

ZT500PXI Arbitrary Waveform Generator

Introduction:

This instrument driver provides programming support for ZT500PXI Arbitrary Waveform Generator.

It contains functions for opening, configuring, taking measurements from, and closing the instrument.

Assumptions:

To successfully use this module, the following conditions must be met:

For GPIB instrument drivers:

- the instrument is connected to the GPIB.
- the GPIB address supplied to the initialize function must match the GPIB address of the instrument.

For VXI instrument drivers:

- the instrument is installed in the VXI mainframe and you are using one of the following controller options:
 - Embedded controller
 - MXI
 - MXI2
 - GPIB-VXI
- the logical address supplied to the initialize function must match the logical address of the instrument.

For RS-232 instrument drivers:

- the instrument is connected to the RS-232 interface.
 - the COM port, baud rate, parity, and timeout supplied to the initialize function must match the settings of the instrument.
-

Error and Status Information:

Each function in this instrument driver returns a status code that either indicates success or describes an error or warning condition.

Your program should examine the status code from each call to an instrument driver function to determine if an error occurred.

The general meaning of the status code is as follows:

Value	Meaning
0	Success
Positive Values	Warnings
Negative Values	Errors

The description of each instrument driver function lists possible error codes and their meanings

How To Use This Document:

Use this document as a programming reference manual.
It describes each function in the

ZT500PXI Arbitrary Waveform Generator

instrument. The functions appear in alphabetical order, with a description of the function and its C syntax, a description of each parameter, and a list of possible error codes.

Function Tree Layout:

Class/Panel Name:	Function Name:
Initialize	zt500pxi_initialize
Configure	
Vertical	zt500pxi_conf_vertical
Horizontal	zt500pxi_conf_horizontal
Segment	zt500pxi_conf_segment
Sequence	zt500pxi_conf_sequence
Trigger	
Source	zt500pxi_conf_trig_source
Initiate	zt500pxi_conf_trig_init
Advance	zt500pxi_conf_trig_advance
Data	
Write Segment	zt500pxi_write_segment
Write Sequence	zt500pxi_write_sequence
Operate	
Run	zt500pxi_operate_run
Trigger	zt500pxi_operate_trigger
Utility	
Error	zt500pxi_util_error
Information	zt500pxi_util_info
Close	zt500pxi_close

ZT500PXI Arbitrary Waveform Generator

Instrument driver for the ZT500PXI.

The following functions are in alphabetical order.

zt500pxi_close

```
ViStatus zt500pxi_close (ViSession instrumentSession);
```

Purpose

Closes the VISA communication session opened by Initialize.

Valid Values
ZT500PXI_CONF_HORIZ_REF_LOCAL
ZT500PXI_CONF_HORIZ_REF_PXI

Return Value

This control contains the status code returned by the function call.

zt500pxi_conf_segment

```
ViStatus zt500pxi_conf_segment (ViSession instrumentSession,  
                                unsigned int size, unsigned short loop,  
                                unsigned int *count);
```

Purpose

Configures memory segmentation.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

size

Variable Type unsigned int

Selects maximum data size used within each memory segment.

Valid Values:

4 < size < maximum size in increments of 2

maximum size = 2097152 or 4194304, depending on your memory options.

loop

Variable Type unsigned short

When segment advance mode is non-triggered, selects the number of times each segment will be performed before going to the next segment.

Valid Values:

1 to 65535

count

Variable Type unsigned int (passed by reference)

Returns the number of memory segments. This is determined by dividing the memory size by the segment size.

Valid Values:
2^n

n = 0 to 16

Return Value

This control contains the status code returned by the function call.

zt500pxi_conf_sequence

```
ViStatus zt500pxi_conf_sequence (ViSession instrumentSession,  
                                unsigned short loop);
```

Purpose

Configures sequencing.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

loop

Variable Type unsigned short

Sets the sequence loop mode. Single causes the instrument to step through the sequence table once. Continuous causes the instrument to continuously repeat the entire equence table.

Valid Values:

ZT500PXI_CONF_SEQ_LOOP_SINGLE
ZT500PXI_CONF_SEQ_LOOP_CONT

Return Value

This control contains the status code returned by the function call.

zt500pxi_conf_trig_advance

```
ViStatus zt500pxi_conf_trig_advance (ViSession instrumentSession,  
                                      unsigned short advance);
```

Purpose

Sets the instrument up to trigger the sequence or to trigger each segment.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

advance

Variable Type unsigned short

Sets the sequence advance mode. Auto sequence advance automatically steps through the loaded segments. Triggered sequence advance stops and waits for a trigger at the end of each segment.

Valid Values:

ZT500PXI_CONF_TRG_ADV_AUTO
ZT500PXI_CONF_TRG_ADV_TRG

Return Value

This control contains the status code returned by the function call.

zt500pxi_conf_trig_init

ViStatus zt500pxi_conf_trig_init (ViSession instrumentSession,
 unsigned short mode,
 unsigned int delay);

Purpose

Sets up the trigger initiate or arm.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

mode

Variable Type unsigned short

Initiates the waveform sequencer response to trigger activity.

Valid Values:

ZT500PXI_CONF_TRG_MODE_OFF
ZT500PXI_CONF_TRG_MODE_NORMAL
ZT500PXI_CONF_TRG_MODE_GATED
ZT500PXI_CONF_TRG_MODE_DELAY

delay

Variable Type unsigned int

In delay trigger mode, selects the amount of time the output will be delayed following an active trigger event.

Valid Values:

The variable is an unsigned 32 bit number, with entry 0 as a 0, entry 1 as the MSW and entry 2 as the LSW. The delay time must be entered as a hexadecimal number. For example, 1000 usec delay requires the user to input 000016 in entry 1 and 03E816 in entry 2.

Return Value

This control contains the status code returned by the function call.

zt500pxi_conf_trig_source

```
ViStatus zt500pxi_conf_trig_source (ViSession instrumentSession,  
                                     unsigned short source,  
                                     unsigned short slope,  
                                     unsigned short EXTOut,  
                                     unsigned short PXI0Out);
```

Purpose

Sets the trigger source.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

source

Variable Type unsigned short

Sets the source for the trigger event for the instrument.

Valid Values:

ZT500PXI_CONF_TRG_SOURCE_SW
ZT500PXI_CONF_TRG_SOURCE_PXI0
ZT500PXI_CONF_TRG_SOURCE_PXI1
ZT500PXI_CONF_TRG_SOURCE_PXI2
ZT500PXI_CONF_TRG_SOURCE_PXI3
ZT500PXI_CONF_TRG_SOURCE_PXI4
ZT500PXI_CONF_TRG_SOURCE_PXI5
ZT500PXI_CONF_TRG_SOURCE_PXI6
ZT500PXI_CONF_TRG_SOURCE_PXI7

ZT500PXI_CONF_TRG_SOURCE_STAR
ZT500PXI_CONF_TRG_SOURCE_EXT

slope

Variable Type unsigned short

Sets the slope of the selected trigger for the instrument to rising edge or falling edge.

Valid Values:

ZT500PXI_CONF_TRG_SLOPE_RISE
ZT500PXI_CONF_TRG_SLOPE_FALL

EXTOut

Variable Type unsigned short

When trigger source is not EXT, enabling the EXT out routes the trigger signal to the front panel EXT connector.

Valid Values:

ZT500PXI_CONF_TRG_EXT_DISABLE
ZT500PXI_CONF_TRG_EXT_ENABLE

PXI0Out

Variable Type unsigned short

When trigger source is not PXI0, enabling the PXI0 out routes the trigger signal to the PXI0 backplane connector.

Valid Values:

ZT500PXI_CONF_TRG_PXI0_DISABLE
ZT500PXI_CONF_TRG_PXI0_ENABLE

Return Value

This control contains the status code returned by the function call.

zt500pxi_conf_vertical

```
ViStatus zt500pxi_conf_vertical (ViSession instrumentSession,  
                                unsigned short amplitude,  
                                unsigned short bandwidth);
```

Purpose

Set up the vertical parameters.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

amplitude

Variable Type unsigned short

Selects the output voltage range in millivolts. When driving a low impedance load (50 or 75 ohms), this command sets the peak-to-peak voltage. When driving a high impedance load, this command sets the peak voltage.

An example of this would be setting the output voltage to 1000, would result in a 0.5V_{peak} output (1V_{peak-to-peak}) for a 50 W load and a 1V_{peak} output (±1V or 2V_{peak-to-peak}) for a high impedance load.

Valid Values:

The parameter is specified in millivolts and may vary from 100 (0.1V) to 3500 (3.5V).

bandwidth

Variable Type unsigned short

Sets the output filter on the selected channel. The user can decide whether to use the 10 MHz filter, bypass the filter, or let the unit decide (automatic filter select mode). The filters are 5 pole, lowpass Bessel (linear phase) filters with varying cutoff frequencies.

Valid Values:

ZT500PXI_CONF_VERT_BW_BYPASS = 0x0000
ZT500PXI_CONF_VERT_BW_100MHZ = 0x0001
ZT500PXI_CONF_VERT_BW_50MHZ = 0x0002
ZT500PXI_CONF_VERT_BW_25MHZ = 0x0003

Return Value

This control contains the status code returned by the function call.

zt500pxi_initialize

```
ViStatus zt500pxi_initialize (ViRsrc resourceName,  
                             unsigned short IDQuery,  
                             unsigned short resetDevice,  
                             ViPSession instrumentSession);
```

Purpose

The initialize function sets up the visa session and establishes communications with the instrument.

Parameter List

resourceName

Variable Type ViRsrc

Specifies the interface and address of the device that is to be initialized (Instrument Descriptor). The exact grammar to be used in this input is:

PXI[board]::logical address::INSTR

IDQuery

Variable Type unsigned short

Selects whether to do a valid instrument check. It checks to see if the instrument is a ZT500 or not.

Do Query = 1

Skip Query = 0

resetDevice

Variable Type unsigned short

The reset parameter is ignored on the ZT500PXI. It is included for a placeholder only.

Controls whether an instrument reset is performed upon initialization. A reset will return all registers to their default condition.

off - 0 - Don't Reset

on - 1 - Reset Device

instrumentSession

Variable Type ViSession (passed by reference)

This returns an Instrument Session handle that is used in all subsequent function calls to differentiate between different sessions of this instrument driver. Each time this function is invoked a Unique Session is opened. It is possible to have more than one session open for the same resource.

Return Value

This control contains the status code returned by the function call.

zt500pxi_operate_run

ViStatus zt500pxi_operate_run (ViSession instrumentSession,

```
unsigned short run);
```

Purpose

Controls waveform generation.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

run

Variable Type unsigned short

Enables or disables waveform generation.

Valid Values:

ZT500PXI_OP_RUN_ENABLE

ZT500PXI_OP_RUN_DISABLE

Return Value

This control contains the status code returned by the function call.

zt500pxi_operate_trigger

```
ViStatus zt500pxi_operate_trigger (ViSession instrumentSession);
```

Purpose

Temporarily overrides the current trigger source setting and forces a trigger event.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

Return Value

This control contains the status code returned by the function call.

zt500pxi_util_error

```
ViStatus zt500pxi_util_error (ViSession instrumentSession,
```

```
unsigned short *error);
```

Purpose

Returns the instrument error code.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

error

Variable Type unsigned short (passed by reference)

The next entry from the unit error log. If no errors are pending, an 0xFFFF is reported.

Error

0504 Host response buffer overrun
0940 EEPROM Controller EEPROM timeout during read
0941 EEPROM Controller EEPROM timeout during write
0D40 Mfg Info invalid unit mfg info
0D41 Mfg Info invalid baseboard mfg info
0D42 Mfg Info invalid TRSE mfg info
1140 Memory invalid memory page
150A Command Processor invalid command
1540 Command Processor invalid mfg info device during read
1541 Command Processor invalid mfg info device during write
1542 Command Processor command locked out
1543 Command Processor invalid parameter
1548 Command Processor invalid command SW version, Unit P/N or Unit serial #
15C0 Command Processor invalid command to the arbitrary waveform generator
15C1 Command Processor query not supported
15C2 Command Processor only query supported
1904 Flash Flash programming or verify failed
1907 Flash command locked out
1980 Flash timeout while erasing
1981 Flash timeout while writing
1982 Flash flash verify failed
1D40 AWG Controller Sequencer PLL unlocked
1D41 AWG Controller Signal DAC PLL unlocked
1D80 AWG Controller invalid sample rate to ZT500SM
1DC0 AWG Controller invalid run/stop code
1DC1 AWG Controller invalid bandwidth setting
1DC2 AWG Controller invalid segment count
1DC3 AWG Controller invalid segment length
1DC4 AWG Controller invalid sample rate
1DC5 AWG Controller invalid voltage range
1DC6 AWG Controller invalid Range DAC setting
1DC7 AWG Controller invalid Offset DAC setting

1DC8 AWG Controller invalid advance mode setting
1DC9 AWG Controller invalid trigger mode
1DCA AWG Controller invalid loop mode setting
1DCB AWG Controller invalid memory type select
1DCC AWG Controller invalid memory page
1DCD AWG Controller invalid segment or segment address
21C1 Trigger Controller invalid PXI trigger control
21C2 Trigger Controller invalid retigger mode setting
21C3 Trigger Controller invalid set run trigger control
21CC Trigger Controller invalid front panel trigger control

Return Value

This control contains the status code returned by the function call.

zt500pxi_util_info

```
ViStatus zt500pxi_util_info (ViSession instrumentSession,  
                             char IDString[],  
                             unsigned short *serialNumber);
```

Purpose

Returns the ID string and driver revision.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

IDString

Variable Type char []

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,ZT500PXI,S/N nnn,Version n.nn"

serialNumber

Variable Type unsigned short (passed by reference)

Returns the instrument serial number.

Return Value

This control contains the status code returned by the function call.

zt500pxi_write_segment

```
ViStatus zt500pxi_write_segment (ViSession instrumentSession,  
                                unsigned short segmentNumber,  
                                unsigned int dataLength, short data[]);
```

Purpose

Downloads a waveform array into the selected segment.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

segmentNumber

Variable Type unsigned short

Selects the segment number to write to.

Valid Values:
0 to 65535

dataLength

Variable Type unsigned int

Selects the length of the waveform array to download.

Valid Values:
Should be less than or equal to the segment length set in the Configure Segment function.

data

Variable Type short []

Passes in the waveform array to download to the instrument.

Return Value

This control contains the status code returned by the function call.

zt500pxi_write_sequence

```
ViStatus zt500pxi_write_sequence (ViSession instrumentSession,  
                                  unsigned short dataLength,  
                                  short data[]);
```

Purpose

Downloads the sequence table to the instrument.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between different sessions of this instrument driver.

dataLength

Variable Type unsigned short

Sets the length of the sequence with loaded pattern/waveform segments in the sequence table.

Valid Values:

1 to 65535

data

Variable Type short []

Passes in the sequence array to download to the instrument.

Return Value

This control contains the status code returned by the function call.