

Partnership Courtyard, The Ramparts,

Dundalk, Ireland

Version 6.6.0.0

March 12, 2021

www.measuresoft.com

+353 42 933 2399

This document is the copyright of Measuresoft and may not be modified, copied or distributed in any form whatsoever without the prior permission of Measuresoft.

Measurement Computing User Manual

1. Installation and Setup 3

2. Configuration 5

3. Channel Configuration (Analogs and Counters) 6

3.1 Enable Channel 6

3.2 Tag 6

3.3 Description 6

3.4 Units 7

3.5 Device Specific Button 7

3.6 Scaling 7

3.6.1 Auto Scaling 7

3.7 Significant Change 8

3.8 Event Checking 8

3.8.1 High Limit 8

3.8.2 Low Limit 8

3.9 Alarm Checking 8

3.9.1 Enable Alarm Checking 8

3.9.2 Drive Common Alarm 9

3.9.3 Alarm Limit 9

3.9.4 Warning and Limit 9

3.9.5 Hysteresis 9

3.9.6 Priority 9

3.9.7 Alarm Delay 9

3.9.8 Alarm Message 9

4. Channel Configuration (Digitals) 10

4.1 Enable Channel 11

4.2 Tag 11

4.3 Description 11

4.4 LowState Description 11

4.5 HighState Description 12

4.6 Event Checking 12

4.7 Alarm Checking 12

4.7.1 AlarmState 12

4.7.2 Alarm Priority 12

4.7.3 Common Alarm 12

4.7.4 Alarm Delay 12

4.7.5 Alarm Message 12

5. Advanced Device Configuration 13

5.1 AutoEnable Device 13

5.2 Scan Rate 13

5.3 Save Outputs 13

5.4 Device Specific Button 14

6. Using Measurement Computing Channels 15

7. Driver Error Codes 16

# Installation and Setup

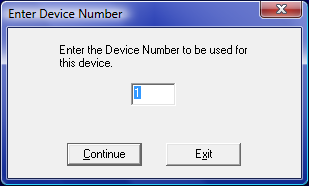
## 1Gb of additional RAM is required to run the Measurement Computing driver on the server machine.

* 1. *Measurement Computing InstaCal/Universal Library* must be installed.
  2. Insert the CD labelled *Measurement Computing InstaCal/Universal Library*, which comes with your Measurement Computing Controller board, into your CD-ROM drive.

Note: If you are performing this installation under Windows , you must be logged on as an administrator.

The Installation program will provide you with further information through the course of the installation.

* 1. The Measurement Computing driver can now be installed by running setup from the Drivers\Measurement\_Computing installation folder.
  2. Follow the instructions in the setup. When prompted with the following



**1.7**  The appropriate driver is now installed on your system.

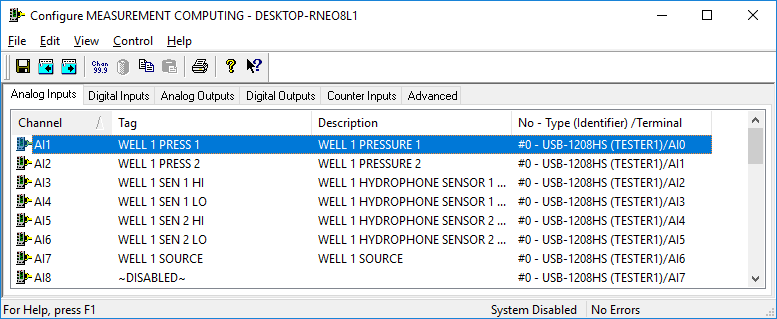
You can now use the Measurement Computing hardware.

# Configuration

The first time the system is configured it is necessary to enable and configure all devices you require. To configure a particular device select the ***Devices*** option from the main menu followed by the appropriate device.

This will launch an application to configure the device. You will be presented with a set of tabs as shown below.

To configure a channel select a group of channels by clicking on the appropriate channel tab.



The board number, board identifier and board terminal associated with each channel is displayed on the right hand side of the channel list. The Measurement Computing driver can handle multiple boards. Board numbers are assigned using the Instacal utility.

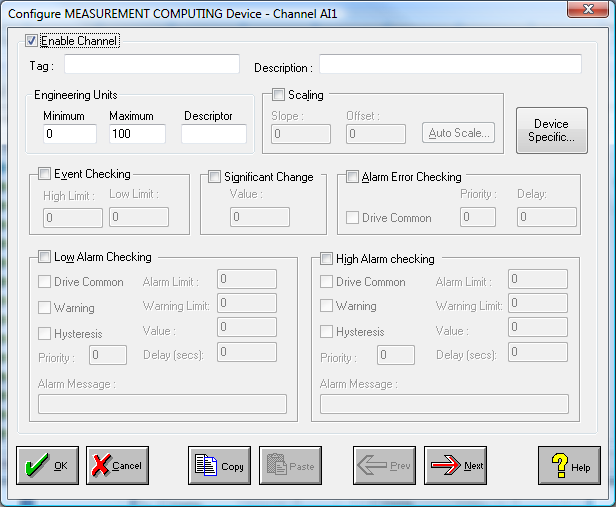
From the list provided select a channel and double-click.

Alternatively, you can select a channel and then click on the Configure Channel button.

This will launch a channel configuration dialog which enables you to configure individual channels.

If you enable a channel which has no associated board number, board identifier and board terminal, an error is generated which will prevent scanning so it is important to only configure channels with an associated board number, board identifier and board terminal.

# Channel Configuration (Analogs and Counters)



## Enable Channel

The Enable Channel check box must be checked to enable, and allow this channel to be configured and ultimately included with all other configured channels in the overall system.

## Tag

The Tag field is a 12 character alphanumeric field that can contain channel information or wiring schedule references.

## Description

The Description field is a 32 character alphanumeric field in which a description of the channel can be detailed.

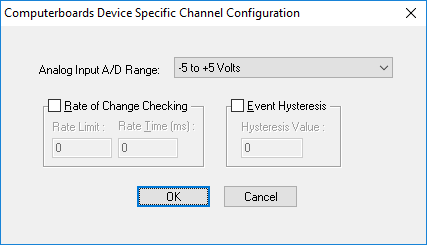
## Units

A 4 character field available to describe the units of the output.

## Device Specific Button

If the Device Specific Button is visible, then click on it to configure channel features that are specific only to the type of device you are configuring.

For analog inputs the following is displayed.

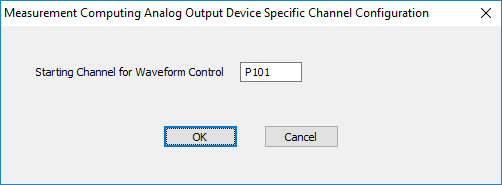


The channel specific Analog Input A/D Range is only used if sampling is being performed at 100Hz or slower.

Rate of Change is used to generate events for high speed rate of change checking. The rate limit is the rate of change being detected and rate time is the time period over which the rate should occur to generate an event. The rate time must be less than 5 seconds.

Event Hysteresis is used to eliminate multiple events on noisy signals.

For Analog outputs the following is displayed



The Starting channel for waveform control is an optional User Analog Channel which can be used for controlling waveforms on one or more Measurement Computing Analog outputs.

The User Analog channels are configured in the following order

1. Pulse Enable
2. Pulse Command
3. Pulse Count Reset
4. Waveform Type

0 – Triangle

1 – Sine

2 - Square

3 – Sweep Triangle

4 – Sweep Sine

5 – Sweep Square

6 – Ramp and Hold

1. Number of Waveform repititions/cycles
2. Waveform Amplitude in V
3. Waveform DC Start Offset in V
4. Waveform Start Offset in degrees 0 to 360
5. Waveform frequency in Hz
6. Waveform Length in Seconds
7. Waveform Starting Frequency for Sweep Waves
8. Waveform Ending Frequency for Sweep Waves
9. Ramp Time for Ramp and Hold
10. Hold Time for Ramp and Hold
11. Delay at end of each Pulse
12. Feedback on Waveform Active
13. Feedback on Pulse Count since last reset
14. Feedback on Pulse Count in current series of pulses
15. Feedback on Number of Pulses to be generated
16. Feedback on Number of Seconds since last Pulse

## Scaling

NB. Scaling is only available to Analogue channels that are not Output channels.

Some transducers give a number of pulses, or a frequency output proportional to their full scale range. To enable the Scaling utility check the Scaling Check box. The Slope and Offset values can be entered directly into the text boxes. The formula applied is:

y = mx + c where: m is SLOPE

x is the measured value.

c is the OFFSET

### Auto Scaling

Click On the Auto Scale button if you want the scale and offset values calculated automatically. A dialog box will be displayed. Enter the values in the text boxes. The low measured value, and the high measured value, the output range of the transducer. When the fields have been completed, and assuming the System is enabled click on the Apply button. Under the heading Current Values the actual measured value will be shown, as well as the Engineering Value.

Click on OK to accept the scaling, or Cancel to abort the Auto Scaling feature.

NB. Scaling will not be applied to the channel, even if the system is enabled, until the system is next enabled or the InstruNet Device is reconfigured

## Significant Change

Significant changes on InstruNet channels do not trigger logger events.

The significant change status of a channel can be monitored from one scan to the next.

## Event Checking

Event checking is used, if required to trigger a logger to record information on a number of channels during an event. Check the Event Checking check box if this channel is to trigger an event. Events are detected on inputs using data acquired at 1Khz. Events are detected on outputs using the configured scan rate.

### High Limit

A value, in engineering units, entered in this text box will define the level that, if exceeded, will cause an event trigger.

### Low Limit

A value entered in this text box will define the level that if the channel result falls below will cause an event trigger.

## Alarm Checking

Alarm checking is available on all channels throughout the system. Low Alarm and High Alarm levels can be configured independent of each other. If the channel output exceeds the High Alarm limit then an alarm will be triggered as it will if the output goes below the Low Alarm limit. Alarms and warnings are detected at the configured scan rate.

When monitoring channels, if the high or low alarm is triggered, then the fact will be annotated alongside the other channel information in the Channel Monitor. To configure the Alarm Checking section of the device complete the options as follows for either or both the High Alarm and Low Alarm checking.

### Enable Alarm Checking

Check either the Low Alarm Checking or High Alarm Checking or both check boxes to enable the facility.

### Drive Common Alarm

A common alarm is a single digital output which will switch on when any channel with the Drive Common Alarm enabled goes into an alarm state. Check this box if a link to the Common Alarm is required.

### Alarm Limit

Specifies the value which will trigger this alarm. for Low Alarm Checking it will be any value <= the Alarm Limit and for High Alarm Checking it will be any value >= the Alarm Limit.

### Warning and Limit

If required, a warning can be displayed when a channel reaches a limit close to the alarm limit. For low alarm checking, the limit must be less than the alarm limit. For high alarm checking, the warning limit must be less than the alarm limit

### Hysteresis

Hysteresis can prevent 'noisy' channels from reporting multiple alarms when the average reading is close to the alarm threshold. Check the box if this feature if needed. Enter the value of the dead band in the corresponding value field.

### Priority

Enter or edit the number in the text box to allocate the priority of this alarm. Alarm priority ranges are from 0 to 255.

### Alarm Delay

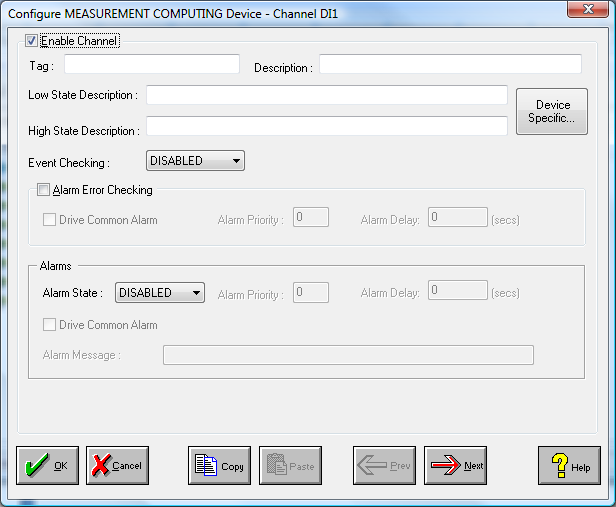
Enter the time, in seconds, between the channel value entering the alarm state and the system flagging an alarm.

### Alarm Message

An Alarm Message can be defined to be displayed on the Status line of the Main Window when a channel goes into an alarm state. Enter the message, up to 32 characters, that is to appear in the event of an alarm.

# Channel Configuration (Digitals)

Counter and digital inputs are not scanned at sample speed speeds faster than 100Hz but at the slower scan rate specified in advanced section below.



## 

## Enable Channel

The Enable Channel check box must be checked to enable, and allow this channel to be configured and ultimately included with all other configured channels in the overall system.

## Tag

The Tag field is an 12 character alphanumeric field that can contain channel information or wiring schedule references.

## Description

The Description field is a 32 character alphanumeric field in which a description of the channel can be detailed.

## LowState Description

A 32 character field in which to enter a description of the low state of the channel.

## HighState Description

A 32 character field in which to enter a description of the high state of the channel.

## Event Checking

Event checking is used, if required to trigger a logger to record information on an event. If this facility is required click on the drop down list box and select OFF, HIGHSTATE, or LOWSTATE as appropriate. Events are detected on inputs using data acquired at 1Khz. Events are detected on outputs using the configured scan rate.

## Alarm Checking

### AlarmState

Alarm checking is available on all channels throughout the system. To configure alarm checking on this channel click on the drop down box and select OFF, HIGHSTATE, or LOWSTATE as appropriate. If the channel's output state changes to an AlarmState an alarm will be triggered on the channel. When monitoring channels, if the alarm is triggered, the fact will be annotated alongside the other channel information in the Channel Monitor . Alarms and warnings are detected at the configured scan rate.

### Alarm Priority

Enter the priority of the alarm triggered by this channel. Alarm priority ranges are from 0 to 255.

### Common Alarm

Channels can be configured to trigger a Common Alarm. A common alarm is a single digital output which will switch on when any channel with the Drive Common Alarm enabled goes into an alarm state.

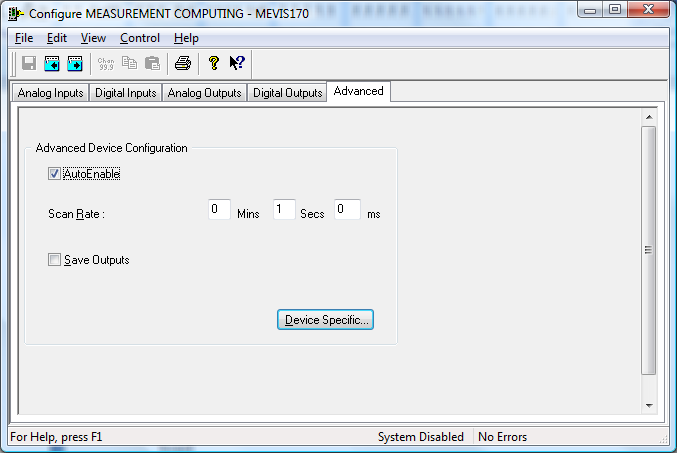
### Alarm Delay

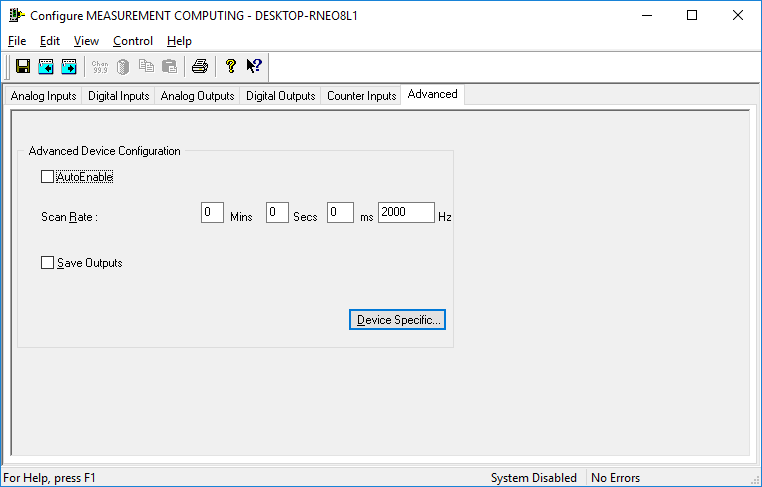
Enter the time, in seconds, between the channel value entering the alarm state and the system flagging an alarm.

### Alarm Message

An Alarm Message can be defined to be displayed on the Status line of the Main Window when a channel goes into an alarm state.

# Advanced Device Configuration





When the system is enabled and the device is enabled this window will display the Achieved Scan Rate . If the device is not scanning then any error associated with the device will be displayed instead.

## AutoEnable Device

To ensure that the device is enabled on the system check the Enable Device box.

## Scan Rate

To set the rate at which the device will scan, edit the text boxes associated with the Scan Rate field. The scan rate can also be specified in Hz for scan rates faster than 1kHz

## Save Outputs

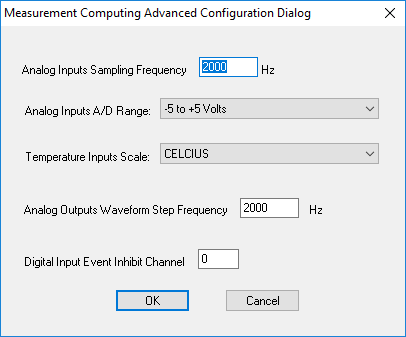
To enable this utility check the Save Outputs flag. All values in output channels are saved to disk when the system is disabled. The next time the system is restarted the values which were previously in output channels will be restored to the appropriate channel number.

**By Tag**

Channel values can be saved and restored to channels using the channel tag instead of the channel number. In this way, channels can be rearranged within the modules and as long as the channel tags remain the same, the correct channel values will be restored to the appropriate channel number.

## Device Specific Button

If the Device Specific Button is visible and the systems is disabled, then click on it to configure features that are specific only to this device.



Sampling frequency is the rate at which samples are collected from the board. Ensure that the number of analog input channels by the sampling frequency can be handled by the board. Where the sampling frequency is faster than the scan rate, samples are averaged.

The Analog Input A/D Range is only used if sampling is being performed at speeds higher than 100Hz.

Temperature scale is the scale for temperature analog inputs.

Analog Outputs Waveform Step Frequency determines the maximum resolution of an output waveform. In the above example up to 2000 points are used per second of waveform.

The Digital Input Event Inhibit Channel issued to suppress event logging. This field is the number of the digital input channel. If it is zero, no channel is used. The digital channel must be in a high state to suppress events.

# Using Measurement Computing Channels

With the exception of logged analog voltage inputs, Measurement Computing channels including thermocouple channels are only processed at the scan rate specified in the Advanced Device settings. Therefore alarms and calculations of Measurement Computing channels an only be performed at a maximum speed of 100Hz.

Loggers on the other hand will process Measurement Computinganalog inputs at speeds faster than 100Hz. It is important to make sure that the buffer allocated to logging is adequate. If not enough logging buffers are allocated, you may experience poor performance such as erratic mouse movement.

When loggers are recording Measurement Computing input channels, they only start logging when the system is enabled and the Measurement Computing driver has started. If a logger remains active when the system is disabled and subsequently re-enabled, the logger will stop logging. Therefore when disabling the system, you must disable all loggers that log Measurement Computing input channels.

# Driver Error Codes

Each entry in the list has four parts:

* Error code number
* Symbolic name
* Error message
* Explanation of the error

|  |  |  |
| --- | --- | --- |
| 1 | **BADBOARD** | Invalid board number |
|  | The BoardNum argument that was specified does not match any of the boards that are listed in the configuration file. Run the configuration program to check which board numbers are configured. | |
| 2 | **DEADDIGITALDEV** | Digital device is not responding. |
|  | The digital device on the specified board is not responding. Either the board was installed incorrectly or the board is defective. Run the configuration program and make sure that the correct board was installed. | |
| 3 | **DEADCOUNTERDEV** | Counter device is not responding. |
|  | The counter device on the specified board is not responding. Either the board was installed incorrectly or the board is defective. Run the configuration program and make sure that the correct board was installed. | |
| 4 | **DEADDADEV** | D/A is not responding. |
|  | The D/A device on the specified board is not responding. Either the board was installed incorrectly or the board is defective. Run the configuration program and make sure that the correct board was installed. | |
| 5 | **DEADADDEV** | A/D is not responding. |
|  | The A/D device on the specified board is not responding. Either the board was installed incorrectly or the board is defective. Run the configuration program and make sure that the correct boards was installed. | |
| 6 | **NOTDIGITALCONF** | Selected board does not have digital I/O. |
|  | A digital I/O function or method was called with a board number that referred to a board that does not support digital I/O. Run the configuration program to see which type of board that board number refers to. | |
| 7 | **NOTCOUNTERCONF** | Selected board does not have a counter. |
|  | A counter function or method was called with a board number that referred to a board that does not have a counter. Run the configuration program to see which type of board that board number refers to. | |
| 8 | **NOTDACONF** | Selected board does not have a D/A. |
|  | An analog output function or method was called with a board number that referred to a board that does not have an analog output (D/A). Run the configuration program to see which type of board the board number refers to. | |
| 9 | **NOTADCONF** | Selected board does not have an A/D. |
|  | An analog input function or method was called with a board number that referred to a board that does not have an analog input (A/D). Run the configuration program to see which type of board that board number refers to. | |
| 10 | **NOTMUXCONF** | Selected board does not have thermocouple inputs. |
|  | A thermocouple input function or method was called with a board number that does not support thermocouple inputs, or is not connected to an EXP board. Run the configuration program to view/change the board configuration. | |
| 11 | **BADPORTNUM** | Invalid digital port number. |
|  | The Port number that was specified for a digital I/O function does not exist on the specified board. | |
| 12 | **BADCOUNTERDEVNUM** | Invalid counter device. |
|  | The Counter Number that was specified for a counter function does not exist on the board that was specified. | |
| 13 | **BADDADEVNUM** | Invalid D/A device. |
|  | The D/A channel that was specified for an analog output function does not exist on the board that was specified. | |
| 14 | **BADSAMPLEMODE** | Invalid sample mode. |
|  | A sample mode that is not supported on this board (SINGLEIO, DMAIO or BLOCKIO) was specified in the Options argument. Try running the function without setting any of the Sample Mode options. | |
| 15 | **BADINT** | Board configured for invalid interrupt level. |
|  | No interrupt was selected in InstaCal and one is required, or the board is set for "compatible mode" and the interrupt level selected is not supported in this mode. Interrupts above 7 are not valid in compatible mode. Either change the switch setting on the board to "enhanced mode", or change the interrupt level with the configuration program to something less than 8. | |
| 16 | **BADADCHAN** | Invalid A/D channel number. |
|  | An invalid channel argument was passed to an analog input function or method. The range of valid channel numbers depends on which A/D board you are using - refer to the board manual. For some boards it also depends on how the board is configured (with a switch). For those boards run the configuration program and check how many channels the board is configured for. | |
| 17 | **BADCOUNT** | Invalid count. |
|  | An invalid Count argument was specified to a function or method. If this error occurs during cbAInScan()/AInScan(), increasing the Count should correct the problem. For boards using DMAIO, adjust the data buffer and Count above (HighChan-LowChan+1)\*Rate/100 for CONTINUOUS mode scans. However, those boards using BLOCKIO, require a user buffer and Count large enough to hold at least one half FIFO worth of samples (typically, 512 samples) for CONTINUOUS mode scans. | |
| 18 | **BADCNTRCONFIG** | Invalid counter configuration specified. |
|  | An invalid Config argument was passed to cbC8254Config/C8254Config. The only legal values are HIGHONLASTCOUNT, ONESHOT, RATEGENERATOR, SQUAREWAVE, SOFTWARESTROBE and HARDWARESTROBE. | |
| 19 | **BADDAVAL** | Invalid D/A value. |
|  | An invalid D/A value was passed as an argument/parameter to an analog output function or method. The only legal values are 0 to 4,095 for 12-bit boards or 0 to 65,535 for 16-bit boards (see the "[Visual Basic signed integers](mk:@MSITStore:C:\Program%20Files%20(x86)\Measurement%20Computing\DAQ\ULHelp.chm::/Users_Guide/Counter_Boards/CTR-hardware.htm)" discussion at the beginning of the "Counter Boards" section in the Universal Library User's Guide). | |
| 20 | **BADDACHAN** | Invalid D/A channel number. |
|  | An invalid D/A channel was passed as an argument/parameter to an analog output function or method. The legal range of values depends on which D/A board you are using. Refer to the board manual to find how many D/A channels it has. | |
| 22 | **ALREADYACTIVE** | Background operation already in progress. |
|  | An attempt was made to start a second background process on the same board before the first one had completed. Background processes are started whenever the BACKGROUND option is used by cbCStoreOnInt()/CStoreOnInt(). To stop a background operation, call cbStopBackground()/StopBackground(). To wait for a background process to complete. To wait for a background process to complete call cbGetStatus()/GetStatus() and wait for status=IDLE. | |
| 23 | **PAGEOVERRUN** | DMA transfer crossed page boundary, may have gaps in data. |
|  | When a DMA transfer crosses a 64K memory page boundary on boards without FIFO buffers, there may be a small gap (missing samples) in the data. For applications requiring high speed transfers of greater than 32K samples, please select a board with a FIFO buffer. For boards without, check the data for gaps and do not specify rates over that at which gapless data may be taken. This is system-specific, so you must determine the rate by experimentation. | |
| 24 | **BADRATE** | Invalid sampling rate. |
|  | Invalid sampling rate argument was specified. The rate was either zero, a negative number or it was higher than the selected board supports. Refer to board-specific information for board maximum rates. | |
| 25 | **COMPATMODE** | Board switches set for Compatible mode. |
|  | An operation was attempted that is not possible when the board's switch is" set for 'compatible' operation. The Most likely causes are due to using the BLOCKIO option or the pre-triggering functions. Either turn off the 'compatible' mode switch on the board or don't use the BLOCKIO option or the pre-triggering functions. | |
| 26 | **TRIGSTATE** | Incorrect initial trigger state - trigger must start at TTL low. |
|  | Boards that use "polled gate" triggering require that the trigger be "off" when a pre-trigger functions is first called. It then waits for the trigger signal. Make sure that the Trigger Input line (usually D0) is held at TTL low before calling the pre-trigger function. | |
| 27 | **ADSTATUSHUNG** | A/D is not responding. |
|  | The A/D board is not responding as it should. Usually indicates some kind of hardware problem - either defective hardware or more than one board at the same base address. | |
| 28 | **TOOFEW** | Trigger occurred before the requested number of samples were collected. |
|  | A pre-trigger function or method was called and the trigger signal occurred before the requested number of samples could be collected. This is only a warning message. The function or method continued anyway. The data that was returned to the array will contain fewer than the expected number of points. The function or method will return the actual number of pre-trigger points and the total number of points. You can use these two values to find your way around the data in the array. | |
| 29 | **OVERRUN** | Data overrun - data was lost. |
|  | Data was lost during an analog input because the computer could not keep up with the A/D sampling rate. This typically can only happen with the file input functions or methods, or by using SINGLIO mode. Possible solutions include lowering the sampling rate, defragmenting the "streamer" file, switching to a RAM disk, or lowering the count. | |
| 30 | **BADRANGE** | Invalid voltage or current range. |
|  | Invalid Range argument was specified to an analog input or output function or method. The board does not support the gain you specified. Refer to board-specific information for a list of allowable ranges. | |
| 31 | **NOPROGGAIN** | This A/D board does not have programmable gain. |
|  | Invalid Range argument was passed to an analog input function or method. The selected board does not support programmable gains so the only valid Range argument is 0. (This argument is ignored for these board types in later versions of the library.) | |
| 32 | **BADFILENAME** | Specified file name is not valid. |
|  | The FileName argument/parameter that was passed to a file function is not valid. It is either an empty string or a NULL pointer. | |
| 33 | **DISKISFULL** | Disk is full, could not complete operation. |
|  | A file operation failed before completing because the disk that it was writing to is full. Try erasing some files from the disk. If this error occurred during either cbFileAInScan()/FileAInScan() or cbFilePretrig()/FilePretrig() it indicates another problem. The disk space for these commands should have been previously allocated with the MAKESTRM.EXE program. If this error is generated when data is being collected it indicates that you did not allocate a large enough file with MAKESTRM.EXE. | |
| 34 | **COMPATWARN** | Board switch set to compatible mode - sampling speed may be limited. |
|  | The board's switch is set for "compatible mode." When in "compatible mode," BLOCKIO transfers are not possible. BLOCKIO sampling was specified but it has automatically been changed to DMAIO transfers. The maximum sampling rate will be limited to the maximum rate for DMA transfers. Change the "compatible mode" switch on the board if you want to use BLOCKIO transfers. | |
| 35 | **BADPOINTER** | Pointer is not valid. |
|  | An invalid (NULL) pointer was passed as an argument/parameter to a function or method. | |
| 37 | **RATEWARNING** | Sample rate may be too fast for SINGLEIO mode. |
|  | The specified sampling rate MAY be too high. The maximum allowable sampling rate depends very much on the computer that the program is running on. This warning is generated based on the slowest CPU speed. Your computer may be able to sustain faster rates, but, you should expect the computer to lock up (fail to respond to keyboard input) if you do exceed the sampling rate your computer can sustain. | |
| 38 | **CONVERTDMA** | CONVERTDATA cannot be used with DMAIO and BACKGROUND. |
|  | The CONVERTDATA and BACKGROUND options can not be used together when the board is transferring data via DMA. Possible solutions include: Use cbAConvertData()/AConvertData() to convert the data after it is collected. Don't use the BACKGROUND option. Use the BLOCKIO option if your A/D board supports it. Use the SINGLEIO option if your computer is fast enough to support the selected sampling rate. | |
| 39 | **DTCONNECTERR** | Board does not support the DTCONNECT option. |
|  | The DTCONNECT option was passed to an analog input function or method. The selected board does not support that option. | |
| 40 | **FORECONTINUOUS** | CONTINUOUS can only be run with BACKGROUND. |
|  | The CONTINUOUS option was passed to a function or method without also setting the BACKGROUND option. This is not allowed. Any time you set the CONTINUOUS option you must also set the BACKGROUND option. | |
| 41 | **BADBOARDTYPE** | This function or method cannot be used with this board. |
|  | An attempt was made to call a function or method for a board that does not support that function or method. | |
| 42 | **WRONGDIGCONFIG** | Digital port not configured correctly for requested operation. |
|  | Some of the digital bits or ports (FIRSTPORTA - EIGHTHPORTCH) must be configured as inputs OR outputs but not both. An attempt was made to use a digital input function or method on a port or bit that was configured as an output or vice versa. Use cbDConfigPort()/DConfigPort() or cbDConfigBit()/DConfigBit() to switch a port's (or bits) direction. If the board you are using contains configurable port types and you do not call cbDConfigPort()/DConfigPort() or cbDConfigBit()/DConfigBit() in your program, then all of the configurable ports will be in an unknown state (input or output). | |
| 43 | **NOTCONFIGURABLE** | This digital port is not configurable (it's an In/Out port). |
|  | cbDConfigPort()/DConfigPort() or cbDConfigBit()/DConfigBit() was called for a port that is not configurable. Check the PortNum argument passed to cbDConfigPort() and make sure that it is in the range FIRSTPORTA - EIGHTHPORTCH. If PortNum is AUXPORT, make sure your hardware supports configuration of this port type. If not then there is no need to call this function or method. | |
| 44 | **BADPORTCONFIG** | Invalid digital port configuration. |
|  | The Direction argument passed to cbDConfigPort()/DConfigPort() or cbDConfigBit()/DConfigBit() is invalid. It must be set to either DIGITALIN or DIGITALOUT. | |
| 45 | **BADFIRSTPOINT** | First point number is not valid. |
|  | The FirstPoint argument to cbFileRead ()/FileRead() is invalid. It is either a negative number or it is larger then the number of points in the file. | |
| 46 | **ENDOFFILE** | Attempted to read past the end of the file. |
|  | cbFileRead()/FileRead() attempted to read beyond the end of the file. Check the file length with cbFileGetInfo()/FileGetInfo() and make sure that the FirstPoint and Count arguments to cbFileRead()/FileRead() are correct for that file length. | |
| 47 | **NOT8254CTR** | This board does not have an 8254 counter. |
|  | cbC8254Config()/C8254Config()was called for a board that has a counter but not an 8254 counter. This function or method can only be used with an 8254 counter. | |
| 48 | **NOT9513CTR** | This board does not have a 9513 counter. |
|  | cbC9513Config()/C9513Config()was called for a board that has a counter but not a 9513 counter. This function or method can only be used with a 9513 counter. | |
| 49 | **BADTRIGTYPE** | Invalid TrigType. |
|  | cbATrig()/(ATrig()) was called with an invalid TrigType argument. It must be set to either TRIGABOVE or TRIGBELOW. | |
| 50 | **BADTRIGVALUE** | Invalid TrigValue. |
|  | cbATrig()/ATrig() was called with an invalid TrigValue argument/parameter. It must be in the range 0 - 4,095 for 12-bit boards or 0 to 65,535 for 16-bit boards (see the "[Visual Basic signed integers](mk:@MSITStore:C:\Program%20Files%20(x86)\Measurement%20Computing\DAQ\ULHelp.chm::/Users_Guide/Counter_Boards/CTR-hardware.htm)" discussion at the beginning of the "Counter Boards" section in the Universal Library User's Guide.) | |
| 52 | **BADOPTION** | Invalid option specified for this function or method. |
|  | The Option argument contains an option that is not valid for this function or method. | |
| 53 | **BADPRETRIGCOUNT** | Invalid PreTrigCount specified. |
|  | Either cbAPretrig()/APretrig()or cbFilePretrig()/FilePretrig() was called with an invalid PretrigCount argument. The pre-trigger count must not be <0, and must be less than TotalCount – 512. It also must be less than 32k for cbAPretrig()/APretrig(), and less than 16k for cbFilePretrig()/FilePretrig(). | |
| 55 | **BADDIVIDER** | Invalid FOutDivider value. |
|  | The FOutDivider argument to cbC9513Init()/C9513Init() is not valid. It must be in the range 0 to 15. | |
| 56 | **BADSOURCE** | Invalid FOutSource value. |
|  | The FOutSource argument to cbC9513Init() (C9513Init()) is not valid. It must be one of the following values: CTRINPUT1, CTRINPUT2, CTRINPUT3, CTRINPUT4, CTRINPUT5, GATE1, GATE2, GATE3, GATE4, GATE5, FREQ1, FREQ2, FREQ3, FREQ4, FREQ5 (for example 0 to 15). | |
| 57 | **BADCOMPARE** | Invalid compare value. |
|  | One or both of the compare arguments to cbC9513Init()/C9513Init() are not valid. They must be set to (CB)ENABLED or (CB)DISABLED (1 or 0). | |
| 58 | **BADTIMEOFDAY** | Invalid TimeOfDay value. |
|  | The TimeOfDay argument to cbC9513Init()/C9513Init() is not valid. It must be set to either ENABLED or DISABLED (1 or 0). | |
| 59 | **BADGATEINTERVAL** | Invalid Gate Interval value. |
|  | The GateInterval argument to cbCFreqIn()/CFreqIn() is not valid. It must be greater than 0. | |
| 60 | **BADGATECNTRL** | Invalid Gate Control value. |
|  | The GateControl argument to cbC9513Config()/C9513Config() is not valid. It must be in the range 0 to 7. | |
| 61 | **BADCOUNTEREDGE** | Invalid Counter Edge value. |
|  | The CounterEdge argument to cbC9513Config()/C9513Config() is not valid. It must be set to either POSITIVEEDGE or NEGATIVEEDGE. | |
| 62 | **BADSPCLGATE** | Invalid SpecialGate value. |
|  | The SpecialGate argument to cbC9513Config()/C9513Config() is not valid. It must be set to either (CB)ENABLED or (CB)DISABLED (1 or 0). | |
| 63 | **BADRELOAD** | Invalid Reload value. |
|  | The Reload argument to cbC9513Config()/C9513Config() is not valid. It must be set to either LOADREG or LOADANDHOLDREG. | |
| 64 | **BADRECYCLEFLAG** | Invalid Recycle Mode value. |
|  | The RecycleMode argument to cbC9513Config()/C9513Config() is not valid. It must be set to either (CB)ENABLED or (CB)DISABLED (1 or 0). | |
| 65 | **BADBCDFLAG** | Invalid BCD Mode value. |
|  | The BCDMode argument/parameter to cbC9513Config()/C9513Config() is not valid. It must be set to either (CB)ENABLED or (CB)DISABLED (1 or 0). | |
| 66 | **BADDIRECTION** | Invalid Count Direction value. |
|  | The CountDirection argument to cbC9513Config() (C9513Config()) is not valid. It must be set to either COUNTUP or COUNTDOWN. | |
| 67 | **BADOUTCONTROL** | Invalid Output Control value. |
|  | The OutputControl argument to cbC9513Config() (C9513Config()) is not valid. It must be set to either ALWAYSLOW, HIGHPULSEONTC, TOGGLEONTC, DISCONNECTED or LOWPULSEONTC. | |
| 68 | **BADBITNUMBER** | Invalid BitNum specified. |
|  | The BitNum argument to cbDBitIn() or cbDBitOut() (DBitIn() or DBitOut()) is not valid. The valid range of bit numbers depends on the selected board. If it is a DIO24 compatible board, the maximum bit number is 23. If it's a DIO96, the maximum bit number is 95. Refer to board-specific information in the Universal Library User's Guide or in your hardware manual. | |
| 69 | **NONEENABLED** | None of the counter channels were enabled. |
|  | None of the counter channels were marked as (CB)ENABLED in the CntrControl array that was passed to cbCStoreOnInt()/CStoreOnInt(). At least one of the counter channels must be enabled. | |
| 70 | **BADCTRCONTROL** | An element of Cntr Control array not set to ENABLED or DISABLED |
|  | One of the elements of the CntrControl array that was passed to cbCStoreOnInt()/(CStoreOnInt()) was set to something other then (CB)ENABLED or (CB)DISABLED. The array must have at least ten elements, and the first ten elements must be set to either (CB)ENABLED or (CB)DISABLED. | |
| 71 | **BADEXPCHAN** | Invalid EXP channel specified. |
|  | An invalid channel was passed to one of the thermocouple input commands. The channel number when using an EXP board must be ≥16. The maximum allowable channel number depends on which EXP board is being used (and how many of them). Refer to the board manual to find the number of channels. | |
| 72 | **WRONGADRANGE** | Board set to wrong A/D range for reading thermocouples. |
|  | A thermocouple input function or method was called to read an EXP board input. The EXP board is connected to an A/D board with hardware selected gain that is set to the wrong range. When using EXP boards with thermocouples, the A/D must be set to the –5 to +5 volt range when available. When using RTD sensors, the range is 0 to 10V when available. | |
| 73 | **OUTOFRANGE** | Temperature input is out of range. |
|  | A thermocouple input function or method returned an invalid temperature. This usually indicates an open connection in the thermocouple or its connection to the mux board. | |
| 74 | **BADTEMPSCALE** | Invalid temperature scale specified |
|  | The Scale argument to a thermocouple input function or method is not valid. It must be set to either CELSIUS, FAHRENHEIT, KELVIN, VOLTS or NOSCALE.  Set to VOLTS to read the voltage input of a thermocouple. Refer to board-specific information in the Universal Library User's Guide to determine if the hardware supports this option.  Set to NOSCALE to retrieve raw data from a device. Specifying NOSCALE returns calibrated data, however a cold junction compensation (CJC) correction factor is not applied to the returned values. Refer to board-specific information in the Universal Library User's Guide to determine if the hardware supports this option. | |
| 76 | **NOQUEUE** | Specified board does not have channel/gain queue. |
|  | The function or method that was called requires that the board has a channel/gain queue. The specified board does not have a queue. | |
| 77 | **CONTINUOSCOUNT** | Count must be > the packet size to use continuous mode. |
|  | The Count argument is not valid for continuous mode. Using BLOCKIO mode, the Count argument must be large enough to cause at least one interrupt. This is usually half the size of the boards FIFO (typical sizes are 256, 512, and 1,024). Refer to the board-specific information in the Universal Library User's Guide. | |
| 78 | **UNDERRUN** | D/A FIFO went empty during output. |
|  | The specified D/A output rate could not be sustained. Try increasing the size of the data buffer or reducing the update rate to eliminate the error. | |
| 79 | **BADMEMMODE** | Invalid memory mode specified. |
|  | The memory mode that was selected with cbMemSetDTMode()/MemSetDTMode() is not one of the valid modes. | |
| 80 | **FREQOVERRUN** | Measured frequency too high for selected gating interval. |
|  | The GateInterval argument used with cbCFreqIn()/CFreqIn() is too large to measure the frequency of the signal connected to the counter. The counter is overflowing. Decrease the gating interval to eliminate the error. | |
| 81 | **NOCJCCHAN** | A CJC Channel must be configured to make temperature measurements. |
|  | When the board was installed with the InstaCal installation program, no Cold Junction Compression (CJC) channel was selected. To use the temperature measurement functions or methods with thermocouples, you must first select a CJC channel on the A/D board and then rerun the InstaCal installation program. | |
| 82 | **BADCHIPNUM** | Invalid ChipNum specified. |
|  | An invalid ChipNum argument was used with cbC9513Init()/C9513Init(). If the board is CTR05, set ChipNum to 0. If the board is a CTR10, set ChipNum to either 0 or 1. | |
| 83 | **DIGNOTENABLED** | The digital I/O on this board is not enabled. |
|  | When the board was installed with the InstaCal installation program, the expansion digital I/O was set to DISABLED. To use these digital I/O lines, you must enable the digital I/O on the board (with a jumper) and then re-run the InstaCal installation program and set the digital I/O to ENABLED. | |
| 84 | **CONVERT16BITS** | CONVERTDATA option can not be used with 16-bit A/D converters. |
|  | When using a 16-bit A/D (DAS1600/16), if you try to use the CONVERTDATA option with cbAInScan()/AInScan() or call cbAConvertData()/AConvertData(), this error is returned. (This has been updated in later versions of the library so that it is ignored for boards for which it is inappropriate.) | |
| 85 | **NOMEMBOARD** | The EXTMEMORY option requires that a MEGA-FIFO be attached. |
|  | Attempt to use a cbMem\_() function or Mem\_() method without a MEGA-FIFO board installed. Install MEGA-FIFO with InstaCal. | |
| 86 | **DTACTIVE** | No memory read/write allowed while DT transfer in progress. |
|  | A read or write to a memory board was attempted while data was being transferred via DT-Connect. | |
| 87 | **NOTMEMCONF** | Specified board is not a memory board. |
|  | The specified board is not a memory board. This function or method only works with memory boards. | |
| 88 | **ODDCHAN** | The first channel in scan and number of channels must be even (0, 2, 4, and so on). |
|  | Some boards use a channel/gain queue that require the first channel in the queue and the number of channels in the queue always be an even channel. This error can occur even when you are not in the process of loading the queue. Some boards use the queue automatically with cbAInScan()/AInScan(). On those boards the low channel must be an even number. | |
| 89 | **CTRNOINIT** | Counter was not configured or initialized. |
|  | You attempted to use cbCLoad() or cbCIn() (CLoad() or CIn()) before initializing and configuring the counter. | |
| 90 | **NOT8536CTR** | This board does not have an 8536 counter chip. |
|  | An attempt has been made to use 8536 initialization or configuration on board without an 8536 chip. | |
| 91 | **FREERUNNING** | Board doesn't time A/D sampling. Collecting at fastest possible speed. |
|  | This board does not have an A/D pacer mechanism and you have called cbAInScan()/(AInScan(). The A/D will be sampled in a tight software loop as fast as the CPU can execute the instructions. The speed of sampling is dependent on the computer and the concurrent tasks. | |
| 92 | **INTERRUPTED** | Operation interrupted with <Ctrl-C>. |
|  | A foreground operation was stopped before completion because either <Ctrl-C> or <Ctrl-Break> was pressed. | |
| 93 | **NOSELECTORS** | No selector could be allocated. |
|  | A Windows selector required by the library could not be allocated. Close any open Windows applications that are not required to be running, and try again. | |
| 94 | **NOBURSTMODE** | This board does not support burst mode. |
|  | An attempt was made to use the BURSTMODE option on a board which does not support that option. | |
| 95 | **NOTWINDOWSFUNC** | This function is not available in Windows library. |
|  | The library function you called is not supported in the current revision of Universal Library for Windows Languages. It may be supported in the future. Contact technical support. | |
| 96 | **NOTSIMULCONF** | Board not configured for SIMULTANEOUS option. |
|  | The configuration file of the D/A board in InstaCal must be set for simultaneous update before you use the SIMULTANEOUS option of cbAOutScan()/AOutScan(). The jumpers on the D/A board must be set for simultaneous update before it will work. | |
| 97 | **EVENODDMISMATCH** | An even channel is in an odd slot in the queue, or vice versa. |
|  | The channel gain queue on some A/D boards has a restriction that the channel numbers must be in even queue positions and odd channel numbers must be in odd queue positions. | |
| 98 | **M1RATEWARNING** | Sampling speed to system memory MAY be too fast. |
|  | The A/D board sampling speed you have requested may be too fast for the computer system bus transfer to complete before the next packet is ready for transfer. If this is the case, data will overrun and sample data will be garbled. This warning is initiated whenever you request a sample rate over 625 kHz, AND the sample set is larger than the FIFO buffer on the board, AND an external memory board, such as a MEGA-FIFO is not being used. Your system may be able to handle the rate requested but only experimentation will bear this out. Your system may be capable of the full 1 MHz rate directly to system memory. | |
| 99 | **NOTRS485** | Selected board is not a RS-485 board. |
|  | An attempt was made to call cbRS485()/RS485() with a board that is not RS485 compatible. | |
| 100 | **NOTDOSFUNCTION** | This function is not available in DOS. |
|  | The function that was called is not available in the DOS version of the Universal Library. | |
| 101 | **RANGEMISMATCH** | Bipolar and unipolar ranges cannot be used together in A/D queue. |
|  | The channel/gain queue should only be loaded (via cbALoadQueue()/ALoadQueue()) with all unipolar or bipolar ranges. | |
| 102 | **CLOCKTOOSLOW** | Sampling rate is too high for clock speed; change clock jumper on the board. |
|  | The sampling rate that you requested is too fast. The A/D board pacer might be capable of running at a higher rate. Check the board for an XTAL jumper and, if it is not set for the highest rate, place the jumper in the position for the highest rate. After the jumper is set, re-run InstaCal. | |
| 103 | **BADCALFACTORS** | Calibration factors are invalid - Disabling software calibration. |
|  | The selected board uses software calibration and the stored calibration factors are invalid. Run InstaCal and calibrate the board before using it. | |
| 104 | **BADCONFIGTYPE** | Invalid configuration information type specified. |
|  | An invalid ConfigType argument was passed to either cbGetConfig() or cbSetConfig(). | |
| 105 | **BADCONFIGITEM** | Invalid configuration item specified. |
|  | An invalid ConfigItem argument was passed to either cbGetConfig() or cbSetConfig(). | |
| 106 | **NOPCMCIABOARD** | Cannot access the PCMCIA board. |
|  | Cannot access the specified PCMCIA board. Make sure that the PCMCIA Card & Socket Services are installed correctly and that the board was installed in the system correctly via InstaCal. | |
| 107 | **NOBACKGROUND** | Board does not support background operation. |
|  | The BACKGROUND option was used and the specified board does not support background operation. | |
| 108 | **STRINGTOOSHORT** | The string argument is too short for the string being returned. |
|  | The string passed to a library function or method is to small to contain the string that is being returned. Increase the size of the string to the minimum size specified for the function or method that you are using. | |
| 109 | **CONVERTEXTMEM** | CONVERTDATA not allowed with EXTMEMORY option. |
|  | You requested both the CONVERTDATA and EXTMEMORY option. These options cannot be used together. Collect the data without the CONVERTDATA option. After the data has been collected, read it back from the memory card (cbMemRead()/MemRead()or cbMemReadPretrig()/ MemReadPretrig()), and use cbAConvertData()/AConvertData()) to convert the data. | |
| 110 | **BADEUADD** | Program error bad values used in cbFromEngUnits or cbToEngUnits(). |
|  | Invalid floating point data was used in cbFromEngUnits()/FromEngUnits()or cbToEngUnits/ToEngUnits(). Check the arguments passed to the relevant function or method. | |
| 111 | **DAS16JRRATEWARNING** | Rates greater than 125 kHz must use on board 10 MHz clock. |
|  | If a rate greater than 125 kHz is selected and the on board jumper is set for 1 MHz when using the CIO-DAS16/Jr, this warning is generated. Place the jumper on the 10MHz position and update your InstaCal settings. | |
| 112 | **DAS08TOOLOW\_RATE** | The desired sample rate is below hardware minimum. |
|  | Increase the value of the Rate argument in cbAInScan()/AInScan(). The lowest pacer frequency is the clock frequency (usually 8 MHz ÷ 2) ÷ by 65,535 for the CIO-, PC104 and PCM- DAS08. | |
| 114 | **AMBIGSENSORONGP** | More than one temperature sensor type defined for EXP-GP. |
|  | Thermocouple and RTD types are both defined for an EXP-GP. cbTIn()/(TIn() and cbTInScan()/TInScan()) require that only one be defined to operate. Use InstaCal to set one of the sensor types to "Not Installed". | |
| 115 | **NOSENSORTYPEONGP** | No temperature sensor type defined for EXP-GP. |
|  | Neither Thermocouple nor RTD types are defined for an EXP-GP. cbTIn()/(TIn() and cbTInScan()/TInScan()) require that one and only one be defined to operate. Use InstaCal to set one of the sensor types to a predefined type. | |
| 116 | **NOCONVERSIONNEEDED** | Selected 12-bit board already returns converted data. |
|  | Some 12-bit boards do not need to have their data converted after a call to cbAInScan()/AInScan() with the NOCONVERTDATA option. These boards return no channel tags and therefore return data in its proper format. Calling cbAConvertData()/AConvertData() with data generated from these boards will generate this warning. | |
| 117 | **NOEXTCONTINUOUS** | CONTINUOUS mode cannot be used with EXTMEMORY. |
|  | CONTINUOUS mode is ignored when used with the EXTMEMORY option. | |
| 118 | **INVALIDPRETRIGCONVERT** | cbAConvertPretrigData called after cbAPretrig failed. |
|  | The data you are attempting to convert with cbAConvertPretrigData()/ AConvertPretrigData() can not be converted because cbAPretrig()/APretrig() did not return a complete data set, probably due to an early trigger. | |
| 119 | **BADCTRREG** | Bad counter argument passed to cbCLoad() |
|  | The RegNum argument passed to cbCLoad() (CLoad()) is not a valid register. | |
| 120 | **BADTRIGTHRESHOLD** | Low trigger threshold is greater than high threshold. |
|  | The LowThreshold arguments to cbSetTrigger()/SetTrigger() must be < the HighThreshold. | |
| 121 | **BADPCMSLOTREF** | NO PCM Card was found in the specified slot. |
|  | This is usually caused by swapping PCMCIA cards and not re-running InstaCal. Run InstaCal. | |
| 122 | **AMBIGPCMSLOTREF** | Two identical PCM cards found. Please specify exact slot in InstaCal. |
|  | This error occurs in DOS mode only when InstaCal is configured for a PCMCIA card in "any slot". To correct the problem, run InstaCal, go to the Install menu and pop up the board's menu. Highlight PCMCIA slot and choose either "0" or "1". | |
| 123 | **BADSENSORTYPE** | Invalid sensor type selected in InstaCal. |
|  | The specified sensor type is not included in the allowed list of thermocouple/RTD types. Set the sensor type to a predefined type using InstaCal. | |
| 126 | **CFGFILENOTFOUND** | Cannot find CB.CFG file. |
|  | The CB.CFG file could not be found. This file should be located in the same directory in which you installed the software. | |
| 127 | **NOVDDINSTALLED** | The CBUL.386 virtual device driver is not installed. |
|  | The Windows device driver CBUL.386 is not installed on your system. Normally, it will be automatically installed when you run the standard installation program. The following line should be in your \windows\system.ini file in the [386Enh] section: | |
| 128 | **NOWINDOWSMEMORY** | Requested amount of Windows page-locked memory is not available. |
|  | The Windows device driver could not allocate the required amount of physical memory. This error should not normally occur unless you are collecting very large amounts of data or your system is very memory constrained. If you are collecting a very large block of memory, try collecting a smaller amount. If this is not an option, than consider using cbFileAInScan()/FileAInScan() instead of cbAInScan()/AInScan(). Also, if you are running other programs, try shutting them down. | |
| 129 | **OUTOFDOSMEMORY** | Not enough DOS memory available. |
|  | Try closing down any unneeded programs that are running. | |
| 130 | **OBSOLETEOPTION** | Obsolete option specified for cbSetConfig/cbGetConfig. |
|  | The specified configuration item is no longer supported in the 32 bit version of the Universal Library. | |
| 131 | **NOPCMREGKEY** | No registry entry for this PCMCIA card. |
|  | When running under Windows/NT, there must be an entry in the system registry for each PCMCIA card that you will be using with the system. This is ordinarily taken care of automatically by the Universal Library installation program. If this error occurs, contact technical support for assistance. | |
| 132 | **NOCBUL32SYS** | CBUL32.SYS device driver is not installed. |
|  | The Windows device driver CBUL.SYS is not installed on your system. Normally, it will be automatically installed when you run the MCC standard installation program. Contact technical support for assistance. | |
| 133 | **NODMAMEMORY** | No DMA memory available to device driver. |
|  | The Windows device driver could not allocate the minimum required amount of memory for DMA. If you are sampling at slower speeds, you can specify SINGLEIO in the Options argument to cbAInScan()/(AInScan(). This will prevent the library from attempting to use DMA. In general though, this error should not ordinarily occur. Contact technical support for assistance. | |
| 134 | **IRQNOTAVAILABLE** | IRQ not available. |
|  | The Interrupt Level that was specified for the board (in InstaCal) conflicts with another board in your computer. Try switching to a different interrupt level. | |
| 135 | **NOT7266CTR** | This board does not have an LS7266 counter. |
|  | This function or method can only be used with a board that contains an LS7266 chip. These chips are used on various quadrature encoder input boards. | |
| 136 | **BADQUADRATURE** | Invalid Quadrature argument passed to cbC7266Config(). |
|  | The Quadrature argument must be set to either NO\_QUAD, X1\_QUAD, X2\_QUAD, or X4\_QUAD. | |
| 137 | **BADCOUNTMODE** | Invalid counting mode specified. |
|  |  | |
| 138 | **BADENCODING** | Invalid DataEncoding argument passed to cbC7266Config(). |
|  | The DataEncoding argument must be set to either BCD\_ENCODING or BINARY\_ENCODING. | |
| 139 | **BADINDEXMODE** | Invalid IndexMode argument passed to cbC7266Config() |
|  | The IndexMode argument must be set to either INDEX\_DISABLED, LOAD\_CTR, LOAD\_OUT\_LATCH, or RESET\_CTR. | |
| 140 | **BADINVERTINDEX** | Invalid InvertIndex argument passed to cbC7266Config() |
|  | The InvertIndex argument must be set to either (CB)ENABLED or (CB)DISABLED. | |
| 141 | **BADFLAGPINS** | Invalid FlagPins argument passed to cbC7266Config() |
|  | The FlagPins argument must be set to either CARRY\_BORROW, COMPARE\_BORROW, CARRYBORROW\_UPDOWN, or INDEX\_ERROR. | |
| 142 | **NOCTRSTATUS** | This board does not support cbCStatus() |
|  | This board does not return any status information. | |
| 143 | **NOGATEALLOWED** | Gating cannot be used when indexing is enabled. |
|  | Gating and indexing can not be used simultaneously. If Gating is set to (CB)ENABLED, then IndexMode must be set to INDEX\_DISABLED. | |
| 144 | **NOINDEXALLOWED** | Indexing not allowed in non-quadrature mode |
|  | Indexing is not supported when the Quadrature argument is set to NO\_QUAD. | |
| 145 | **OPENCONNECTION** | Temperature input has open connection |
|  |  | |
| 146 | **BMCONTINUOUSCOUNT** | Count must be integer multiple of packet size for Continuous mode. |
|  |  | |
| 147 | **BADCALLBACKFUNC** | Invalid pointer to callback function or delegate passed as argument. |
|  |  | |
| 148 | **MBUSINUSE** | Metrabus in use |
|  |  | |
| 149 | **MBUSNOCTLR** | Metrabus I/O card has no configured controller card |
|  |  | |
| 150 | **BADEVENTTYPE** | Invalid event type specified for this board. |
|  | Although this board does support cbEnableEvent()/EnableEvent(), it does not support one or more of the event types specified. | |
| 151 | **ALREADYENABLED** | Event handler is already enabled for this event type. |
|  | There is already an event handler bound to one or more of the events specified. To attach the new handler to the event type, first disable and disconnect the current handler using cbDisableEvent()/DisableEvent(). | |
| 152 | **BADEVENTSIZE** | Invalid event count has been specified |
|  | The ON\_DATA\_AVAILABLE event requires an event count greater than (0). | |
| 153 | **CANTINSTALLEVENT** | Unable to install event handler |
|  | An internal error occurred while trying to setup the event handling. | |
| 154 | **BADBUFFERSIZE** | Buffer is too small for operation |
|  | The memory allocated by cbWinBufAlloc()/WinBufAlloc() is too small to hold all the data specified in the operation. | |
| 155 | **BADAIMODE** | Invalid analog input mode |
|  | Invalid analog input mode (RSE, NRSE, DIFF). | |
| 156 | **BADSIGNAL** | Invalid signal type specified |
|  | The specified signal type does not exist, or is not valid for signal direction specified. | |
| 157 | **BADCONNECTION** | message |
|  | The specified connection does not exist, or is not valid for the signal type and direction specified. | |
| 158 | **BADINDEX** | Invalid index specified. |
|  | For Index >0, indicates that the specified index is beyond the end of the internal list of output connections assigned to the specified signal type. | |
| 159 | **NOCONNECTION** | Invalid connection |
|  | No connection is assigned to the specified signal. | |
| 160 | **BADBURSTIOCOUNT** | Count cannot be greater than the FIFO size for BURSTIO mode. Furthermore, Count must be integer multiple of number of channels in scan. |
|  | When using BURSTIO mode, the count entered cannot be larger than the FIFO size. | |
| 161 | **DEADDEV** | Device has stopped responding. |
|  | Check cable connections to USB device and to your computer's USB port. | |
| 163 | **INVALIDACCESS** | Required access or privilege not acquired for specified operation. Please check for other users of device and restart application. |
|  | You are currently not the device owner and therefore cannot change the state or configuration of the Ethernet device with functions such as cbAOut()/AOut(), cbDBitOut/DBitOut(), cbAInScan()/AInScan(), cbFlashLED()/FlashLED(), and others. However, you can still read the state or configuration of the Ethernet device with functions such as cbAIn()/AIn(), cbDBitIn()/DBitIn(), and so on. | |
| 164 | **UNAVAILABLE** | Device unavailable at time of request. Please repeat operation. |
|  | You requested an operation that conflicts with an operation in progress on the device. This error usually occurs in multithreaded applications or if you are running multiple applications that access the device. Both types of operations are not supported. | |
| 165 | **NOTREADY** | Device is not ready to send data. Please repeat operation. |
|  | You requested an operation that conflicts with an operation in progress on the device. This error can occur during device initialization. | |
| 169 | **BITUSEDFORALARM** | The specified bit is used for alarm. |
|  | You attempted to set the state of a digital output bit that is configured as an alarm input. | |
| 170 | **PORTUSEDFORALARM** | One or more bits on the specified port are used for alarm. |
|  | You attempted to write to a digital output port that contains a bit configured as an alarm input. | |
| 171 | **PACEROVERRUN** | Pacer overrun; external clock rate too fast. |
|  | You set the external clock rate to a value that is higher than the rate supported by the board. | |
| 172 | **BADCHANTYPE** | Invalid channel type specified. |
|  | You set the channel type to a type that is not supported by the board. | |
| 173 | **BADTRIGSENSE** | Invalid trigger sensitivity specified. |
|  | You set the trigger sensitivity to a value that is not supported by the board. | |
| 174 | **BADTRIGCHAN** | Invalid trigger channel specified. |
|  | You set the trigger channel to a value that is not supported by the board. | |
| 175 | **BADTRIGLEVEL** | Invalid trigger level specified. |
|  | You set the trigger level to a value that is not supported by the board. | |
| 176 | **NOPRETRIGMODE** | Pretrigger mode is not supported for the specified trigger type. |
|  | You selected a trigger source that does not support pre-trigger data acquisitions. | |
| 177 | **BADDEBOUNCETIME** | Invalid debounce timing specified. |
|  | You set the debounce time to a value that is not supported by the board. | |
| 178 | **BADDEBOUNCETRIGMODE** | Invalid debounce trigger mode specified. |
|  | You set the debounce trigger mode to a value that is not supported by the board. | |
| 179 | **BADMAPPEDCOUNTER** | Invalid mapped channel specified. |
|  | You mapped to a counter input channel that is not supported by the board. | |
| 180 | **BADCOUNTERMODE** | Invalid counter mode specified. |
|  | This function cannot be used with the current mode of the specified counter. | |
| 181 | **BADTCCHANMODE** | Single-ended mode cannot be used for temperature input. |
|  | You specified single-ended mode for use with a temperature input. | |
| 182 | **BADFREQUENCY** | Invalid frequency specified. |
|  | You specified a frequency value that is not supported by the board. | |
| 183 | **BADEVENTPARAM** | Invalid event parameter specified. |
|  | You specified an event parameter that is not supported by the board. | |
| 184 | **NONETIFC** | No interface devices were found with the required PAN and channel. |
|  | No interface devices were detected whose PAN ID and RF channel number match those of a remote device. | |
| 185 | **DEADNETIFC** | The interface device(s) with the required PAN and channel has failed. Please check the connection. |
|  | The interface device whose PAN ID and RF channel number match a remote device is not responding. Check the USB connection to the computer. | |
| 186 | **NOREMOTEACK** | The remote device is not responding to commands and queries. Please check the device. |
|  | The wireless remote device is not responding. Check that the device is powered, that its PAN ID and RF channel match the interface device, and that the LEDs are functioning. | |
| 187 | **INPUTTIMEOUT** | The device acknowledged the operation, but has not completed before the timeout. |
|  | The operation was acknowledged but has timed out before it was completed. | |
| 188 | **MISMATCHSETPOINTCOUNT** | Number of setpoints is not equal to number of channels with a setpoint flag set. |
|  | Set the number of setpoints equal to the number of channels with a setpoint flag set. | |
| 189 | **INVALIDSETPOINTLEVEL** | Setpoint level is outside channel range. |
|  | You specified a setpoint level that is outside of the range supported by the board. | |
| 190 | **INVALIDSETPOINTOUTPUTTYPE** | Setpoint Output Type is invalid. |
|  | You specified a setpoint output type that is not supported by the board. | |
| 191 | **INVALIDSETPOINTOUTPUTVALUE** | Setpoint Output Value is outside channel range. |
|  | You specified a setpoint output value that is outside of the range supported by the board. | |
| 192 | **INVALIDSETPOINTLIMITS** | Setpoint Comparison Limit B greater than Limit A. |
|  | Set the setpoint comparison value for limit A to be larger than the value set for limit B. | |
| 193 | **STRINGTOOLONG** | The string length entered is too long for this operation. |
|  | Enter a string up to the maximum number of characters specified for the function or method that you are using. | |
| 194 | **INVALIDLOGIN** | An invalid user name or password has been entered. |
|  | Check that the password and user name entered were correct. If either has been lost, use the device reset button to reset the device to default values. | |
| 195 | **SESSIONINUSE** | Device session is already in use. |
|  | Another user is currently logged in to a device session. Only one device session can be opened at a time. | |
| 196 | **NOEXTPOWER** | External power is not connected. |
|  | External power is required. Connect the device to an external power supply. | |
| 197 | **BADDUTYCYCLE** | Invalid duty cycle specified. |
|  | You attempted to set the duty cycle to a value not supported by the hardware. | |
| 200-299 | **200-299** | Internal 16-bit error |
|  | Internal error occurred in the library. Refer to the specific errors below: | |
| 201 | **CANT\_LOCK\_DMA\_BUF** | DMA buffer could not be locked. |
|  | There is not enough physical memory to lock down enough DMA memory for this operation. Try closing out other applications, or installing additional RAM. | |
| 202 | **DMA\_IN\_USE** | DMA already controlled by another driver. |
|  | The DMA controller is currently being used by another device, such as another DMA board or the floppy drive. | |
| 203 | **BAD\_MEM\_HANDLE** | Invalid Windows memory handle. |
|  | The memory handle supplied is invalid. Memory handles supplied to library functions and methods should be allocated using cbWinBufAlloc()/WinBufAlloc(), and should not be de-allocated until BACKGROUND operations using this buffer are complete or cancelled with cbStopBackground()/StopBackground(). | |
| 300-399 | **300-399** | Internal 32-bit error. See specific errors below. |
|  |  | |
| 304 | **CFG\_FILE\_READ\_FAILURE** | Error reading from configuration file. |
|  | The program was unable to read the configuration file CB.CFG. Confirm that CB.CFG was not deleted, moved, or renamed since the software installation. | |
| 305 | **CFG\_FILE\_WRITE\_FAILURE** | Error writing to configuration file. |
|  | The program was unable to write to the configuration file CB.CFG. Confirm that CB.CFG is present and that its attributes are not set for Read-only. Also, check that not more than one application is trying to access this file. | |
| 308 | **CFGFILE\_CANT\_OPEN** | Cannot open configuration file. |
|  | The program was unable to open the configuration file CB.CFG. Confirm that CB.CFG was not deleted, moved, or renamed since the software installation. | |
| 325 | **BAD\_RTD\_CONVERSION** | Overflow of RTD conversion. |
|  | Either cbTIn()/TIn() or cbTInScan()/TInScan() returned an invalid temperature conversion. Confirm that the configuration matches the RTD type, and physical EXP board settings; pay particular attention to gain settings and RTD base resistance. Also, check that the RTD leads are securely attached to the EXP terminals. Finally, confirm that the board is measuring reasonable voltages via cbAIn()/AIn(). | |
| 326 | **NO\_PCI\_BIOS** | PCI BIOS not present on the PC. |
|  | Could not locate the BIOS for the PCI bus. Consult PC supplier for proper installation of the PCI BIOS. | |
| 327 | **BAD\_PCI\_INDEX** | Specified PCI board not detected. |
|  | The specified PCI board was not detected. Check that the PCI board is securely installed into the PCI slot. Also, run InstaCal to locate/set valid base address and configuration. | |
| 328 | **NO\_PCI\_BOARD** | Specified PCI board not detected. |
|  | The specified PCI board was not detected. Check that the PCI board is securely installed into the PCI slot. Also, run InstaCal to locate/set valid base address and configuration. | |
| 334 | **CANT\_INSTALL\_INT** | Cannot install interrupt handler. IRQ already in use. |
|  | The device driver could not enable requested interrupt. Check that the selected IRQ is not already in use by another device. This error can also occur if a FOREGROUND scan was aborted; in such cases, rebooting the PC will correct the problem. | |
| 339 | **CANT\_MAP\_PCM\_CIS** | Unable to access Card Information Structure. |
|  | A resource conflict between the specified PCMCIA or PC-Card device and another device prevents the system from allocating sufficient resources to map the onboard CIS. | |
| 343 | **NO\_USB\_BOARD** | Specified USB board not detected. |
|  |  |  |
| 344 | **NOMOREFILES** | No more files in the directory. |
|  | The end of the log file was reached before the file header was read. | |
| 345 | **BADFILENUMBER** | No file exists for the specified file number. |
|  | The specified binary file number does not exist. | |
| 347 | **LOSSOFDATA** | The file may not contain all of the data from the logging session because the logging session was not terminated properly. |
|  | The log file may be incomplete if the logging session is not properly terminated. Always end a logging session by pressing the data logging button until the LED turns off. Possible data loss may occur if the end of the log file is reached before the file header is read. | |
| 348 | **INVALIDBINARYFILE** | The file is not a valid MCC binary file. |
|  | The binary file was not logged from an MCC USB device with data logging capability, or the binary file was logged during a data logging session that was not properly terminated and is missing information. | |
| 349 | **INVALIDDELIMITER** | Invalid delimiter specified for CSV file extension. |
|  | When converting a binary log file to a comma-separated values text file (.CSV), the delimiter character must be set to a comma. | |
| 350 | **NO\_BTH\_BOARD** | Specified Bluetooth board not detected. |
|  |  |  |
| 351 | **NO\_NET\_BOARD** | Specified Network board not detected. |
|  |  |  |
| 400-499 | **PCMCIA error** | Card & Socket Service error. Contact the manufacturer. |
|  |  | |
| 500-599 | **Internal DOS error** | Contact the manufacturer. |
|  |  | |
| 600-699 | **Internal Windows error** | Refer to specific errors below. |
|  |  | |
| 603 | **WIN\_CANNOT\_ENABLE\_INT** | Cannot enable interrupt. IRQ already in use. |
|  | The device driver could not enable requested interrupt. Check that the selected IRQ is not already in use by another device. This error can also occur if a FOREGROUND scan was aborted; in such cases, rebooting the PC will correct the problem. | |
| 605 | **WIN\_CANNOT\_DISABLE\_INT** | Cannot disable interrupts. |
|  | The device driver was unable to disable the IRQ. This can occur when interrupts are generated too fast for the PC to complete servicing. For example, sampling at high frequencies (above ~2 kHz) with scan mode set for SINGLEIO can lead to this error. Frequently, an OVERRUN error accompanies this condition. | |
| 606 | **WIN\_CANT\_PAGE\_LOCK\_BUFFER** | Insufficient memory to page lock data buffer. |
|  | There is not enough physical memory to lock down the entire data buffer. Try closing out other applications, selecting smaller data buffers, or installing additional RAM. | |
| 630 | **NO\_PCM\_CARD** | PCM card not detected. |
|  | The specified PCMCIA card was not detected. Confirm that the PCM card is securely plugged into PCMCIA slot. If the board continues to return this error, run InstaCal to reset the configuration. | |
| 801 | **INVALIDGAINARRAYLENGTH** | The number of elements in the gain array must equal the number of channels in the scan. |
|  | This error is generated when WinBufToEngArray() is called with the number of elements in gainArray is not equal to the number of channels specified. Make sure that the number of elements in the array is the same as the number of channels in the scan. | |
| 802 | **INVALIDDIMENSION0LENGTH** | The length of dimension 0 in the data array must equal the number of channels in the scan. |
|  | This error is generated when WinBufToEngArray() is called with the length of dimension 0 of EngUnits not equal to the number of channels specified. Make sure that the length of dimension 0 in the array is the same as the number of channels in the scan. | |
| 1000 | **NOTEDSSENSOR** | No TEDS sensor was detected on the specified channel. |
|  | Connect a TEDS sensor to the specified channel. | |
| 1001 | **INVALIDTEDSSENSOR** | Connected TEDS sensor to the specified channel is not supported. |
|  | Connect a TEDS sensor that is supported by the hardware to the specified channel. | |
| 1002 | **CALIBRATIONFAILED** | Calibration failed. |
|  | The attempt to calibrate the device has failed. | |
| 1003 | **BITUSEDFORTERMINALCOUNTSTATUS** | The specified bit is used for terminal count status. |
|  | The terminal count status must be disabled for a digital bit before it can be used for timer output or DIO operations. | |
| 1004 | **PORTUSEDFORTERMINALCOUNTSTATUS** | One or more bits on the specified port are used for terminal count status. |
|  | The terminal count status must be disabled for all digital bits before the port can be used for digital operations. | |
| 1005 | **BADEXCITATION** | Invalid excitation specified |
|  | Refer to board-specific information for valid values. | |
| 1006 | **BADBRIDGETYPE** | Invalid bridge type specified |
|  | Refer to board-specific information for valid values. | |
| 1007 | **BADLOADVAL** | Invalid load value specified |
|  | Refer to board-specific information for valid values. | |
| 1008 | **BADTICKSIZE** | Invalid tick size specified |
|  | Refer to board-specific information for valid values. | |
| 1009 | **MINSLOPEVALREACHED** | Minimum slope value reached |
|  | Refer to board-specific information for valid values. | |
| 1010 | **MAXSLOPEVALREACHED** | Maximum slope value reached |
|  | Refer to board-specific information for valid values. | |
| 1011 | **MINOFFSETVALREACHED** | Minimum offset value value reached |
|  | Refer to board-specific information for valid values. | |
| 1012 | **MAXOFFSETVALREACHED** | Maximum offset value value reached |
|  | Refer to board-specific information for valid values. | |
| 1013 | **BTHCONNECTIONFAILED** | Bluetooth connection failed |
|  | Verify the Bluetooth connection. | |
| 1014 | **INVALIDBTHFRAME** | Invalid Bluetooth frame |
|  | The specified Bluetooth frame is invalid. | |
| 1015 | **BADTRIGEVENT** | Invalid trigger event specified |
|  | Refer to board-specific information for valid values. | |
| 1016 | **NETCONNECTIONFAILED** | Network connection failed |
|  |  | |
| 1017 | **DATASOCKETCONNECTIONFAILED** | Data socket connection failed. |
|  |  | |
| 1019 | **NETTIMEOUT** | Network device did not respond within expected time. |
|  |  | |
| 1020 | **NETDEVNOTFOUND** | Network device not found. |
|  |  | |
| 1021 | **INVALIDCONNECTIONCODE** | Invalid connection code. |
|  |  | |
| 1022 | **CONNECTIONCODEIGNORED** | Connection code ignored. |
|  |  | |
| 1023 | **NETDEVINUSE** | Network device already in use. |
|  |  | |
| 1024 | **NETDEVINUSEBYANOTHERPROC** | Network device already in use by another process. |
|  |  | |
| 1025 | **SOCKETDISCONNECTED** | Socket disconnected. |
|  |  | |
| 1026 | **BOARDNUMINUSE** | Board Number already in use. |
|  |  | |
| 1027 | **DEVALREADYCREATED** | Specified DAQ device already created. |
|  |  | |
| 1028 | **BOARDNOTEXIST** | Tried to release a board which doesn't exist. |
|  |  | |
| 1029 | **INVALIDNETHOST** | Invalid host specified. |
|  |  | |
| 1030 | **INVALIDNETPORT** | Invalid port specified. |
|  |  | |
| 1031 | **INVALIDNETIFC** | Invalid interface specified. |
|  |  | |
| 1032 | **INVALIDAIINPUTMODE** | Invalid input mode specified. |
|  |  | |
| 1033 | **AIINPUTMODENOTCONFIGURABLE** | Input mode not configurable. |
|  |  | |
| 1034 | **INVALIDEXTPACEREDGE** | Invalid external pacer edge specified. |
|  |  | |
| 1035 | **CMREXCEEDED** | Common-mode voltage range exceeded. |
|  |  | |

# Service Dependency Command

To ensure that the Measurement Computing driver is up and running before the service starts, enter the following command from the command line.

sc config Orchestrator depend=cbul32