Labview Examples

Reference Manual for EDRE Labview Interface Driver Version 1.0

Eagle Technology – Cape Town, South Africa

Copyright © 1999-2000

www.eagle.co.za

COPYRIGHT
All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any from or any means, electronic, mechanical, by photographing, recording, or otherwise without prior written permission.

Copyright © 2000 Eagle Technology

Information furnished in this manual is believed to be accurate and reliable; however no responsibility is assumed for its use, or any infringements of patents or other rights of third parties which may result from its use.

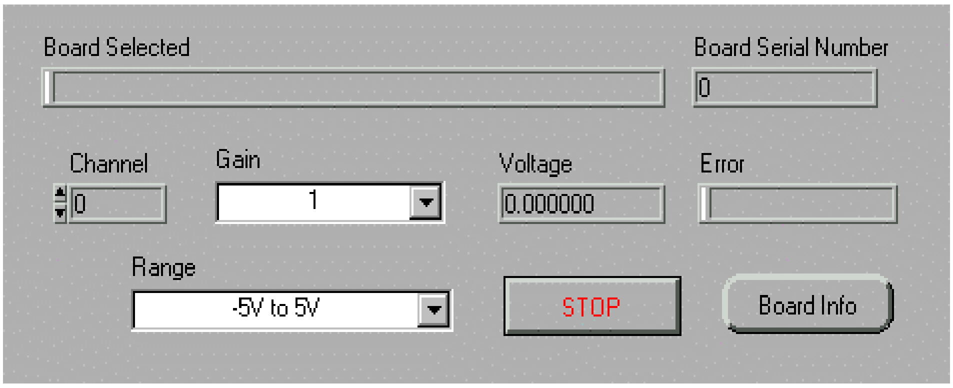
Trademarks and Logos in this manual are the property of their respective owners.

1. Examples

1.1 Analogue to Digital Conversion

These VI examples demonstrate how to perform AD conversion.

1.1.1 Read AD Single



Board Selected

Board Serial Number: 0

Channel: 0

Gain: 1

Voltage: 0.000000

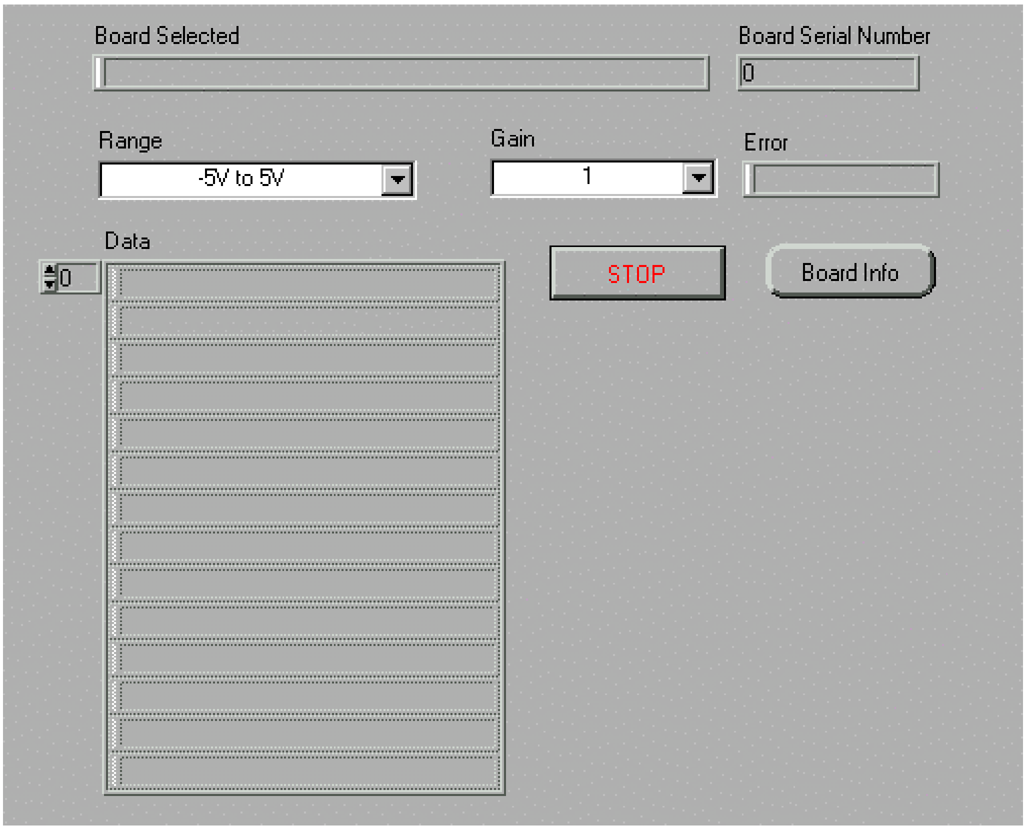
Error

Range: -5V to 5V

This example demonstrates how to read a single analogue value from the AD card. The example starts by asking the user to select the board to use for sampling. The user should then select the channel to sample, the gain at which the sampling should take place, and the voltage range of the AD card. These are then used to read an AD voltage repeatedly at 300ms intervals.

This example includes a “board info” button which, when pressed, displays the current status of and information about the AD board being used. The sampling continues until the stop button is pressed.

1.1.2 Read AD Polled All



Board Serial Number: 0

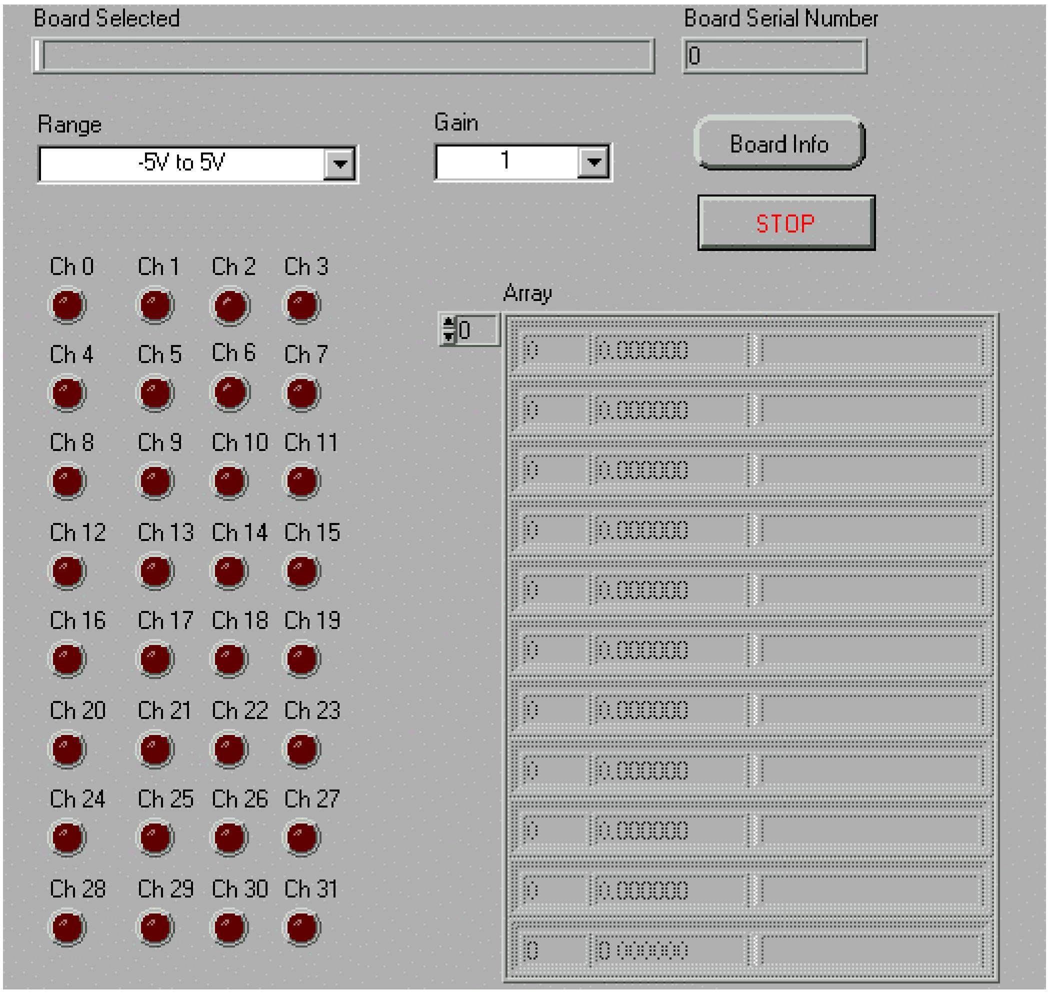
Range: -5V to 5V

Gain: 1

This example demonstrates how to read and display multiple values after AD conversion. The example starts by asking the user to select the board to use for sampling. The user should then select the gain and input voltage range to use for the sampling. The example then samples all the channels every 300ms and displays the results as an array of strings where each string contains the channel number and the voltage read on that channel.

This example includes a “board info” button which, when pressed, displays the current status of and information about the AD board being used. The sampling continues until the stop button is pressed.

1.1.3 Read AD Polled



Board Selected

Board Serial Number: 0

Range: -5V to 5V

Gain: 1

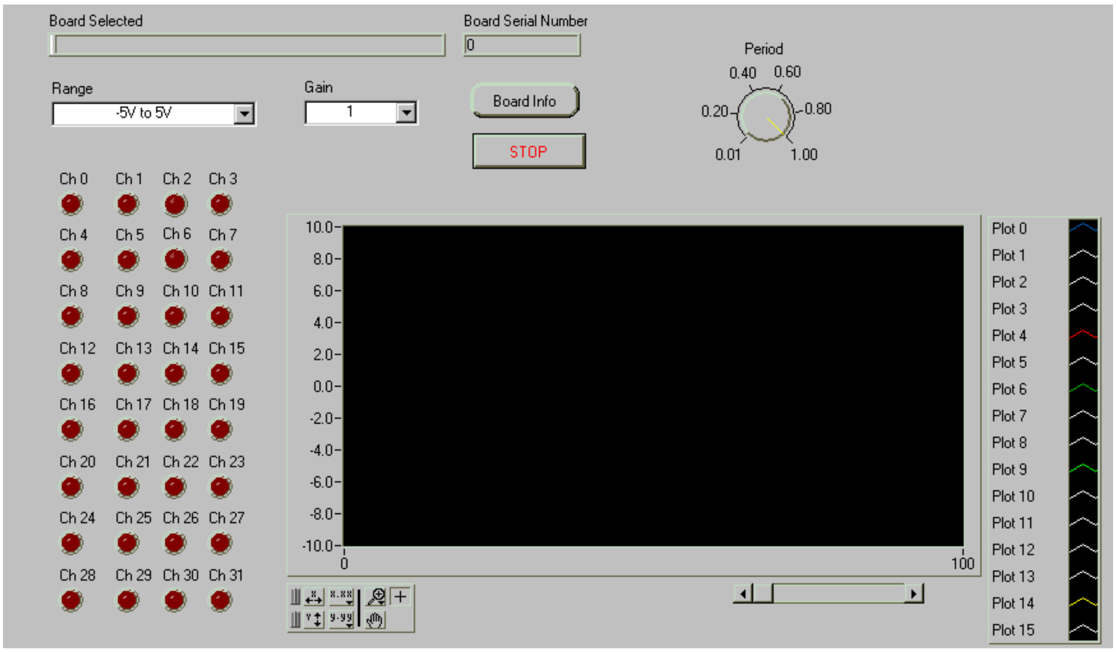
Array: 0

```
0  0.000000
0  0.000000
0  0.000000
0  0.000000
0  0.000000
0  0.000000
0  0.000000
0  0.000000
0  0.000000
0  0.000000
0  0.000000
```

This example is a slightly mode advanced version of "Read AD Polled All" in that it allows the user to select the channels to sample. Firstly, the example starts by asking the user to select the board to use for sampling. The user should then select the gain and input voltage range to use for the sampling. The user can then select which channels he/she wants to sample by pressing any of the channel selection switches. The example then samples each of the requested channels every 300ms and displays the results as an array which indicates the channel number, the voltage read on that channel and an error string indicating the error, if any, that occurred in taking that sample.

This example includes a "board info" button which, when pressed, displays the current status of and information about the AD board being used. The sampling continues until the stop button is pressed.

1.1.4 Read AD Polled Graph



Board Selected

Board Serial Number: 0

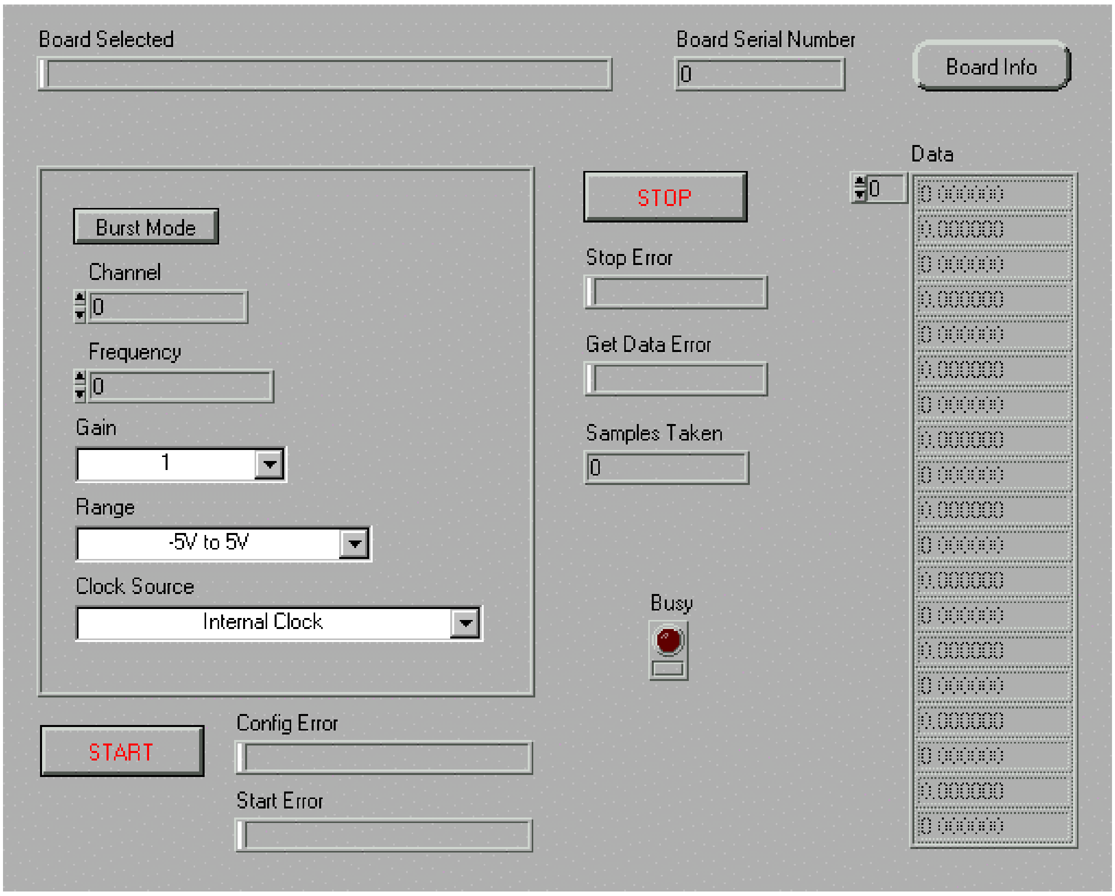
Range: -5V to 5V

Gain: 1

This example adds two new features to those present in "Read AD Polled". Firstly, it allows the user to set the sampling interval that is used to sample the data, and secondly it graphs the output voltages on a graph. As before, the user begins by selecting the board to use for sampling when the example is started. The user should then select the gain and input voltage range to use for the sampling. The user can then select which channels he/she wants to sample by pressing any of the channel selection switches. The user also selects the time period to wait between samples. The example then samples each of the requested channels every interval as specified by the user and displays the results as a graph of voltages.

This example includes a "board info" button which, when pressed, displays the current status of and information about the AD board being used. The sampling continues until the stop button is pressed. The user can change the channels that are being sampled, as well as the sampling period, while the program is running.

1.1.5 AD Block



Board Selected:

Board Serial Number: 0

Channel: 0

Frequency: 0

Gain: 1

Range: -5V to 5V

Clock Source: Internal Clock

Stop Error:

Get Data Error:

Samples Taken: 0

Config Error:

Start Error:

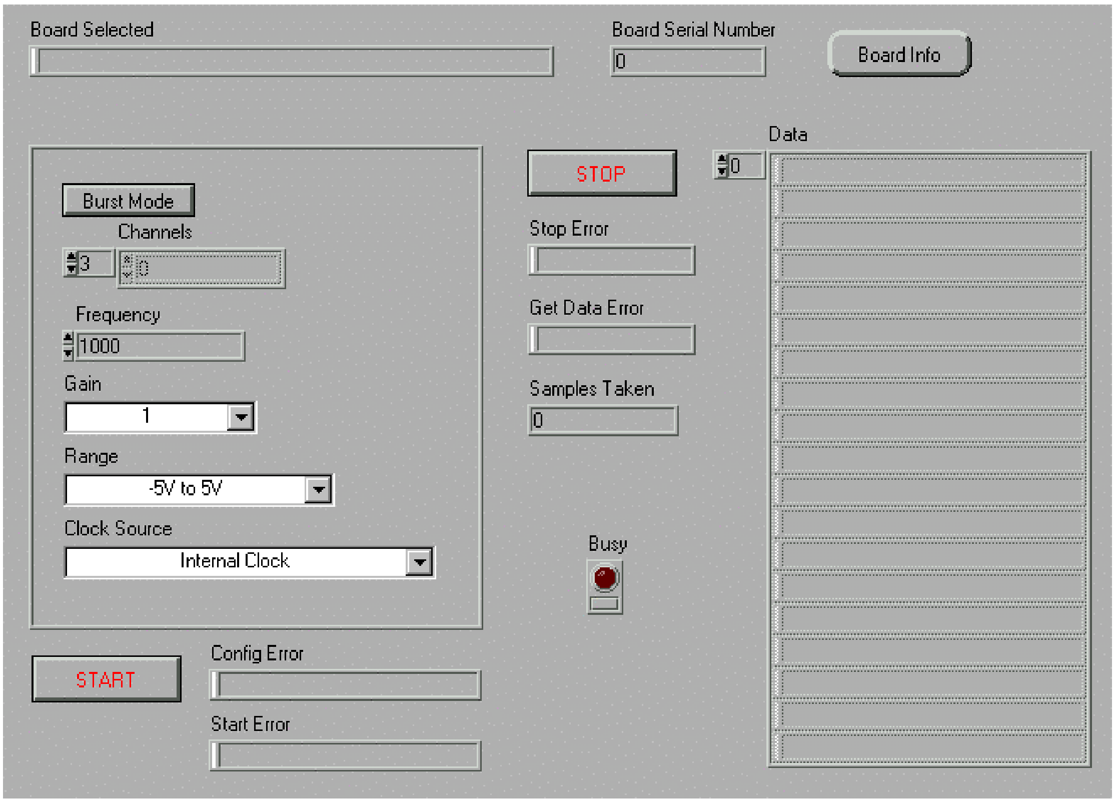
Data: 0

```
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
```

This example demonstrates how to achieve block sampling of data. Block sampling is required if large amounts of samples need to be taken every second. The example starts by asking the user to select the board to use for sampling. The user should then configure the channel to use, the sampling frequency to be used for sampling the channel, the input voltage range of the signal and the gain to apply to the incoming signal before sampling. The user should also specify the clock source to be used. The clock source can be either the internal frequency generator, the external clock line where the user supplies the clock pulses, or the internal frequency generator gated with the external clock line. These three options of clock sources should cover sampling at a fixed frequency and at a user specified frequency.

The user should then press the start button. The start button initiates two operations. Firstly, it configures the card, with the error of this operation displayed in the “Config Error” string. Secondly, it starts the sampling, with the error of this call displayed in the “Start Error” string. At this stage the busy light should be on, and sampling will continue until the stop button is pressed. This stops the sampling, displaying the error in the “Stop Error” string and then the data is retrieved from the device driver, which displays its error in the “Get Data Error” string if an error did occur. The number of samples that were taken between the start button being pressed and the stop button being pressed is displayed in the “Samples Taken” indicator. Finally, all the samples that were taken are displayed in an array of voltages.

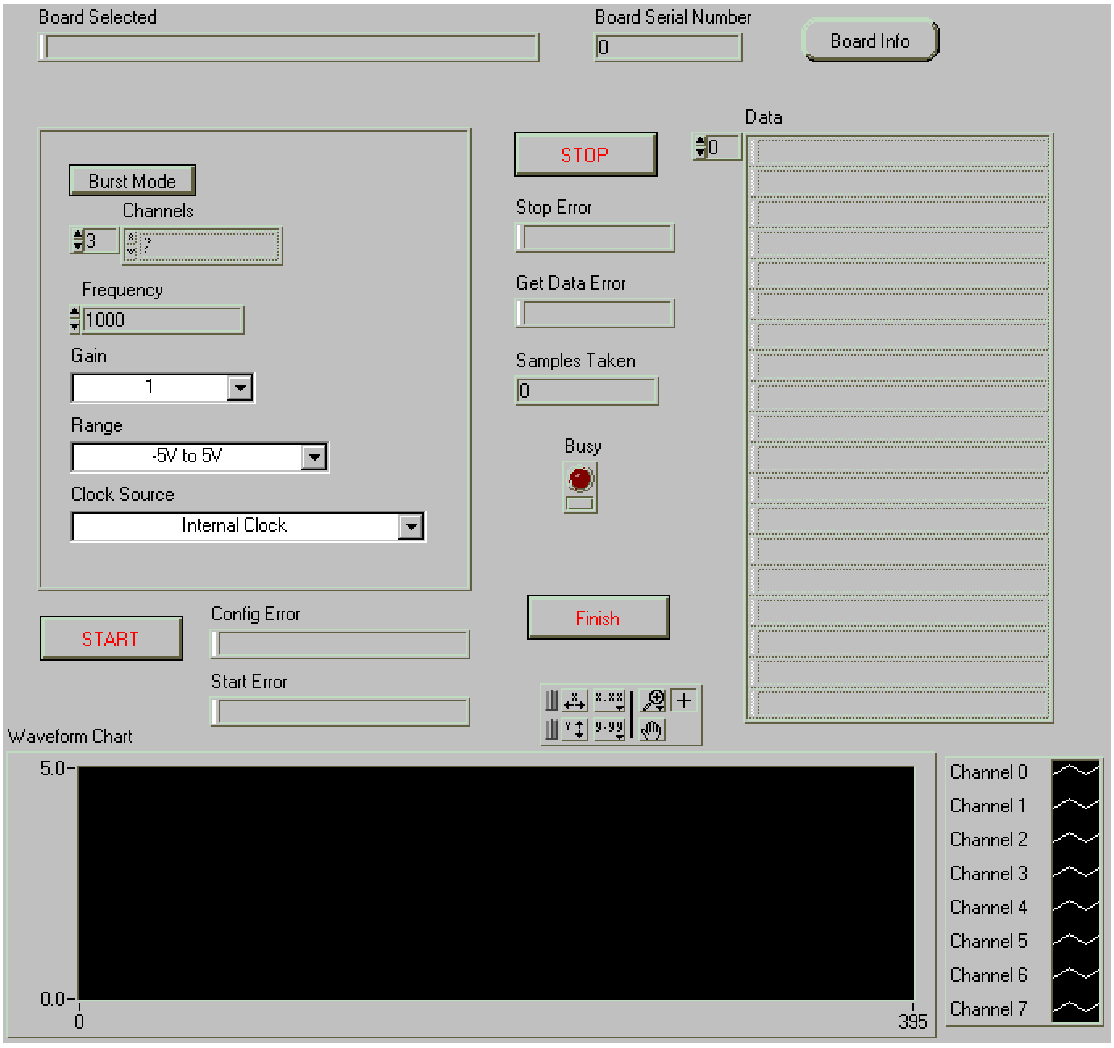
1.1.6 AD Block Multi



This example is a more advanced version of the “AD Block” example. It allows the user to select multiple channels to use for sampling. The example starts by asking the user to select the board to use for the sampling. The user should then configure the sampling frequency to be used for sampling the channel, the input voltage range of the signal and the gain to apply to the incoming signal before sampling. The user specifies the channels to sample in the channel array. Thus, if the array contains the values {0, 1, 5, 9}, then these channels will be sampled by the example in this order. The user can also enable burst mode. In burst mode, on every clock pulse all the channels in the channel array are sampled. If burst mode is disabled, then at a clock pulse only a single channel is sampled, with the next channel in the channel array only being sampled at the following clock pulse. The user should also specify the clock source to use. The clock source can be either the internal frequency generator, the external clock line where the user supplies the clock pulses, or the internal frequency generator gated with the external clock line. These three options of clock source should cover sampling at a fixed frequency and at a user specified frequency.

The user should then press the start button. The start button initiates two operations. Firstly, it configures the card, with the error of this operation displayed in the “Config Error” string. Secondly, it starts the sampling, with the error of this call displayed in the “Start Error” string. At this stage the busy light should be on, and sampling will continue until the stop button is pressed. This stops the sampling, displaying the error in the “Stop Error” string and then it retrieves the data from the device driver, which displays its error in the “Get Data Error” string if an error did occur. The number of samples that were taken between the start button being pressed and the stop button being pressed is displayed in the “Samples Taken” indicator. Finally, all the samples that were taken are displayed in an array of strings. Because each voltage belongs to a different channel, each string contains both the channel number and the voltage read on that channel.

1.1.7 AD Block Graph



Board Serial Number: 0

Channels: 3

Frequency: 1000

Gain: 1

Range: -5V to 5V

Clock Source: Internal Clock

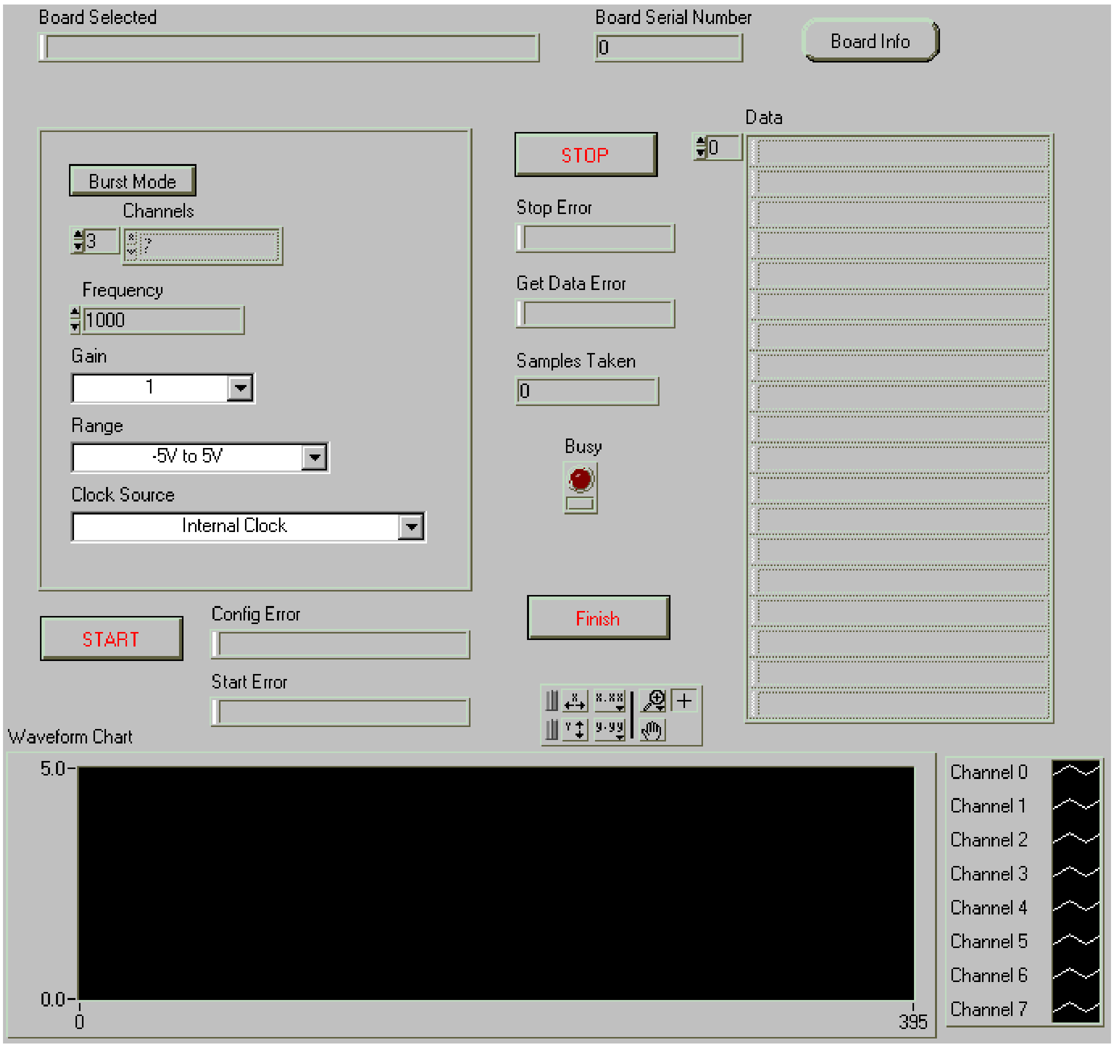
Samples Taken: 0

Data: 0

This example is an enhanced version of “AD Block Multi”, since it also graphs the output of the sampling operation on a graph. The example starts by asking the user to select the board to use for the sampling. The user should then configure the sampling frequency to be used for sampling the channel, the input voltage range of the signal and the gain to apply to the incoming signal before sampling. The user specifies the channels to sample in the channel array. Thus, if the array contains the values {0, 1, 5, 9}, then these channels will be sampled by the example in this order. The user can also enable burst mode. In burst mode, on every clock pulse all the channels in the channel array are sampled. If burst mode is disabled, then at a clock pulse only a single channel is sampled, with the next channel in the channel array only being sampled at the following clock pulse. The user should also specify the clock source to use. The clock source can be either the internal frequency generator, the external clock line where the user supplies the clock pulses, or the internal frequency generator gated with the external clock line. These three options of clock source should cover sampling at a fixed frequency and at a user specified frequency.

The user should then press the start button. The start button initiates two operations. Firstly, it configures the card, with the error of this operation displayed in the “Config Error” string. Secondly, it starts the sampling, with the error of this call displayed in the “Start Error” string. At this stage the busy light should be on, and sampling will continue until the stop button is pressed. This stops the sampling, displaying the error in the “Stop Error” string and then it retrieves the data from the device driver, which displays its error in the “Get Data Error” string if an error did occur. The number of samples that were taken between the start button being pressed and the stop button being pressed is displayed in the “Samples Taken” indicator. Finally, all the samples that were taken are displayed in an array of strings. Because each voltage belongs to a different channel, each string contains both the channel number and the voltage read on that channel. The additional graph displays the data as a strip chart of values for each channel.

1.1.8 AD Block Graph File



Board Selected

Board Serial Number: 0

Channels: 3

Frequency: 1000

Gain: 1

Range: -5V to 5V

Clock Source: Internal Clock

Stop Error

Get Data Error

Samples Taken: 0

Config Error

Start Error

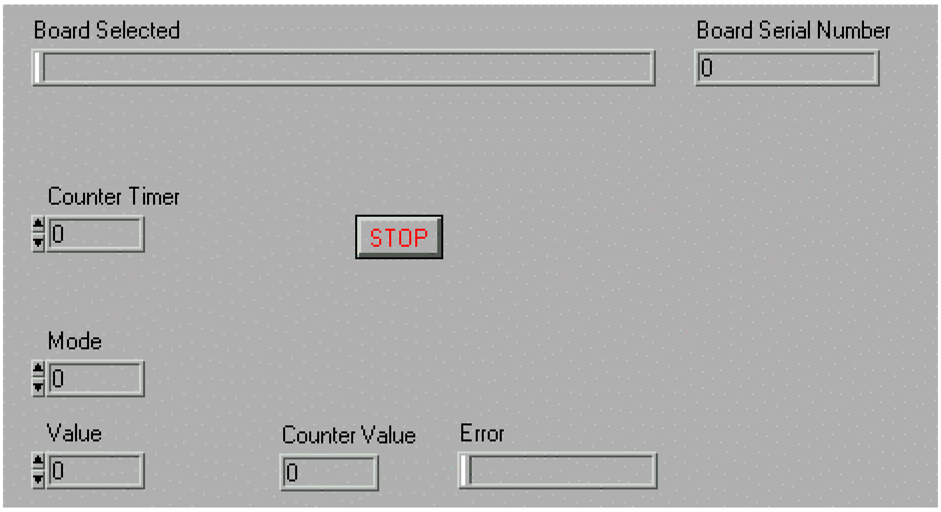
Data: 0

This example is similar to “AD Block Graph” although it saves the data in a spreadsheet readable format in a file as selected by the user. The example starts by asking the user to select the board to use for the sampling. The user should then configure the sampling frequency to be used for sampling the channel, the input voltage range of the signal and the gain to apply to the incoming signal before sampling. The user specifies the channels to sample in the channel array. Thus, if the array contains the values {0, 1, 5, 9}, then these channels will be sampled by the example in this order. The user can also enable burst mode. In burst mode, on every clock pulse all the channels in the channel array are sampled. If burst mode is disabled, then at a clock pulse only a single channel is sampled, with the next channel in the channel array only being sampled at the following clock pulse. The user should also specify the clock source to use. The clock source can be either the internal frequency generator, the external clock line where the user supplies the clock pulses, or the internal frequency generator gated with the external clock line. These three options of clock source should cover sampling at a fixed frequency and at a user specified frequency.

The user should then press the start button. The start button initiates two operations. Firstly, it configures the card, with the error of this operation displayed in the “Config Error” string. Secondly, it starts the sampling, with the error of this call displayed in the “Start Error” string. At this stage the busy light should be on, and sampling will continue until the stop button is pressed. This stops the sampling, displaying the error in the “Stop Error” string and then it retrieves the data from the device driver, which displays its error in the “Get Data Error” string if an error did occur. The number of samples that were taken between the start button being pressed and the stop button being pressed is displayed in the “Samples Taken” indicator. Finally, all the samples that were taken are displayed in an array of strings. Because each voltage belongs to a different channel, each string contains both the channel number and the voltage read on that channel. The additional graph displays the data as a strip chart of values for each channel. The example will then ask the user to select a file to which to write the data. This data is saved in a format that should be understood by all spreadsheet programs.

1.2 Counter/Timer

1.2.1 Counter Timer



Board Selected

Board Serial Number 0

Counter Timer 0

Mode 0

Value 0

Counter Value 0

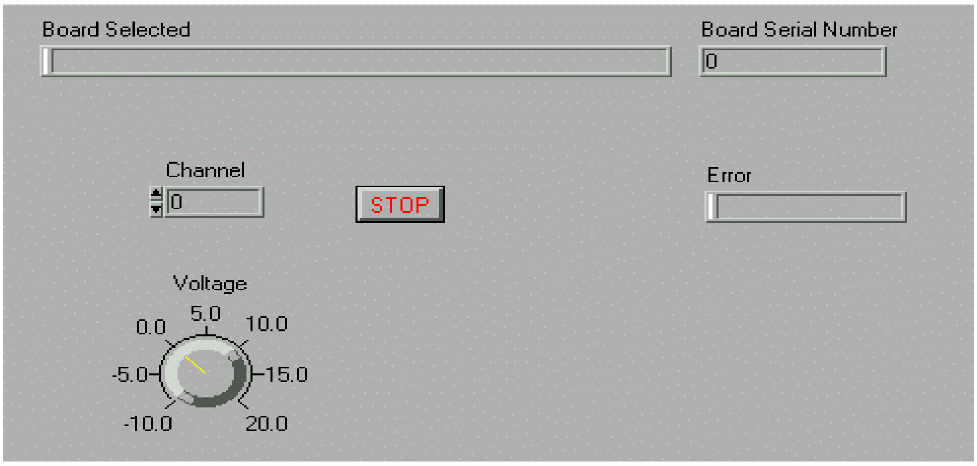
Error

This example demonstrates how to write to the clock/timer. The example starts by asking the user to select the board to use for the sampling. The user should then select the counter timer on which to operate. Every 300ms the example will read the value contained in the counter timer and display it to the user. If the user has external clock pulse generating circuitry connected to the clock input of the counter / timer, then this value should decrement. The user can also change the mode and starting value of the counter timer.

When the user presses the write button, then the values in the “Mode” and “Value” controls are written into the counter / timer. In the arrival of the next clock pulse at the counter / timer the value written should be loaded into the counter timer, and this new value will be displayed in the “Counter Value” indicator.

1.3 Digital to Analogue Conversion

1.3.1 Write Single DA


Board Selected

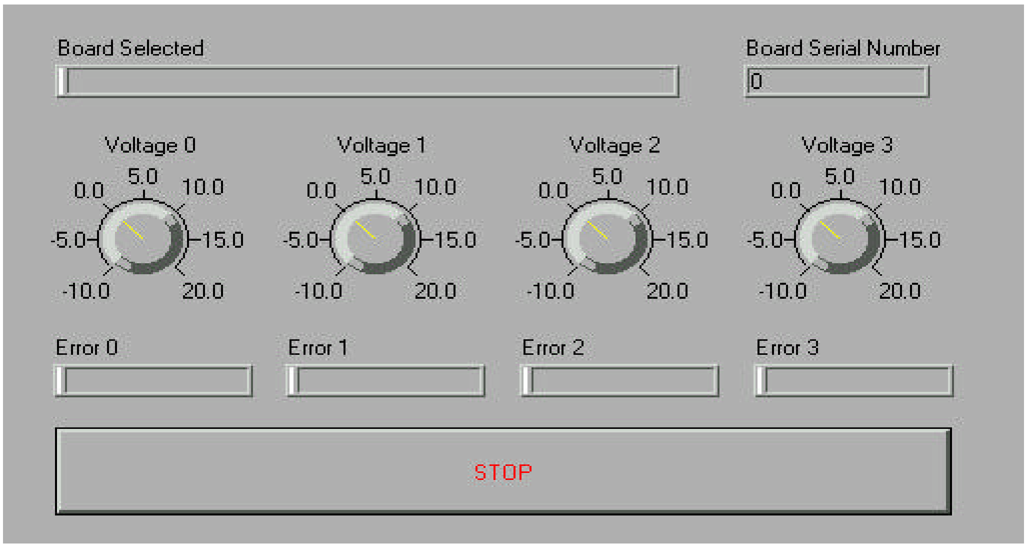
Board Serial Number 0

Channel 0

Error

This example demonstrates how to perform a single DA conversion and write the results to specified channel. The example starts by asking the user to select the board to use for the sampling. The user should then configure the output voltage to be written and the channel to which to output this voltage. An error indicator displays the error string if any error occurred writing the voltage output. The selected voltage or any new voltage will be written out to the DA channel every 300ms until the stop button is pressed.

1.3.2 Write 4 DA Channels


Board Selected

Board Serial Number 0

Error 0

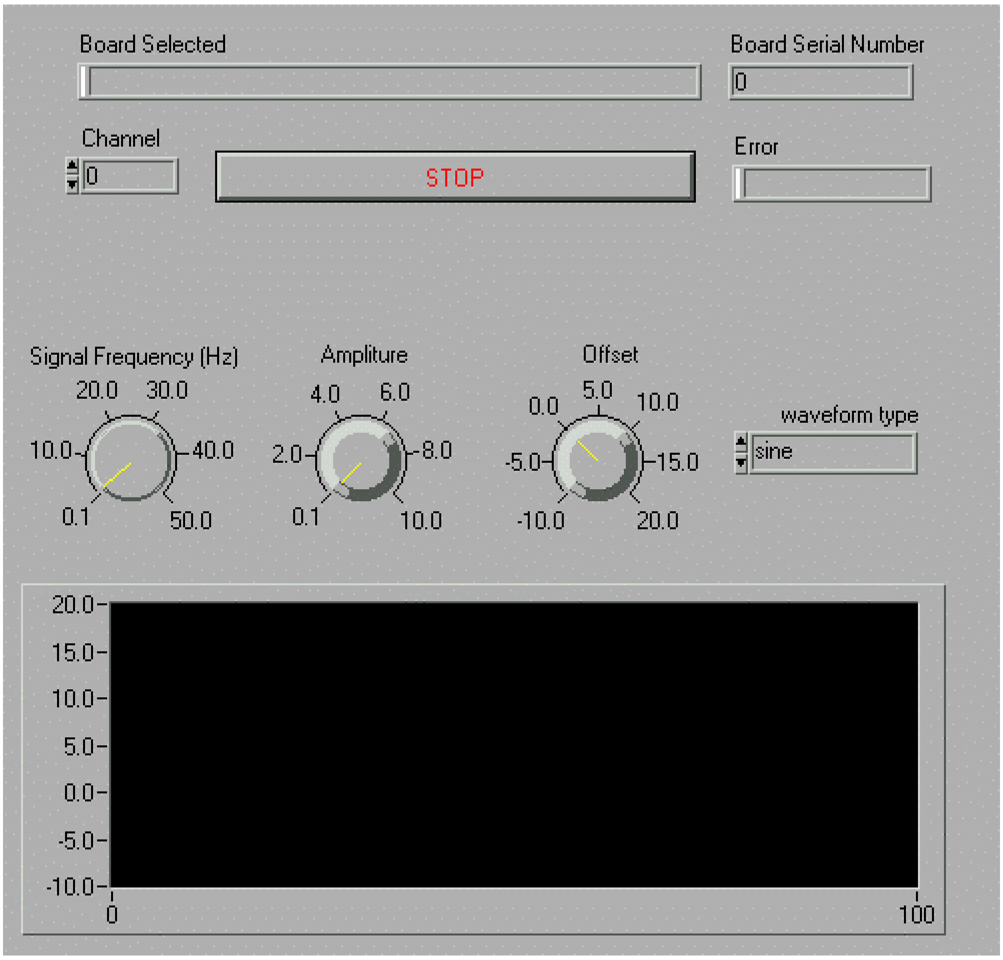
Error 1

Error 2

Error 3

This example is the same as "Write Single DA" with the exception that it supports 4 DA output channels simultaneously. The example starts by asking the user to select the board to use for sampling. The user should then supply the output voltage for each of 4 channels to be written to. An error indicator for each DA channel displays the error string if any error occurred writing that voltage output. The selected voltages or any new voltages will be written out to the DA channel every 300ms until the stop button is pressed.

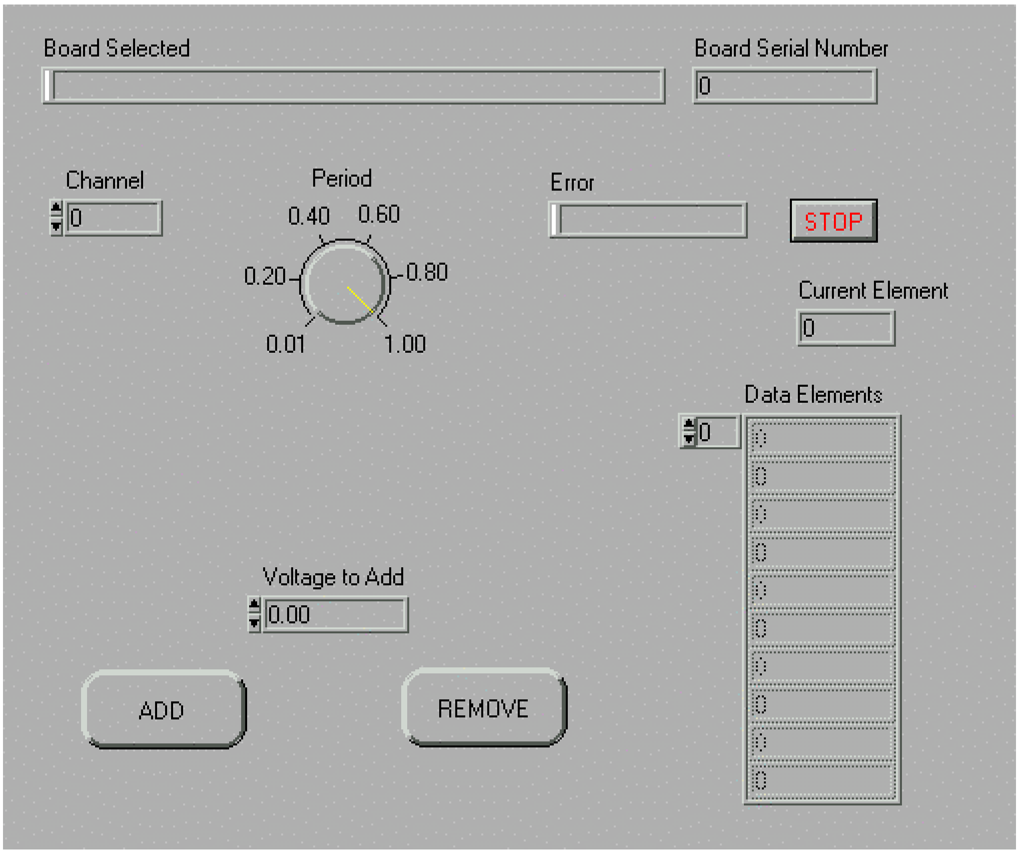
1.3.3 Waveform Output



This example demonstrates how to generate different output waveforms. The example starts by asking the user to select the board to use for sampling. The user should then set the output channel, the signal frequency, the amplitude, the voltage offset and the waveform type to use to generate the waveform. The generated waveform is displayed on the waveform chart.

The output waveform can be changed at any time, and will continue to be generated until the user clicks on the stop button. This example is capable of generating sine, cosine, increasing ramp, decreasing ramp, sawtooth, triangular and square waveforms.

1.3.4 Pattern DA Output


Board Selected

Board Serial Number 0

Channel 0

Error

Current Element 0

Data Elements 0

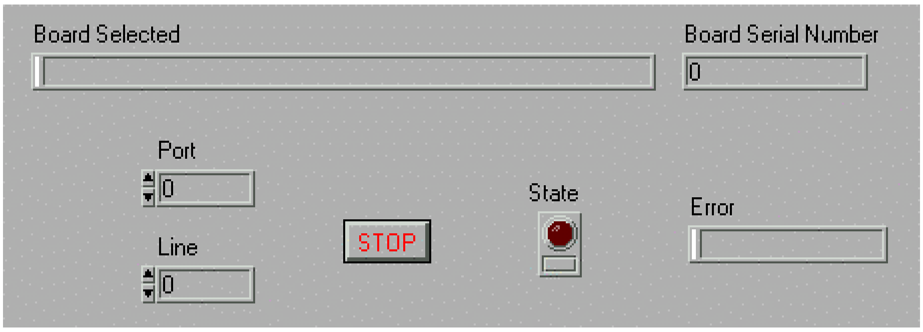
Voltage to Add 0.00

This example is a more general adaptation of the “Waveform Output” example. It is capable of generating any user-defined output required. The example starts by asking the user to select the board to use for sampling. The user should then select the output channel, and the signal period. To enter the output waveform, the user should type in the waveform in the “Voltage to Add” control and click on the Add button. This adds the voltage to the bottom of the array of voltages.

The remove button removes the last voltage in the array of voltages. The example operation loops through all the voltages present in the array and writes these to the selected channel. A write operation takes place every interval as specified by the user with the Period control. The error indicator displays the error, if any, that has occurred writing the data and the “Current Element” displays the last written index in the array of “Data Elements” that was written to the DA channel.

1.4 Digital Input/Output

1.4.1 Read Single Line



Board Selected

Board Serial Number: 0

Port: 0

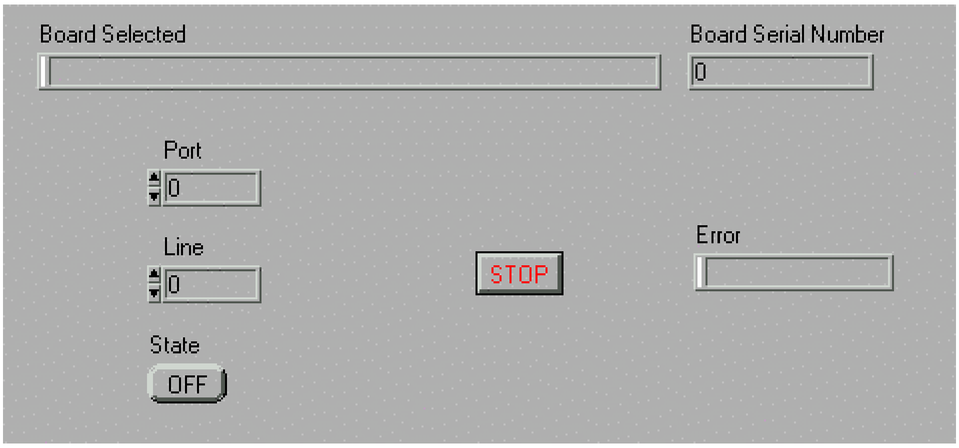
Line: 0

State

Error

This example asks the user to select a board to use for sampling. The user should then specify the DIO port to read, and the DIO line on that port. The example then loops through every 300ms reading the specific port and uses the state light to indicate whether the line on this port is high or low. The error string, if any, is displayed in the Error indicator.

1.4.2 Write Single Line



Board Selected

Board Serial Number: 0

Port: 0

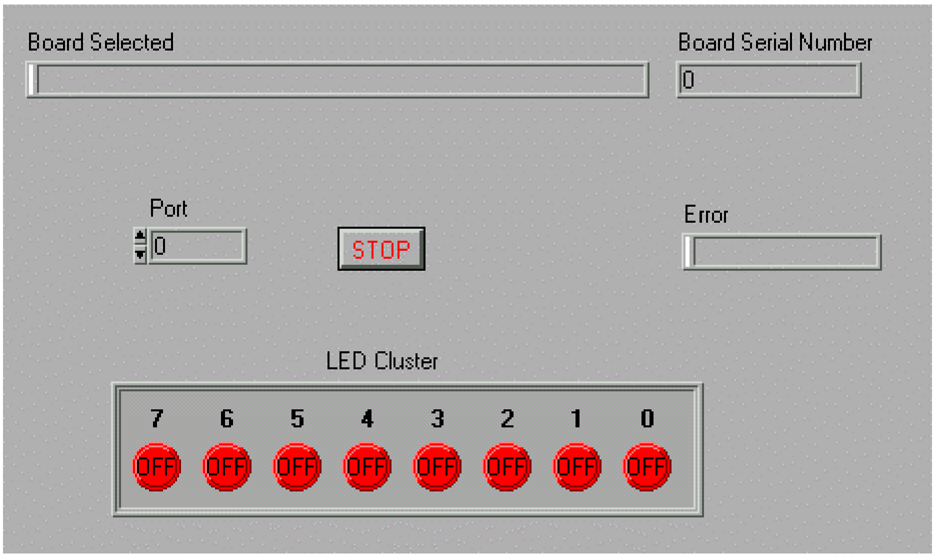
Line: 0

State: OFF

Error

This example asks the user to select a board to use for sampling. The user should then specify the DIO port to write to, and the DIO line on that port that the user wants to write to. The example then loops through every 300ms writing the bit value as specified by the state button to the specific port and line. The error string, if any error occurs, is displayed in the Error indicator.

1.4.3 Read Single Port



Board Selected

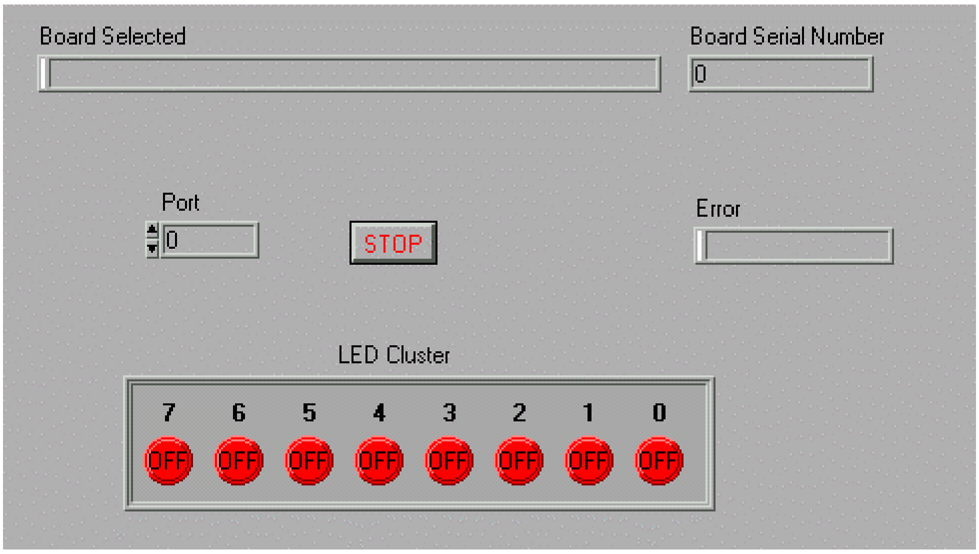
Board Serial Number: 0

Port: 0

Error

This example asks the user to select a board to use for sampling. The user should then specify the DIO port to read. The example then loops through every 300ms reading the specific port and uses the eight state lights to indicate whether the bits on the port are high or low. The error string, if any, is displayed in the Error indicator.

1.4.4 Write Single Port



Board Selected

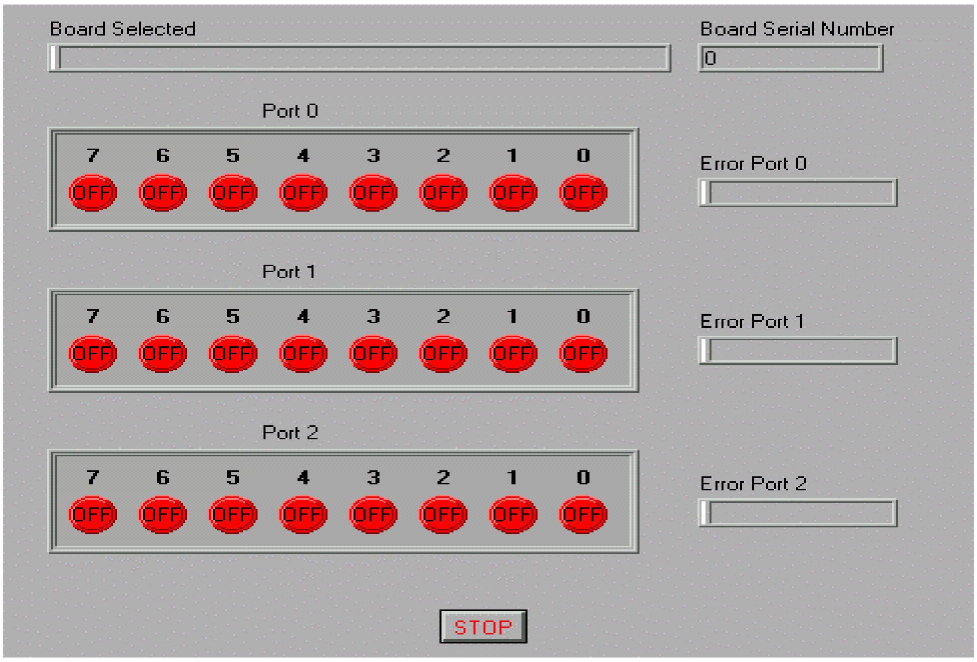
Board Serial Number: 0

Port: 0

Error

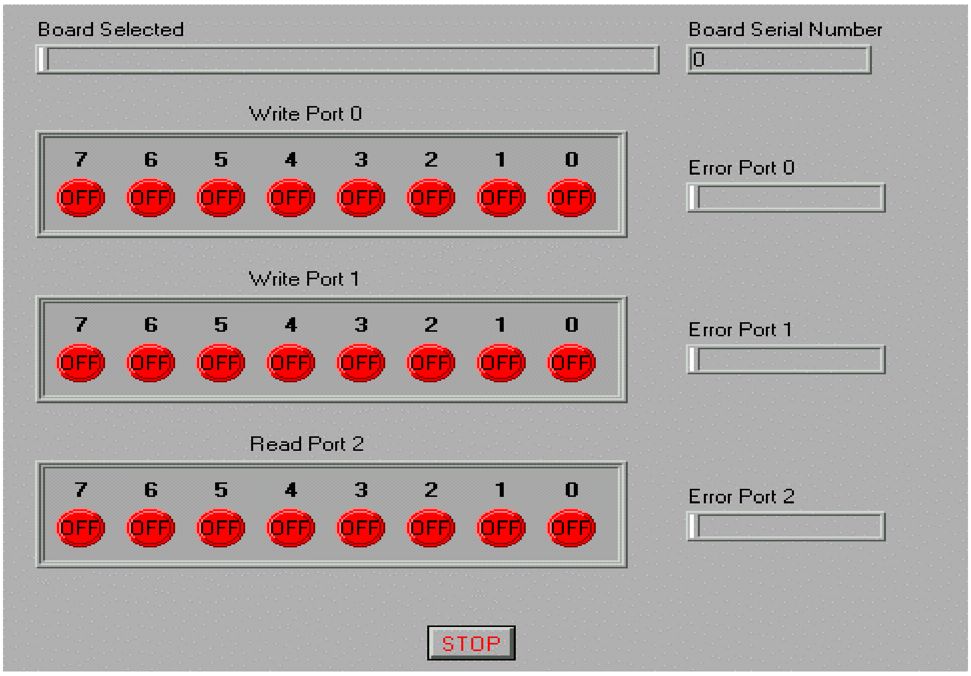
This example asks the user to select a board to use for sampling. The user should then specify the DIO port to write to. The example then loops through every 300ms writing the bit pattern as specified by the user using the button controls to the specific port. The error string, if any error occurs, is displayed in the Error indicator.

1.4.5 Write 3 Port



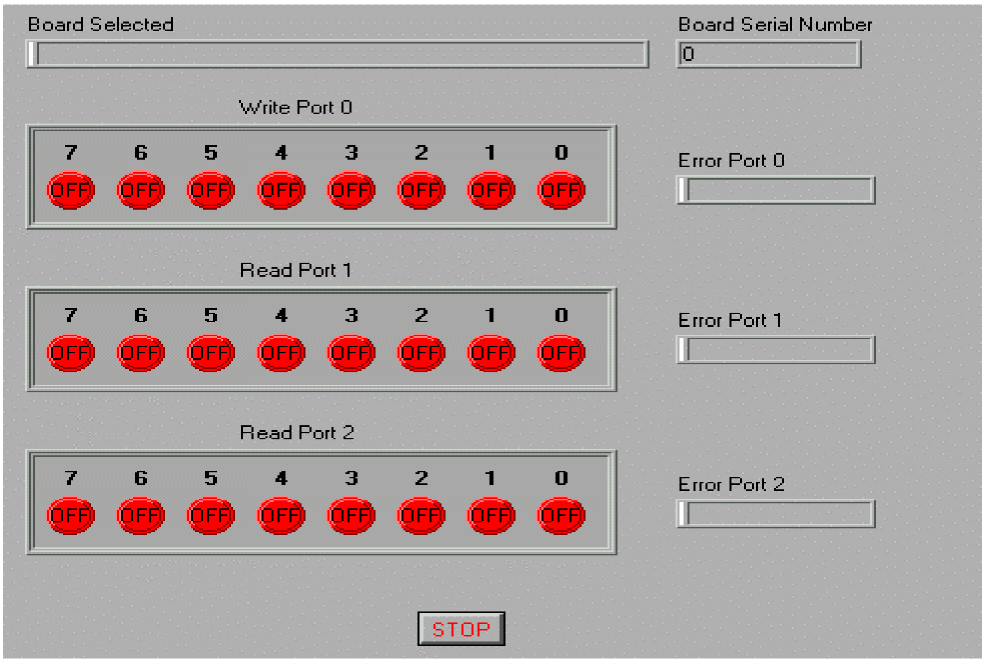
This example asks the user to select a board to use for sampling. The example loops through every 300ms writing the bit pattern as specified by the user using the three button controls to the corresponding port. The error string, if any error occurs, is displayed in the Error indicator for each of three ports.

1.4.6 Write 2 Read 1 Port



This example is a combination of “Read Single Port” and “Write Single Port” with Port 0 and Port 1 being written to and Port 2 being read from.

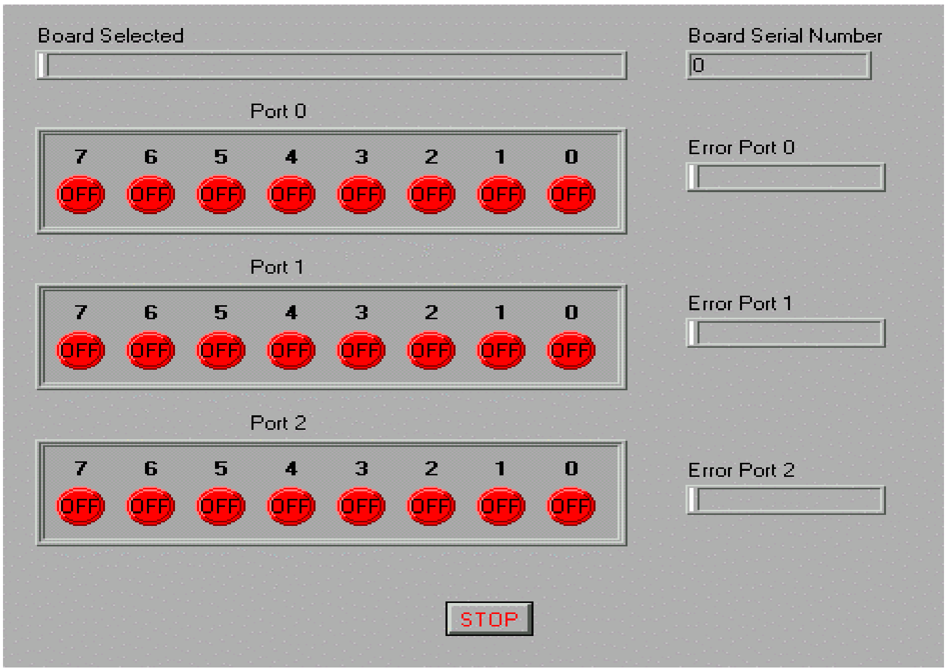
1.4.7 Write 1 Read 2 Port



Board Serial Number: 0

This example demonstrates how to write to 1 port and read and display content of the other 2 ports on selected board. The value to be written is set using the led (switches) and the read value displayed on relevant LED’s.

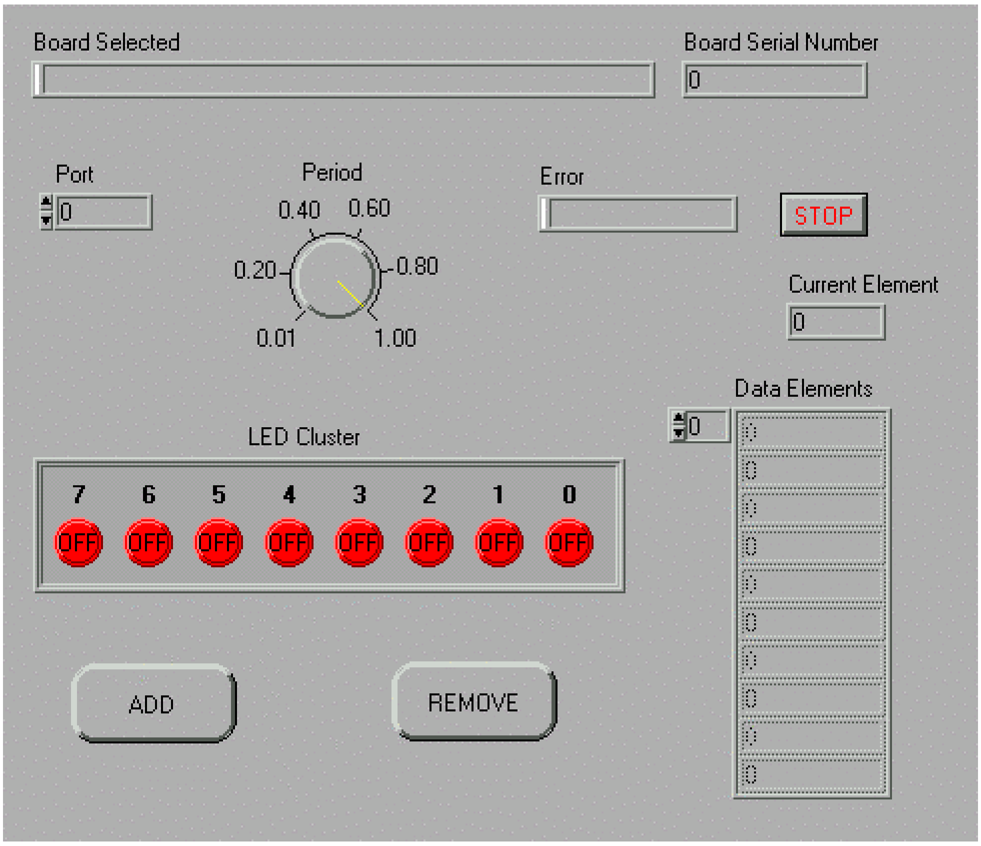
1.4.8 Read 3 Ports



Board Serial Number: 0

This example demonstrates how to read from 3 ports on the selected board. The example asks the user to select a board to use for sampling. The example outputs the values in each of the three ports in the three sets of LED indicators. The content is displayed using eight LED’s for each port, corresponding to each bit in a port. The LED may either be on or off depending on bit value.

1.4.9 Pattern Output

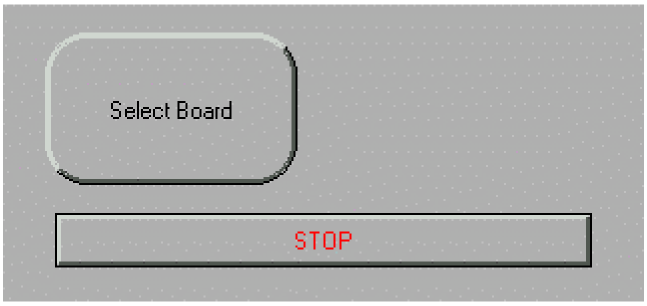


This example is capable of generating any user defined DIO output required. The example asks the user to select a board to use for sampling. The user should then select the output port to write to, and the signal period to use. To enter the output waveform, the user should click in the bit pattern using the LED buttons and click on the Add button. This adds the bit pattern to the bottom of the array of outputs.

The remove button removes the last bit pattern in the array of bit patterns. The example operation loops through all the values present in the array and writes these to the selected port. A write operation takes place every interval as specified by the user with the Period control. The error indicator displays the error, if any, that has occurred writing the data and the “Current Element” displays the last written index in the array of “Data Elements” that was written to the DIO port.

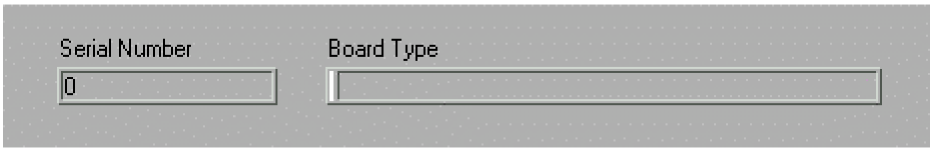
1.5 General

1.5.1 Example: Board Info



This example shows how to use the Board Info Utility VI. This example asks the user to select a board to use for sampling. It then displays all the information it has about the board, driver and windows API.

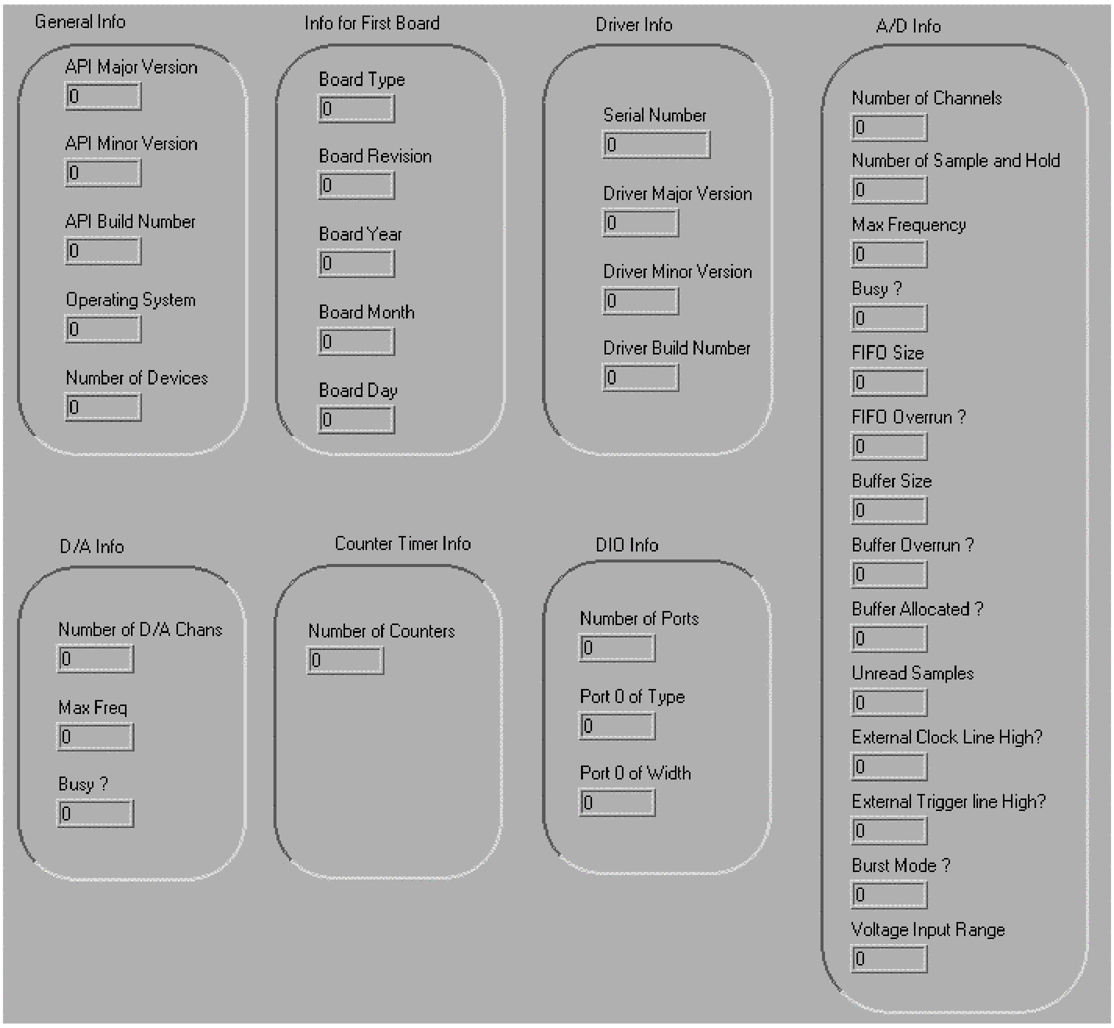
1.5.2 Example: Select Board


Serial Number: 0

Board Type:

This example shows how to use the Select Board Utility VI. This VI allows the user to select the board to use for sampling from a list rather than having to type in the board's serial number or a specific board index.

1.5.3 Example: Simple Query



General Info

API Major Version
0

API Minor Version
0

API Build Number
0

Operating System
0

Number of Devices
0

Info for First Board

Board Type
0

Board Revision
0

Board Year
0

Board Month
0

Board Day
0

Driver Info

Serial Number
0

Driver Major Version
0

Driver Minor Version
0

Driver Build Number
0

A/D Info

Number of Channels
0

Number of Sample and Hold
0

Max Frequency
0

Busy ?
0

FIFO Size
0

FIFO Overrun ?
0

Buffer Size
0

Buffer Overrun ?
0

Buffer Allocated ?
0

Unread Samples
0

External Clock Line High?
0

External Trigger line High?
0

Burst Mode ?
0

Voltage Input Range
0

D/A Info

Number of D/A Chans
0

Max Freq
0

Busy ?
0

Counter Timer Info

Number of Counters
0

DIO Info

Number of Ports
0

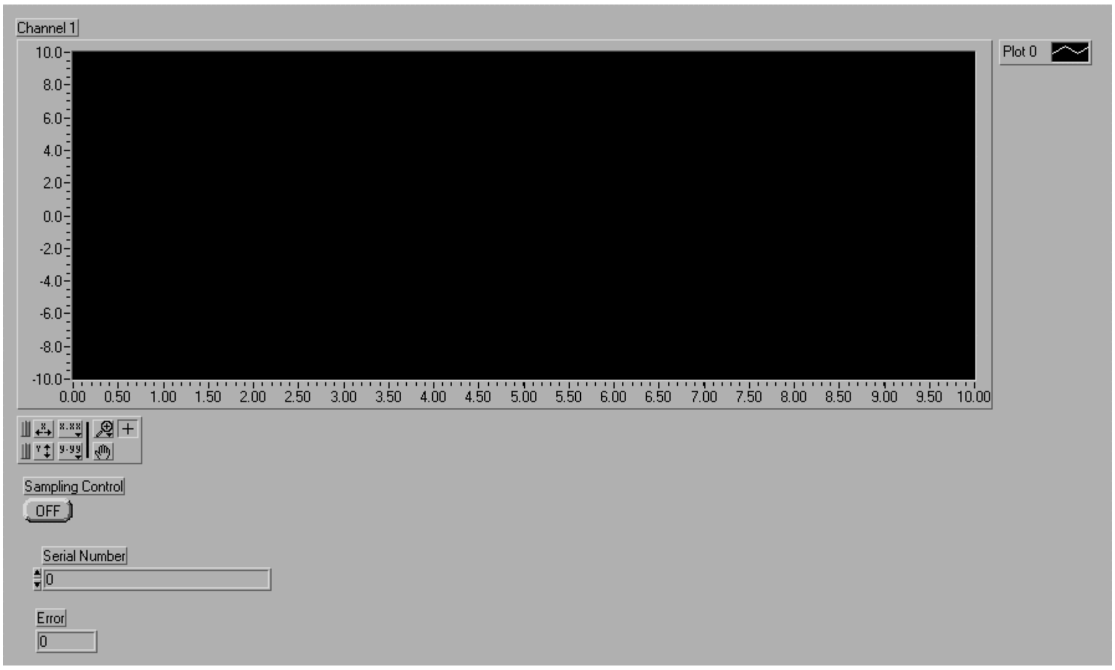
Port 0 of Type
0

Port 0 of Width
0

This example shows how to use the Simple Query Driver VI's. This example calls a whole collection of driver VI's to get information about the Board, the driver and the API on which it is running.

1.6 PCI 30fg

1.6.1 Example: AD Poll



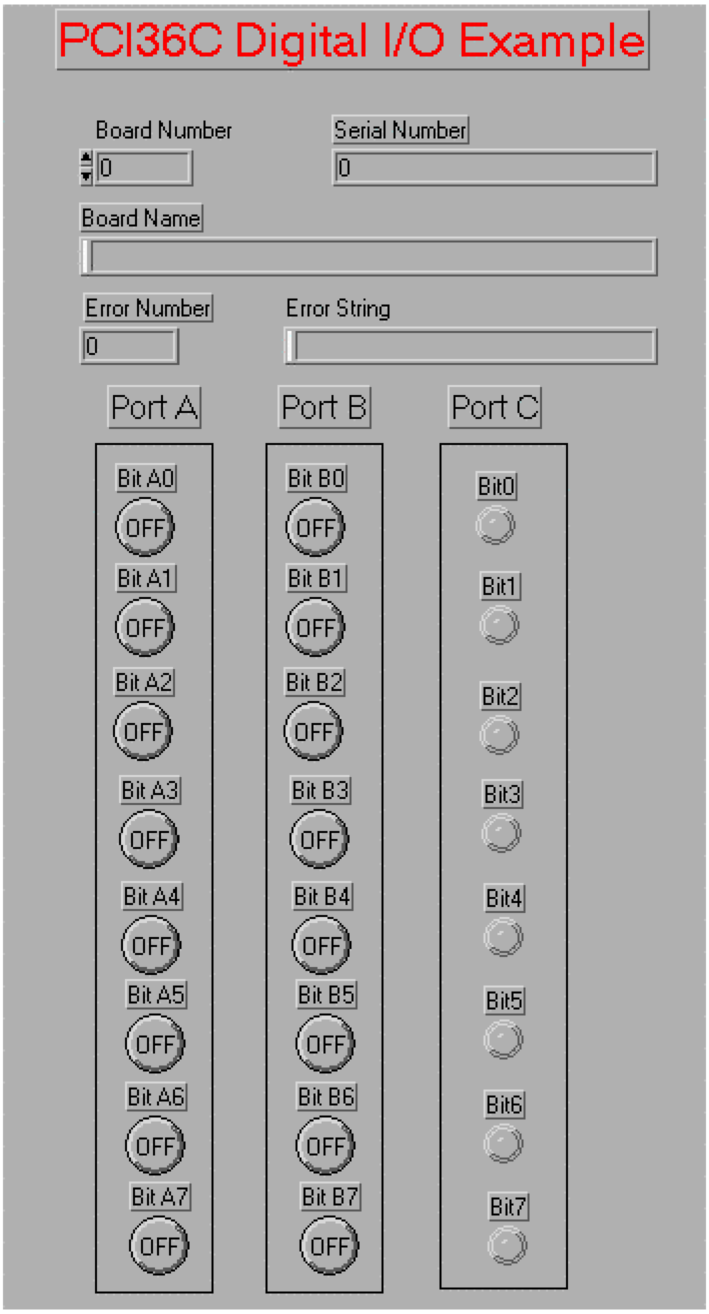
Serial Number: 0

Error: 0

This is an example specific to the PCI30FG card, demonstrating basic usage of the card to sample and graph a single channel.

1.7 PCI 36c

1.7.1 Example: PCI 36C



PCI36C Digital I/O Example

Board Number 0

Serial Number 0

Board Name

Error Number 0

Error String

This is an Example specific to the PCI36C. It shows the basic elements in writing to two DIO ports and reading from a third one.