ENOB" = Effective number of bits (FFT only) "FEDG" = Number of Falling Edges "FOV" = Falling edge overshoot "FPR" = Falling edge preshoot "FREQ" = Frequency "FTCR" = Falling edge crossing time "FTIM" = Fall time "HIGH" = High "LOW" = Low "MAX" = Maximum "MIN" = Minimum "MID" = Mid "NDUT" = Negative duty cycle "NWID" = Negative pulse width "PDUT" = Positive duty cycle "PER" = Period "PHAS" = Phase "PTP" = Peak-to-peak "PWID" = Positive width "REDG" = Number of Rising Edges "ROV" = Rising edge overshoot "RPR" = Rising edge preshoot "RTCR" = Rising edge crossing time "RTIM" = Rise time "SDEV" = Standard deviation "SFDR" = Spurious-free dynamic range (FFT only) "SNDR" = Signal-to-noise+distortion (FFT only) "SNR" = Signal-to-noise (FFT only) "THD" = Total harmonic distortion (FFT only) "TMAX" = Time of maximum "TMIN" = Time of minimum
setCalc*N*LimitLower getCalc*N*LimitLower	ao ai	-inf to +inf	The lower boundary for limit tests
setCalc*N*LimitUpper getCalc*N*LimitUpper	ao ai	-inf to +inf	The upper boundary for limit tests
setCalcLimitNumber getCalcLimitNumber	longout longin	0 to 2147483647	Sets or gets the number of limit test failures until limit test aborts. 0 runs the limit test until aborted, n stops the limit test upon the n^{th} failure.
Calc*N*LimitClear	bo	0 or 1	Clears limit test statistics; this will also reset the count for limit number abort. A transition from 0 to 1 causes the clear, after which the value returns to 0

Process Variable	Type	Values	Description
setCalc*N*MaskLower getCalc*N*MaskLower	mbbo mbbi	string	The lower boundary for mask tests: “CALC1” = calculate channel 1 “CALC2” = calculate channel 2 “CALC3” = calculate channel 3 “CALC4” = calculate channel 4 “REF1” = reference channel 1 “REF2” = reference channel 2 “REF3” = reference channel 3 “REF4” = reference channel 4
setCalc*N*MaskUpper getCalc*N*MaskUpper	mbbo mbbi	string	The upper boundary for mask tests: “CALC1” = calculate channel 1 “CALC2” = calculate channel 2 “CALC3” = calculate channel 3 “CALC4” = calculate channel 4 “REF1” = reference channel 1 “REF2” = reference channel 2 “REF3” = reference channel 3 “REF4” = reference channel 4
Calc*N*MaskGen	bo	0 or 1	Generates reference waveforms for mask testing based on calculation source.
setCalcMaskVertOffset getCalcMaskVertOffset	ao ai	0 to full scale	Sets the vertical offset in volts used in generate reference masks
setCalcMaskHorzOffset getCalcMaskHorzOffset	ao ai	0 to sweep time	Sets the horizontal offset in seconds used in generate reference masks
setCalc*N*MeasType getCalc*N*MeasType	stringout stringin	string	The type of measurement to store in a calculation channel: “AC” = AC RMS “AMPL” = Amplitude “AVER” = Average “CAV” = Cycle average “CFR” = Cycle frequency “CPER” = Cycle period “CRMS” = Cycle RMS “DC” = DC RMS “ENOB” = Effective number of bits (FFT only) “FEDG” = Number of Falling Edges “FOV” = Falling edge overshoot “FPR” = Falling edge preshoot “FREQ” = Frequency “FTCR” = Falling edge crossing time “FTIM” = Fall time “HIGH” = High “LOW” = Low “MAX” = Maximum “MIN” = Minimum “MID” = Mid “NDUT” = Negative duty cycle “NWID” = Negative pulse width “PDUT” = Positive duty cycle “PER” = Period

Process Variable	Type	Values	Description
			"PHAS" = Phase "PTP" = Peak-to-peak "PWID" = Positive width "REDG" = Number of Rising Edges "ROV" = Rising edge overshoot "RPR" = Rising edge preshoot "RTCR" = Rising edge crossing time "RTIM" = Rise time "SDEV" = Standard deviation "SFDR" = Spurious-free dynamic range (FFT only) "SNDR" = Signal-to-noise+distortion (FFT only) "SNR" = Signal-to-noise (FFT only) "THD" = Total harmonic distortion (FFT only) "TMAX" = Time of maximum "TMIN" = Time of minimum
Calc*N*MeasClear	bo	0 or 1	Clears measurement history. A transition from 0 to 1 causes the clear, after which the value returns to 0
Calc*N*LimRepMax	ai	-inf to +inf	Gets the maximum value of the waveform reported from the calculation limit test.
Calc*N*LimRepMin	ai	-inf to +inf	Gets the minimum value of the waveform reported from the calculation limit test.
Calc*N*LimRepAvg	ai	-inf to +inf	Gets the average value of the waveform reported from the calculation limit test.
Calc*N*LimRepSdev	ai	-inf to +inf	Gets the standard deviation of the waveform reported from the calculation limit test.
Calc*N*LimRepLast	ai	-inf to +inf	Gets the last value of the waveform reported from the calculation limit test.
Calc*N*LimRepCount	longin	0 to 2147483647	Gets the number of waveforms captured in the calculation limit test.
Calc*N*LimRepFail	longin	0 to 2147483647	Gets the number of waveform capture failures in the calculation limit test.

Reference Process Variables

The following table lists the PVs for the reference channel settings that configure the waveform storage channels. These PVs are duplicated for each reference channel. In the following table, *N* is substituted with the appropriate channel number from 1 to 4 (for example: Ref1Enable).

Process Variable	Type	Values	Description
setRef*N*Range	ao	-inf to +inf	Range in Vpp for selected reference channel,

Process Variable	Type	Values	Description
getRef*N*Range	ai		Reference range is set automatically when the reference waveform is stored
setRef*N*Offset getRef*N*Offset	ao ai	-inf to +inf	Offset in Volts for selected reference channel, Reference offset is set automatically when the reference waveform is stored
Ref*N*Store	mbbo	string	Store a waveform from the selected channel to the reference channel. Source channels: “INP1” = input channel 1 “INP2” = input channel 2 “INP3” = input channel 3 (4 channel only) “INP4” = input channel 4 (4 channel only) “CALC1” = calculate channel 1 “CALC2” = calculate channel 2 “CALC3” = calculate channel 3 “CALC4” = calculate channel 4 “REF1” = reference channel 1 “REF2” = reference channel 2 “REF3” = reference channel 3 “REF4” = reference channel 4
Ref*N*Upload	aao	Array	Upload a signed short integer waveform data in codes into the selected reference channel.
Ref*N*ScaledUpload	aao	Array	Upload a floating point scaled waveform data array in Volts into the selected scaled reference channel.

Measure Process Variables

The following table lists the PVs for the Measure functionality. Up to 32 measurements are performed automatically after each waveform acquisition, and the results updated in the corresponding measurement PVs. In the following table, *N* is substituted with the appropriate measurement number from 1 to 32 (for example: Meas1Enable). The corresponding measurements can also be viewed through ZScope™ or the ztscopeM drivers as four lists of eight measurements.

Process Variable	Type	Values	Description
setMeas*N*Source getMeas*N*Source	mbbo mbbi	string	Source for selected measurement number: “INP1” = input channel 1 “INP2” = input channel 2 “INP3” = input channel 3 (4 channel only) “INP4” = input channel 4 (4 channel only) “CALC1” = calculate channel 1 “CALC2” = calculate channel 2 “CALC3” = calculate channel 3 “CALC4” = calculate channel 4 “REF1” = reference channel 1 “REF2” = reference channel 2 “REF3” = reference channel 3 “REF4” = reference channel 4

Process Variable	Type	Values	Description
setMeas*N*Type getMeas*N*Type	stringout stringin	string	Type for selected measurement number: “NONE” = none selected “AC” = AC RMS “AMPL” = Amplitude “AVER” = Average “CAV” = Cycle average “CFR” = Cycle frequency “CPER” = Cycle period “CRMS” = Cycle RMS “DC” = DC RMS “ENOB” = Effective number of bits (FFT only) “FEDG” = Number of Falling Edges “FOV” = Falling edge overshoot “FPR” = Falling edge preshoot “FREQ” = Frequency “FTCR” = Falling edge crossing time “FTIM” = Fall time “HIGH” = High “LOW” = Low “MAX” = Maximum “MIN” = Minimum “MID” = Mid “NDUT” = Negative duty cycle “NWID” = Negative pulse width “PDUT” = Positive duty cycle “PER” = Period “PHAS” = Phase “PTP” = Peak-to-peak “PWID” = Positive width “REDG” = Number of Rising Edges “ROV” = Rising edge overshoot “RPR” = Rising edge preshoot “RTCR” = Rising edge crossing time “RTIM” = Rise time “SDEV” = Standard deviation “SFDR” = Spurious-free dynamic range (FFT only) “SNDR” = Signal-to-noise+distortion (FFT only) “SNR” = Signal-to-noise (FFT only) “THD” = Total harmonic distortion (FFT only) “TMAX” = Time of maximum “TMIN” = Time of minimum
Meas*N*Result	ai	-Inf to +Inf	Measurement result in appropriate units
Meas*N*Status	longin	0000_{16} to $0F03_{16}$	Status of selected measurement number: 0000_{16} = measurement valid Bit 0 = invalid waveform Bit 1 = edge not found Bit 8 = over-voltage maximum Bit 9 = over-voltage minimum Bit 10 = low signal level Bit 11 = measurement command invalid

Operate Process Variables

The following table lists the PVs for the instrument operation involving the controls for waveform acquisition.

Process Variable	Type	Values	Description
setOpInitCont getOpInitCont	bo bi	0 or 1	The instrument's continuous acquisition mode: 0 = single capture, 1 = continuous capture.
OpInitiate	bo	0 or 1	Initiate waveform acquisition.
OpAbort	bo	0 or 1	Aborts waveform acquisition.
OpForceCap	bo	0 or 1	Force waveform capture (all triggers): 0 = no operation, 1 = force capture A transition from 0 to 1 causes the waveform acquisition to complete, after which the value returns to 0
OpArm	bo	0 or 1	Manual arm for software arm source: 0 = disarm, 1 = arm This only applies when the arm source is set to software
OpAutoSetup	bo	0 or 1	Automatically setup range, offset, trigger level for enabled channels: 0 = no operation, 1 = force auto setup A transition from 0 to 1 causes the auto setup, after which the value returns to 0

Waveform Process Variables

The following table lists the PVs for the acquired waveforms. In the following table, *N* is substituted with the appropriate channel number (for example: Inp1Wave). The ZT4x11 has two input channels and the ZT4x12 has four input channels. There are four calculate channels and four reference channels. Waveform PVs can only be read (downloaded) from the instrument. . Due to the hardware architecture it is not possible to read waveforms while initiated continuously.

The size of the waveform allowed to be transmitted via channel access is limited by EPICS_CA_MAX_ARRAY_BYTES. To avoid waveforms that are larger than that, set the number of waveform elements allowed in the PV file. Scaled Wave and Time PVs are doubles, so they require 8 bytes per point, and Waves are integers, so they require 4 bytes per point. In the PV file set the maximum values as required and upload to the instrument. Default values are 4096 and 2048, based on the default EPICS_CA_MAX_ARRAY_BYTES of 16384.

Process Variable	Type	Values	Description
Inp*N*WavePoints	longin	10 to 65535	The number of actual data points stored in Inp*N*Wave, Inp*N*ScaledWave and Inp*N*ScaledTime.

Process Variable	Type	Values	Description
Inp*N*Wave	waveform	-2147483648 to 2147483647	Signed short integer waveform data in codes for selected input channel
Inp*N*ScaledWave	waveform	-inf to +inf	Floating point scaled waveform data in Volts for selected input channel
Inp*N*ScaledTime	waveform	0 to 100	Floating point scaled time (X-axis) waveform data in seconds for selected input channel
Inp*N*Timestamp	ai	0.0 to 0.9999999	Relative timestamp in seconds for acquired waveform, 1 second wrap period, 100 ns resolution
Calc*N*WavePoints	longin	10 to 65535	The number of actual data points stored in Calc*N*Wave, Calc*N*ScaledWave and Calc*N*ScaledTime.
Calc*N*Wave	waveform	-2147483648 to 2147483647	Signed short integer waveform data in codes for selected calculate channel
Calc*N*ScaledWave	waveform	-inf to +inf	Floating point scaled waveform data in Volts for selected calculate channel
Calc*N*ScaledTime	waveform	-inf to +inf	Floating point scaled time (X-axis) waveform data for selected calculate channel
Ref*N*WavePoints	longin	10 to 32768	The number of actual data points stored in Ref*N*Wave, Ref*N*ScaledWave and Ref*N*ScaledTime.
Ref*N*Wave	waveform	-2147483648 to 2147483647	Signed short integer waveform data in codes for selected reference channel
Ref*N*ScaledWave	waveform	-inf to +inf	Floating point scaled waveform data in Volts for selected reference channel
Ref*N*ScaledTime	waveform	-inf to +inf	Floating point scaled time (X-axis) waveform data for selected reference channel

Utility Process Variables

The following table lists the PVs for the miscellaneous instrument utilities. For the save and recall operations in the following table, *N* is substituted with the appropriate state number from 1 to 8 (for example: UtilSave1).

Process Variable	Type	Values	Description
UtilID	stringin	ID string	Identification string of following format: “ZTEC,model_number,serial_number,version”
UtilReset	bo	0 or 1	Instrument reset: 0 = no operation, 1 = reset A transition from 0 to 1 causes the instrument reset, after which the value returns to 0
UtilSave	longout	1 to 31	Instrument state save: Causes the instrument state to be saved to one of thirty one locations in non-volatile memory.

Process Variable	Type	Values	Description
UtilRecall	longout	1 to 31	Instrument state recall: Causes the instrument state to be recalled from one of thirty one locations in non-volatile memory.
UtilStatus	longin	00_{16} to FF_{16}	Instrument status: 1 = latched event Bit 2 = error log not empty Bit 3 = questionable event Bit 4 = message available Bit 5 = standard event Bit 6 = master summary status Bit 7 = operation event
UtilFreqFault	longin	00_{16} to FF_{16}	Frequency fault status results: 1 = latched event Bit 0 = sample clock unlocked Bit 2 = memory 1-2 clock unlocked Bit 3 = memory 3-4 clock unlocked(4 channel only)
UtilTestResult	longin	0000_{16} to $FFFF_{16}$	Test status results: 1 = latched event Bit 0 = baseboard register test failed Bit 1 = n/a Bit 2 = baseboard ROM test failed Bit 3 = n/a Bit 4 = reference oscillator test failed Bit 5 = DRAM test failed Bit 6 = flash memory test failed Bit 8 = digitizer 1 test failed Bit 9 = digitizer 2 test failed (4 channel only)
UtilOperation	longin	0000_{16} to $FFFF_{16}$	Operation status results: 1 = latched event Bit 0 = calibrating Bit 1 = settling Bit 2 = ranging Bit 3 = sweeping Bit 4 = measuring Bit 5 = waiting for trigger Bit 6 = waiting for arm Bit 8 = trigger event Bit 9 = data capture event Bit 10 = limit test event Bit 11 = auto download event

Process Variable	Type	Values	Description
UtilStandard	longin	00_{16} to FF_{16}	Standard status events: 1 = latched event Bit 0 = operation complete Bit 1 = request control Bit 2 = query error Bit 3 = device dependent error Bit 4 = execution error Bit 5 = command error Bit 6 = user request Bit 7 = power on
UtilQuesReg	longin	0000_{16} to $FFFF_{16}$	Questionable status: 1 = latched event Bit 0 = voltage fault Bit 5 = frequency fault Bit 8 = calibration failure Bit 9 = test failure
UtilVoltFault	longin	000_{16} to FFF_{16}	Voltage status faults: 1 = latched event Bit 0 = input 1 overload Bit 1 = input 2 overload Bit 2 = input 3 overload (4 channel only) Bit 3 = input 4 overload (4 channel only) Bit 4 = input 1 overvoltage Bit 5 = input 2 overvoltage Bit 6 = input 3 overvoltage (4 channel only) Bit 7 = input 4 overvoltage (4 channel only) Bit 8 = accessory 1 fault Bit 9 = accessory 2 fault Bit 10 = accessory 3 fault (4 channel only) Bit 11 = accessory 4 fault (4 channel only)
UtilCalResult	longin	00_{16} to FF_{16}	Calibration status results: 1 = latched event Bit 0 = calibration storage failed Bit 1 = offset zero failed Bit 2 = offset scale failed Bit 3 = null balance failed Bit 4 = gain balance failed Bit 5 = trigger level failed
UtilErrCount	longin	0 to 32	Number of errors in error log, 0 = no errors
UtilErrNext	longin	0 to -32767	Current error code, processing clears current error and retrieves next error.
UtilErr	stringin	error string	Error string explaining current error code
UtilTemp	ai	0.0 to 100.0	Internal temperature in degrees C
UtilMem	longin	16,777,216 or 67,108,864	Waveform capture memory in samples per channel, varies by installed memory option
setUtilSelfID getUtilSelfID	bo bi	0 or 1	Instrument HST LED state: 0 = LAN status, 1 = blink Allows instrument identification.

Process Variable	Type	Values	Description
UtilLockState	bi	0 or 1	EPICS lock state 0 = unlocked, full control available through epics interface 1 = locked, epics interface is read-only
UtilSCPISend	stringout	string	Sends the entered SCPI string to the instrument.
UtilSCPIRecv	stringin	string	Receives the current instrument string output buffer.
setOutCoerce getOutCoerce	bo bi	0 or 1	Selects whether set parameters update their values based on instrument coercion.
setRestoreState getRestoreState	longout longin	0 to 31	Sets or gets the location to load the instrument state from on initialization. 0 location restores the instrument to reset conditions. These locations can be saved to using the UtilSave*N* PVs.
UtilRefresh	bo	0 or 1	Manual trigger to update all PVs.

ZTEC Instruments