

Introduction:

This instrument driver provides programming support for ZTEC Instruments PXI Common.

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

ZTEC Instruments PXI Common

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	zt_pxi_init
Open	zt_pxi_open
ID Query	zt_pxi_id_query
Command	zt_pxi_command
Query	zt_pxi_query
Block Write	zt_pxi_block_write
Block Read	zt_pxi_block_read
Error Message	zt_pxi_error_message
Driver Revision	zt_pxi_revision_query
Send	zt_pxi_send
Receive	zt_pxi_receive
Close	zt_pxi_close

ZTEC Instruments PXI Common

This instrument module provides programming support for ZTEC PXI instruments.

The following functions are in alphabetical order.

zt_pxi_block_read

```
ViStatus zt_pxi_block_read (ViSession instrumentSession,  
                           unsigned short address, unsigned short size,  
                           unsigned short data[]);
```

Purpose

This function reads a block of data from the instrument.

Parameter List

instrumentSession

Variable Type ViSession

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

address

Variable Type unsigned short

This control sets the start address for the block transfer. This block data start address is typically ZT_PXI_BLOCK_BUFFER (0x4085).

size

Variable Type unsigned short

This control sets the size of the block data to write. The maximum size is ZT_PXI_BUFFER_SIZE (0x800).

data

Variable Type unsigned short []

This control sets the array for the block data read from the instrument.

Return Value

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

zt_pxi_block_write

```
ViStatus zt_pxi_block_write (ViSession instrumentSession,  
                             unsigned short address,  
                             unsigned short size,  
                             unsigned short data[]);
```

Purpose

This function writes a block of data to the instrument.

Parameter List

instrumentSession

Variable Type ViSession

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

address

Variable Type unsigned short

This control sets the start address for the block transfer. This block data start address is typically ZT_PXI_BLOCK_BUFFER (0x4085).

size

Variable Type unsigned short

This control sets the size of the block data to write. The maximum size is ZT_PXI_BUFFER_SIZE (0x800).

data

Variable Type unsigned short []

This control sets the array for the block data sent to the instrument.

Return Value

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

zt_pxi_close

```
ViStatus zt_pxi_close (ViSession instrumentSession);
```

Purpose

This function closes the instrument session, removes the entry from the resource manager, and frees memory.

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.

zt_pxi_command

```
ViStatus zt_pxi_command (ViSession instrumentSession,  
                        unsigned short command,  
                        unsigned short numberParameters,  
                        unsigned short parameters[],  
                        unsigned short *response);
```

Purpose

This function sends a command to the instrument, including all command parameters. The command handshaking process ensures that the command is accepted and a response is returned.

Parameter List

instrumentSession

Variable Type ViSession

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

command

Variable Type unsigned short

This control sets the command code to send to the instrument.

numberParameters

Variable Type unsigned short

This control sets the number of parameters for the command sent to the instrument.

parameters

Variable Type unsigned short []

This control sets the parameter array for the command sent to the instrument.

response

Variable Type unsigned short (passed by reference)

This pointer provides the address for the returned command response.

Return Value

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

zt_pxi_error_message

```
ViStatus zt_pxi_error_message (ViSession instrumentSession,  
                               ViStatus statusCode,  
                               char errorMessage[]);
```

Purpose

This function interprets a status code and returns the corresponding error message.

Parameter List

instrumentSession

Variable Type ViSession

This Instrument Session handle is used to differentiate between

different sessions of this instrument driver.

statusCode

Variable Type ViStatus

Status value to interpret.

errorMessage

Variable Type char []

Error message interpreted from the status code.

Return Value

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

zt_pxi_id_query

```
ViStatus zt_pxi_id_query (ViSession instrumentSession, char queryCode[],  
                        unsigned short manufacturerID,  
                        unsigned short modelCode, char IDString[]);
```

Purpose

This Function queries the instrument and closed the session if its the incorrect instrument.

Parameter List

instrumentSession

Variable Type ViSession

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

queryCode

Variable Type char []

This control is the command to query the instrument for its ID string.

manufacturerID

Variable Type unsigned short

This control is the expected manufacturer id.

modelCode

Variable Type unsigned short

This control is the expected model code of the instrument.

IDString

Variable Type char []

This control is the expected return of the ID query command. If the returned string is longer than the passed string, the rest is ignored.

Return Value

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

zt_pxi_init

```
ViStatus zt_pxi_init (ViSession instrumentSession);
```

Purpose

This function initializes the PCI bridge.

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.

zt_pxi_open

```
ViStatus zt_pxi_open (ViPSession instrumentSession,  
                      ViRsrc resourceName);
```

Purpose

This function opens a visa session to an instrument.

The default timeout value is set to 35 seconds.

Parameter List

instrumentSession

Variable Type ViSession (passed by reference)

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

resourceName

Variable Type ViRsrc

This control is used to pick which instrument to open a session with.

Return Value

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

zt_pxi_query

```
ViStatus zt_pxi_query (ViSession instrumentSession,  
                      unsigned short query,  
                      unsigned short numberQueryParameters,  
                      unsigned short queryParameters[],  
                      unsigned short *response,  
                      unsigned short numberResponseParameters[],  
                      unsigned short responseParameters[]);
```

Purpose

This function sends a query to the instrument, including all query parameters. The command handshaking process ensures that the command is accepted, a response and response parameters are returned.

Parameter List

instrumentSession

Variable Type ViSession

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

query

Variable Type unsigned short

This control sets the query code to send to the instrument.

numberQueryParameters

Variable Type unsigned short

This control sets the number of parameters for the query sent to the instrument.

queryParameters

Variable Type unsigned short []

This control sets the parameter array for the query sent to the instrument.

response

Variable Type unsigned short (passed by reference)

This pointer provides the address for the returned query response.

numberResponseParameters

Variable Type unsigned short []

This pointer provides the address for the returned number of query parameters.

responseParameters

Variable Type unsigned short []

This pointer provides the address for the query parameter array.

Return Value

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

zt_pxi_receive

```
ViStatus zt_pxi_receive (ViSession instrumentSession,  
                        char formatString[], ...);
```

Purpose

This Function receives a squence of data from the instrument.

Parameter List

instrumentSession

Variable Type ViSession

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

formatString

Variable Type char []

String containing the format string that specifies the type of the received data. If insufficient arguments exist for the format, the behavior is undefined. If the format is exhausted while arguments remain, the excess arguments are evaluated but are otherwise ignored.

Example Format String:

"%hu%f" interprets the received data as an unsigned short and a floating point

parameters

Variable Type any type (passed by value)

This specifies one or more pointers to store the received data. If there are multiple parameters, they must be separated by commas.

Return Value

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

zt_pxi_revision_query

```
ViStatus zt_pxi_revision_query (ViSession instrumentSession,  
                                char driverRevision[]);
```

Purpose

This function reads a block of data from the instrument.

Parameter List

instrumentSession

Variable Type ViSession

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

driverRevision

Variable Type char []

String containing the driver revision information in the following format:

"Rev X, Date, CVI Y"

Where X is the revision number, Date is the date created, and Y is the National Instruments CVI version used to create the driver.

Return Value

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

zt_pxi_send

```
ViStatus zt_pxi_send (ViSession instrumentSession, char formatString[],  
                      ...);
```

Purpose

This Function sends a command/query and parameters to the instrument.

Parameter List

instrumentSession

Variable Type ViSession

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

formatString

Variable Type char []

String containing the command/query code and the format string that specifies the type of subsequent parameters. If insufficient arguments exist for the format, the behavior is undefined. If the format is exhausted while arguments remain, the excess arguments are evaluated but are otherwise ignored.

Example Format String:

"0x8100%hu%f" indicates the command is 0x8100 followed by an unsigned short parameter and a floating point parameter

parameters

Variable Type any type (passed by value)

This specifies one or more parameters that are to be converted and written to the instrument. If there are multiple parameters, they must be separated by commas.

Return Value

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