

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*

EIP 1230A/1231A VXIbus Frequency Counter

Introduction:

This instrument driver provides programming support for EIP 1230A/1231A VXIbus Frequency Counter.
It contains functions for opening, configuring, taking measurement from, and closing the instrument.

Assumptions:

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

For GPIB instruments drivers:
- the instrument is connected to the GPIB.
- the GPIB address supplied to the initialize function must match the GPIB address of the instrument.
- there must be at least one unused device name in the handler.

For VXI instruments drivers:
- the instrument is installed in the VXI mainframe and you are using one of the following controller options:
  Embedded controller
    AT-MXI
    MC-MXI
    GPIB-VXI
- the logical address supplied to the initialize function must match the logical address of the instrument.
- if you are using GPIB-VXI, there must be at least one unused device name in the handler.

For RS-232 instruments 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.

LabWindows Libraries:

In order to use this instrument driver outside of the LabWindows environment, it must be linked with the following LabWindows libraries:

For GPIB instruments drivers:
- GPIB Library
- Format and I/O Library
- LabWindows System Library

For VXI instruments drivers:
- The appropriate VXI Library:

VXI386.LIB for embedded controllers
VXIAT.LIB for the AT-MXI
VXIMC.LIB for the MC-MXI
GPIB_C.LIB for GPIB-VXI (C users)
GPIB_QB.LIB for GPIB-VXI (QuickBASIC users)
- Format and I/O Library
- LabWindows System Library

For RS-232 instruments drivers:
- RS-232 Library
- Format and I/O Library
- LabWindows System Library

National Instruments strongly recommends that you use the LWMake utility to build any standalone executable program based on this intrument driver.

Error Codes:

Error codes are returned as the return value of each instrument driver function. A program should examine this value after each call to an instrument driver function to determine if any error occurred. Possible error codes and their meanings are listed with the corresponding instrument driver function.

How To Use This Document:

This document is intended to be used as a programming reference manual. It describes each function in the

EIP 1230A/1231A VXIbus Frequency Counter

instrument driver. The functions are listed in alphabetical order with a description of the function, C and QuickBASIC syntax of the function, a description of each parameter, and possible error codes.

Function Tree Layout:

EIP 1230A/1231A VXIbus Frequency Counter Function Name:

- Initialize — eip123x_init
- Configuration
  - Bands (1230 & 1231)
    - Bands for EIP1230 A — eip123x_band_eip1230
    - Band for EIP1231 A — eip123x_band_eip1231
    - Center Frequency — eip123x_center_freq
    - Frequency Window Limits — eip123x_freq_win_lim
  - Sample Rate — eip123x_sample_rate
  - Resolution — eip123x_resolution
  - Frequency Average — eip123x_freq_avr
  - Frequency Offset — eip123x_freq_offset

Min Pulse Repetition	eip123x_min_freq_rep
Inhibit Input	eip123x_inhibit_input
Time Base Source	eip123x_time_base
Trigger Set Up	eip123x_trig_setup
Special Functions	
Special Function Select	eip123x_special_funct
Terminate All Specials	eip123x_terminate_spec
Measurement	
Hold Function	eip123x_hold_func
Set Measurement Type	eip123x_set_meas_type
Trigger Function	eip123x_trigger_func
Get Measurement	eip123x_get_measurement
Save/Recall Instr Setup	
Save/Recall Instr Setup	eip123x_counter_setup
Save Default Setup	eip123x_save_default_setup
Get General Setup Info	eip123x_get_gen_setup_info
Low Level	
Write to Instrument	eip123x_write
Read Instrument Data	eip123x_read
Reset instrument	eip123x_reset
Error Clear/Query	eip123x_error_query
Revision Query	eip123x_revision_query
Close	eip123x_close

Instrument : EIP 1230A/1231A VXIbus Frequency Counter

Instrument Name: EIP123xA Pulsed Microwave
Frequency Counter

Description: This instrument module provides
programming support for the
EIP1230A and EIP1231A Microwave
Frequency counters. The module
is divided into the following
Classes:

Functions/Classes:

(1) Initialize
Initialize the instrument and set to default
configuration.

(2) Configuration
This class contains the functions required to
configure the instrument.

(2a) Bands (EIP1230A & EIP1231A)
This sub-class contains the band
configuration functions for both
instruments. It also contains the
Frequency Windows function which may
only be used with Band 2, and the
Center Frequency Function which is

available for Bands 2 & 3.

(3) Special Functions
This class contains the Special Functions which this instrument driver supports.

(4) Measurement Functions
This class contains the functions necessary to capture and display measurements.

(5) Save and Recall Instrument Setup
This class contains functions that store and recall the current instrument setup

(6) Low Level Functions:
This Class provides lower level functions to communicate with the instrument, and change instrument parameters.

(7) Close
Take the instrument offline.

Following functions are in alphabetical order.

Function Name : eip123x_band_eip1230

C Syntax : int eip123x_band_eip1230(Instrument_ID, Primary_Band,
Sub_Band_GHz);
int Instrument_ID;
int Primary_Band;
int Sub_Band_GHz;

QuickBASIC Syntax : Error.Code% = eip123x.band.eip1230(Instrument.ID%, _
Primary.Band%, Sub.Band.GHz%)

Purpose : Panel Name: Bands for EIP1230 A

Description: This function contains the band configuration functions for the EIP1230A

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Primary Band
0 - 250 MHz -
1 - 1 GHz -
2 - 26.5 GHz - (Default)
3 - 170 GHz -
Control Name: Primary Band (EIP 1230A)

Description: This parameter controls the frequency measurement range for the EIP1230A model. Select a specific band according to the following table.

0: 100 Hz - 250 MHz (CW only)
1: 250 MHz - 1 GHz
2: 950 MHz - 26.5 GHz
3: 26.5 GHz - 170 GHz (Optional)

Variable Type: Integer

Valid Range: 0 - 100 Hz - 250 MHz (CW only)
1 - 250 MHz - 1 GHz
2 - 950 MHz - 26.5 GHz
3 - 26.5 GHz - 170 GHz (Optional)

Default Value: 2 - 950 MHz - 26.5 GHz

Sub Band (GHz)
1 - 26.5 - 40 - (Default)
2 - 33 - 50 -
3 - 40 - 60 -
4 - 50 - 75 -
5 - 60 - 90 -
6 - 75 - 110 -
7 - 90 - 140 -
8 - 110 - 170 -
Control Name: Sub Band (FOR BAND 3 ONLY)

Description: This parameter controls the frequency measurement range of Band 3 and is set according to the remote sensor being used. Select the appropriate subband according to the following table.

1: 26.5 to 40 GHz
2: 33 to 50 GHz
3: 40 to 60 GHz
4: 50 to 75 GHz
5: 60 to 90 GHz
6: 75 to 110 GHz
7: 90 to 140 GHz
8: 110 to 170 GHz

Variable Type: Integer

Valid Range: 1 - 26.5 to 40 GHz
2 - 33 to 50 GHz
3 - 40 to 60 GHz
4 - 50 to 75 GHz
5 - 60 to 90 GHz
6 - 75 to 110 GHz
7 - 90 to 140 GHz
8 - 110 to 170 GHz

Default Value: 1

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Primary Band out of range
-3 Sub Band out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized

330 Sub Band parameter valid only for the optional Band 26.5 - 170 GHz Set Sub Band back to default 26.5 - 40 GHz

350 This Panel valid only for EIP 1230A

Function Name : eip123x_band_eip1231

C Syntax : int eip123x_band_eip1231(Instrument_ID, Band_EIP_1231A);
int Instrument_ID;

int Band_EIP_1231A;

QuickBASIC Syntax : Error.Code% = eip123x.band.eip1231(Instrument.ID%, _
Band.EIP.1231A%)

Purpose : Panel Name: Bands for EIP1231 A

Description: This function contains the band configuration functions for the EIP1231A

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Band (EIP 1231A)
0 - 250 MHz -
1 - 1 GHz -
2 - 20 GHz - (Default)

Control Name: Band (EIP 1231A)

Description: This parameter controls the frequency measurement range for the EIP 1231A model. Select a specific band according to the following table.

0: 100 Hz - 250 MHz (CW only)
1: 250 MHz - 1 GHz
2: 950 MHz - 20 GHz

Variable Type: Integer

Valid Range: 0 - 100 Hz - 250 MHz (CW only)
1 - 250 MHz - 1 GHz
2 - 950 MHz - 20 GHz

Default Value: 2

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Band out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized

360 This Panel Valid only for EIP 1231A

Function Name : eip123x_center_freq

C Syntax : int eip123x_center_freq(Instrument_ID, Center_Freq_MHz);
int Instrument_ID;
double Center_Freq_MHz;

QuickBASIC Syntax : Error.Code% = eip123x.center.freq(Instrument.ID%, _
Center.Freq.MHz#)

Purpose : Panel Name: Center Frequency

Description: Controls the center frequency mode
This function controls the center frequency mode of operation in which the counter looks for a signal in the vicinity of the center frequency value. This mode can be used to reduce acquisition time or when measuring a particular signal in a multiple signal environment. This mode of operation is available in Bands 2 and 3 only. Select Band 2 center frequency in the range of 900 MHz to 26.7 GHz for the model 1230A, and in the range of 900 MHz to 20.5 GHz for the model 1231A. The counter will lock on signals 50 MHz from the entered center frequency. Select Band 3 center frequency in the range of the subband currently selected. The counter will lock on to signals

2 GHz from the entered frequency, but will not reject signals outside of this range. If a signal is applied that is more than 2 GHz from the entered frequency an erroneous reading may result. The value entered by the user is truncated to 10 MHz.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Center Freq (MHz)

Control Name: Center Frequency

Description: This parameter controls the center frequency mode of operation in which the counter looks for a signal in the vicinity of the center frequency value. This mode can be used to reduce acquisiton time when measuring a particular signal in a multiple signal environment. This mode is available in either Band 2 or Band 3.

Variable Type: Real

Valid Range: Dependent upon the selected band - See below.

- Band 2: 900 MHz - 26,700 MHz (EIP1230A)
  900 MHz - 20,500 MHz (EIP1231A)

- Band 3: The range of the Sub Band currently selected:

Subband 1: 26,500 MHz to 40,000 MHz
Subband 2: 33,000 MHz to 50,000 MHz
Subband 3: 40,000 MHz to 60,000 MHz
Subband 4: 50,000 MHz to 75,000 MHz
Subband 5: 60,000 MHz to 90,000 MHz
Subband 6: 75,000 MHz to 110,000 MHz
Subband 7: 90,000 MHz to 140,000 MHz
Subband 8: 110,000 MHz to 170,000 MHz

Default Value: 900

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Center Frequency out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized
236 Error interpreting instrument response

340 Function valid only for bands 2 & 3 in the EIP 1230 A, and band 2 in EIP 1231 A.

Function Name : eip123x_close

C Syntax : int eip123x_close(Instrument_ID);
int Instrument_ID;

QuickBASIC Syntax : Error.Code% = eip123x.close(Instrument.ID%)

Purpose : Panel Name: Close

Description: The close routine removes the entry for the instrument from the Instrument Table.

Note: The instrument must be reinitialized to use it again.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range

221 Unable to close device
232 Device not initialized

Function Name : eip123x_counter_setup

C Syntax : int eip123x_counter_setup(Instrument_ID, Mode, Setup_Register);
int Instrument_ID;
int Mode;
int Setup_Register;

QuickBASIC Syntax : Error.Code% = eip123x.counter.setup(Instrument.ID%, _ Mode%, Setup.Register%)

Purpose : Panel Name: Save/Recall Instr Setup

Description: This function selects and implements the Store and Recall functions. The Store command causes the instrument to save it’s current setup in one of its 10

internal storage registers (0-9).
After power-up or initialization,
the instrument restores itself
from register 0.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned
by the Initialize function to
select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of
this Instrument

The Maximum Instances of this
Instrument is the maximum number
instruments that can be controlled
by this driver in a single system.
The value is set internally in the
driver. The typical value is 10.

Mode

1 - Store

0 - Recall (Default)

Control Name: Mode

Description: This function causes the counter's
setup to be stored into or
recalled from one of its
10 internal storage registers.
Register 0 is the instrument's
default register. At turn-on
the instrument will be configured
per setup data from register 0.

Store: Saves the current counter setup
in the storage register specified
in the register box.

Recall: Recalls a counter setup which
has been previously stored
in the specified storage register.

Variable Type: Integer

Valid Range: 1 - Store
0 - Recall

Default Value: 0 - Recall

Setup Register

Control Name: Setup Register

Description: Specifies which instrument storage register is to be stored or recalled.

Variable Type: Integer

Valid Range: 0 through 9

Default Value: 0

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0	Success
-1	Instrument ID out of range
-2	Mode out of range
-3	Setup Register out of range
230	Error writing to instrument
232	Instrument not initialized

Function Name : eip123x_error_query

C Syntax : int eip123x_error_query(Instrument_ID, Action,
123x_Error_Number, Error_Message);
int Instrument_ID;
int Action;
int *123x_Error_Number; !! Invalid control label name.
char *Error_Message;

QuickBASIC Syntax : Error.Code% = eip123x.error.query(Instrument.ID%, _
Action%, 123x.Error.Number%, Error.Message$)

Purpose : Panel Name: Error Clear/Query

Description: Either clears pending error indications or reads the current error number. An error number of 0 means that there are no errors.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number of instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Action
1 - Clear Error
0 - Read Error (Default)
Control Name: Action

Description: This switch determines the action performed by the function.

Variable Type: Integer

Valid Range: 0 - Read last 123x error
1 - Cause the 123x to clear its internal error indications.

Default Value: 0 - Read

123x Error Number
Control Name: Error Code

Description: This output returns the error code read from the instrument error queue.

Variable Type: Integer *

Error Message
Control Name: Error Message

Description: This output returns the error message string read from the instrument's error message queue.

Variable Type: String (char *)

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized
236 Error interpreting instrument response

Function Name : eip123x_freq_avr

C Syntax : int eip123x_freq_avr(Instrument_ID, Frequency_Average);
int Instrument_ID;
int Frequency_Average;

QuickBASIC Syntax : Error.Code% = eip123x.freq.avr(Instrument.ID%, _
Frequency.Average%)

Purpose : Panel Name: Frequency Average

Description: This function controls the number of measurements to be averaged on frequency and pulse measurements. Select in the range of 1 to 99.

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Frequency Average
Control Name: Frequency Average

Description: This parameter controls the number of measurements to be averaged on frequency and pulse parameters. Frequency averaging is typically used to reduce random measurement error.

Variable Type: Integer

Valid Range: 01 to 99

Default Value: 01

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0	Success
-1	Instrument ID out of range
-2	Frequency Average out of range
230	Error writing to instrument
231	Error reading from instrument
232	Instrument not initialized

Function Name : eip123x_freq_offset

C Syntax : int eip123x_freq_offset(Instrument_ID, Freq_Multiplier, Freq_Offset__GHz);
int Instrument_ID;
int Freq_Multiplier;
double Freq_Offset__GHz;

QuickBASIC Syntax : Error.Code% = eip123x.freq.offset(Instrument.ID%, _ Freq.Multiplier%, Freq.Offset..GHz#)

Purpose : Panel Name: Frequency Offset

Description: Configures frequency measurement parameters. This function allows the entry of a positive or negative frequency to 1 kHz resolution into the offset

frequency register. This parameter controls the contant B in the formula: Output Freq = (M * measured freq) + B where M is the frequency multiplier and B is the frequency offset. Select in the range -99.99 999 GHz to +99.99 999 GHz.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Freq Multiplier

Control Name: Multiplier

Description: This parameter controls the value of the constant M in the formula:

Output Frequency = (M * measured frequency) + B

Where M is the frequency multiplier and B is the frequency offset.

Variable Type: Integer

Valid Range: 1 to 99

Default Value: 1

Freq Offset (GHz)

Control Name: Offset Frequency

Description: Allows the entry of a positive or negative frequency to 1 kHz resolution into the offset frequency register. This parameter controls the constant B

in the formula:

Output Frequency = (M * measured frequency) + B

Where M is the frequency multiplier and B is the frequency offset.

Variable Type: real

Valid Range: -99.99 999 GHz to +99.999 999 GHz

Default Value: 0

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0	Success
-1	Instrument ID out of range
-2	Frequency Multiplier out of range
-3	Frequency Offset out of range
230	Error writing to instrument
231	Error reading from instrument
232	Instrument not initialized

Function Name : eip123x_freq_win_lim

C Syntax : int eip123x_freq_win_lim(Instrument_ID, Low_Limit_MHz, High_Limit_MHz);
int Instrument_ID;
double Low_Limit_MHz;
double High_Limit_MHz;

QuickBASIC Syntax : Error.Code% = eip123x.freq.win.lim(Instrument.ID%, _ Low.Limit.MHz#, High.Limit.MHz#)

Purpose : Panel Name: Frequency Window Limits

Description: This function controls the high and low limits of the frequency window that is searched in Band 2. Select Low and High Limit in the range of 900 MHz to 26.7 GHz for

the model 1230A, and in the range of 900 MHz to 20.5 GHz for the model 1230A. The value entered by the user is truncated to 10 MHz resolution. This function is only available in Band 2, and Low limit must always be less than High limit.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Low Limit (MHz)

Control Name: Low Limit

Description: This parameter controls the low end of the frequency window that is searched for a signal in Band 2. Select Low Limit in the range of 900 MHz to 26700 MHz for Model 1230A and in the range of 900 MHz to 20500 MHz for Model 1231A.

The value entered by the user is trincated to 10 MHz resolution. This function is only available in Band 2. Low Limit must always be lower than High Limit.

Variable Type: Real

Default Value: 900 MHz

High Limit (MHz)

Control Name: High Limit

Description: This parameter controls the high

end of the frequency window that is searched for a signal in Band 2. Select the High Limit in the range of 900 MHz to 26,700 MHz for Model 1230A and in the range of 900 MHz to 20,500 MHz for Model 1231A.

The value entered by the user is trincated to 10 MHz resolution. This function is only available in band 2. Low Limit must always be lower than High Limit.

Variable Type: Real

Default Value: 26,700 MHz for EIP 1230A

- This is the value that the control is set to if you are using a 1231A change the default setting to: 20,500 MHz for EIP 1231A

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0	Success
-1	Instrument ID out of range
-2	Low Limit out of range
-3	High Limit out of range
230	Error writing to instrument
231	Error reading from instrument
232	Instrument not initialized
236	Error interpreting instrument response
380	This panel vaid only for Band 2.

Function Name : eip123x_get_gen_setup_info

C Syntax : int eip123x_get_gen_setup_info(Instrument_ID, Setup, Display);

```
int Instrument_ID;
int Setup;
char *Display;
```

QuickBASIC Syntax : Error.Code% = eip123x.get.gen.setup.info(Instrument.ID%, _
Setup%, Display$)

Purpose : Panel Name: Get General Setup Info:

Description: This panel returns the current value of the parameter selected. Parameter definitions are below.

Average Val: This parameter controls the number of measurements to be averaged on frequency and pulse parameters.

Band: This parameter controls the frequency measurement range for the EIP1230A/1231A models. Select a specific band according to the following table.

0: 100 Hz - 250 MHz (CW only)
1: 250 MHz - 1 GHz
2: 950 MHz - 26.5 GHz (EIP1230A)
950 MHz - 20 GHz (EIP1231A)
3: 26.5 GHz - 170 GHz (Optional)

Sub Band: This parameter controls the frequency measurement range of Band 3 and is set according to the remote sensor being used. Select the apprpriate sybband according to the following table.

1: 26.5 GHz to 40 GHz
2: 33 GHz to 50 GHz
3: 40 GHz to 60 GHz
4: 50 GHz to 75 GHz
5: 60 GHz to 90 GHz
6: 75 GHz to 110 GHz
7: 90 GHz to 140 GHz
8: 110 GHz to 170 GHz

Center Freq: This parameter controls the center frequency mode of operation in which the counter looks for a signal in the vicinity of the center frequency value. This mode can be uded to reduce acquisiton time or when measuring a particylar signal in a multiple signal environment. This mode is available in either Band 2 or Band 3. The value is returned in Hz.

Resolution: This parameter controls the frequency measurement resolution.

Select the desired resolution according ro the following table:

2: 100 Hz
3: 1 kHz
4: 10 kHz
5: 100 kHz
6: 1 MHz
7: 10 MHz
8: 100 MHz
9: 1 GHz

Low/high limit: This parameter controls the lo/hi end of the frequency window that is searched for a signal in Band 2. Select Low Limit in the range of 900 MHz to 26700 MHz for Model 1230A and in the range of 900 MHz to 20500 MHz for Model 1231A. The values are returned in Hz.

Synch Line: Selects the TTL TRG line to be used in the Synchronous Trigger Protocol. The SYNC message enables the SYNCHRONOUS TTLRG trigger protocol

Asynch Line: Selects the TTL TRG line pair to be used in the Asynchroneous Trigger Protocol as follows:

0: 0 & 1
1: 2 & 3
2: 4 & 5
3: 6 & 7

Inhibit Line: Selects the input to be used for the measurement inhibit function as follows:

0: VXIbus ECLTRG0 line
1: VXIbus ECLTRG1 line
2: Front Panel

MinPRF: This parameter controls the minimum pulse repetition frequency of the pulsed signals that can be acquired and measured by the counter. For example, if 500 Hz is selected, the counter will measure signals with a pulse repetition frequency of 500 Hz or greater. The value is returned in Hz.

Multiplier: This parameter controls the value of the constant M in the formula:

Output Frequency = (M * measured frequency) + B

Where M is the frequency multiplier and B is the frequency offset.

Offset: Allows the entry of a positive or negative frequency to 1 kHz resolution into the offset frequency register. This parameter controls the constant B in the formula:

Output Frequency = (M * measured frequency) + B

Where M is the frequency multiplier and B is the frequency offset.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Setup

0 - Average Val-
1 - Band -
2 - Subband -
3 - Center Freq-
4 - Resolution -
5 - Low Limit -
6 - High Limit -
7 - Asynch Line-
8 - Synch Line -
9 - Inhibit Line

10 - MinPRF -
11 - Multiplier -
12 - Offset - (Default)

Control Name: Setup

Description: Selects the Setup Parameter to be viewed, returns the current setting of the selected parameter

Variable Type: Integer

Valid Range: 0 - Avg Value
1 - Band
2 - Subband
3 - Center Freq
4 - Resolution
5 - Low Limit
6 - High Limit
7 - Async Line
9 - Sync Line
10 - MIN PRF
11 - Multiplier
12 - Offset

Default Value: 12 - Offset

Display

Control Name: Display

Description: This function displays the current status of the parameter selected. Parameter definitions are below.

Average Val: This parameter controls the number of measurements to be averaged on frequency and pulse parameters.

Band: This parameter controls the frequency measurement range for the EIP1230A/1231A models. Select a specific band according to the following table.

0: 100 Hz - 250 MHz (CW only)
1: 250 MHz - 1 GHz
2: 950 MHz - 26.5 GHz (EIP1230A)
950 MHz - 20 GHz (EIP1231A)
3: 26.5 GHz - 170 GHz (Optional)

Sub Band: This parameter controls the frequency measurement range of Band 3 and is set according to the remote sensor being used. Select the apprpriate sybband according to the following table.

1: 26.5 GHz to 40 GHz
2: 33 GHz to 50 GHz
3: 40 GHz to 60 GHz
4: 50 GHz to 75 GHz
5: 60 GHz to 90 GHz
6: 75 GHz to 110 GHz
7: 90 GHz to 140 GHz
8: 110 GHz to 170 GHz

Center Freq: This parameter controls the center frequency mode of operation in which the counter looks for a signal in the vicinity of the center frequency value. This mode can be uded to reduce acquisiton time or when measuring a particylar signal in a multiple signal environment. This mode is available in either Band 2 or Band 3. The value is returned in Hz.

Resolution: This parameter controls the frequency measurement resolution. Select the desired resolution according ro the following table:

2: 100 Hz
3: 1 kHz
4: 10 kHz
5: 100 kHz
6: 1 MHz
7: 10 MHz
8: 100 MHz
9: 1 GHz

Low/high limit:This parameter controls the lo/hi end of the frequency window that is searched for a signal in Band 2. Select Low Limit in the range of 900 MHz to 26700 MHz for Model 1230A and in the range of 900 MHz to 20500 MHz for Model 1231A. The values are returned in Hz.

Synch Line: Selects the TTL TRG line to be used in the Synchronous Trigger Protocol. The SYNC message enables the SYNCHRONOUS TTLRG trigger protocol

Asynch Line: Selects the TTL TRG line pair to be used in the Asynchroneous Trigger Protocol as follows:

0: 0 & 1
1: 2 & 3
2: 4 & 5
3: 6 & 7

Inhibit Line: Selects the input to be used for the measurement inhibit function as follows:

0: VXIbus ECLTRG0 line
1: VXIbus ECLTRG1 line
2: Front Panel

MinPRF: This parameter controls the minimum pulse repetition frequency of the pulsed signals that can be acquired and measured by the counter. For example, if 500 Hz is selected, the counter will measure signals with a pulse repetition frequency of 500 Hz or greater. The value is returned in Hz.

Multiplier: This parameter controls the value of the constant M in the formula:

Output Frequency = (M * measured frequency) + B

Where M is the frequency multiplier and B is the frequency offset.

Offset: Allows the entry of a positive or negative frequency to 1 kHz resolution into the offset frequency register. This parameter controls the constant B in the formula:

Output Frequency = (M * measured frequency) + B

Where M is the frequency multiplier and B is the frequency offset.

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Setup out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized
236 Error interpreting instrument response

Function Name : eip123x_get_measurement

C Syntax : int eip123x_get_measurement(Instrument_ID, Measurement);
int Instrument_ID;
double *Measurement;

QuickBASIC Syntax : Error.Code% = eip123x.get.measurement(Instrument.ID%, _
Measurement#)

Purpose : Panel Name: Get Measurement

Description: This function reads the measurement back from the instrument. All measurement configuration and triggering must be implemented before this function is called to insure accurate data is returned from the instrument.

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Measurement
Control Name: Result

Description: Outputs the measurement result, all measurement configuration and triggering must be implemented before this function is called to insure accurate data is returned from the instrument.

Variable Type: Real

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range

231 Error reading from instrument
232 Instrument not initialized
236 Error interpreting instrument response

Function Name : eip123x_hold_func

C Syntax : int eip123x_hold_func(Instrument_ID, Hold);
int Instrument_ID;
int Hold;

QuickBASIC Syntax : Error.Code% = eip123x.hold.func(Instrument.ID%, Hold%)

Purpose : Panel Name: Hold Function

Description: This function selects the status of the Hold command.

When Hold is On, the counter takes one reading and waits for a Reset or Trigger command. Upon receipt of one of these commands, the counter will perform another measurement and, when requested, output the new reading over the VXI bus. Note that a measurement result will be output only once; thus, when the instrument is in

Hold, the user must trigger the counter before requesting another reading. If no signal is found, the counter will output a zero.

When the Hold command is Off, data is read out to the bus in the normal way. The reading is automatically updated according to the specified sample rate, and the counter can output successive readings without requiring a Reset or Trigger command each time.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Hold

1 - On

0 - Off (Default)

Control Name: Hold

Description: Holds the last result if On. Hold On puts the counter into triggered operational mode.

Variable Type: Integer

Valid Range: 1 - On

0 - Off

Default Value: 0 - Off

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Hold out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized

Function Name : eip123x_inhibit_input

C Syntax : int eip123x_inhibit_input(Instrument_ID,
Inhibit_Input_Source);
int Instrument_ID;
int Inhibit_Input_Source;

QuickBASIC Syntax : Error.Code% = eip123x.inhibit.input(Instrument.ID%, _
Inhibit.Input.Source%)

Purpose : Panel Name: Inhibit Input

Description: This function selects the source
of the signal controlling when the
counter can detected a signal.

This feature allows the counter
to perform frequency measurements
at specific points in time.

Arguments 0, 1, and 2 correspond
to the VXIbus ECLTRG0 line, VXIbus
ECLTRG1 line, and front panel,
respectively.

Note: The Inhibit Input is active
high (i.e., the counter is
inhibited by a logic high signal
and enabled by a logic low
signal).

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned
by the Initialize function to
select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Inhibit Input Source
0 - VXI ECLTRG0-
1 - VXI ECLTRG1-
2 - Front Panel- (Default)
Control Name: Inhibit Input

Description: Selects the input to be used for the measurement inhibit function.

Variable Type: Integer

Valid Range: 0 - VXIbus ECLTRG0 line
1 - VXIbus ECLTRG1 line
2 - Front Panel

Default Value: 2 - Front Panel

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Inhibit Input out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized

Function Name : eip123x_init

C Syntax : int eip123x_init(Logical_Address, Interface, ID_Query, Reset_Device, Instrument_ID, Model_Number);
int Logical_Address;
int Interface;

int ID_Query;
int Reset_Device;
int *Instrument_ID;
int *Model_Number;

QuickBASIC Syntax : Error.Code% = eip123x.init(Logical.Address%, Interface%, _
ID.Query%, Reset.Device%, Instrument.ID%, _
Model.Number%)

Purpose : Panel Name: Initialize

Description: This routine performs the following initialization actions:

- Opens the instrument by storing information about the instrument in the Instrument Table.

- Initializes the VXI interface (NI-VXI or NI-GPIB) accordingly.

- Performs an identification query on the Instrument to determine the model number.

- Sends initialization commands to the instrument.

- Returns an Instrument ID which is used to differentiate between instruments of the same model type. This value will be used to identify the instrument in subsequent calls.

Parameter List : Logical Address
Control Name: Logical Address

Description: Specifies the Logical Address of the VXI instrument. This value must corrrespond to the current internal setting of the instrument.

Variable Type: Integer

Valid Range: 0 - 255

Default Value: 18

Interface
1 - NI-VXI (Default)
2 - NI-GPIB

Control Name: Interface

Description: Specifies which protocol is used to communicate with the VXI mainframe.

NI-VXI should be selected for MC-MXI, AT-MXI, or VXIpc-386.

NI-GPIB should be selected for the GPIB-VXI.

Variable Type: Integer

Valid Range: 1 - NI VXI
2 - NI GPIB

Default Value: 1 - NI VXI

ID Query
1 - Do Query (Default)
0 - Skip Query
Control Name: ID Query

Description: Specifies if an ID Query is sent to the instrument during the initialization procedure.

Variable Type: Integer

Valid Range: 1 - Do Query
0 - Skip Query

Default Value: 1 - Do Query

NOTE: Under normal circumstances the ID Query insures that the instrument initialized is the type supported by this driver. However circumstances may arise where it is undesirable to send an ID Query to the instrument. In those cases; set this control to "Skip Query" and this function will initialize the selected interface and the command arrays in the driver, without doing an ID Query.

Reset Device
1 - Reset Device (Default)
0 - Don’t Reset
Control Name: Reset Device

Description: Specifies if the instrument is to be reset to its power-on settings during the initialization procedure.

Variable Type: Integer

Valid Range: 1 - Reset Device
0 - Don’t Reset

Default Value: 1 - Reset Device

NOTE: If you do not want the instrument reset. Set this control to "Don’t Reset" while initializing the instrument.

Instrument ID
Control Name: Instrument ID

Description: Returns an Instrument ID that is used in all subsequent function calls to select the instrument at the address specified.

If more than one instrument of the same model type is used, this ID will be used to differentiate between them. The first instrument of this model type initialized will be assigned an Instrument ID of 1.

Variable Type: Integer

Model Number
Control Name: Model Number

Description: Returns the instrument model number. If the user opted to skip the ID query, the instrument is assumed to be a 1230, otherwise the value is determined from the instrument.

Variable Type: Integer

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Logical Address out of range
-2 Interface out of range

-3 ID Query out of range
-4 Reset Device out of range

220 Unable to open instrument
221 Unable to close instrument
222 Unable to initialize VXI library
223 Instrument identification query failed
230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized
233 Error configuring GPIB address
236 Error interpreting instrument response
239 Error setting time out
241 Error configuring repeat addressing

Function Name : eip123x_min_freq_rep

C Syntax : int eip123x_min_freq_rep(Instrument_ID,
Min_Pulse_Rep_Freq, MinPRF_Units);
int Instrument_ID;
double Min_Pulse_Rep_Freq;
int MinPRF_Units;

QuickBASIC Syntax : Error.Code% = eip123x.min.freq.rep(Instrument.ID%, _
Min.Pulse.Rep.Freq#, MinPRF.Units%)

Purpose : Panel Name: Min Pulse Repetition:

Description: This function controls the minimum pulse repetition frequency of the pulsed signals that can be acquired and measured by the counter. For example if a MINPRF of 500 Hz is selected, the counter will only measure signals with a pulse repetition frequency of 500 Hz or greater.

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number

instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Min Pulse Rep Freq

Control Name: Min Pulse Repetition

Description: This parameter controls the minimum pulse repetition frequency of the pulsed signals that can be acquired and measured by the counter. For example, if 500 Hz is selected, the counter will measure signals with a pulse repetition frequency of 500 Hz or greater.

This parameter affects the acquisition speed indirectly via two internal processes: the waiting time for a pulse at each frequency step in the frequency range being searched, and the waiting time for a pulse when taking measurements before declaring a "signal lost" condition.

When Min Pulse Repetition is set to a low value either center frequency of frequency limits to minimize signal aquisition time.

Variable Type: Real

Valid Range: 1.0 Hz to 2.0 kHz

Default Value: 2.0 kHz

MinPRF Units

0 - Hz -
1 - kHz - (Default)
2 - MHz -
3 - GHz -

Control Name: Minimum PRF Units

Description: Selects the units for the Minimum Pulse Repetition Frequency specified by the Min PRF control.

Valid Range: 0 - Hz
1 - kHz

2 - MHz
3 - GHz

Default Value: 0 - Hz

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Min Pulse Rep Freq out of range
-3 MinPRF Units out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized

Function Name : eip123x_read

C Syntax : int eip123x_read(Instrument_ID, Number_Bytes_To_Read, Read_Buffer, NumBytes_Read);
int Instrument_ID;
int Number_Bytes_To_Read;
char *Read_Buffer;
long *NumBytes_Read;

QuickBASIC Syntax : Error.Code% = eip123x.read(Instrument.ID%, _ Number.Bytes.To.Read%, Read.Buffer$, _ NumBytes.Read&)

Purpose : Panel Name: Read Instrument Data

Description: Reads data from instrument buffer and returns it to the specified variable in memory.

NOTES: (1) Because the instrument may return both numeric and text data in response to queries this function returns the data in string format.

(2) If valid data is not available at the instrument's output buffer when this function is called the instrument will hang up and the function will not

return until it times out. If the time-out is disabled this function will hang indefinitely and it may be neccessary to reboot the computer to break out.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number of instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Number Bytes To Read

Control Name: Number of Bytes to Read

Description: The number of bytes specified by this control should be greater than or equal to the number of bytes which are to be read from the instrument. If the actual number of bytes to be read is greater than the number this control specifies then multiple reads will be necessary to empty the instrument's output buffer.

If the instrument's output buffer is not emptied the instrument may return invalid data when future reads are performed.

Variable Type: Integer

Default Value: 50 (See NOTE 2)

NOTES: (1) If NO DATA is available at the instrument's output buffer when this function is called the instrument will hang up and the function will not return until it times out. If the time-out is disabled, this function

will hang indefinitely and it may be neccessary to reboot the computer.

(2) If the number of bytes expected is greater than 50 the value of this control may be increased. If the actual number of bytes read is larger than the declared size of the read buffer a run-time error will be generated.

Read Buffer

Control Name: Read Buffer

Description: The incoming data from the instrument is placed into this variable.

Variable Type: String

NOTES: (1) Because the Instrument may return both numeric and text data in response to queries this function returns the data in string form.

(2) This function does not overwrite any old data left in the string variable from the last time the function was called. String data in LabWindows is terminated with an ASCII null(Hex 0) and string manipulation functions will only recognize data before the ASCII null.

(3) The declared size of the string variable must be greater than the actual number of bytes read from the instrument if it is not a run-time error will be generated.

NumBytes Read

Control Name: Number of Bytes Read

Description: This variable contains the actual number of bytes read from the instrument. This is the value which is returned by the read function. The global variable "ibcnt" holds the result of GPIB reads and the global variable "bytes" holds the result of VXI word serial reads.

Variable Type: Long

NOTE: If the actual number of bytes read is less than the number of bytes specified in number of bytes to read then the output buffer is probably empty. If the number of bytes read is 0 and the instrument returns a 231 error code the read function failed. The most probable cause for the read to fail is because there was no data available at the instrument's output buffer.

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized
236 Error interpreting instrument response

Function Name : eip123x_reset

C Syntax : int eip123x_reset(Instrument_ID);
int Instrument_ID;

QuickBASIC Syntax : Error.Code% = eip123x.reset(Instrument.ID%)

Purpose : Panel Name: Reset

Description: This function resets the instrument to its default state.

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number of instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized
236 Error interpreting instrument response

Function Name : eip123x_resolution

C Syntax : int eip123x_resolution(Instrument_ID, Resolution);
int Instrument_ID;
int Resolution;

QuickBASIC Syntax : Error.Code% = eip123x.resolution(Instrument.ID%, _
Resolution%)

Purpose : Panel Name: Resolution

Description: This function controls the frequency measurement resolution. Select the desired resolution according to the following table:

Resolution	Freq	Gate Time
2	100 Hz (Band 0 only)	10 ms
3	1 kHz	1 ms
4	10 kHz	100 ys
5	100 kHz	10 ys
6	1 MHz	1ys
7	10 MHz	1ys
8	100 MHz	1ys

9 1 GHz 1ys

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Resolution

2 - 100 Hz -
3 - 1 kHz - (Default)
4 - 10 KHz -
5 - 100 KHz -
6 - 1 MHz -
7 - 10 MHz -
8 - 100 MHz -
9 - 1 GHz -

Control Name: Resolution

Description: This parameter controls the frequency measurement resolution. Select the desired resolution according to the following table:

Resolution	Freq	Gate Time
2	100 Hz (Band 0 only)	10 ms
3	1 kHz	1 ms
4	10 kHz	100 us
5	100 kHz	10 us
6	1 MHz	1 us
7	10 MHz	1 us
8	100 MHz	1 us
9	1 GHz	1 us

Variable Type: Integer

Valid Range: 2 - 100 Hz
3 - 1 kHz
4 - 10 kHz
5 - 100 kHz

6 - 1 MHz
7 - 10 MHz
8 - 100 MHz
9 - 1 GHz

Default Value: 3 - 1 kHz

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Resolution out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized

Function Name : eip123x_revision_query

C Syntax : int eip123x_revision_query(Instrument_ID, Driver_Revision, Firmware_Revision);
int Instrument_ID;
char *Driver_Revision;
char *Firmware_Revision;

QuickBASIC Syntax : Error.Code% = eip123x.revision.query(Instrument.ID%, _ Driver.Revision$, Firmware.Revision$)

Purpose : Panel Name: Revision Query

Description: This function returns the revision numbers of the instrument driver and instrument firmware, and tells the user what firmware the driver is compatible with, i.e., Driver Revision "A1.0" is compatible with Firmware Revision(s) "3.00".

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number of instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Driver Revision
Control Name: Driver Revision

Description: This control returns the Instrument Driver Revision.

Variable Type: String (char *)

Firmware Revision
Control Name: Firmware Revision

Description: This control returns the Instrument's Firmware Revision.

Variable Type: String (char *)

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized
236 Error interpreting instrument response

Function Name : eip123x_sample_rate

C Syntax : int eip123x_sample_rate(Instrument_ID,

Sample_Rate_mSec);
int Instrument_ID;
long Sample_Rate_mSec;

QuickBASIC Syntax : Error.Code% = eip123x.sample.rate(Instrument.ID%, _
Sample.Rate.mSec&)

Purpose : Panel Name: Sample Rate

Description: This function sets a delay between measurement values (0 to 100 S to a 10 mS resolution). Signal tracking is performed before each measurement. This process adds approximately 100 ms to the delay between. This means that even if the Sample Rate is set to 0 there will still be a 100 ms delay between measurements.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Sample Rate (mSec)

Control Name: Sample Rate

Description: Sets a delay between measurement values

Variable Type: Long

Valid Range: 0 to 100,000 ms, by 10 ms resolution

Default Value: 0 ms (maximum sampling rate)

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Sample Rate out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized

Function Name : eip123x_save_default_setup

C Syntax : int eip123x_save_default_setup(Instrument_ID);
int Instrument_ID;

QuickBASIC Syntax : Error.Code% = eip123x.save.default.setup(Instrument.ID%)

Purpose : Panel Name: Save Default Setup

Description: The counter's setup is recalled from Save/Recall register 0 at power-up or upon receipt of the the initialize command. The register can be changed by the user in order to alter the instrument's default settings. This function restores register 0 to the instrument's factory default settings.

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number

instruments that can be controlled
by this driver in a single system.
The value is set internally in the
driver. The typical value is 10.

Return Value : Control Name: Error Code

Description: Displays status relating to the
function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Mode out of range
-3 Setup Register out of range

230 Error writing to instrument
232 Instrument not initialized

Function Name : eip123x_set_meas_type

C Syntax : int eip123x_set_meas_type(Instrument_ID,
Measurement_Type);
int Instrument_ID;
int Measurement_Type;

QuickBASIC Syntax : Error.Code% = eip123x.set.meas.type(Instrument.ID%, _
Measurement.Type%)

Purpose : Panel Name: Set Measurement Type

Description: This function selects which type
of measurement will be output
by the instrument when the Get
Measurement function is used:
Frequency, Width, or Period.
Also, use Set Measurement Type
before using the trigger
if a triggered mode is selected.

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned
by the Initialize function to
select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Measurement Type
0 - Frequency - (Default)
1 - Width -
2 - Period -
Control Name: Measurement

Descriptor: This parameter controls the measurement to output.

Variable Type: Integer

Valid Range: 0 - Frequency
1 - Width
2 - Period

Default Value: 0 - Frequency

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Measurement Type out of range

230 Error writing to instrument
232 Instrument not initialized

Function Name : eip123x_special_funct

C Syntax : int eip123x_special_funct(Instrument_ID, Select_Special, Returned_Reading);
int Instrument_ID;
int Select_Special;

double *Returned_Reading;

QuickBASIC Syntax : Error.Code% = eip123x.special.funct(Instrument.ID%, _
Select.Special%, Returned.Reading#)

Purpose : Panel Name: Special Functions

Description: This function selects the special function from the special functions which this driver supports.

Call the Terminate all specials function to terminate all active special special functions.

Functions supported by this driver are:

Read Relative Frequency:
When this function is activated, the counter assigns a negative value to the last frequency reading and enters it into the frequency offset register (overwriting any existing frequency offset value). The last input frequency in this case means the actual frequency of the input signal, not including the effects of a frequency multiplier or other special function. The counter outputs the the relationship, i.e. the difference, between the last input frequency and the current one, subject to any other functions activated. It continues to do so until the function is terminated.

Display IF Readings:
When this function is activated, the counter assigns a negative value to the local oscillator (LO) frequency and enters it into the frequency offset register (overwriting any existing frequency offset value). The counter then subtracts the LO frequency from the input frequency and outputs the resulting IF. It continues to do so until the function is terminated.

100 MHz Self-Test

When this function is activated, the counter will count its internal VCO. This function is use to verify that the count chain, gate generator, and the VCO are operational.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Select Special

0 - Self Test (Default)
1 - Relative Frq
2 - Display IF

Control Name: Select Special

Description: Selects or terminates the Special functions. Functions which are supported are:

100 MHz Self-Test: When this function is activated, the counter counts an internally generated test signal. The counter returns a frequency reading of 100 MHz to indicate it passed Self-Test.

Read Relative Frequency: When this function is activated, the counter assigns a negative value to the last frequency reading and enters it into the frequency offset register (overwriting any existing frequency offset value). The last input frequency in this case means the actual frequency of the input signal, not including the effects of a frequency multiplier or other special function. The counter

outputs the the relationship, i.e. the difference, between the last input frequency and the current one,subject to any other functions activated. It continues to do so until the function is terminated.

Display IF Readings:
When this function is activated, the counter assigns a negative value to the local oscillator (LO) frequency and enters it into the frequency offset register (overwriting any existing frequency offset value). The counter then subtracts the LO frequency from the input frequency and outputs the resulting IF. It continues to do so until the function is terminated.

Variable Type: Integer

Valid Range: 0 - 100 MHz Self-Test
1 - Read Relative Frequency
2 - Display IF Readings

Default Value: 0 - 100 MHz Self-Test

Returned Reading
Control Name: Returned Reading

Description: Outputs the Self-Test, Relative or IF frequency.

Variable type: Double

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Select Special out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized
236 Error interpreting instrument response

Function Name : eip123x_terminate_spec

C Syntax : int eip123x_terminate_spec(Instrument_ID);
int Instrument_ID;

QuickBASIC Syntax : Error.Code% = eip123x.terminate.spec(Instrument.ID%)

Purpose : Panel Name: Terminate All Specials

Description: This Function terminates all previously implemented special functions.

Parameter List : Instrument ID

Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0	Success
-1	Instrument ID out of range
230	Error writing to instrument
231	Error reading from instrument
232	Instrument not initialized

Function Name : eip123x_time_base

C Syntax : int eip123x_time_base(Instrument_ID, Time_Base_Source);
int Instrument_ID;
int Time_Base_Source;

QuickBASIC Syntax : Error.Code% = eip123x.time.base(Instrument.ID%, _
Time.Base.Source%)

Purpose : Panel Name: Time Base Source

Description: This function selects the source of the 10 MHz frequency reference signal.

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Time Base Source
1 - External
0 - Internal (Default)
Control Name: Time Base Source

Description: Controls the source of the 10 MHz frequency reference signal.

Variable Type: Integer

Valid Range: 0 - Internal
1 - External

Default Value: 0 - Internal

Return Value : Control Name: Error Code

Description: Displays status relating to the

function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Time Base Source out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized

Function Name : eip123x_trig_setup

C Syntax : int eip123x_trig_setup(Instrument_ID, Sync_Trig_Line,
Async_Trig_Line);
int Instrument_ID;
int Sync_Trig_Line;
int Async_Trig_Line;

QuickBASIC Syntax : Error.Code% = eip123x.trig.setup(Instrument.ID%, _
Sync.Trig.Line%, Async.Trig.Line%)

Purpose : Panel Name: Trigger Set Up

Description: This function selects the
triggering mode, configures
the INT/EXT time base reference,
and selects the TTLTRG line pair
to be used in the Asynchronous
Trigger Protocol. The valid range
is 0 to 3, corresponding to TTLTRG
pairs 0 & 1, 2 & 3, 4 & 5, and
6 & 7 respectively. Issuing
ASYNC 9 disables the protocol.
Or it selects the TTLTRG line
to be used in the Synchronous
Trigger Protocol. the valid range
is 0 to 7. Issuing SYNC 9 disables
the protocol.

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned
by the Initialize function to
select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Sync Trig Line
0 - 0 -
1 - 1 -
2 - 2 -
3 - 3 -
4 - 4 -
5 - 5 -
6 - 6 -
7 - 7 -
9 - Disabled - (Default)
Control Name: Sync Trig Line

Description: Selects the TTL TRG line to be used in the Synchronous Trigger Protocol. The SYNC message enables the SYNCHRONOUS TTLRG trigger protocol

Variable Type: Integer

Valid Range: 0 - Line 0
1 - Line 1
2 - Line 2
3 - Line 3
4 - Line 4
5 - Line 5
6 - Line 6
7 - Line 7
9 - Protocol Disabled

Default Value: 9 - Protocol Disabled

Async Trig Line
0 - 0 & 1 -
1 - 2 & 3 -
2 - 4 & 5 -
3 - 6 & 7 -
9 - Disabled - (Default)
Control Name: Async Trig Line

Description: Selects the TTL TRG line pair to be used in the Asynchroneous Trigger Protocol as follows:

0: 0 & 1
1: 2 & 3
2: 4 & 5
3: 6 & 7

Variable Type: Integer

Valid Range: 0 - Line pair 0 & 1
1 - Line pair 2 & 3
2 - Line pair 4 & 5
3 - Line pair 6 & 7
9 - Protocol Disabled

Default Value: 9 - Protocol Disabled

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Sync Trig Line out of range
-3 Async Trig Line out of range

230 Error writing to instrument
231 Error reading from instrument
232 Instrument not initialized

Function Name : eip123x_trigger_func

C Syntax : int eip123x_trigger_func(Instrument_ID, Function);
int Instrument_ID;
int Function;

QuickBASIC Syntax : Error.Code% = eip123x.trigger.func(Instrument.ID%, _
Function%)

Purpose : Panel Name: Trigger Function

Description: This function selects and implements one of the two internal trigger functions, trigger or reset.

Trigger triggers a new measurement cycle. If a

measurement cycle is in progress
it will be aborted.

Reset resets
the converter and restarts the
measurement process. If a
measurement is found, a
measurement will be taken, even if
the counter is in Hold.

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned
by the Initialize function to
select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of
this Instrument

The Maximum Instances of this
Instrument is the maximum number
instruments that can be controlled
by this driver in a single system.
The value is set internally in the
driver. The typical value is 10.

Function
1 - Trigger
0 - Reset (Default)
Control Name: Function

Description:
1. Reset: Resets the converter and
restarts the signal acquisition
process. If a signal is found,
a measurement will be taken,
even if the counter is in Hold.

2. Trigger: Triggers a new
measurement cycle, If a
measurement cycle is in
progress, it will be aborted.

Variable Type: Integer

Valid Range: 1 - Trigger
0 - Reset

Default Value: 0 - Reset

Return Value : Control Name: Error Code

Description: Displays status relating to the function call.

Variable Type: Integer

Error Codes:

0 Success

-1 Instrument ID out of range
-2 Function out of range

230 Error writing to instrument
232 Instrument not initialized

Function Name : eip123x_write

C Syntax : int eip123x_write(Instrument_ID, Write_Buffer);
int Instrument_ID;
char *Write_Buffer;

QuickBASIC Syntax : Error.Code% = eip123x.write(Instrument.ID%, _
Write.Buffer$)

Purpose : Panel Name: Write To Instrument

Description: Writes commands and queries to the instrument to modify parameters and query device settings.

Parameter List : Instrument ID
Control Name: Instrument ID

Description: Accepts the Instrument ID returned by the Initialize function to select the desired instrument.

Variable Type: Integer

Valid Range: 1 to Maximum Instances of this Instrument

The Maximum Instances of this Instrument is the maximum number of instruments that can be controlled by this driver in a single system. The value is set internally in the driver. The typical value is 10.

Write Buffer
Control Name: Write Buffer

Description: The user can use this control to send common commands and queries to the instrument. This control can also be used to write any valid command to the instrument.

Variable Type: String

Default Value: "*RST"

NOTES: (1) The command or query to be sent to the instrument may be a literal enclosed in double quotes i.e. "*RST" or may be contained in a variable of type string.

(2) If the number of bytes to be written is greater than the size of "cmd" in the source code then the size of cmd must be increased. The original size of cmd is 50 bytes (cmd[50]) the largest cmd may be declared as is 32,767 bytes.

Return Value : Control Name: Error Code

Description: Displays the results of the function call.

Variable Type: Integer

Error Codes:

0	Success
-1	Instrument ID out of range
230	Error writing to instrument
232	Instrument not initialized

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP